Myths and Counterexamples in Mathematical Programming

Harvey J. Greenberg

hjgreenberg@gmail.com

February 20, 2010
(Previous: September 16, 2009)

Available at the INFORMS Computing Society
Mathematical Programming Glossary
http://glossary.computing.society.informs.org/.

Contact the author to be notified of updates.

Cartoon courtesy of John Zakour ▷

Contents

Acknowledgments

I thank the authors of the many papers I used in this compilation. I specifically thank those who contributed clarifications, suggestions, and updates: Karen Aardal, Mihai Anitescu, Hossein Arsham, Tiravat Assavapokee, Katy Azoury, Michel Balinski, Jonathan Bard, Amitabh Basu, András A. Benczúr, Add Ben-Israel, Harold P. Benson, Lorenz Biegler, Charles Blair, Winfried Bruns, Sharlee Climer, William L. Cooper, Gerard P. Cornuéjols, Yu-Hong Dai, Marbelly Davila, Michael A.H. Dempster, John Dennis, Matthias Ehrgott, James Flynn, Bennett Fox, Saul I. Gass, Fred Glover, Jochen Gorski, Richard C. Grinold, Julian Hall, Alan J. Hoffman, Allen Holder, David S.P. Hopkins, T. C. Hu, X.-D. Hu, Raj Jagannathan, Santosh N. Kabadi, Kathrin Klamroth, Ed Klotz, Sven Leyffer, Arjen Lenstra, Jan-Karel Lenstra, Steven H. Low, Timothy J. Lowe, Walter F. Mascarenhas, Olvi Mangasarian, Ken McKinnon, Nimrod Megiddo, José A. Moreno-Pérez, Jim Morris, Frederic H. Murphy, Katta G. Murty, Wlodzimierz Ogryczak, James B. Orlin, Michael O'Sullivan, Anand A. Paul, Matthew J. Realff, Romeo Rizzi, Sheldon Ross, Tim Roughgarden, Peter Salamon, Avijit Sarkar, Galen H. Sasaki, Lex Schrijver, Andyás Sebö, Suvrajeet Sen, Hanif Sherali, Danny Sleator, Jeannette Song, Ralph Steuer, Arie Tamir, Philippe L. Toint, Craig Tovey, Frank Vanden Berghen, Hercules Vladimirou, Heiko Vogel, Andreas Wächter, Roman L. Weil, Jr., Richard Weber, Ward Whitt, H. Paul Williams, Gerhard J. Woeginger, Henry Wolkowicz, Giacomo Zambelli, Milan Zeleny, and Majid Zohrehbandian. I thank Tod Morrison for his help with LaTeX. I am also grateful for the support and encouragement from Allen Holder. Lastly, I thank the Auraria Library services, especially Geraldine Archuleta, Interlibrary Loan, and Elaine Jurries, Science/Engineering Bibliographer.

An added thank you to David Weisman for rebuilding *Myths* January 2014 using TexLive to solve the problem: *non-italic Computer Modern fonts are not searchable.* This pdf is fully searchable.

Morality Code

Citations

Cite this in book format:

> H. J. Greenberg. *Myths and Counterexamples in Mathematical Programming*. INFORMS Computing Society, http://glossary.computing.society.informs.org, February, 2010. (ongoing, first posted October 2008).

Here is the BibTeX entry:

```
@BOOK{GreenbergMyths,
    AUTHOR = {H. J. Greenberg},
    TITLE = {Myths and Counterexamples in Mathematical Programming},
    PUBLISHER = {INFORMS Computing Society},
    YEAR = {2010},
    month = {February},
    address = {\url{http://glossary.computing.society.informs.org}},
    note = {(ongoing, first posted October 2008)}
}
```

Hyperlinks

I have put hyperlinks to help navigate this document, but there are some imperfections. If you do not see what you should, according to the link you clicked, check the next page. Sometimes, when a Myth falls on the top of a new page, the link takes you to the previous page.

Opportunity Knocks

I add a note where I can identify publishable work to be done. (See Table of Contents.) If you take my advice, please send me the result (I shall credit you with the entry), cite this book, and consider submitting to *INFORMS Journal on Computing*.

Enjoy!

Introduction

This is an update of what I began in 1996 and is posted with the ICS *Mathematical Programming Glossary*[4] since October 2008. I follow the terms and notation in the *Glossary*, presenting examples in Linear Programming (LP), Integer Programming (IP), Dynamic Programming (DP), Nonlinear Programming (NLP), Multiple-Objective Programming (MOP), and Special Forms (SF). These comprise the sections that follow, but they are not a partition of mathematical programming in general. Many problems overlap; for example, a problem could be represented as an LP and a DP. Further, network problems are scattered in all of these. I placed an entry where I thought it should go for what we teach. Thus, most network problems are in the LP section, and a dynamic problem is in DP only if dynamic programming is the underlying methodology, not just that the model is dynamic.

The use of counterexamples to disprove some result that *seems* as though it is true is an old technique to deepen our understanding of the underlying concepts. One of the most impressive books I read in graduate school was *Counterexamples in Analysis*[2], by Gelbaum and Olmsted. Since then, similar books have appeared[1, 3, 5, 6, 7, 8, 9].

Pedagogically, one could put a theorem out to the students of the form: $P \to Q$, then list some counterexamples to Q. The goal is for the student to discover P that makes Q true. What are the properties of the pathologies? Some myths are counterexamples to previously-published claims. Although that renders the original claim obsolete (unless repaired), it is included to demonstrate the construction of a counterexample in what appeared to be a valid result, not only to the author but also to at least two referees and one editor. What property did they all miss, and where does it present a flaw in the alleged proof?

The myths and counterexamples I present here are not restricted to mathematical constructs. I have also included some practices that have grown in the folklore to dispel myths about "good" models, solutions, and computational efficiency. One class of myth to challenge our intuition is that the objective value cannot worsen when we improve resources and/or relax constraints. I list these as *better is worse* in the index. A related type of myth is *more for less*, also in the index.

I use fairly standard notation for mathematical objects (though they have no universal standard), some of which are shown in Table 1.

Table 1: Notation

(a, b)	open interval	$\{x : a < x < b\}$
$[a, b]$	closed interval	$\{x : a \le x \le b\}$
$\mathbb{R}$	set of real values	$(-\infty, \infty)$
$\mathbb{Z}$	set of integer values	$\{\dots, -2, -1, 0, 1, 2, \dots\}$
$\mathbb{Q}$	set of rational values	$\left\{\frac{p}{q} : p, q \in \mathbb{Z} : q > 0\right\}$

I use $\mathbb{R}_+, \mathbb{Z}_+, \mathbb{Q}_+$ to restrict the values to be non-negative. For example, $\mathbb{R}_+ = [0, \infty)$. I use $\mathbb{R}^n, \mathbb{Z}^n, \mathbb{Q}^n$ to denote n-vectors whose coordinates belong to the indicated set. For example, $\mathbb{Z}^n = \{x = (x_1, \dots, x_n) : x_j \in \mathbb{Z} \text{ for } j = 1, \dots, n\}$. These can be combined. For example, $\mathbb{Q}^n_+ = \{x \in \mathbb{R}^n_+ : x_j \in \mathbb{Q} \text{ for } j = 1, \dots, n\}$.

Following the *Glossary* notation, the general form of a mathematical program is given by:

$$\min f(x) : x \in X,\ g(x) \ge 0,\ h(x) = 0,$$

where $\emptyset \ne X \subseteq \mathbb{R}^n$, $f : X \to \mathbb{R}$, $g : X \to \mathbb{R}^m$, $h : X \to \mathbb{R}^M$. (The sense of optimization could be max.) The functional relations are called *constraints*.

I welcome suggestions for future versions.

General References

[1] M. Capobianco and J. C. Molluzzo. *Examples and Counterexamples in Graph Theory.* North-Holland, New York, NY, 1978.

[2] B. R. Gelbaum and J. M. H. Olmsted. *Counterexamples in Analysis.* Holden-Day, San Francisco, CA, 1964.

[3] B. R. Gelbaum and J. M. H. Olmsted. *Theorems and Counterexamples in Mathematics.* Springer-Verlag, New York, NY, 1990.

[4] A. Holder, editor. *Mathematical Programming Glossary.* INFORMS Computing Society, http://glossary.computing.society.informs.org, 2006–10.

[5] J. P. Romano and A. F. Siegel. *Counterexamples in Probability and Statistics.* Wadsworth & Brooks/Cole, Monterey, CA, 1986.

[6] S. Ya. Serovaiskii. *Counterexamples in Optimal Control Theory.* Inverse and Ill-Posed Problems. VSP, The Netherlands, 2004.

[7] L. A. Steen and J. A. Seebach, Jr. *Counterexamples in Topology.* Holt, Rinehart and Winston, Inc., New York, NY, 1971.

[8] J. M. Stoyanov. *Counterexamples in Probability.* John Wiley & Sons, New York, NY, 2nd edition, 1997.

[9] G. L. Wise and E. B. Hall. *Counterexamples in Probability and Real Analysis.* Oxford University Press, New York, NY, 1993.

Linear Programming

The general form of a linear program (LP) is the optimization of a linear function subject to a system of linear equations and inequalities. The *standard form* is

$$\min\ cx:\ Ax = b,\ x \geq 0,$$

where $\text{rank}(A) = m =$ number of equations. This form is particularly useful when considering the simplex method.

When talking about duality, I use the *canonical form*:

$$\min\ cx:\ Ax \geq b,\ x \geq 0.$$

(No rank condition on A.) This renders the dual prices non-negative, giving the dual canonical form:

$$\max\ \pi b:\ \pi A \leq c,\ \pi \geq 0.$$

Unless stated otherwise, or implied from context, the LP in question could be any linear system; it need not be in standard or canonical form.

The *standard simplex method* is the original pivot-selection rule by Dantzig, applied to the standard form — a variable with the greatest reduced cost (rate of improvement) is chosen to enter the basis. An alternative is the *best-gain* criterion, which evaluates the actual gain of each candidate to enter the basis by computing its change in level and multiplying by the rate of improvement.

A constraint is *redundant* if its removal does not change the set of feasible points. An inequality is an *implied equality* if it must hold with equality in every feasible solution.

LP Myth 1. *All redundant constraints can be removed.*

The reason this is incorrect is that once a redundancy is removed, the other constraints may no longer be redundant.

Counterexample. $x, y \geq 0$ and $x - y = 0$. Each non-negativity constraint is redundant, but they cannot both be removed. The redundancy of $x \geq 0$ follows from the equation and the non-negativity of y: $x = y \geq 0$.

Practical use was first reported by Tomlin and Welch[76], and that led to a theory of *common dependency sets* by Greenberg[36].

LP Myth 2. *A degenerate basis implies there is a (weakly) redundant constraint.*

Counterexample. Consider $y \geq 0$, $x \geq 1$, $x + y \leq 1$. The only feasible point is $(x, y) = (1, 0)$ with slack and surplus variables both 0. Thus, each of the possible feasible bases is degenerate, but no constraint is redundant.

Sierksma and Tijssen[71] generalized this: If a face of dimension $n - 1$ or $n - 2$ is degenerate, the defining linear inequalities are not minimal — that is, the system must contain either a

redundant inequality or an implied equality. Note the special conditions on dimension. For $n \geq 3$, it cannot apply generally to an extreme point (face of 0 dimension). A pyramid is a counterexample for $n = 3$. The pyramid's top extreme point is degenerate because it is the intersection of 4 planes, but none of the defining inequalities is redundant or an implied equality.

LP Myth 3. *If an LP has an optimal solution, there is an extreme point of the feasible region that is optimal.*

Counterexample. Arsham[3, #9] provides the following: $\max x_1 + x_2 : x_1 + x_2 \leq 5$. The feasible set is a polyhedron with no extreme point. This occurs because we do not require the variables to be non-negative.

The myth's statement is true when the LP is in standard form. Converting the example to standard form increases the dimension:

$$\begin{gathered} \max u_1 - v_1 + u_2 - v_2 : \\ u_1 - v_1 + u_2 - v_2 + x_3 = 5, \\ u_1, v_1, u_2, v_2, x_3 \geq 0, \end{gathered}$$

where we have augmented the slack variable, x_3, and we have partitioned each of the original variables into their positive and negative parts:

$$x_1 = u_1 - v_1 \text{ and } x_2 = u_2 - v_2.$$

(Be sure to see LP Myth 13.)

In this higher-dimensional space, it is true that an extreme point is optimal — in particular, $(u_1, v_1, u_2, v_2, x_3) = (5, 0, 0, 0, 0)$. In fact, there are three extreme points; the other two are (0,0,5,0,0) and (0,0,0,0,5). Each of these three extreme points is optimal for some objective value coefficients, spanning all that render the LP optimal (vs. unbounded).

LP Myth 4. *If one knows that an inequality constraint must hold with equality in every optimal solution, it is better to use the equality in the constraint because it will reduce the solution time.*

First, it is not necessarily the case that it will reduce the solution time — the solver could get a first feasible solution faster with the inequality formulation. Second, even if the tighter version solves faster (perhaps by pre-solve reduction), it is generally better to let the model tell you the answer than for you to wire the result. Your intuition could be wrong, or there could be a data entry error that goes undetected with the equality constraint. A better approach is to attach a back-end report to examine all things "known" to be true and flag the violations. Thus, if an inequality is slack and you expected it to be tight, you can investigate why the model did what it did.

LP Myth 5. *In a dynamic LP, each period should be the same duration.*

This is tacitly implied in many textbook examples. The reality is that we know more about what is likely to happen tomorrow than next year. In general, data can provide forecasts for demands, supplies, and other model parameters, but the accuracy tends to be less as the time is further into the future. One may have, for example, a 5-year planning model with the first 12 time periods being months, the next 4 periods being quarters, and the last 3 being years.

LP Myth 6. *Maximizing an absolute value can be converted to an equivalent LP.*

Consider the conversion of the NLP with free variables:

$$\max \sum_j c_j |x_j| : Ax = b$$

to a standard LP:

$$\max \sum_j c_j x_j^+ + \sum_j c_j x_j^- : Ax^+ - Ax^- = b,\ x^+, x^- \geq 0.$$

Shanno and Weil[70] point out that this equivalence is not correct if $c \not\leq 0$.

Counterexample. max $|x| : -4 \leq x \leq 2$, where x is a free variable. (Add slack variables to put into equality form.) The associated LP is

$$\max\ x^+ + x^- : -x^+ + x^- + s_1 = 4,\ x^+ - x^- + s_2 = 2,\ x^+, x^-, s \geq 0.$$

The LP is unbounded (let $x^+ = 4 + \theta$, $x^- = \theta \to \infty$), but the original NLP is optimized by $x = -4$.

Shanno and Weil note that the unboundedness problem is avoided with the simplex method by adding the restricted basis entry condition: $x_j^+ x_j^- = 0$ for all j. When $c \leq 0$, this condition is satisfied anyway, but for $c_j > 0$, it must be forced.

Rao[60] points out that $c \geq 0$ means the objective function is convex, which implies there is an extreme point that is optimal, but there could be (and generally are) local maxima. On the other hand, $c \leq 0$ means the objective function is concave, so local maxima is not an issue.

Kaplan[48] proposed the following modification. Bound the variables by a single constraint:

$$\sum_j x_j^+ + \sum_j x_j^- \leq M,$$

where M is large enough to make this redundant when the NLP has a solution. Then, he purported that if this constraint is not active at the LP optimum (that is, if the slack variable is basic), it solves the NLP. If it is active (that is, if the slack variable is nonbasic), the NLP is unbounded. Unfortunately, this simple fix does not always work.

Counterexample. Ravindran and Hill[61] provide the following:

$$\max\ |x_1| : x_1 - x_2 = 2.$$

Kaplan's LP is:

$$\begin{aligned} \max\ & x_1^+ - x_1^- : x^+, x^-, s \geq 0, \\ x_1^+ - x_1^- - x_2^+ + x_2^+ &= 2 \\ x_1^+ + x_1^- + x_2^+ + x_2^+ + s &= M. \end{aligned}$$

The simplex method obtains the basic solution with $x_1^+ = 2$ and $s = M - 2$ (and all other variables zero). Thus, this does not solve the NLP. The problem here is that the LP can have only two basic variables, and the original polyhedron has no extreme points.

The unboundedness is not the real issue. Ravindran and Hill note that we could add the constraint $-6 \le x_1 \le 4$. Then, the LP solution is the same, but the original problem is solved by $x = (-6, -8)$.

For $c \le 0$, the NLP is equivalent to minimization of the form:

$$\min \sum_j |\alpha_j x_j - \beta_j| : x \in P,$$

where P is the polyhedron. This is equivalent to the LP:

$$\min \sum_j v_j : x \in P,\ v_j \ge \alpha_j x_j - \beta_j,\ v_j \ge -\alpha_j x_j + \beta_j.$$

This is the common LP equivalent, and it uses two properties: $|z| = \max\{z, -z\}$ and $\min\{v : v = |z|\} = \min\{v : v \ge |z|\}$. This latter property fails for maximization. The Shanno-Weil example would become

$$\max v : v \ge x,\ v \ge -x,\ -6 \le x \le 4,$$

which is unbounded.

Opportunity Knocks

There remains the issue of how we can use LP to maximize a linear function of absolute values, where the coefficients (c) could be positive. For $c > 0$, we know this is an instance of the hard problem of maximizing a convex function on a polyhedron, and there can be local maxima at some vertices. However, is there some special structure to exploit?

LP Myth 7. *The expected value of the second-stage of a stochastic linear program with recourse is a differentiable function, provided that the random variable is continuous.*

My thanks to Suvrajeet Sen for suggesting this.

The 2-stage recourse LP model is defined here as:

$$\min cx + \mathbf{E}_\theta[h(x, \theta)] : x \ge 0,\ Ax = b,$$

where θ is a random variable, and the *recourse function* is the LP value:

$$h(x, \theta) = \min\{Cy : y \ge 0,\ By = \theta - Tx\}.$$

The myth asserts that h is differentiable in x, provided the probability distribution function of θ is continuous. (It is obvious that h is not generally differentiable for a discrete distribution function since then h is piece-wise linear.)

Counterexample. Sen[69] provides the following: let $\theta = (d_1, d_2, d_3)$ be demands in the second stage for three destinations, and let the first stage determine supplies from two sources, so h is the optimal value of a transportation problem:

$$\begin{aligned} h(x, \theta) = \min & \textstyle\sum_{i,j} C_{ij} y_{ij} : y \ge 0, \\ y_{i1} + y_{i2} + y_{i3} \le x_i & \quad \text{for } i = 1, 2 \\ y_{1j} + y_{2j} \ge d_j & \quad \text{for } j = 1, 2, 3. \end{aligned}$$

Suppose d_1, d_2 are deterministic and $d_3 \in (0, D)$ for some finite $D > 0$. Let the unit shipping cost matrix be

$$C = \begin{bmatrix} 0 & 1 & 2 \\ 3 & 2 & 2 \end{bmatrix}.$$

Suppose $\bar{x} = (d_1, d_2 + D)$. Then, the following are alternative dual-optimal solutions:

$$\lambda = (-3, 0,\ 3, 2, 2) \text{ and } \lambda' = (-1, 0,\ 1, 2, 2).$$

(Supply prices are $-(\lambda_1, \lambda_2)$, and demand prices are $(\lambda_3, \lambda_4, \lambda_5)$.) Sen proves that these are optimal for all $d_3 \in (0, D)$. The subgradient of h thus includes subgradients $(-3, 0)$ and $(-1, 0)$, so the recourse function is not differentiable at $\bar{x}$.

Sen extends earlier works to establish necessary and sufficient conditions for h to be differentiable.

LP Myth 8. **new** *For a multistage stochastic program with non-anticipativity constraints, there exist optimal dual multipliers that are also non-anticipative.* ***next new*** ▷

My thanks to Suvrajeet Sen for suggesting this.

Non-anticipativity constraints require recourse variables to be independent of the history of outcomes. (See Beasley[7] for a succinct introduction and example.)

Counterexample. Higle and Sen[43] consider a 3-stage LP:

$$\min \sum_{t=1}^{3} c_t x_t : \ -1 \le x_t \le 1,\ t = 1, 2, 3,\ x_t \ge x_{t+1},\ t = 1, 2.$$

Let c be random with four equally-likely values:

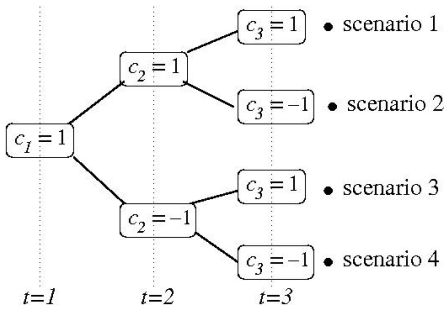
$$c \in \{(1, 1, 1),\ (1, 1, -1),\ (1, -1, 1),\ (1, -1, -1)\}.$$

Indexing the objective coefficients as c_{it} for scenario i at stage t, let x_{it} be the associated decision variable. Thus, the recourse LP is

$$\min \tfrac{1}{4} \sum_{i=1}^{4} \sum_{t=1}^{3} c_{it} x_{it} : \ -1 \le x_{it} \le 1,\ t = 1, 2, 3,\ x_{it} \ge x_{i\,t+1},\ t = 1, 2$$
$$x_{11} = x_{21} = x_{31} = x_{41},\ x_{12} = x_{22},\ x_{32} = x_{42},\ x_{13} = x_{33},\ x_{23} = x_{43}, \qquad \text{(LP.1)}$$

where (LP.1) comprise the non-anticipativity constraints. These are due to the commonality: $c_{i1} = 1$ for all $i = 1, \ldots, 4$, $c_{12} = c_{22}$, and $c_{32} = c_{42}$.

The scenario tree, shown on the right, illustrates that each path through time corresponds to a scenario. There is a bifurcation at a node when there is an event that changes the cost coefficient. For example, at $t = 1$, events can cause $c_2 = 1$ or $c_2 = -1$. However, to avoid clairvoyance, the decision variable, x_2, must be the same for scenarios 1 and 2, and for scenarios 3 and 4, since the cost is the same within each of those groupings. That is the "commonality" that yields the non-anticipativity constraints.



Let $u = (u_1, u_2, u_3)$ denote the dual variables for the non-anticipativity constraints associated with $t = 1$, and let $v = (v_1, v_2)$ be those associated with $t = 2$. The myth asserts that the dual variables associated with the non-anticipativity constraints are themselves non-anticipative — that is, $u_1 = u_2 = u_3$. However, the dual solution has $u = (\frac{1}{4}, -\frac{1}{2}, -\frac{1}{4})$, giving a contradiction.

Higle and Sen model the non-anticipativity constraints differently, but primal-equivalent to (LP.1):

$$x_{i1} - \tfrac{1}{4}\sum_{k=1}^{4} x_{k1} = 0 \qquad \text{for } i = 1, \ldots, 4 \qquad \text{(LP.2a)}$$
$$x_{i2} - \tfrac{1}{2}(x_{12} + x_{22}) = 0 \qquad \text{for } i = 1, 2 \qquad \text{(LP.2b)}$$
$$x_{i2} - \tfrac{1}{2}(x_{32} + x_{42}) = 0 \qquad \text{for } i = 3, 4 \qquad \text{(LP.2c)}$$

The dual variables now measure the rate of deviation from a group's average. Intuition may suggest that this averaging induces a non-anticipative dual stochastic process.

However, an optimal dual value has $u = (0, 0, \frac{3}{8}, \frac{1}{8})$ and $v = (0, \frac{1}{4}, 0, \frac{1}{2})$, which contradict non-anticipativity. (Higle and Sen obtain different optimal dual values, but they show all optimal dual values are non-anticipative.)

Higle and Sen prove that the optimal non-anticipativity dual variables are non-anticipative if, and only if, the expected value of perfect information equals zero. In the example, $EVPI = -1\frac{1}{2}$.

LP Myth 9. *A feasible infinite-horizon LP can be approximated by truncating to a finite horizon.*

The infinite-horizon model has the form:

$$f^* = \max \sum_{t=0}^{\infty} c^t x^t : x \geq 0,\ A^0 x^0 = b^0,\ A^{t+1} x^{t+1} - B^t x^t = b^t,\ \text{ for } t = 0, 1, \ldots$$

One associated finite-horizon model is the truncation:

$$f^*(T) = \max \textstyle\sum_{t=0}^{T} c^t x^t : x \geq 0,\ A^0 x^0 = b^0,$$
$$A^{t+1} x^{t+1} - B^t x^t = b^t,\ \text{ for } t = 0, 1, \ldots, T-1,$$
$$x^T \in \mathcal{T},$$

where $\mathcal{T}$ is an *end condition*.

Consider a stationary model, where $A^t = A$, $B^t = B$, and $b^t = b$ for all t. One could define $x^t = x^T$ for all $t > T$, in which case the end condition simply requires $(A - B)x^T = b$. For this case, Grinold[39] provides the following:

Counterexample. Let $A = 1$, $B = 1.6$, $b = 1$, and $c^t = (\frac{1}{4})^t$. Then, $x^t = \frac{1.6^{t+1} - 1}{0.6}$ is feasible, and $\sum_{t=0}^{\infty} c^t x^t = 2.222\ldots$ However, $(A - B)x = 1,\ x \geq 0$ has no solution.

Grinold provides another counterexample, where x^T is not required to satisfy $(A - B)x^T = b$.

Counterexample. Let $c^t = \left((\frac{1}{2})^t, (\frac{1}{2})^t, 0, 0 \right)$, $b = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$,

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}, \; B = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

The truncated-horizon model is

$$\max \textstyle\sum_{t=0}^{T} (\frac{1}{2})^t (x_1^t + x_2^t) : x \geq 0$$

$$\begin{array}{rcrcrcrcll} x_1^0 & & & + & x_3^0 & & & = & 1 & \\ & - & x_2^0 & & & + & x_4^0 & = & 0 & \\ x_1^t & & & + & x_3^t & & x_2^{t-1} & = & 1 & \text{for } t = 1, \ldots, T \\ & - & x_2^t & & & + & x_4^t - x_2^{t-1} & = & 0 & \text{for } t = 1, \ldots, T \end{array}$$

Let $x^t = (1, 0, 0, 0)^\intercal$ for $t = 0, \ldots, T-1$ and $x^T = (1, \theta, 0, \theta)$. This is feasible for all $\theta > 0$, and the objective value satisfies

$$f^*(T) \geq \sum_{t=0}^{T} c^t x^t = \sum_{t-0}^{T} (\tfrac{1}{2})^t + \theta (\tfrac{1}{2})^T = (\theta - 1) (\tfrac{1}{2})^T + 2.$$

Letting $\theta \to \infty$, we conclude that the truncated LP is unbounded for all finite T. However, the infinite-horizon objective is bounded, with optimal value 2.

Grinold provides greater analysis and conditions for finite-horizon approximations. He extends his work by analyzing four methods to correct end effects[40]. Also see Evers[24]. For more infinite-horizon myths see DP entries (starting at p. 100).

Even when the myth's statement holds, a software anomaly can arise with discounting.

Counterexample. The following infinite-horizon model is unbounded:

$$\max \sum_{t=0}^{\infty} (\tfrac{1}{2})^t x_t : x \geq 0, \; x_t + \sum_{s=0}^{t-1} x_s \leq 2^{t+1}, \quad \text{for } t = 0, 1, \ldots$$

Letting $x_t = 2^t$, each constraint holds and the objective equals ∞. An associated finite-horizon model is the truncation:

$$f^*(T) = \max \sum_{t=0}^{T} (\tfrac{1}{2})^t x_t : x \geq 0, \; x_t + \sum_{s=0}^{t-1} x_s \leq 2^{t+1}, \quad \text{for } t = 0, 1, \ldots, T.$$

For T sufficiently large, the objective coefficient, $(\frac{1}{2})^t$, becomes zero in the computer. Thus, the computed value of $f^*(T)$ is bounded. In particular, both MATLAB® and CPLEX® reach this at $T = 20$, giving $f^*(T) = 43$ for all $T \geq 20$.

We can add stationary bounds, $x^t \leq U$ for $t > \tau$, so the infinite-horizon model is bounded. For $\tau > 20$, the problem persists: the truncated optima converge to the incorrect limit due to the computer's limitation of precision.

LP Myth 10. *The duality theorem applies to infinite LPs.*

An infinite LP is one with an infinite number of variables and constraints, which arises naturally in infinite-horizon models. The duality theorem includes the following implications:

1. If x is primal-feasible, y is dual-feasible, and they satisfy complementary slackness, they are optimal in their respective LPs.
2. If the primal and dual have optimal solutions, their objective values are equal.

Counterexample. Hopkins[46] rebukes the first implication with the following:

$$\begin{array}{c} \min\ x_1 : x \geq 0, \\ \begin{array}{rcrcrcl} x_1 & - & x_2 & & & \geq & 1 \\ & & x_2 & - & x_3 & \geq & 0 \\ & & & \ddots & & & \end{array} \end{array}$$

Its dual is given by:

$$\begin{array}{c} \max\ y_1 : y \geq 0, \\ \begin{array}{rcrcrcl} y_1 & & & & & \leq & 1 \\ -y_1 & + & y_2 & & & \leq & 0 \\ & - & y_2 & + & y_3 & \leq & 0 \\ & & & \ddots & & & \end{array} \end{array}$$

A primal-feasible solution is $x = (2, 1, 1, \dots)$; a dual-feasible solution is $y = (1, 1, \dots)$. They satisfy complementary slackness, but x is not optimal for the primal since $x = (1, 0, 0, \dots)$ is also primal-feasible with lower objective value.

Hopkins identifies the cause: the sequence $\left\{\sum_{i=1}^{T}\sum_{j=1}^{T} y_i A_{ij} x_j\right\}_{T\to\infty}$ is not absolutely convergent. (Hopkins proves that absolute convergence is a sufficient condition for duality to hold.)

Counterexample. Grinold and Hopkins[41] rebuke the second implication with the following:

$$\begin{array}{c} \min \sum_{t=0}^{\infty} \left(\frac{1}{2}\right)^t z_t : x_0 = 1,\ y_0 + z_0 = 1 \\ -2y_{t-1} + x_t = 0,\ -2x_{t-1} + y_t + z_t = 0 \quad \text{for } t = 1, 2, \dots \\ x_t, y_t, z_t \geq 0 \quad \text{for } t = 0, 1, \dots \end{array}$$

The objective is bounded below by zero. A feasible solution is $x_t = y_t = 2^t$, $z_t = 0$, and it is optimal since its objective value is zero.

The dual is

$$\begin{array}{c} \max\ u_0 + v_0 : \\ u_t - 2v_{t+1} \leq 0,\ v_t - 2u_{t+1} \leq 0,\ v_t \leq \left(\frac{1}{2}\right)^t, \quad \text{for } t = 0, 1, 2, \dots \end{array}$$

Since $v_0 \leq 1$ and $u_0 \leq 2v_1 \leq 1$, the objective value is bounded above by 2. A feasible solution is $u_t = v_t = \left(\frac{1}{2}\right)^t$, and it is optimal since its objective value is 2.

Moreover, the complementary slackness conditions are satisfied:

$$(u_t - 2v_{t+1})x_t = 0,\ (v_t - 2u_{t+1})y_t = 0,\ (v_t + (\tfrac{1}{2})^t)z_t = 0,\ \text{for } t = 0, 1, 2, \ldots$$

The failure of equal objective values can be attributed to the correction by u_{t+1}, v_{t+1} in the dual. The truncation yields an optimal value of zero because the last constraints do not have that correction:

$$u_T \le 0,\ v_T \le 0.$$

This back-propagates to render $u_0 = v_0 = 0$.

Also see Evers[24, §6.9,§6.20].

LP Myth 11. *If the optimal value of a slack variable is zero, the associated constraint is binding.*

As suggested by H. P. Williams, this myth reflects confusion in terminology. An inequality constraint is *active* at a point if it holds with equality; it is *binding* if its removal changes the solution.

Counterexample. $\max\ x_1 : x \ge 0,\ x_1 + 2x_2 \le 3,\ 2x_1 + x_2 \le 3,\ x_1 + x_2 \le 2.$

The (unique) optimal solution is at $x^* = (1, 1)$, and all slack variables are zero. Although the last constraint is active, it is not binding (it is redundant).

LP Myth 12. *If the primal and dual are both degenerate, they cannot both have alternative optima.*

Suggested by H. P. Williams, this myth violates the established fact:

> *If the primal and dual LPs have optimal solutions, they have a strictly complementary optimal solution.*

Counterexample.

Primal	Dual
$\max\ 0x : x \ge 0,\ x_1 \le 0,\ x_2 \ge 0.$	$\min\ 0\pi : \pi \ge 0,\ \pi_1 \ge 0,\ \pi_2 \le 0.$

Primal optima are of the form $(0, x_2) : x_2 \ge 0$; dual optima are of the form $(\pi_1, 0) : \pi_1 \ge 0$.

LP Myth 13. *It is a good idea to convert free variables to the standard form by the expression: $x = u - v$, where u is the positive part and v is the negative part of x.*

Too often students (and new graduates) do this, perhaps thinking it is necessary due to the text they used. However, all solvers handle free variables directly.

For a simplex method, the conversion requires a change in basis whenever x needs to change sign. This is an unnecessary pivot, wasting time and space. Recognition of free variables allows the solver to put all free variables into the basis at the start (dealing with linear dependence, if that should be a problem). Once in the basis, a free variable cannot block an entrant, so it simply stays there. Some solvers also use the free variable to eliminate a row (and restore it

after a solution is obtained). Thus, it is never a good idea to perform this conversion when using a simplex method.

For an interior method, this causes the optimality region to be unbounded (if it is not empty). Whatever the value of x^*, there is an infinite number of values of u^* and v^* that yield the same difference, $u^* - v^*$. During the iterations, it is not unusual for u and v to diverge, while maintaining a constant difference, and this divergence can cause numerical problems for the algorithm (especially for convergence detection).

LP Myth 14. *The standard simplex method does not select a dominated column to enter the basis.*

Consider LP in canonical form:

$$\max\ cx : x \geq 0,\ Ax \leq b.$$

A column, j, is *dominated* if there exists $k \neq j$ such that

$$c_k \geq c_j \text{ and } A_k \leq A_j.$$

Counterexample. Blair[13] provides the following:

$$\begin{array}{rrcrcrcrcl} \max & 5x_1 & + & 3x_2 & + & x_3 & + & x_4 & & \\ & x_1 & - & x_2 & + & 5x_3 & + & 3x_4 & \leq & 10 \\ & 3x_1 & + & x_2 & + & x_3 & + & x_4 & \leq & 40 \\ & -2x_1 & + & x_2 & - & 3x_3 & - & 3x_4 & \leq & 10 \\ & & & & & x \geq 0. & & & & \end{array}$$

After adding slack variables to convert to standard form, the first simplex tableau is:

	Level	x_1	x_2	x_3	x_4	s_1	s_2	s_3
←	10	1	−1	5	3	1	0	0
	40	3	1	1	1	0	1	0
	10	−2	1	3	3	0	0	1
	0	5	3	1	1	0	0	0
		↑						

The first pivot exchange is $s_1 \leftarrow x_1$:

	Level	x_1	x_2	x_3	x_4	s_1	s_2	s_3
	10	1	−1	5	3	1	0	0
←	10	0	4	−14	−8	−3	1	0
	30	0	−1	13	9	2	0	1
	50	0	8	−24	−14	−5	0	0
			↑					

Column 3 is dominated by column 4, but it enters the basis next:

Level	x_1	x_2	x_3	x_4	s_1	s_2	s_3
$12\frac{1}{2}$	1	0	$1\frac{1}{2}$	1	$\frac{1}{4}$	$\frac{1}{4}$	0
$2\frac{1}{2}$	0	1	$-3\frac{1}{2}$	−2	$-\frac{3}{4}$	$\frac{1}{4}$	0
$32\frac{1}{2}$	0	0	$9\frac{1}{2}$	7	$1\frac{1}{4}$	$\frac{1}{4}$	1
69	0	0	4	2	1	−2	0
			↑				

One way to look at Blair's example is that the dominance conditions are not generally preserved as the basis changes.

Another view is to drop the first two columns entirely and consider a 2-variable LP with an initial basis that is slack. The values of A do not affect the selection of the basis entrant. With equal costs, the first variable (x_3) is selected, which is dominated by the second (x_4).

Level	x_3	x_4	s_1	s_2	s_3
10	5	3	1	0	0
40	1	1	0	1	0
10	3	3	0	0	1
0	1	1	0	0	0
	↑				

LP Myth 15. *The affine scaling algorithm converges to an optimum extreme point.*

Counterexample. Mascarenhas[53] provides the following:

$$\begin{gathered}\min\ x_1 : x_1, x_2 \geq 0 \\ \alpha x_1 + \beta x_2 - x_3 \geq 0 \\ \beta x_1 + \alpha x_2 - x_3 \geq 0 \\ -x_1 - x_2 + x_3 \geq -1,\end{gathered}$$

where $\alpha = 0.39574487$ and $\beta = 0.91836049$. The optimal solution is at the extreme point $x^* = (0, 0, -1)$. The essence of the counterexample is Mascarenhas' proof that there exists a half-line such that starting there and using a step size of 0.999, causes all even iterates to be in the half-line, and they converge to zero.

LP Myth 16. *At optimality, $\pi^* b = cx^*$ — that is, the inner product of the optimal dual variables on the constraints and the right-hand side values equals the optimal primal objective value.*

While this is true for standard and canonical forms, it fails when primal bounds are handled directly. Consider the primal-dual LPs:

Primal	Dual
$\min\ cx : 0 \leq x \leq U,\ Ax \geq b.$	$\max\ \pi b - \mu U : \pi, \mu \geq 0,\ \pi A - \mu \leq c.$

At optimality, $cx^* = \pi^* b - \mu^* U$, so one must be careful to subtract $\mu^* U$ from $\pi^* b$ to obtain the correct equation.

Support for handling bounds directly, rather than including them in other constraints, is an example of how optimization software may use different conventions than in the theory. Such deviations from theory in the world of optimization software include reporting dual prices and/or reduced costs as the negative of their theoretically-correct values. One must check the manual or run a small test case to see how they are reported in any particular solver. (ANALYZE[34] reports theoretically-correct values, changing solver-values as needed.)

LP Myth 17. *Once the simplex method reaches an optimal vertex, it terminates.*

The fallacy is that the Basic Feasible Solution (BFS) reached must be both primal and dual optimal for the tableau to be terminal.

Counterexample. Gerard Sierksma provided the following (converted to standard form):

$$\begin{array}{rcrcrcl} & & \max\ x_1 + x_2 : x, s \geq 0 & & & & \\ x_1 & & & + & s_1 & = & 1 \\ & + & x_2 & + & s_2 & = & 1 \\ x_1 & + & x_2 & - & s_3 & = & 2 \end{array}$$

The extreme point $(1, 1)$ is optimal and corresponds to three BFSs:

basic	level	s_2	s_3
x_1	1	-1	-1
x_2	1	1	0
s_1	0	1	1
$-z$	2	0	1
			$\uparrow$

basic	level	s_1	s_3
x_1	1	1	0
x_2	1	-1	-1
s_2	0	1	1
$-z$	2	0	1
			$\uparrow$

basic	level	s_1	s_2
x_1	1	1	0
x_2	1	0	1
s_3	0	1	1
$-z$	2	-1	-1
		Terminal	

Only the third of these is both primal and dual optimal; the other two are not terminal. The reason is the myopic nature of rates, oblivious to the degeneracy:

Tableau 1

Δx_1	=	Δs_3
Δx_2	=	0
Δs_1	=	$-\Delta s_3$
Δz	=	Δs_3

Tableau 2

Δx_1	=	0
Δx_2	=	Δs_3
Δs_2	=	$-\Delta s_3$
Δz	=	Δs_3

Tableau 3

Δx_1	=	$-\Delta s_1$
Δx_2	=	$-\Delta s_2$
Δs_3	=	$-\Delta s_1 - \Delta s_2$
Δz	=	$-\Delta s_1 - \Delta s_2$

Tableau 1 sees a rate of change in the objective value as $+1$ per unit of increase in s_3 (keeping $s_2 = 0$). The linear equations show that the net rate of change in the objective value (z) is $+1$, which is its reduced cost. Similarly, tableau 2 sees a rate of change in the objective value as $+1$ per unit of increase in s_3 (keeping $s_1 = 0$). The linear equations show that the net rate of change in the objective value (z) is $+1$, which is its reduced cost. The third tableau has s_3 in the basis, so it responds to changes in either of the first two slack variables. Any increase in one slack value causes a decreases in its corresponding variable while keeping the other primary variable at 1 — for example,

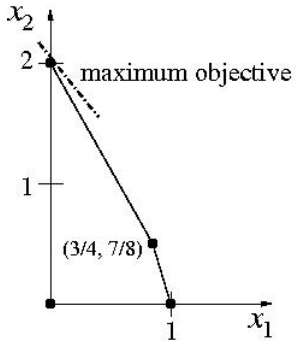
$$\Delta s_1 > 0 \Rightarrow \Delta x_1 = -\Delta s_1 < 0 \text{ and } \Delta x_2 = 0.$$

(The value of s_3 also decreases at the same rate, which does not affect the objective value.) The net effect is that the objective value decreases at that same unit rate, as indicated by the reduced cost. The same analysis applies to increasing s_2.

LP Myth 18. *In the absence of degeneracy, the standard simplex method does not repeat a basis exchange.*

Saaty[68] presented this conjecture with some supporting intuition. In the absence of degeneracy, this has a unique choice of departing variable for the exchange. However, Goldman and Kleinman[32] found the following:

Counterexample. This is a special case of the family of counterexamples in [32]:



$$\max\ 3x_1 + 2x_2 :$$
$$7x_1 + 2x_2 \le 7$$
$$3x_1 + 2x_2 \le 4$$
$$x_1, x_2 \ge 0.$$

Adding slack variables $s = (s_1, s_2)$, and starting at $x = (0, 0)$, the standard simplex iterations are:

Iteration	Vertex	Basic Variables	Basis Exchange
0	$(0, 0)$	$s_1,\ s_2$	$s_1 \leftarrow x_1$
1	$(1, 0)$	$x_1,\ s_2$	$s_2 \leftarrow x_2$
2	$(\,^3/_4\,,\,^7/_8\,)$	$x_1,\ x_2$	$x_1 \leftarrow s_1$
3	$(0, 2)$	$s_1,\ x_2$	

LP Myth 19. *The standard simplex method does not revisit a basic feasible solution (that is, cycle) as it pivots to an optimum.*

Hoffman[44] gave the first example of cycling in the standard simplex method, which has 11 variables and 3 equations.

Counterexample. The following is due to Beale[6], with only 7 variables and 3 equations.

x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
$(\mathbf{1/4})$	-60	$\mathbf{-1/250}$	9	1			0
$\mathbf{1/2}$	-90	$\mathbf{-1/50}$	3		1		0
		1				1	1
$\mathbf{-3/4}$	150	$\mathbf{-1/50}$	6	$\bullet$	$\bullet$	$\bullet$	0

↑

1	-240	$\mathbf{-4/250}$	36	4			0
	$(\mathbf{30})$	$\mathbf{3/50}$	-15	-2	1		0
		1				1	1
$\bullet$	-30	$\mathbf{-7/50}$	33	3	$\bullet$	$\bullet$	0

↑

1		$(\mathbf{8/250})$	-84	-12	8		0
	1	$\mathbf{1/500}$	$\mathbf{-1/2}$	$\mathbf{-1/15}$	$\mathbf{1/30}$		0
		1				1	1
$\bullet$	$\bullet$	$\mathbf{-2/250}$	18	1	1	$\bullet$	0

↑

$\mathbf{25/8}$		1	$\mathbf{-525/2}$	$\mathbf{-75/2}$	28		0
$\mathbf{-1/160}$	1		$(\mathbf{1/40})$	$\mathbf{1/120}$	$\mathbf{-1/60}$		0
$\mathbf{-25/8}$			$\mathbf{525/2}$	$\mathbf{75/2}$	-25	1	1
$\mathbf{1/4}$	$\bullet$	$\bullet$	-3	-2	3	$\bullet$	0

↑

$\mathbf{-125/2}$	10500	1		$(\mathbf{50})$	-150		0
$\mathbf{-1/4}$	40		1	$\mathbf{1/3}$	$\mathbf{-2/3}$		0
$\mathbf{-125/2}$	-10500			-50	150	1	1
$\mathbf{-1/2}$	120	$\bullet$	$\bullet$	-1	1	$\bullet$	0

↑

$\mathbf{-5/4}$	210	$\mathbf{1/50}$		1	-3		0
$\mathbf{1/6}$	-30	$\mathbf{-1/150}$	1		$(\mathbf{1/3})$		0
		1				1	1
$\mathbf{-7/4}$	330	$\mathbf{1/50}$	$\bullet$	$\bullet$	-2	$\bullet$	0

↑

Next tableau is same as first.

Hall and McKinnon[42] established the following form for a class of cycling examples with the same dimensions as Beale's example — four variables, three inequality constraints, one of which is just for bounding:

$$\max\ cx:\ x \geq 0,\ A^1x + A^2y \leq 0,\ x_1 + x_2 \leq 1,$$

where $c = (a, b)$ such that $a > 0 > b$, and A^1, A^2 and are 2×2 blocks such that $A^i_{11} + A^i_{22} = A^i_{21}A^i_{12} - A^i_{11}A^i_{22} = -1$ for $i = 1, 2$. In particular, they provide the following:

Counterexample.

$$\begin{array}{rcrcrcrcl} \max\ 2.3\,x_1 & + & 2.15\,x_2 & - & 13.55\,x_3 & - & 0.4\,x_4 & : & x \geq 0 \\ 0.4\,x_1 & + & 0.2\ x_2 & - & 1.4\ x_3 & - & 0.2\,x_4 & \leq & 0 \\ -7.8\,x_1 & - & 1.4\ x_2 & + & 7.8\ x_3 & + & 0.4\,x_4 & \leq & 0 \\ x_1 & + & x_2 & & & & & \leq & 1. \end{array}$$

The optimal solution is $(0, 1, 0, 1)$.

Using the standard max reduced cost for entry, Hall and McKinnon use the largest pivot value to select the variable to leave the basis (among those with minimum ratio). Starting with the basis of surplus variables, $\{x_5, x_6, x_7\}$, the example cycles after six iterations. An important difference with Beale's example is that Hall and McKinnon establish a *family* of smallest examples, for which the above is one instance.

Hall and McKinnon also provide a library of test problems at http:\www.maths.ed.ac.uk/hall/PublicLP/. The above example is called HAMCK26E. The library also includes examples of a related phenomenon, called *stalling*, where the objective remains constant for a large number of iterations.

Also see Gass and Vinjamuri[28] for more cycling examples.

LP Myth 20. *A simplex method using steepest-edge column selection does not cycle.*

Counterexample. Using the same construction approach as in LP Myth 19, Hall and McKinnon[42] provide the following:

$$\begin{array}{rcrcrcrcl} \max\ x_1 & + & 1.75\,x_2 & - & 12.25\,x_3 & - & 0.5\,x_4 & : & x \geq 0 \\ 0.4\,x_1 & + & 0.2\ x_2 & - & 1.4\ x_3 & - & 0.2\,x_4 & \leq & 0 \\ -7.8\,x_1 & - & 1.4\ x_2 & + & 7.8\ x_3 & + & 0.4\,x_4 & \leq & 1 \\ & - & 20\ x_2 & + & 156\ x_3 & + & 8\ x_4 & \leq & 0. \end{array}$$

Here are the tableaux that form the 6-cycle, where the last row in each tableau is the reduced cost divided by the Euclidean norm of the tableau column vector. (This is the initial rate of change in the objective value with respect to change in total distance. Further, it is scale-free and accounts for the geometry of the basis in the sense that $T_j = B^{-1}A_j$. See Greenberg and Kalan[38] for how this measure can be computed without solving $B\,T_j = A_j$ explicitly.) The steepest-edge rule chooses the maximum of these to enter the basis. (The departing variable remains chosen by largest pivot.)

x_1	x_2	x_3	x_4	x_5	x_6	x_7	RHS
(**0.4**)	0.2	−1.4	−0.2	1			0
−7.8	−1.4	7.8	0.4		1		0
	−20	156	8.0			1	1
1	1.75	−12.25	−0.5	•	•	•	0
0.128	0.09	−0.08	−0.06				0

1	0.5	−3.5	−0.5	2.5			0
	(**2.5**)	−19.5	−3.5	19.5	1		0
	−20	156	8	0		1	1
•	1.25	−8.75	0	−2.5	•	•	0
	0.06	−0.06	0	−0.13			0

1		(**0.4**)	0.2	−1.4	−0.2		0
	1	−7.8	−1.4	7.8	0.4		0
		0	−20	156	8	1	1
•	•	1	1.75	−12.25	−0.5	•	0
		0.13	0.09	−0.08	−0.06		0

19.5	1		(**2.5**)	−19.5	−3.5		0
2.5		1	0.5	−3.5	−0.5		0
0			−20	156	8	1	1
−2.5	•	•	1.25	−8.75	0	•	0
−0.13			0.06	−0.06	0		0

−1.4	−0.2	1		(**0.4**)	0.2		0
7.8	0.4		1	−7.8	−1.4		0
156	8			0	−20	1	1
−12.25	−0.5	•	•	1	1.75	•	0
−0.08	−0.06			0.13	0.09		0

−19.5	−3.5	19.5	1		(**2.5**)		0
−3.5	−0.5	2.5		1	0.5		0
156	8	0			−20	1	1
−8.75	0	−2.5	•	•	1.25	•	0
−0.06	0	−0.13			0.06		0

The next pivot exchange is $x_4 \leftarrow x_6$, which returns to the initial tableau.

The odd iterates have two candidates to enter the basis (that is, two reduced costs are positive). The one with greatest steepest-edge is opposite the one with greatest reduced cost. Then, there is only one positive entry in the column (0.4), which dictates the variable to leave the basis. The even iterates have only one candidate to enter the basis but two candidates to leave. The greatest pivot element is 2.5 (vs. 0.5).

LP Myth 21. *A simplex method does not cycle for an assignment problem.*

"A simplex method" is taken to mean any sequence of (adjacent) basic feasible solutions that enters a basic variable with negative reduced cost. This need not be the standard simplex method, which selects one with the most negative reduced cost.

Counterexample. Gassner[29] provides a 4×4 with costs:

$$c = \begin{bmatrix} 3 & 5 & 5 & 11 \\ 9 & 7 & 9 & 15 \\ 7 & 7 & 11 & 13 \\ 13 & 13 & 13 & 17 \end{bmatrix}.$$

Begin with the diagonal assignment: $x_{11} = x_{22} = x_{33} = x_{44} = 1$. Let the additional 3 basic (degenerate) variables be x_{12}, x_{23}, and x_{34}. Here is the initial (abbreviated) tableau:

			Nonbasic								
	Basic	level	x_{13}	x_{14}	x_{21}	x_{24}	x_{31}	x_{32}	x_{41}	x_{42}	x_{43}
	x_{11}	1	0	0	1	0	1	0	1	0	0
	x_{22}	1	−1	−1	1	0	1	1	1	1	0
	x_{33}	1	0	−1	0	−1	1	1	1	1	1
	x_{44}	1	0	0	0	0	0	0	1	1	1
	x_{12}	0	1	1	−1	0	−1	0	−1	0	0
←	x_{23}	0	**1**	1	0	1	−1	−1	−1	−1	0
	x_{34}	0	0	1	0	1	0	0	−1	−1	−1
		38	−2	2	4	4	0	−2	2	0	−2
			↑								

There are three candidates for entering the basis; select x_{13}. Then, there are two candidates to leave the basis; select x_{23}. The pivot results in the following tableau:

			Nonbasic								
	Basic	level	x_{14}	x_{21}	x_{23}	x_{24}	x_{31}	x_{32}	x_{41}	x_{42}	x_{43}
	x_{11}	1	0	1	0	0	1	0	1	0	0
	x_{22}	1	0	1	1	1	0	0	0	0	0
	x_{33}	1	−1	0	0	−1	1	1	1	1	1
	x_{44}	1	0	0	0	0	0	0	1	1	1
←	x_{12}	0	0	−1	−1	−1	0	1	0	**1**	0
	x_{13}	0	1	0	1	1	−1	−1	−1	−1	0
	x_{34}	0	1	0	0	1	0	0	−1	−1	−1
		38	4	4	2	6	−2	−4	0	−2	−2
										↑	

The next entering variable is x_{42}, which has reduced cost = −2 (not the most negative). In each of the subsequent tableaux, Gassner selects an entrant with reduced cost = −2, although some have a reduced cost = −4, which would be selected by the standard simplex method.

			Nonbasic								
	Basic	level	x_{12}	x_{14}	x_{21}	x_{23}	x_{24}	x_{31}	x_{32}	x_{41}	x_{43}
	x_{11}	1	0	0	1	0	0	1	0	1	0
	x_{22}	1	0	0	1	1	1	0	0	0	0
	x_{33}	1	−1	−1	1	1	0	1	0	1	1
	x_{44}	1	−1	0	1	1	1	0	−1	1	1
	x_{42}	0	1	0	−1	−1	−1	0	1	0	0
	x_{13}	0	1	1	−1	0	0	−1	0	−1	0
←	x_{34}	0	1	1	−1	−1	0	0	**1**	−1	−1
		38	2	4	2	0	4	−2	−2	0	−2
									↑		

			Nonbasic								
	Basic	level	x_{12}	x_{14}	x_{21}	x_{23}	x_{24}	x_{31}	x_{34}	x_{41}	x_{43}
	x_{11}	1	0	0	1	0	0	1	0	1	0
	x_{22}	1	0	0	1	1	1	0	0	0	0
	x_{33}	1	−1	−1	1	1	0	1	0	1	1
	x_{44}	1	0	1	0	0	1	0	1	0	0
←	x_{42}	0	0	−1	0	0	−1	0	−1	**1**	1
	x_{13}	0	1	1	−1	0	0	−1	0	−1	0
	x_{32}	0	1	1	−1	−1	0	0	1	−1	−1
		38	4	6	0	−2	4	−2	2	−2	−4
										↑	

			Nonbasic								
	Basic	level	x_{12}	x_{14}	x_{21}	x_{23}	x_{24}	x_{31}	x_{34}	x_{42}	x_{43}
	x_{11}	1	0	1	1	0	1	1	1	−1	−1
	x_{22}	1	0	0	1	1	1	0	0	0	0
	x_{33}	1	−1	0	1	1	1	1	1	−1	0
	x_{44}	1	0	1	0	0	1	0	1	0	0
	x_{41}	0	0	−1	0	0	−1	0	−1	1	1
←	x_{13}	0	1	0	−1	0	−1	−1	−1	1	**1**
	x_{32}	0	1	0	−1	−1	−1	0	0	1	0
		38	4	4	0	−2	2	−2	0	2	−2
											↑

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{14}	x_{21}	x_{23}	x_{24}	x_{31}	x_{34}	x_{42}
	x_{11}	1	1	1	1	0	0	0	0	0	0
	x_{22}	1	0	0	0	1	1	1	0	0	0
	x_{33}	1	−1	0	0	1	1	1	1	1	−1
	x_{44}	1	0	0	1	0	0	1	0	1	0
←	x_{41}	0	−1	−1	−1	**1**	0	0	1	0	0
	x_{43}	0	1	1	0	−1	0	−1	−1	−1	1
	x_{32}	0	1	0	0	−1	−1	−1	0	0	1
		38	6	2	4	−2	−2	0	−4	−2	4
						↑					

Standard simplex enters x_{31}

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{14}	x_{23}	x_{24}	x_{31}	x_{34}	x_{41}	x_{42}
	x_{11}	1	1	1	1	0	0	0	0	0	0
	x_{22}	1	1	1	1	1	1	−1	0	−1	0
	x_{33}	1	0	1	1	1	1	0	1	−1	−1
	x_{44}	1	0	0	1	0	1	0	1	0	0
	x_{21}	0	−1	−1	−1	0	0	1	0	1	0
	x_{43}	0	0	0	−1	0	−1	0	−1	1	1
←	x_{32}	0	0	−1	−1	−1	−1	**1**	0	1	1
		38	4	0	2	−2	0	−2	−2	2	4
								↑			

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{14}	x_{23}	x_{24}	x_{32}	x_{34}	x_{41}	x_{42}
	x_{11}	1	1	1	1	0	0	0	0	0	0
	x_{22}	1	1	0	0	0	0	1	0	0	1
	x_{33}	1	0	1	1	1	1	0	1	−1	−1
	x_{44}	1	0	0	1	0	1	0	1	0	0
	x_{21}	0	−1	0	0	1	1	−1	0	0	−1
	x_{43}	0	0	0	−1	0	−1	0	−1	1	1
←	x_{31}	0	0	−1	−1	−1	−1	1	0	1	1
		38	4	−2	0	−4	−2	2	−2	4	6
							↑				

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{14}	x_{21}	x_{23}	x_{32}	x_{34}	x_{41}	x_{42}
	x_{11}	1	1	1	1	0	0	0	0	0	0
	x_{22}	1	1	0	0	0	0	1	0	0	1
	x_{33}	1	1	1	1	−1	0	1	1	−1	0
	x_{44}	1	1	0	1	−1	−1	1	1	0	1
	x_{24}	0	−1	0	0	1	1	−1	0	0	−1
←	x_{43}	0	−1	0	−1	1	1	−1	−1	1	0
	x_{31}	0	−1	−1	−1	1	0	0	0	1	0
		38	2	−2	0	2	−2	0	−2	4	4
							↑				

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{14}	x_{21}	x_{32}	x_{34}	x_{41}	x_{42}	x_{43}
	x_{11}	1	1	1	1	0	0	0	0	0	0
	x_{22}	1	1	0	0	0	1	0	0	1	0
	x_{33}	1	1	1	1	−1	1	1	−1	0	0
	x_{44}	1	0	0	0	0	0	0	1	1	1
←	x_{24}	0	0	0	1	0	0	1	−1	−1	−1
	x_{23}	0	−1	0	−1	1	−1	−1	1	0	1
	x_{31}	0	−1	−1	−1	1	0	0	1	0	0
		38	0	−2	−2	4	−2	−4	6	4	2
					↑						

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{21}	x_{24}	x_{32}	x_{34}	x_{41}	x_{42}	x_{43}
	x_{11}	1	1	1	0	−1	0	−1	1	1	1
	x_{22}	1	1	0	0	0	1	0	0	1	0
	x_{33}	1	1	1	−1	−1	1	0	0	1	1
	x_{44}	1	0	0	0	0	0	0	1	1	1
	x_{14}	0	0	0	0	1	0	1	−1	−1	−1
	x_{23}	0	−1	0	1	1	−1	0	0	−1	0
←	x_{31}	0	−1	−1	1	1	0	1	0	−1	−1
		38	0	−2	4	2	−2	−2	4	2	0
								↑			

			Nonbasic								
	Basic	level	x_{12}	x_{13}	x_{21}	x_{24}	x_{31}	x_{32}	x_{41}	x_{42}	x_{43}
	x_{11}	1	0	0	1	0	1	0	1	0	0
	x_{22}	1	1	0	0	0	0	1	0	1	0
	x_{33}	1	1	1	−1	−1	0	1	0	1	1
	x_{44}	1	0	0	0	0	0	0	1	1	1
←	x_{14}	0	1	1	−1	0	−1	0	−1	0	0
	x_{23}	0	−1	0	1	1	0	−1	0	−1	0
	x_{34}	0	−1	−1	1	1	1	0	0	−1	−1
		38	−2	−4	6	4	2	−2	4	0	−2
			↑								

The next pivot brings us back to the initial tableau, thus completing the cycle. (Also see Gass[27, Chap. 10].)

Gassner proved that a simplex method cannot cycle for $n < 4$, so the above is an example of a smallest assignment problem for which a simplex method cycles.

Opportunity Knocks

To my knowledge, there is no example of an assignment problem that cycles with the standard simplex method. You may want to construct one or prove that no such counterexample exists.

LP Myth 22. *When applying the simplex method to minimum-cost flows on a directed, generalized network, the strongly convergent pivot rule out-performs the lexicographic rule for selecting a departing variable from the basis.*

The *strongly convergent pivot rule* was introduced by Elam, Glover, and Klingman[23] for the LP model:

$$\min\ cx:\ Ax = b,\ 0 \le x \le U,$$

where A is the node-arc incidence matrix (with weights), and x is the arc flow. Orlin[58] proves it is equivalent to the lexicographic rule (though not at all obvious). He also cites related works.

LP Myth 23. *Suppose LP is solved and π_i is the dual price associated with the i^{th} constraint. Then, the same solution is obtained when removing the constraint and subtracting $\pi_i A_{i\bullet} x$ from the objective.*

The reason this incorrect is because other solutions might exist to the revised LP. This error has caused some to say that a tax is equivalent to a prohibition in the sense that the dual price can be used as a tax in an LP that adds the tax to the objective and removes the prohibition constraint.

Counterexample. $\min\ x + 2y : 0 \le x, y \le 10,\ x + y = 1$. The solution is $(x^*, y^*) = (1, 0)$ with dual price, $\pi = 1$ for the last constraint. Then, the *tax equivalent* is:

$$\min y : 0 \le x, y \le 10.$$

The solutions are of the form $(x, 0)$, where x is arbitrary in $[0, 10]$. Using a simplex method, the solution obtained will be one of the extremes: $x = 0$ or $x = 10$, neither of which is the original solution. In fact, the basic solution $(10, 0)$ violates the original constraint.

A motivating application is the control of emissions of some pollutant. In an LP, there may be a prohibition constraint:

$$\max\ cx : x \geq 0,\ Ax = b,\ dx \leq \delta,$$

where d_j is the rate of emission caused by activity j, and δ is the limit. The tax model has the form:

$$\max\ cx - \tau dx : x \geq 0,\ Ax = b,$$

where τ is the shadow price associated with the prohibition constraint (equal to an extreme dual-variable value). Although the prohibition solution is optimal in this tax model, there may be other optimal solutions that violate the limit.

Consider a numerical example for electricity generation by three sources: scrubbed coal, oil, and uranium. The variables are fuel purchases and generation. The prohibition is a limit on sulfur emissions (LSU) while satisfying electricity demand (DEL). The B-rows balance fuels.

	Purchase			*Generate*					*Dual*
	PCL	*POL*	*PUR*	*GSC*	*GOL*	*GUR*			*Price*
COST	18	15	20	0.9	0.6	0.4	=	min	
BCL	1			−1			≥	0	18
BOL		1			−1		≥	0	15
BUR			1			−1	≥	0	20
DEL				0.3	0.3	0.4	≥	10	67.5
LSU				0.2	0.6		≤	6	−8.25
bound					25	10			
level	15	5	10	15	5	10			

The solution to this LP generates all the electricity it can from uranium, which is 4 units, and the remaining 6 units from the only combination of oil and scrubbed coal to satisfy both the demand and the sulfur limit: $GSC = 15$ and $GOL = 5$. The issue is whether the sulfur-limit constraint can be replaced by a tax on sulfur emissions.

The tax model adds 8.25 times the LSU coefficients to the objective:

$$COST + 8.25(0.2\,GSC + 0.6\,GOL).$$

The tax model and its two optimal solutions are:

	Purchase			*Generate*					*Dual*
	PCL	*POL*	*PUR*	*GSC*	*GOL*	*GUR*			*Price*
COST	18	15	20	2.55	5.55	0.4	=	min	
BCL	1			−1			≥	0	18
BOL		1			−1		≥	0	15
BUR			1			−1	≥	0	20
DEL				0.3	0.3	0.4	≥	10	67.5
bound					25	10			
level[1]	20	0	10	20	0	10			
level[2]	0	20	10	0	20	10			

The tax LP has alternative solutions with extremes that contain the original limit of 6 units of sulfur emissions. At one extreme (level^1), the company uses no oil; it generates the 6 units of remaining electricity (after nuclear generation) by scrubbed coal. This complies with the sulfur limit with slack: the amount of sulfur emitted is only 4 units. At the other extreme (level^2), the company uses no scrubbed coal. This violates the sulfur limit: the amount emitted is 12 units. (This is the solution to the original model without the sulfur limit constraint; the prohibition was specified to disallow this.)

Because the 'equivalent' tax model could result in a violation, the tax might be levied at slightly more than the dual price of \$8.25. In that case, however, the result is overly conservative, resulting in much less sulfur emission than was deemed necessary for good health while raising the cost above its minimum.

The problem is the bang-bang phenomenon with linear models: solutions respond to data changes by an all-or-nothing principle. This reflects the fact that constant rates of substitution cause trade-offs that are marginally beneficial to be globally beneficial; only a constraint can stop the negotiation.

LP Myth 24. *Let $z(t) = \min\{cx : x \geq 0,\ Ax = b + th\}$, where h is a (fixed) m-vector. Then, z is piece-wise linear, where the break-points occur wherever there must be a basis change.*

The fallacy is the last sentence. The reason that this is not correct is that not every change in basis implies the slope must change.

Counterexample. $\min x - y : x, y \geq 0,\ x - y = t$. Because $z(t) = t$ for all t, there is only one linearity interval (no breakpoints). However, for t positive, we must have x basic, and for t negative, we must have y basic. At $t = 0$ there are two optimal bases, and the basis must change as t varies in either of the two directions. Thus, although the basis must change (to be feasible), the point at which this occurs (namely, at $t = 0$) is not a breakpoint of z.

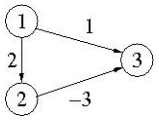
Note: the interior approach gives the correct answer (that is, the slope changes when the optimal partition changes). In the example, the optimal support has both $x > 0$ and $y > 0$, no matter what the value of t. Thus, the optimal partition does not change.

LP Myth 25. *Dijkstra's shortest path algorithm is correct, even with negative arc-costs, as long as there are no negative cycles.*

The usual counterexample to the correctness of Dijkstra's algorithm is with a negative cycle, for which there is no shortest path. What if there is no cycle?

Counterexample. Yen[78] provides the following:

Dijkstra's algorithm obtains the path $(1, 3)$, whereas the shortest path from 1 to 3 is $(1, 2, 3)$.

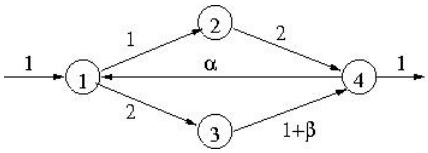


LP Myth 26. *Alternative, integer-valued optima in a shortest path problem correspond to alternative shortest paths.*

Counterexample. Consider the following network, where the LP is to ship one unit from node 1 to node 4 along the least costly route. An optimal solution is the shortest path,

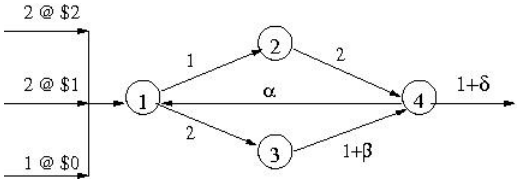
$1 \to 2 \to 4$, with a cost of \$3. There are two parameters, α, β, whose values can create alternative optima. We assume $\alpha \geq -3$ to avoid a negative cycle, and we assume $\beta \geq 0$.

If $\beta = 0$, another shortest path is $1 \to 3 \to 4$. The two shortest paths correspond to two basic optima in the LP formulation, consistent with the myth. However, when $\alpha = -3$, we have a zero-cost cycle: $1 \to 2 \to 4 \to 1$. Any solution can be augmented by an arbitrary amount of flow around the cycle without changing the total cost.



The essence of the myth rings true — there are two *simple* paths corresponding to two basic optima. However, the alternative optima with positive flow around the cycle spoils the result being literally true. One must consider zero-cost cycles as a caveat in how the statement is worded. The issue runs deeper in separating true alternative optima from *frivolous* ones. In particular, the dual always has alternative optima of the form $\pi' = \pi + K$, where π is any dual solution and $K > 0$. This is frivolous because they do not convey any true alternatives in the underlying economics.

To illustrate the difference between true versus frivolous alternative dual optima, consider a 3-tier supply, shown on the right. The dual price at node 4 depends on the demand parameter $\delta \geq 0$.



For $\delta = 0$, the initial supply step can be basic, giving a basic dual price of $\pi_4 = 3$ (and $\pi_1 = 0$). Another basic optimum has the initial supply step out of the basis at its upper bound of one unit, and the second supply step is in the basis (at zero level), giving $\pi_1 = 1$. The price at node 4 then becomes $\pi_4 = 4$. We have another interval of optimal prices at $\delta = 2$. Optimal dual prices are never unique, but when $\delta \neq 0, 2, 4$, alternatives are frivolous in that we could simply add any constant to all of them to obtain an alternative optimum. That notion of "alternative" does not correspond to a real alternative; it is an artifact of the modeling.

To summarize, we have the following cases (for $\alpha \geq -3$, $\beta \geq 0$, $\delta \geq 0$):

	Primal	Dual
unique	$\alpha > -3$, $\beta > 0$	never
frivolous	$\alpha = -3$, $\beta > 0$	$\delta \neq 0, 2, 4$
true alternatives	$\alpha > -3$, $\beta = 0$	$\delta = 0, 2, 4$

Opportunity Knocks

The distinction between true and frivolous alternative optima can be difficult to represent precisely. There is practical benefit to doing so. Besides ruling out some solutions as frivolous, one may want to know some *generating set* that brings an exponential number of alternatives down to a linear number in terms of more basic dimensions. For example, suppose an m-regional model has two alternatives within each region (but distributions among regions are completely determined by specifying one of the 2^m alternative optima). The total number of alternative optima is 2^m, but I suggest that there are circumstances where the distributions

associated with combinations are not of much interest compared to knowing each of the $2m$ alternatives. Syntactically, a modeling language could allow some notion of *blocks* or *submodels* that make this practical. Also see IP Myth 16.

LP Myth 27. *In a standard assignment problem, it is always optimal to assign the person-to-job that has the least cost.*

If this were true, we would have a greedy algorithm that recursively assigns the pair of least cost among unassigned pairs. As illustrated with the following counterexample, the optimality of an assignment depends upon *relative* costs. The one with least cost may eliminate an alternative savings that is greater when considering second-least costs.

Counterexample.

1	2
10	15

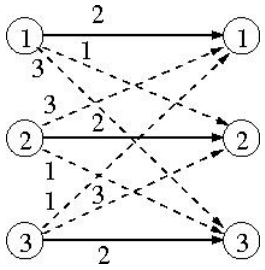
This is a 2×2 problem, and the issue is whether to assign Person 1 to Job 1 since that is the least cost.

If we assign Person 1 to Job 1, that cost is only 1, but we must then assign Person 2 to Job 2. That yields a total cost of 16. The optimal assignment is to assign Person 1 to Job 2 and Person 2 to Job 1, for a total cost of 12.

LP Myth 28. *Given an assignment problem with a non-optimal (but feasible) assignment, its cost can be reduced by swapping some pair of assignments.*

The following counterexample is adapted from Bertsekas[11].

Counterexample. There are 3 people to be assigned to 3 jobs. The current assignment is shown below with the solid arcs, having total cost = 6.



Numbers next to arcs are costs.

Here are the possible pair-wise swaps:

Old	New	Δcost
{1-1, 2-2}	{1-2, 2-1}	0
{1-1, 3-3}	{1-3, 3-1}	0
{2-2, 3-3}	{2-3, 3-2}	0

Every pair of swaps leaves the cost unchanged, but an optimal assignment is {1-2, 2-3, 3-1}, having total cost = 3.

LP Myth 29. *A transportation problem with unique shipping costs has a uniquely optimal shipment.*

Counterexample. Rubin and Wagner[66] pointed this out after noticing that managers apply this myth in practice. They provided the following:

	Supplier 1	Supplier 2	Demand
Market 1	55 0 10	5 10 0	10
Market 2	65 5 5	15 10 10	15
Market 3	75 10 0	25 0 10	10
Supply	20	20	

The upper number in each cell is the unit shipping cost. For example, each unit shipped from Supplier 1 to Market 1 is \$55. The lower-left number is the shipment in one optimal solution, and the lower-right number is the shipment in another optimal solution.

Note that the unit costs are all different, yet there are alternative optimal shipments. (The minimum total cost is \$1,275.)

LP Myth 30. *The optimal dual price of a demand constraint equals the increase in the minimum total cost if that demand is increased one unit.*

This fails if the solution is not at a *compatible basis*[37] (in the presence of primal degeneracy).

Counterexample. The following is taken from Rubin and Wagner[66].

	Supplier 1	Supplier 2	Demand	Price
Market 1	55 10	10 $0^{\ddagger}$	10	$55^{\dagger}$,$55^{\ddagger}$
Market 2	65 $0^{\dagger}$	15 10	10	$65^{\dagger}$,$60^{\ddagger}$
Market 3	80 0	25 10	10	$75^{\dagger}$,$70^{\ddagger}$
Supply	20	20		†Basis 1
Price	0, 0	50, 45		‡Basis 2

The cell values are unit costs and the (unique) optimal shipment levels. Two (basic) dual prices are shown.

If Market 2 demand increases, the first basis is compatible, and the change in the minimum total cost is indeed \$65. This can be achieved by sending one unit from Supplier 1 (which has excess). The basis is compatible with this change because the shipment level, x_{12}, can increase from its degenerate basic value, 0. On the other hand, if the solver obtains Basis 2, the \$60 dual price understates the increase in minimum total cost.

However, if we want to know the rate of savings from decreasing the demand in Market 2, we obtain the minimum optimal dual price (among the alternative optima) of the demand constraint. It is given by Basis 2 by letting the basic shipment level, x_{21}, increase by 1, balanced by decreasing x_{11} and x_{22} to 9.

The importance of using the wrong dual price for a marginal demand change is that the computed change in the minimum total cost may not be correct. One must have the maximum dual price to compute the effect of a demand increase, and one must have the minimum dual price to compute the effect of a demand decrease. (More details are in [35].)

For non-network LPs the myth can fail by having the correct slope (that is, $\partial f^*(b)/\partial b_i = \pi_i$), but the slope changes at $\Delta b_i < 1$, so the effect of a full unit change cannot be measured precisely with the shadow price.

LP Myth 31. *An increase in a demand requirement (with concomitant increase in supply) increases the minimum total cost.*

This is called the "more-for-less paradox." The following transportation problem is from Charnes and Klingman[21] (also see [74]).

Counterexample. There are 3 suppliers, with supplies shown in the last column, and 4 destinations, with demands shown in the last row. The cell values are optimal flows (blank is zero) and the boxed cell values in the NW corner are costs. The modified problem is to increase demand 1 and supply 2 by 9 units. The new optimal flow is shown on the right, and the total cost has decreased from \$152 to \$143, despite the increase in total flow, from 55 to 64.

[1] 11	[6]	[3] 7	[5] 2	20
[7]	[3]	[1] 10	[6]	10
[9]	[4] 13	[5]	[4] 12	25
11	13	17	14	55

Original Problem
Min Cost = \$152

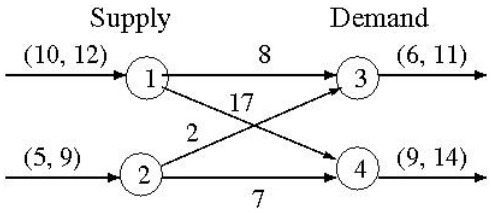
[1] 20	[6]	[3]	[5]	20
[7]	[3] 2	[1] 17	[6]	19
[9]	[4] 11	[5]	[4] 14	25
20	13	17	14	64

Modified Problem
Min Cost = \$143

The underlying economics is that the greater flow can take advantage of low-cost activities. In the transportation example, shipments from supplier 1 to destination 1 have the lowest cost, but the original demand is not enough to ship all of the availability supply; supplier 1 must ship to other destinations. In the revised problem, supplier 1 can ship all of its units to destination 1, and the other destinations can meet their requirements from other suppliers less expensively.

Dĭneko, B. Klinz, and G. J. Woeginger[22] provide the following 3×3 transportation problem: supply: $s = (0, 1, 1)$, demand: $d = (1, 1, 0)$, and cost: $c_{ij} = 2^{|i-j|}$. The minimum total cost is 4. Increasing the first supply and last demand to $s' = d' = (1, 1, 1)$, the minimum total cost is only 3. They proceed to develop a key condition under which this paradox cannot occur: *there does not exist* i, j, p, q *such that* $c_{ij} + c_{pq} < c_{iq}$. If this condition does not hold, the more-for-less paradox may apply, depending on the data.

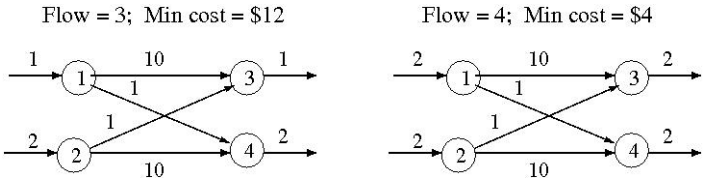
Glover[20, p. 37] gives another example:



The supplies and demands are required ranges, and the arc numbers are unit flow costs.

The minimum feasible flow is 15 units, and the least costly way to send that minimum is $x_{13} = 6$, $x_{14} = 4$, and $x_{24} = 5$, for a total cost of \$151. However, we can ship $x_{13} = 10$ and $x_{24} = 9$, for a total cost of \$143. We thus ship more for less!

Michael Hennebry provided the small example shown at the right.



Another form of the more-for-less paradox also arises with modeling requirement constraints as equations, rather than with inequalities. The problem need not be a network.

Counterexample. The following is a diet problem with 3 foods and 2 nutrient requirements, given by Arsham[5, 2]:

$$\begin{gathered} \min 40x_1 + 100x_2 + 150x_3 : \\ x_1 + 2x_2 + 2x_3 = 10 \\ 3x_1 + x_2 + 2x_3 = 20 \\ x_1, x_2, x_3 \geq 0. \end{gathered}$$

The optimal diet is $x = (6, 2, 0)$ with a minimum total cost of \$440. If we increase the second nutrient requirement to 30, the optimal diet becomes $x = (10, 0, 0)$ with a minimum total cost of \$400.

The diet problem usually has the canonical form:

$$\min\ cx :\ Ax \geq b,\ x \geq 0$$

(perhaps with bounds on the levels of foods, as $L \leq x \leq U$). To require $Ax = b$ does not give the flexibility of allowing over-satisfaction of nutrient requirements, even though it could be quite healthy to do so. This principle carries over to other situations, where modeling with equations is not the appropriate representation. (Also see Charnes, Duffuaa, and Ryan[18].)

Arsham[4] provides another vantage, with some focus on production problems. Ryan[67] addresses economies of scale and scope, using goal programming for multiple market structures.

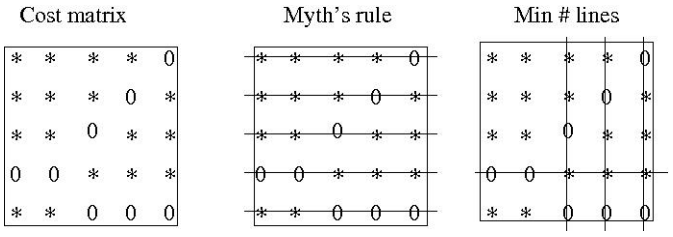
Opportunity Knocks

Does the more-for-less paradox extend to generalized networks? What about nonlinear costs?

LP Myth 32. *The line-drawing step of the Hungarian method for the assignment problem can be replaced by:* cover as many zeroes as possible with each line.

There have been several variants of the Hungarian algorithm — see Kuhn[52]. The original Hungarian method is to cover the zeroes with a minimum number of lines. This myth suggests another criterion, which turns out not to guarantee an optimal solution.

Counterexample. Storøy and Sørevik[77] provide the following 5×5 (* denotes non-zero):



The line-drawing rule starts by covering the three zeroes in row 5, followed by covering the two zeroes in row 4. Thus, a total of five lines must be drawn to cover all zeroes. Since this equals the number of rows (and columns), the Hungarian method's next step is to create an optimal solution from the covered zeroes. This is not possible.

The minimum number of lines is four, and the Hungarian method continues to subtract the minimum uncovered element (adding it to those covered by two lines).

LP Myth 33. *The Stepping Stone Method always produces an optimal distribution.*

This clever, early algorithm by Charnes and Cooper[17] specifically requires equality constraints (with total supply equal to total demand). It was extended to the general *node-bounded problem* by Charnes and Klingman[20]:

$$\begin{aligned} &\min \textstyle\sum_{i,j} c_{ij}x_{ij} : x \geq 0 \\ &\underline{s}_i \leq \textstyle\sum_j x_{ij} \leq \overline{s}_i, \ \forall i \\ &\underline{d}_j \leq \textstyle\sum_i x_{ij} \geq \overline{d}_j, \ \forall j, \end{aligned}$$

where $0 \leq \underline{s} \leq \overline{s}$ (supply out-flow bounds) and $0 \leq \underline{d} \leq \overline{d}$ (demand in-flow bounds).

Charnes, Glover, and Klingman[19] illustrated that the Stepping Stone Method need not terminate with an optimal solution if the constraints are the following special case of the node-bounded problem:

$$\min \textstyle\sum_{i,j} c_{ij}x_{ij} : x \geq 0, \ \sum_j x_{ij} \geq a_i, \ \sum_i x_{ij} \geq b_j.$$

Counterexample. Charnes, Glover, and Klingman gave a counterexample for each case:

$\sum_i a_i = \sum_j b_j$

1	6	3	5	20
7	3	1	6	10
8	3	4	3	25
11	13	17	14	

$\sum_i a_i < \sum_j b_j$

2	4	3
1	1	1
2	5	1
3	4	

$\sum_i a_i > \sum_j b_j$

1	1	2	5
6	5	1	6
2	7	1	

Each table gives the data in the form:

c_{11}	$\cdots$	c_{1n}	a_1
$\vdots$		$\vdots$	$\vdots$
c_{m1}	$\cdots$	c_{mn}	a_m
b_1	$\cdots$	b_m	

The solutions given by the Stepping Stone Method are the associated x_{ij}:

$\sum_{i,j} c_{ij}x_{ij} = 127$

11	0	9	0
0	2	8	0
0	11	0	14

$\sum_{i,j} c_{ij}x_{ij} = 13$

2	1
0	3
1	0

$\sum_{i,j} c_{ij}x_{ij} = 27$

2	3	0
0	4	2

Here are feasible solutions with lower costs:

$\sum_{i,j} c_{ij}x_{ij} = 118$

20	0	0	0
0	2	17	0
0	11	0	14

$\sum_{i,j} c_{ij}x_{ij} = 12$

3	0
0	4
1	0

$\sum_{i,j} c_{ij}x_{ij} = 15$

2	7	0
0	0	6

LP Myth 34. *The standard free-float formula for an activity in an activity-on-arc network equals the maximum leeway for scheduling the activity without affecting any the earliest start time of any later activity.*

The standard formula for the free float (FF) activity (i, j) is:

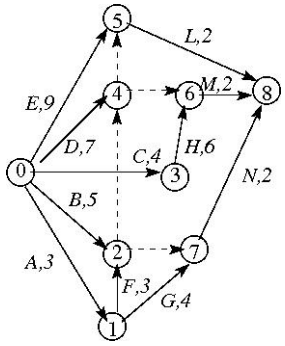
$$FF_{ij} = ES_j - EC_i \tag{LP.3}$$

where ES = earliest start time, EC = earliest completion time.

The statement is true in the absence of dummy arcs, but it can be an underestimate when all successors of some activity in the activity-on-arc network are dummy arcs.

Counterexample. Zhao and Tseng[81] provide the following (numbers on arcs are activity durations):

Activity	(i,j)	incorrect FF_{ij}	correct FF_{ij}
B	$(0,2)$	1	2
D	$(0,4)$	0	2
F	$(1,2)$	0	1



The incorrect values are from (LP.3). For example, $FF_{02} = ES_2 - (ES_0 + 5) = 6 - (0 + 5) = 1$. The maximum leeway, however, is 2. If we delay starting activity B by 2 time units, that will delay reaching node 2 by 2 time units. But since all arcs out of node 2 are dummy arcs, no activity is immediately affected. Instead, the float limit of 2 comes from tracing the paths out of node 2. Path $2 \to 7 \to 8$ gives a limit of 2 time units — that is, increasing the start of activity B by t delays the start of activity N by $t - 2$ for $t \geq 2$. Similarly, the path $2 \to 4 \to 5 \to 8$ reveals that the start of activity L will be delayed by $t - 9$, and the path $2 \to 4 \to 6 \to 8$ reveals that the start of activity M will be delayed by $t - 10$. The binding limit is from the first path, which yields the correct float value of 2.

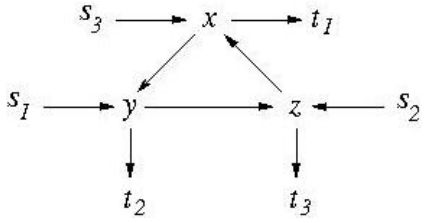
Similarly, applying (LP.3) to arc $(0, 4)$, we have the incorrect value: $FF_{04} = ES_4 - (ES_0 + 7) = 7 - (0 + 7) = 0$. The correct value is obtained by tracing the paths $4 \to 5 \to 8$ and $4 \to 6 \to 8$. The former path yields a float limit of 2 time units (since activity L earliest start time $= ES_5 = 9$); the latter yields a float limit of 3 time unit (since activity M earliest start time $= ES_6 = 10$). The least of these limits is 2, which is the correct float value.

Zhao and Tseng developed this into an algorithm that follows dummy arcs from a rooted tree to obtain the correct free float values.

LP Myth 35. *The maximum flow of commodities through a network equals the capacity of a minimum disconnecting set.*

This is correct when there is only one commodity and for special cases of more than one. The failure for general numbers of commodities on networks of arbitrary topology was recognized in the 1950's — see Zullo[83] and her bibliography through 1995. The following example is from Ford and Fulkerson[25], and is further discussed by Bellmore, Greenberg, and Jarvis[8].

Counterexample. In the following network, all capacities are 3.



The max-flow is to send $^3/_2$ units along each path from its source to its sink, for a total of $^9/_2$ units. Here are the (unique) paths for each commodity: $s_1 \to y \to z \to x \to t_1$; $s_2 \to z \to x \to y \to t_2$; $s_3 \to x \to y \to z \to t_3$.

The minimum disconnecting is just to break the cycle, say with arc $(x,\ y)$, and the supply arc for the one remaining commodity, which is $(s_1,\ y)$, for a total of 6 units of capacity. There is no 1-arc disconnecting set, so this is a minimum, which implies max-flow < min-cut.

LP Myth 36. **new** *A maximum dynamic-flow on a network defined by a static network with stationary data is temporally-repetitive.* **next new** ▷

The maximum dynamic-flow problem is to find the maximum total flow that reaches the sink(s) within a specified number of time periods, N. The time-expanded network is defined by the given, static network, $G = [V, A]$, with specified sources, $S = \{s_1, \ldots, s_m\} \subset V$, and destinations, $D = \{d_1, \ldots, d_m\} \subset V$. The data are capacities, c_a, and traversal time, τ_a, for each $a \in A$. For each $v \in V$ define $N+1$ nodes, $\{v(t)\}_{t=0}^{N}$. For each arc, $a \in A$, with endpoints (u, v), define the arcs $\{a(t)\}_{t=0}^{N-\tau_a}$ with (time-independent) data (c_a, τ_a) and endpoints $(u(t), v(t+\tau_a))$.

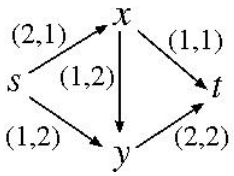
A flow is defined over a set of simple paths, each being from a source to a sink. Let the j^{th} path be $P_j = (a_{i_1}, \ldots, a_{i_{L_j}})$, where the tail of a_{i_1} is in S and the head of $a_{i_{L_j}}$ is in D. The path's total travel time is

$$\sigma_j = \sum_{k=1}^{L_j} \tau_{a_{i_k}}.$$

Let $f_j(t)$ be the flow along path j, starting at time t (restricted to j such that $\sigma_j \leq N$), and let $\{P_j\}_{j=1}^{N_p}$ be the set of paths in G. To satisfy capacity constraints, we must sum flow across each arc at each time period. Let β_{aj} be the time that flow reaches arc $a \in P_j$ along path j, starting at time 0. Then, $\beta_{aj} + t$ is the time it reaches a for flow $f_j(t)$.

For example, the network on the right has one commodity. Dropping the commodity subscripts, there are three paths:

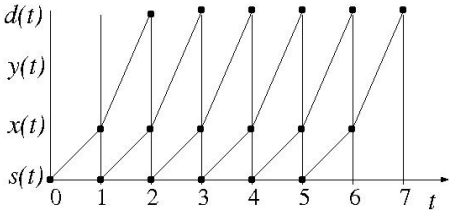
$$\begin{array}{ll} P_1 = (s \to x \to d), & \sigma_1 = 2; \\ P_2 = (s \to y \to d), & \sigma_2 = 4; \\ P_3 = (s \to x \to y \to d), & \sigma_3 = 5. \end{array}$$



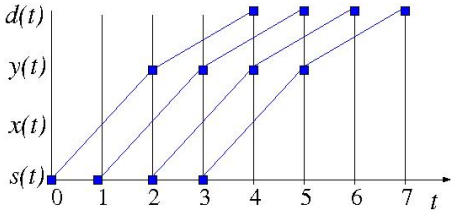
Arc values (c, τ) equal the capacity and traversal time.

The time-expanded network for $N = 7$ has 13 paths:

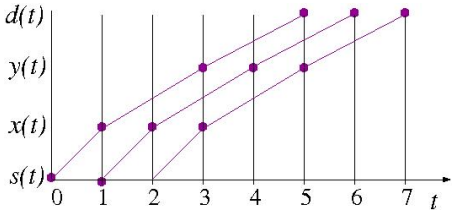
$$\begin{array}{lll} P_1(t) = (s(t) \to x(t+1) \to d(t+2)), & \text{for t=0,...,5} \\ P_2(t) = (s(t) \to y(t+2) \to d(t+4)), & \text{for t=0,...,3} \\ P_3(t) = (s(t) \to x(t+1) \to y(t+3) \to d(t+5)), & \text{for t=0,...,2} \end{array}$$



Expansion of P_1



Expansion of P_2



Expansion of P_3

The arc-chain formulation of the *multi-commodity maximum dynamic-flow problem* is thus:

$$\max \sum_{j=1}^{N_p} \sum_{t=0}^{N-\sigma_j} f_j(t) : \; x \geq 0,$$

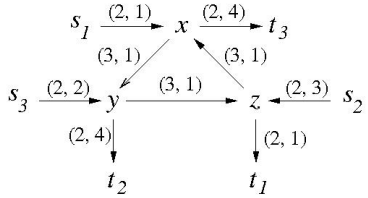
$$\sum_{j:\, a \in P_j} f_j(t - \beta_{aj}) \leq c_a, \qquad t = \underline{t}_a, \ldots, \bar{t}_a, \; a \in A,$$

where the time range is $\underline{t}_a = \max_j \beta_{aj}$, $\bar{t}_a = N + \min_j \{\beta_{aj} - \sigma_j\}$.

Let f^* be a maximum (static) flow in G. A solution is *temporally repetitive* if $f_j(t) = f_j^*$ for $t = 0, 1, \ldots, N - \sigma_j$ and $f_j(t) = 0$ otherwise. The myth's statement asserts that there is a maximum dynamic-flow solution that is temporally repetitive. Ford and Fulkerson[25] proved that the myth's statement is true for one commodity, but Bellmore and Vemuganti[9] provide the following

Counterexample. Consider three commodities ($m = 3$) and $N = 20$.

Path (P_j)	σ_j
$s_1 \to x \to y \to z \to d_1$	4
$s_2 \to z \to x \to y \to d_2$	9
$s_3 \to y \to z \to x \to d_3$	8



There are 42 paths in the dynamic network:

$$P_1(t) = s_1(t) \to x(t+1) \to y(t+2) \to z(t+3) \to d_1(t+4), \quad \text{for } t = 0, \ldots, 16$$
$$P_2(t) = s_2(t) \to z(t+3) \to x(t+4) \to y(t+5) \to d_2(t+9), \quad \text{for } t = 0, \ldots, 11$$
$$P_3(t) = s_3(t) \to y(t+2) \to z(t+3) \to x(t+4) \to d_3(t+8), \quad \text{for } t = 0, \ldots, 12$$

The path-arc arrival times, β (shown on right), determine the potentially binding capacity constraints from the inner arcs:

$$
\begin{array}{ll}
(x,y): & f_1(t-1)+f_2(t-4)\le 3 \quad \text{for } t=4,\ldots,15\\
(y,z): & f_1(t-2)+f_3(t-2)\le 3 \quad \text{for } t=2,\ldots,14\\
(z,x): & f_2(t-3)+f_3(t-3)\le 3 \quad \text{for } t=3,\ldots,14
\end{array}
$$

	arc		
j	(x,y)	(y,z)	(z,x)
1	1	2	na
2	4	na	3
3	na	2	3

The maximum dynamic-flow solution is:

	path		
t	1	2	3
0	2	2	2
1	1	1	1
2	2	2	2
3	1	1	1
4	2	2	2
5	1	1	1

	path		
t	1	2	3
6	2	1	2
7	2	1	1
8	2	2	1
9	1	2	1
10	1	2	2
11	1	1	2

	path			
t	1	2	3	
12	2	0	2	
13	2	0	0	
14	2	0	0	
15	2	0	0	
16	2	0	0	
$\ge$17	0	0	0	
total	28	18	20	66

The total flow is 66. The maximum temporally-repeated flow is:

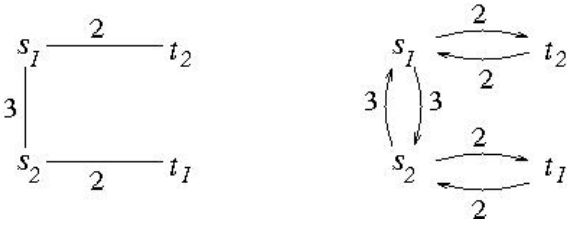
$$
\begin{array}{rcll}
f_1(t) & = & 2 & \text{for } t=0,\ldots,16\\
f_2(t) & = & 1 & \text{for } t=0,\ldots,11\\
f_3(t) & = & 1 & \text{for } t=0,\ldots,12\\
f_j(t) & = & 0 & \text{otherwise.}
\end{array}
$$

The value of this flow is 59. (Bellmore and Vemuganti give the maximum temporally-repeated flow value as 63, but I cannot see it.)

LP Myth 37. *Undirected arcs can be replaced by a pair of oppositely oriented arcs, and there is no loss in generality in obtaining a max-flow or a min-cut.*

This is true for a single-commodity network[25], but it generally fails for multi-commodity networks. The following is given by Bellmore, Greenberg, and Jarvis[8].

Counterexample. In the following network, capacities are shown next to each edge.



In the undirected graph, the max-flow is only 3, sending $^3/_2$ units of each commodity (the min-cut is also 3). After the replacement of each edge with opposite arcs, the max-flow becomes 4 units (also the min-cut value).

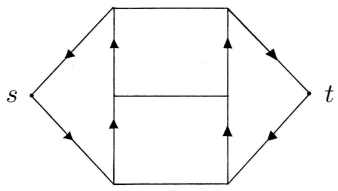
(Note: for a single commodity there is no advantage to sending flow across both arcs since they would cancel out in computing the total flow.)

LP Myth 38. *The maximum two-way flow of a commodity through a network equals its min-cut.*

In this variation of capacitated network flow, some links may be directed (arcs) and some may be undirected (edges). The flow on edges may be in either direction. Two-way flow from node s to node t, denoted $s \leftrightarrow t$, means two paths, one from s to t, denoted $s \to t$, and one from t to s, denoted $t \to s$. A two-way flow is a pair of paths, one in each direction, and the value of the flow is the minimum of all capacities of the links in the paths. A two-way cut for (s, t) is a set of links whose removal removes all paths in both directions, $s \to t$ and $t \to s$.

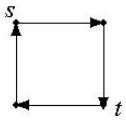
Rothschild and Whinston[63] provide the following:

Counterexample. In the following network (taken from [63]), all capacities are one. We have: two-way max-flow = 1 < two-way min-cut = 2.



Counterexample. T.C. Hu sent me the following:

All capacities are 1, so the two-way max flow = 1, and the two-way min cut = 2.



LP Background — Gomory-Hu Cut Trees

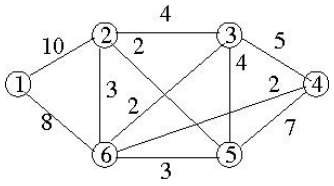
Consider an undirected graph with distinguished nodes s, t. Each edge e has a capacity, c_e, so there is a maximum flow from s to t, which equals the minimum cut that disconnects s from t. The *multi-terminal max-flow/min-cut problem* is to find the max-flow/min-cut between each s, t in the graph. This could be done by solving each of the $\binom{n}{2}$ min-cut problems, but the Gomory-Hu algorithm[33] does this with only $n - 1$ min-cut solutions.

Let V_{st} denote the max-flow/min-cut value between s and t. The Gomory-Hu algorithm produces a *cut-tree* (sometimes called a *Gomory-Hu tree*), whose nodes are those of the original graph and whose edges satisfy:

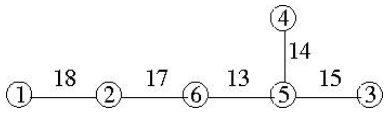
$$V_{st} = \min_{(i,j) \in P_{st}} V_{ij}, \tag{LP.4}$$

where P_{st} = edges in s-t path. The Gomory-Hu algorithm computes the $n-1$ cuts, from which (LP.4) yields all of the $\binom{n}{2}$ min-cut values in the original graph.

Example (taken from [33]):



Capacitated network



Gomory-Hu cut-tree

For example, $V_{14} = 13 = \min\{18, 17, 13, 14\}$. The cut set is $\{(2,3), (2,5)\,(6,3), (6,4), (6,5)\}$, with graph partition $= \{1, 2, 6 \,|\, 3, 4, 5\}$.

A cut-tree has two key properties:

1. Each max-flow/min-cut value in the original graph equals the minimum of the edge values along the unique path connecting them in the cut-tree (that is, equation (LP.4)).
2. Removal of any edge from the cut-tree partitions the original graph into two sets of nodes that comprise a cut set whose value equals the cut-tree edge value.

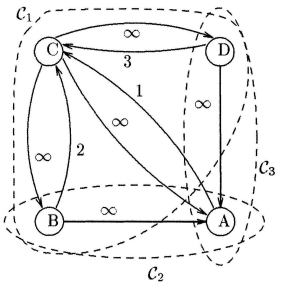
The first property gives the correct value of the min-cut, and hence the max-flow, and the second property gives the actual cut-set for any pair of nodes.

LP Myth 39. *Every connected network has a cut-tree.*

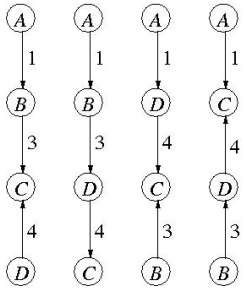
The classical algorithm by Gomory and Hu[33] constructively establishes the existence of a cut-tree for every connected, ***undirected*** graph. This was allegedly extended to directed graphs for the *symmetric case*: the min-cut between two nodes is the lesser of the min-cut from one to the other:

$$\mathcal{V}_{st} = \min\{V_{st}, V_{ts}\}$$

Counterexample. Benczúr[10] provides the following:



Only 3 (of 7) cut-sets are min



Flow trees do not encode min cut-set

Here are the min-cut values:

$$V = \begin{bmatrix} 0 & 1 & 1 & 1 \\ \infty & 0 & 3 & 3 \\ \infty & \infty & 0 & \infty \\ \infty & \infty & 4 & 0 \end{bmatrix} \Rightarrow \mathcal{V} = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 3 & 3 \\ 1 & 3 & 0 & 4 \\ 1 & 3 & 4 & 0 \end{bmatrix}.$$

Since the min-cut value of A is 1 and all other min-cut values are greater than 1, any cut-tree must have A as a leaf. That leaves 9 trees to consider. Of these, 4 are shown with the edge values equal to the associated min-cut values: $V(\mathcal{C}_1 = (A \mid B, C, D)) = 1$, $V(\mathcal{C}_2 = (A, B \mid C, D)) = 3$, and $V(\mathcal{C}_3 = (A, D \mid B, C)) = 4$. Each tree violates the second property to be a cut-tree: the cut-set obtained upon breaking an edge of minimum value in the path between two nodes is not their min-cut.

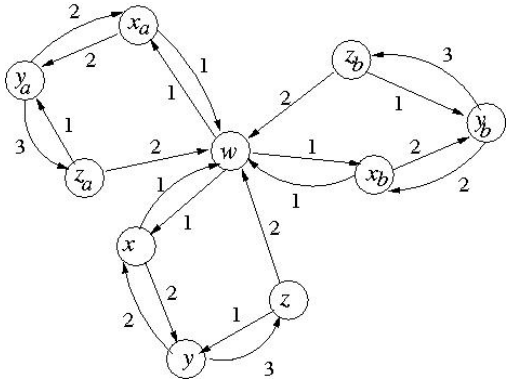
Going from left-to-right, the first two trees' violation is with (D, C). The cut-sets obtained from the edge is $(D \mid C, B, A)$ and $(C \mid D, B, A)$, respectively, but the min cut-set between D and C is $\mathcal{C}_3$. The third tree's violation is with (B, C). The cut-set obtained from the edge is $(B \mid C, D, A)$, but the min cut-set between B and C is $\mathcal{C}_2$. The fourth tree's violation is with (B, D). The cut-set obtained from the edge is $(B \mid D, C, A)$, but the min-cut is $\mathcal{C}_2$.

Now consider the other possible trees. Separating C and D makes their path value 3, which is not the value of their min-cut. The four shown are the only ones satisfying the first property of a cut-tree, showing the correct values of the min-cut using equation (LP.4). Since min-cut = max-flow, these are called *flow trees*.

Rizzi[62] provides the following with additional insight.

Counterexample.

In any tree there must be a leaf. Any cut-tree for this network must therefore have a star cut, $(v \mid \{u \neq v\})$. Suppose z is a leaf and its neighbor is y. The edge value of (z, y) is the star cut value $V(z \mid x, y, w, x_a, \dots) = 3$. If it were a cut-tree, this partition must be the min-cut between z and y. This is not the case, as the min-cut between z and y is $V(z, x \mid y, w, x_a, \dots) = 2$.

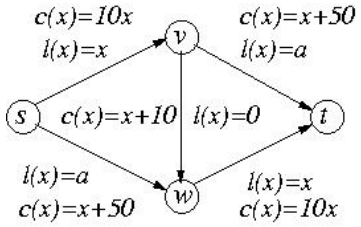


The key property identified by Rizzi is the notion of a *good pair*: (s, t) such that the star cut at t is a min-cut of (s, t), or there is no min-cut of (s, t) (that is, no path $s \to t$ or $t \to s$). Rizzi's network has no good pair, and that is why a cut-tree does not exist.

LP Myth 40. *Removing an arc in a network cannot decrease users' latency or cost.*

This is *Braess' Paradox*[14] applied to traffic flow.

Counterexample. The following is the classical example[15] — also see http://supernet.som.umass.edu/facts/braess.html.



$\ell(x)$ is the latency function of flow, x;
$c(x)$ is the cost.

The equilibrium flow is determined by each driver using the min-latency path. For n users, such that $n < a$, this is $s \to v \to w \to t$. (The users are indifferent among the three paths if $n = a$.) This results in each user experiencing $2n$ units of latency. If we remove arc $(v,\ w)$, the drivers evenly split the use of the two different paths: $s \to v \to t$ and $s \to w \to t$. Their latencies thus reduce to $\frac{1}{2}n + a$ each.

Using the same graph, Steinberg and Zangwill[73] provide the rest of the counterexample, using the cost functions shown. With arc $(v,\ w)$, 6 users evenly split each of the three paths from s to t, so that $x_{sv} = x_{wt} = 4$, while the other arc flows are 2. Thus, each user pays \$92, and the system cost is \$552. Without arc $(v,\ w)$, 6 users split evenly between the two paths. Thus, each user pays \$83, and the system cost is \$498.

A great deal of literature has developed since Braess introduced his paradox in 1968. It has become a cornerstone of traffic equilibrium, as reflected in modern books by Nagurney[54, 57, 55] and Roughgarden[64]. Also see Nagurney[56] and Roughgarden[65] for focus on the Braess paradox and its relatives.

LP Myth 41. *Given strict improvement in the objective value, the standard simplex method does not visit an exponential number of vertices of the feasible polyhedron.*

The falsity of this was first demonstrated by Klee and Minty[51]. The so-called *Klee-Minty polytope* causes the standard simplex method to visit every extreme point, which grows exponentially with the number of variables.

Counterexample. The LP has n variables, n constraints, and 2^n extreme points. The elementary simplex method, starting at $x = 0$, goes through each of the extreme points before reaching the optimum solution at $(0, 0, \ldots, 0, 5^n)$.

$$\begin{array}{rcrcccrcrcr}
\max & 2^{n-1}x_1 & + & 2^{n-2}x_2 & + & \ldots & + & 2x_{n-1} & + & x_n: & \\
& x_1 & & & & & & & & \leq & 5 \\
& 4x_1 & + & x_2 & & & & & & \leq & 25 \\
& 8x_1 & + & 4x_2 & + & x_3 & & & & \leq & 125 \\
& \vdots & & & & & & & & & \vdots \\
& 2^n x_1 & + & 2^{n-1}x_2 & + & \ldots & + & 4x_{n-1} & + & x_n \leq & 5^n
\end{array}$$

$$x \geq 0.$$

Another interesting example of exponential growth is due to Blair[12].

Jeroslow[47] was the first to present the construction of a class of examples for the best-gain basis entrance rule to visit an exponential number of vertices. (Also see Blair[12].)

LP Myth 42. *The worst-case time complexity of the simplex method is exponential and hence worse than the worst-case time complexity of the interior-point method.*

There are several things wrong with this statement. The first thing to note is that there is no "***the*** simplex method" and there is no "***the*** interior-point method." We know that both the standard simplex method and the best-gain rule have exponential time complexity (see LP Myth 41). However, the *Hirsch Conjecture*[82] leaves open the prospect for some simplex

method to be linear in the numbers of variables and constraints. Also, there are interior-point methods that behave better than Karmarkar's original[49] in practice, but have no proof of polynomial complexity.

The second thing to note is the perturbation analysis by Spielman and Teng[72]. In fact, many coefficients are subjected to "random" perturbation due to rounding in their computations from other data.

Now suppose we are talking about the standard simplex method and one of the interior-point methods with a proof of polynomial complexity in the length of the data. Then, the third thing to consider is that the length of the data could be an exponential function of the number of variables. One example of this is a *Linear Programming Relaxation* (LPR) whose coefficients are computed from an aggregation algorithm[31]. The length of the coefficients (number of digits) can be an exponential function of the numbers of variables and constraints.

Thus, one must be careful in how to compare the (theoretical) worst-case time complexities of simplex versus interior methods.

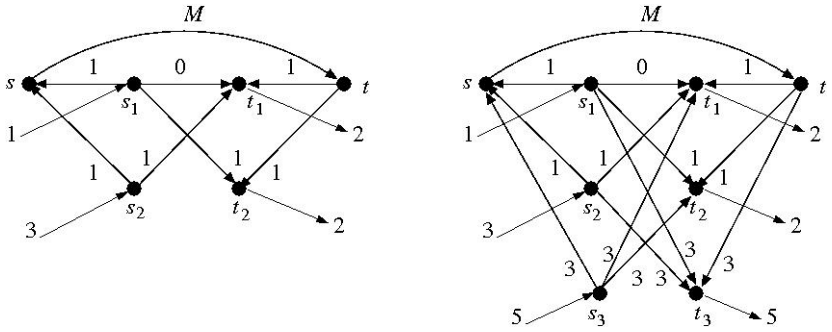
LP Myth 43. *The standard simplex method is polynomial time for min-cost network flow problems.*

Counterexample. Zadeh[79] provides the following. Consider a network, denoted N_n, with $2n+2$ nodes such that n nodes are sources, n are sinks, one node is a super-source, and one is a super-sink; arcs are (s,t), (s_1,t_1), (s_i,s), (t,t_i), $\{(s_i,t_j) : j \neq i\}$ for $i = 1,\ldots,n$. The $2n^2+2$ arc capacities are infinite, but supplies and demands are forced flow values for each (s_i,s) and (t,t_i) from the external supplies (S) and demands (D) for $n > 1$:

$$\begin{array}{llll} S_1 & = 1 & D_1 & = 2 \\ S_2 & = 3 & D_2 & = 2 \\ S_i = D_i & = 2^{i-1} + 2^{i-3} & \text{for } i = 3,\ldots,n & \end{array}$$

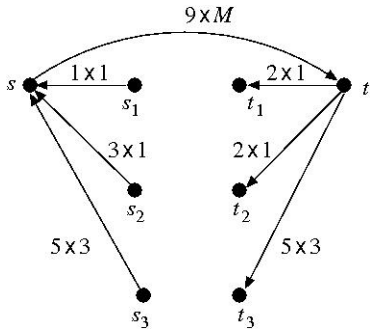
The arc costs are $c(s,t) = M$, $c(s_1,t_1) = 0$, $c(s_i,s) = c(t,t_i) = c(s_i,t_j) = 2^{i-1} - 1$ for $j < i$ and $c(s_i,t_j) = 2^i - 1$ for $j \geq i$ for $i = 1,\ldots,n$. The value of M is sufficiently large to render the use of arc (s,t) prohibitive (that is, $x(s,t)$ is its minimum feasible value in every optimal solution).

Here are N_2 and N_3 — supplies are tailless arcs into s_i; demands are headless arcs out of t_i; and, arc numbers are costs.

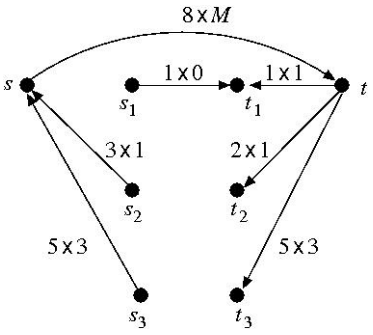


N_2 N_3

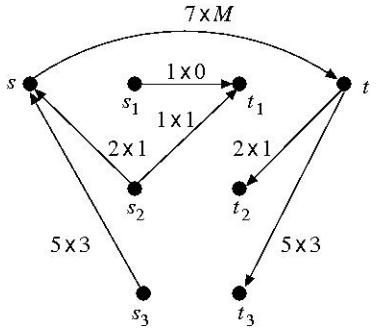
Let the initial feasible basis be the tree with arcs (s,t), $\{(s_i,s),\ (t,t_i) : i = 1,\ldots,n\}$. The basic levels are the associated supplies and demands. The following shows the initial basis for N_3 and the new basis after one iteration. Arc numbers are flow×cost.



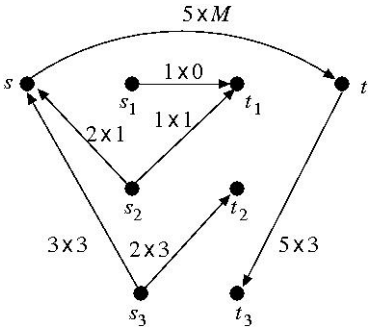
Initial basis for N_3
Total cost $= 38 + 9M$



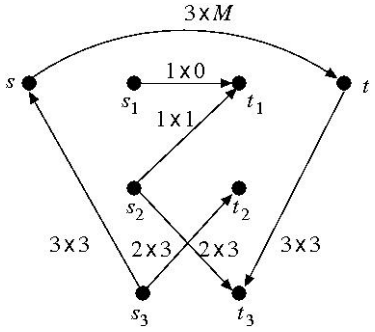
Basis after one iteration
Total cost $= 36 + 8M$



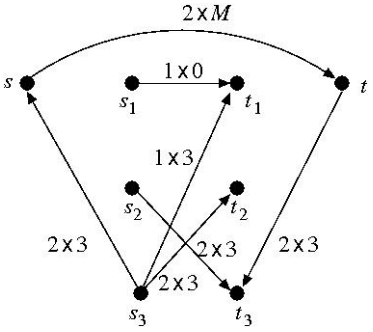
Basis after two iterations
Total cost $= 35 + 7M$



Basis after three iterations
Total cost $= 33 + 5M$

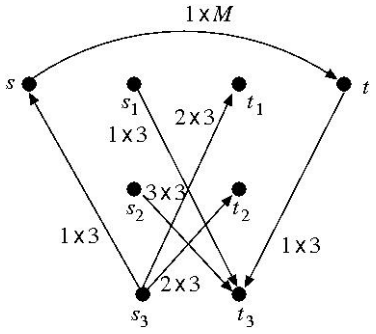


Basis after four iterations
Total cost $= 31 + 3M$



Basis after five iterations
Total cost $= 30 + 2M$

The optimal solution for N_3, shown to the right, has Total cost $= 30 + M$, which is reached in six iterations. (One unit of flow must flow across (s,t) to satisfy demand at t_3).



The choice of arc (s_1, t_1) to enter the basis is because its reduced cost, $-(M+1)$, yields the greatest rate of cost decrease. Every arc of the form (s_i, t_j) reduces a unit of flow across (s,t), so the most negative reduced cost is the one with the least cost. Thus, $c(s_1, t_1) = 0$ is the one. Arc (s_2, t_1), for example, has reduced cost $1 - (M+1)$.

While each iteration selects the arc to enter the basis that has the greatest rate of decrease in flow across (s,t), the actual reduction is limited. For N_3, the reduction is either 1 or 2 each iteration. For $n \geq 3$, the initial flow across (s,t) is

$$\begin{aligned} \textstyle\sum_{i=1}^{n} S_i = \sum_{i=1}^{n} D_i &= 4 + \textstyle\sum_{i=3}^{n} \left(2^{i-1} + 2^{i-3}\right) \\ &= 4 + 5\textstyle\sum_{i=0}^{n-3} 2^i = 4 + 5(2^{n-2} - 1) = 5 \times 2^{n-2} - 1 \end{aligned}$$

Since we do not have arc (s_n, t_n) and the total supply from $s_1, \ldots, s_{n-1}$ is less than D_n, the difference must go across (s,t). That is, the minimum value of $x(s,t)$ is

$$\begin{aligned} D_n - \textstyle\sum_{i=1}^{n-1} S_i &= 2^{n-1} + 2^{n-3} - 1 - 3 - \textstyle\sum_{i=3}^{n-1} \left(2^{i-1} + 2^{i-3}\right) \\ &= 2^{n-1} + 2^{n-3} - 4 - 5(2^{n-3} - 1) \\ &= 2^{n-1} + 2^{n-3} + 1 - 5 \times 2^{n-3} \\ &= 2^{n-1} + 1 - 4 \times 2^{n-3} = 1. \end{aligned}$$

I believe Zadeh's argument claims that the standard simplex method reduces the flow across (s,t) by no more than K each iteration. Since the initial feasible bases has $x(s,t) = O(2^n)$, the number of iterations for N_n to reach the optimal flow of $x(s,t) = 1$ is $O(2^n)$.

Opportunity Knocks

My interpretation of Zadeh's argument could be wrong, and I am unable to present a complete proof that the reduction of $x(s,t)$ in one iteration is limited by some constant, K. I have been unable to reach Zadeh to obtain clarification (and his approach is different). It would be useful to have this completed (or some other counterexample).

Zadeh analyzes other network problems and algorithms using this type of construction. He also provides more pathological examples in [80]. (See Orlin[59] for a polynomial-time simplex algorithm for min-cost network flows.)

LP Myth 44. *The c-diameter of a non-empty polytope of dimension d with f facets cannot exceed $f - d$.*

Let P denote the polytope (that is, bounded polyhedron), and let $V^*(P, c)$ denote the set of vertices that minimize a linear form, cx, on P. The *c-diameter* from a vertex $v \in P$ for a given linear form is defined as the maximum distance from v to $V^*(P, c)$. The *distance* is defined to be the minimum number of edges in a path joining v to $V^*(P, c)$ along which cx is non-increasing. (In terms of LP, the c-diameter is an upper bound on how many vertices the simplex method visits before reaching an optimal vertex.) Denote the c-diameter from v by $\Delta(v, c)$, and the myth asserts $\Delta(v, c) \leq f - d$. This is known as the *monotonic bounded Hirsch conjecture*.

Counterexample. Todd[75] provides the following:

$$P = \{x \in \mathbb{R}^4_+ : Ax \leq b\}, \text{ where } A = \begin{bmatrix} 7 & 4 & 1 & 0 \\ 4 & 7 & 0 & 1 \\ 43 & 53 & 2 & 5 \\ 53 & 43 & 5 & 2 \end{bmatrix}, \; b = \begin{pmatrix} 1 \\ 1 \\ 8 \\ 8 \end{pmatrix}.$$

This is a 4-dimensional polytope with 8 facets, so the myth asserts that the c-diameter cannot exceed 4 for any linear form. Let $c = (1, 1, 1, 1)$, so $V^*(P, c) = \{(0, 0, 0, 0)^\intercal\}$. Todd proves that all non-increasing paths from $v = \frac{1}{19}(1, 1, 8, 8)^\intercal$ to 0 have a distance of 5.

See Klee and Kleinschmidt[50] for an extensive survey of the Hirsch conjecture and related properties of polytopes. Also see Holt and Klee[45] for a counterexample to the strong d-step conjecture.

LP Myth 45. *Determining whether an LP has a degenerate basis has the same complexity as solving the LP.*

LP has a polynomial algorithm, but Chandrasekaran, Kabadi, and Murty[16] prove that the *degeneracy testing problem* is NP-complete. It is easily seen that degeneracy testing is in NP, so it remains to construct a polynomial reduction of an NP-complete problem to degeneracy testing. They use the NP-complete *subset sum problem*[26, SP13]:

$$\text{SS: Given } a_1, \ldots, a_n, b \in \mathbb{Z}_+, \text{ find } x \in \{0, 1\}^n : ax = b.$$

Consider an $n \times 2$ transportation problem with supplies $s = a$ and demands $d = (b, \sum_{i=1}^n a_i - b)$. Then, checking whether SS has a solution is equivalent to checking if the usual algebraic representation of the transportation polytope is degenerate:

$$\sum_j x_{ij} = s_i, \; \sum_i x_{ij} = d_j, \; x \geq 0.$$

Chandrasekaran et al. note that the transportation problem is degenerate whenever there are subsets with total supply equal to total demand — that is, there exist non-empty, proper subsets I, J ($\emptyset \neq I \subset \{1, \ldots, m\}$, $\emptyset \neq J \subset \{1, \ldots, n\}$) such that $\sum_{i \in I} s_i = \sum_{j \in J} d_j$.

My thanks to Katta Murty for providing clarification and the following additional examples of NP-complete (or NP-hard) problems.

1. Find a BFS with the fewest number of positive variables.
2. Find a maximum-cardinality subset of minimally linearly dependent vectors.
3. Find a minimum-cardinality subset of linearly dependent vectors containing a given vector.
4. Find a singular principal submatrix of a square matrix.

LP Myth 46. *In employing successive bound reduction in a presolve, we can fix a variable when its bounds are within a small tolerance of each other.*

The myth is that we can pick a tolerance, say $\tau > 0$, such that if we infer $L \le x \le U$ and $U - L \le \tau$, we can fix x to some value in the interval, such as the midpoint, $\frac{1}{2}(L+U)$. There are a few things wrong with this, as reported by Greenberg[34].

Counterexample. Consider $x \ge 0$ and

$$\begin{array}{rcrcl} \frac{1}{2}x_1 & + & x_2 & = & 1 \\ x_1 & + & x_2 & = & 2. \end{array}$$

This has the unique solution, $x = (2, 0)$, and it is this uniqueness that causes a problem with greater implications.

In successive bound reduction, the most elementary tests evaluate rows to see if just one row alone can tighten a bound on a variable. Initially, the bounds are the original ones: $L^0 = L = (0, 0)$ and $U^0 = U = (\infty, \infty)$. The first iteration results in the inference that $x_1 \le 2$, from the first equation and the fact that $x_2 \ge 0$. It similarly produces an upper bound, $x_2 \le 1$, so $U^1 = (2, 1)$. Still in iteration 1, the second equation causes the inference, $x_1 \ge 1$, because we already have $x_2 \le 1$ when we get there. Thus, $L^1 = (1, 0)$.

At a general iteration, we will have inferred $L_1^k \le x_1 \le 2$ and $0 \le x_2 \le U_2^k$, where $L_1^k < 2$ and $U_2^k > 0$. At the end of iteration k, the inferred bounds are:

$$2 - (\tfrac{1}{2})^k \le x_1 \le 2 \quad \text{and} \quad 0 \le x_2 \le (\tfrac{1}{2})^k .$$

This converges to the unique solution, but it does not reach it finitely. If the iterations go far enough, the bounds become within the tolerance $\tau > 0$. At that point, suppose x is fixed to the interval's midpoint: $x = \frac{1}{2}(L^k + U^k)$.

To see a consequence of this, suppose that the presolve tests feasibility with another tolerance, μ. Let the constraints be of the form $Ax = b$. The rule is: *Declare infeasibility if, for some equation, i,*

$$\begin{aligned} y_i^{\max} &= \max_{L^k \le x \le U^k} A_{i\bullet}x < b_i - \mu \quad \textit{or} \\ y_i^{\min} &= \min_{L^k \le x \le U^k} A_{i\bullet}x > b_i + \mu. \end{aligned}$$

In our example, when $k = \lceil -\log_2 \tau \rceil$, both variables are fixed:

$$x_1 = 2 - (\tfrac{1}{2})^{k+1}, \; x_2 = (\tfrac{1}{2})^{k+1} .$$

Equation 2 passes the feasibility test, but equation 1 has

$$y_1^{\max} = y_1^{\min} = 1 - (\tfrac{1}{2})^{k+2} + (\tfrac{1}{2})^{k+1} = 1 + (\tfrac{1}{2})^{k+2} .$$

Thus, $y_1^{\min} = 1 + (\frac{1}{2})^{k+2}$, so we declare infeasibility if $(\frac{1}{2})^{k+2} > \mu$. Taking logs, this is equivalent to $-(k+2) > \log_2 \mu$. Replacing k, we have that a false infeasibility is declared if

$$-\lceil -\log_2 \tau \rceil - 2 > \log_2 \mu.$$

For example, if $\tau = 2^{-20}$, we declare a *false infeasibility* if $\mu < 2^{-22}$.

This example highlights two things:

1. Tolerances are related. The tolerance to fix a variable should not be substantially less than the infeasibility tolerance.
2. Fix a variable judiciously. When having inferred $x_j \in [L_j, U_j]$, such that $U_j - L_j$ is within tolerance of fixing x_j, do so in the following order of choice:
   (1) If L_j is an original bound, fix $x_j = L_j$;
   (2) If U_j is an original bound, fix $x_j = U_j$;
   (3) If $[L_j, U_j]$ contains an integer, p, fix $x_j = p$;
   (4) If all of the above fail, fix $x_j = \frac{1}{2}(L_j + U_j)$.

LP Myth 47. *A factored form of the basis contains less error for FTRAN after reinversion.*

The Forward Transformation (FTRAN) algorithm solves the forward system, $Bx = b$, by factoring B and updating it after each basis change. Consider the elementary product form: $B = E_1 E_2 \cdots E_k$, where each E_i is an elementary matrix.

Algorithm: Forward Transformation with PFI
Initialize. Set $x^0 = b$.
for $i = 1 : k$ **do**
  Solve $E_i x^i = x^{i-1}$
end for
Exit with x^k the (computed) solution to $Bx = b$.

During the pivoting process, k increases and there are more factors than needed. *Reinversion* is the process of restarting to obtain the minimum number of factors, which equals the number of variables in the basis (except slacks). One reason to reinvert is to "cleanup" the errors that accumulate, which affects the accuracy of solving $Bx_B = b$. (Another reason is to reduce the FTRAN time.)

The essence of the counterexample is cancelation of errors in the first factors that does not cancel in the reinverted factorization.

Counterexample. Consider the 2×3 system:

$$\begin{array}{ccccccc} a_{11}x_1 & + & a_{12}x_2 & + & a_{13}x_3 & = & b_1 \\ a_{21}x_1 & + & a_{22}x_2 & + & a_{23}x_3 & = & b_2 \end{array}$$

Pivoting x_1 on equation 1, then x_2 on equation 2 into the basis, then replacing x_1 with x_3

yields the following elementary factors:

$$E_1 = \begin{bmatrix} a_{11} & 0 \\ a_{21} & 1 \end{bmatrix};\ E_2 = \begin{bmatrix} 0 & a_{12}/a_{11} \\ 1 & a_{22} - a_{21}a_{12}/a_{11} \end{bmatrix}$$

$$E_3 = \begin{bmatrix} a_{13}/a_{11} - a_{22} - a_{12}a)21/a_{11})((a_{23} - a_{13}a)21/a_{11})/a_{12}/a_{11} & 0 \\ \big(a_{23} - a_{13}a_{21}/a_{11}\big)/(a_{12}/a_{11}) & 1 \end{bmatrix}.$$

Collecting computed values and substituting c with a new index whenever there is a new computation, we obtain:

$$E_1 = \begin{bmatrix} a_{11} & 0 \\ a_{21} & 1 \end{bmatrix};\ E_2 = \begin{bmatrix} 0 & c_1 \\ 1 & c_2 \end{bmatrix};\ E_3 = \begin{bmatrix} c_3 & 0 \\ c_4 & 1 \end{bmatrix}.$$

Then, executing FTRAN for b (to get basic levels):

$$\begin{aligned} x^1 &= \begin{pmatrix} b_1/a_{11} \\ b_2 - (b_1/a_{11})a_{12} \end{pmatrix} &= \begin{pmatrix} c_5 \\ c_6 \end{pmatrix} \\ x^2 &= \begin{pmatrix} x_1^1 - (x_2^1/c_2)c_1 \\ x_2^1/c_2 \end{pmatrix} &= \begin{pmatrix} c_7 \\ c_8 \end{pmatrix} \\ x^3 &= \begin{pmatrix} x_1^2/c_3 \\ x_2^2 - (x_1^2/c_3)c_4 \end{pmatrix} &= \begin{pmatrix} c_9 \\ c_{10} \end{pmatrix} \end{aligned}$$

After reinversion, the elementary matrices have the form:

$$E_1 = \begin{bmatrix} a_{13} & 0 \\ a_{23} & 1 \end{bmatrix};\ E_2 = \begin{bmatrix} 0 & c_{11} \\ 1 & c_{12} \end{bmatrix}.$$

Now the FTRAN algorithm yields computed levels:

$$\widehat{B^{-1}b} = \begin{pmatrix} c_{13} \\ c_{14} \end{pmatrix}.$$

Suppose $\beta = B^{-1}b$, the true value of the levels. The issue is whether

$$\left|\left|\begin{pmatrix} \beta_1 - c_{13} \\ \beta_2 - c_{14} \end{pmatrix}\right|\right| = ||\beta - \zeta'|| < \left|\left|\begin{pmatrix} \beta_1 - c_9 \\ \beta_2 - c_{10} \end{pmatrix}\right|\right| = ||\beta - \zeta||\,,$$

where ζ the accumulated error before reinversion, and ζ' is the accumulated error after reinversion.

It is possible that $\zeta = 0$ while $\zeta' \neq 0$ — that is, that we obtain an error-free solution with the original factorization and reinversion introduces error. This can happen by error cancelation. However, even if $||\zeta'|| < ||\zeta||$, the computed levels could have less error, at least for some particular b. For example, let $\beta = (100, 100)^\intercal$, $\zeta = (2, 2)^\intercal$, and $\zeta' = (1, -1)^\intercal$. Then, $||\zeta|| > ||\zeta'||$, yet $||\beta - \zeta|| \cong 138.6 < 141.4 \cong ||\beta - \zeta'||$.

LP References

[1] M. Anitescu. Degenerate nonlinear programming with a quadratic growth condition. *SIAM Journal on Optimization*, 10(4):1116–1135, 2000.

[2] H. Arsham. On the more-for-less paradoxical situations in linear programs: A parametric optimization approach. *Journal of Information & Optimization Sciences*, 17(3):485–500, 1996.

[3] H. Arsham. Tools for modeling validation process: The dark side of LP. Web note, Merrick School of Business, University of Baltimore, http://home.ubalt.edu/ntsbarsh/opre640A/partv.htm, 1996–2008.

[4] H. Arsham. Construction of the largest sensitivity region for general linear programs. *Applied Mathematics and Computation*, 189(2):1435–1447, 2007.

[5] H. Arsham and M. Oblak. Perturbation analysis of general LP models: A unified approach to sensitivity, parametric, tolerance, and more-for-less analysis. *Mathematical and Computer Modelling*, 13(8):79–102, 1990.

[6] E. M. L. Beale. Cycling in the dual simplex method. *Naval Research Logistics Quarterly*, 2(4), 1955.

[7] J. E. Beasley. *OR-Notes*. Brunel University, http://people.brunel.ac.uk/~mastjjb/jeb/or/sp.html, 1999.

[8] M. Bellmore, H. J. Greenberg, and J. J. Jarvis. Multi-commodity disconnecting sets. *Management Science*, 16(6):427–433, 1970.

[9] M. Bellmore and R. R. Vemuganti. On multi-commodity maximal dynamic flows. *Operations Research*, 21(1):10–21, 1973.

[10] A. A. Benczúr. Counterexamples for directed and node capacitated cut-trees. *SIAM Journal on Computing*, 24(3):505–510, 1995.

[11] D. P. Bertsekas. *Linear Network Optimization: Algorithms and Codes*. MIT Press, Cambridge, MA, 1991.

[12] C. Blair. Some linear programs requiring many pivots. Technical Report 867, College of Commerce and Business Administration, University of Illinois, Champaign, IL, 1982. Available at https://netfiles.uiuc.edu/c-blair/www/papers/papers.html.

[13] C. Blair. Random linear programs with many variables and few constraints. *Mathematical Programming*, 34(1):62–71, 1986.

[14] D. Braess. Über ein paradoxon aus der verkehrsplanung. *Unternehmensforschung*, 12:258–268, 1968.

[15] D. Braess, A. Nagurney, and T. Wakolbinger. On a paradox of traffic planning. *Transportation Science*, 39(4):446–450, 2005.

[16] R. Chandrasekaran, S. N. Kabadi, and K. G. Murty. Some NP-complete problems in linear programming. *Operations Research Letters*, 1(3):101–104, 1982.

[17] A. Charnes and W. W. Cooper. The stepping stone method of explaining linear programming calculations in transportation problems. *Management Science*, 1(1):49–69, 1954.

[18] A. Charnes, S. Duffuaa, and M. Ryan. Degeneracy and the more-for-less paradox. *Journal of Information & Optimization Sciences*, 1(1):52–56, 1980.

[19] A. Charnes, F. Glover, and D. Klingman. A note on a distribution problem. *Operations Research*, 18(6):1213–1216, 1970.

[20] A. Charnes and D. Klingman. The distribution problem with upper and lower bounds on the node requirements. *Management Science*, 16(9):638–642, 1970.

[21] A. Charnes and D. Klingman. The "more for less" paradox in the distribution model. *Cahiers du Centre d'Etudes de Recherche Opérationnelle*, 13(1):11–22, 1971.

[22] V. G. Dĭneko, B. Klinz, and G. J. Woeginger. Which matrices are immune against the transportation paradox. *Discrete Applied Mathematics*, 130(3):495–501, 2003.

[23] J. Elam, F. Glover, and D. Klingman. A strongly convergent primal simplex algorithm for generalized networks. *Mathematics of Operations Research*, 4(1):39–59, 1979.

[24] J. J. M. Evers. *Linear programming over an infinite horizon*, volume 8 of *Tilburg studies on economics*. Tilburg University Press, The Netherlands, 1973.

[25] L. Ford and D. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton, NJ, 1963.

[26] M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. freeman, New York, NY, 1979.

[27] S. I. Gass. *Linear Programming: Methods and Applications*. McGraw-Hill, New York, NY, fifth edition, 1975.

[28] S. I. Gass and S. Vinjamuri. Cycling in linear programming problems. *Computers & Operations Research*, 31(2):303–311, 2004.

[29] B. J. Gassner. Cycling in the transportation problem. *Naval Research Logistics Quarterly*, 11(1):43–58, 1964.

[30] F. Glover. *Netforms in Management Science and Decision Support*. University of Colorado, School of Business, Boulder, CO, 1983.

[31] F. Glover and D. A. Babayev. New results for aggregating integer-valued equations. *Annals of Operations Research*, 58:227–242, 1995.

[32] A. J. Goldman and D. Kleinman. Examples relating to the simplex method. *Operations Research*, 12(1):159–161, 1964.

[33] R. E. Gomory and T. C. Hu. Multi-terminal network flows. *Journal of the Society for Industrial and Applied Mathematics*, 9(4):551–570, 1961.

[34] H. J. Greenberg. *A Computer-Assisted Analysis System for Mathematical Programming Models and Solutions: A User's Guide for ANALYZE*, volume 1 of *Operations Research/Computer Science Interfaces Series*. Springer, New York, NY, 1993.

[35] H. J. Greenberg. How to analyze the results of linear programs — Part 2: Price interpretation. *Interfaces*, 23(5):97–114, 1993.

[36] H. J. Greenberg. Consistency, redundancy, and implied equalities in linear systems. *Annals of Mathematics and Artificial Intelligence*, 17:37–83, 1996.

[37] H. J. Greenberg. Chapter 3: Linear programming 1: Basic principles. In T. Gal and H. J. Greenberg, editors, *Advances in Sensitivity Analysis and Parametric Programming*. Kluwer Academic Press, 1997.

[38] H. J. Greenberg and J. E. Kalan. An exact update for Harris' tread. *Mathematical Programming Study*, 4:26–29, 1975.

[39] R. C. Grinold. Finite horizon approximations of infinite horizon linear programs. *Mathematical Programming*, 12(1):1–17, 1977.

[40] R. C. Grinold. Model building techniques for the correction of end effects in multistage convex programs. *Operations Research*, 31(3):407–431, 1983.

[41] R. C. Grinold and D. S. P. Hopkins. Duality overlap in infinite linear programs. *Journal of Mathematical Analysis and Applications*, 41(2):333–335, 1973.

[42] J. A. J. Hall and K. I. M. McKinnon. The simplest examples where the simplex method cycles and conditions where EXPAND fails to prevent cycling. *Mathematical Programming, Series B*, 100(1):133–150, 2004.

[43] J. L. Higle and S. Sen. Duality for multistage convex stochastic programs. *Annals of Operations Research*, 142(1):129–146, 2006.

[44] A. J. Hoffman. Cycling in the simplex algorithm. Report 2974, National Bureau of Standards, Gaithersburg, MD, 1953.

[45] F. Holt and V. Klee. Counterexamples to the strong d-step conjecture for $d \geq 5$. *Discrete & Computational Geometry*, 19(1):33–46, 1998.

[46] D. S. P. Hopkins. *Sufficient conditions for optimality in infinite horizon linear economic models*. Ph. D. thesis, Stanford University, Stanford, CA, 1969.

[47] R. G. Jeroslow. The simplex algorithm with the pivot rule of maximizing criterion improvement. *Discrete Mathematics*, 4(4):367–377, 1973.

[48] S. Kaplan. Comment on a Précis by Shanno and Weil. *Management Science*, 17(11):778–780, 1971.

[49] N. Karmarkar. A new polynomial time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[50] V. Klee and P. Kleinschmidt. The d-step conjecture and its relatives. *Mathematics of Operations Research*, 12(4):718–755, 1987.

[51] V. Klee and G. J. Minty. How good is the simplex algorithm? In O. Shisha, editor, *Inequalities III*, pages 159–175, New York, NY, 1972. Academic Press.

[52] H. W. Kuhn. Variants of the Hungarian method for assignment problem. *Naval Research Logistics Quarterly*, 3:253–258, 1956.

[53] W. F. Mascarenhas. The affine scaling algorithm fails for stepsize 0.999. *SIAM Journal on Optimization*, 7(1):34–46, 1997.

[54] A. Nagurney. *Network Economics: A Variational Inequality Approach*, volume 10 of *Advances in Computational Economics*. Springer, Dordrecht, The Netherlands, revised edition, 1999.

[55] A. Nagurney. *Supernetworks: Decision-Making for the Information Age*. Edward Elgar Publishing, Cheltenham, England, 2002.

[56] A. Nagurney. Supernetworks: Paradoxes, challenges, and new opportunities. In W. H. Dutton, B. Kahin, R. O'Callaghan, and A. W. Wyckoff, editors, *Transforming Enterprise*, pages 229–254, Cambridge, MA, 2004. MIT Press.

[57] A. Nagurney. *Supply Chain Network Economics: Dynamics of Prices, Flows And Profits*. New Dimensions in Networks. Edward Elgar Publishing, Cheltenham, England, 2006.

[58] J. B. Orlin. On the simplex algorithm for networks and generalized networks. *Mathematical Programming Study*, 24:166–178, 1985.

[59] J. B. Orlin. A polynomial time primal network simplex algorithm for minimum cost flows. *Mathematical Programming*, 78(1):109–129, 1997.

[60] M. R. Rao. Some comments on 'linear' programming with absolute-value functionals. *Operations Research*, 21(1):373–374, 1973.

[61] A. Ravindran and T. W. Hill, Jr. A comment on the use of simplex method for absolute value problems. *Management Science*, 19(5):581–582, 1973.

[62] R. Rizzi. Excluding a simple good pair approach to directed cuts. *Graphs and Combinatorics*, 17(4):741–744, 2001.

[63] B. Rothschild and A. Whinston. Maximal two-way flows. *SIAM Journal on Applied Mathematics*, 5(5):1228–1238, 1967.

[64] T. Roughgarden. *Selfish Routing and the Price of Anarchy*. MIT Press, Cambridge, MA, 2005.

[65] T. Roughgarden. On the severity of Braess's paradox: Designing networks for selfish users is hard. *Journal of Computer and System Sciences*, 72(5):922–953, 2006.

[66] D. S. Rubin and H. M. Wagner. Shadow prices: Tips and traps for managers. *Interfaces*, 20(4):150–157, 1990.

[67] M. J. Ryan. The distribution problem, the more for less (nothing) paradox and economies of scale and scope. *European Journal of Operational Research*, 121(1):92–104, 2000.

[68] T. L. Saaty. A conjecture concerning the smallest bound on the iterations in linear programming. *Operations Research*, 11(1):151–153, 1963.

[69] S. Sen. Subgradient decomposition and the differentiability of the recourse function of a two stage stochastic LP with recourse. *Operations Research Letters*, 13(3):143–148, 1993.

[70] D. F. Shanno and R. L. Weil, Jr. 'Linear' programming with absolute value functionals. *Operations Research*, 19(1):120–124, 1971.

[71] G. Sierksma and G. A. Tijssen. Degeneracy degrees of constraint collections. *Mathematical Methods of Operations Research*, 57(3):437–448, 2003.

[72] D. A. Spielman and S.H. Teng. Smoothed analysis: why the simplex algorithm usually takes polynomial time. *Journal of the Association for Computing Machinery*, 51(3):385–463, 2004.

[73] R. Steinberg and W. I. Zangwill. The prevalence of braess' paradox. *Transportation Science*, 17(3):301–318, 1983.

[74] W. Szwarc. The transportation paradox. *Naval Research Logistics Quarterly*, 18:185–202, 1971.

[75] M. J. Todd. The monotonic bounded Hirsch conjecture is false for dimension at least 4. *Mathematics of Operations Research*, 5(4):599–601, 1980.

[76] J. A. Tomlin and J. S. Welch. A pathological case in the reduction of linear programs. *Operations Research Letters*, 2(2):53–57, 1983.

[77] S. Storøy and T. Sørevik. A note on an orthogonal systolic design for the assignment problem. *Parallel Computing*, 17(4–5):523–525, 1991.

[78] J. Y. Yen. On Elmaghraby's "The theory of networks and management science". *Management Science*, 18(1):84–86, 1971.

[79] N. Zadeh. A bad network problem for the simplex method and other minimum cost flow algorithms. *Mathematical Programming*, 5(1):255–266, 1973.

[80] N. Zadeh. More pathological examples for network flow problems. *Mathematical Programming*, 5:217–224, 1973.

[81] T. Zhao and C-L. Tseng. A note on activity floats in activity-on-arrow networks. *Journal of the Operational Research Society*, 54(12):1296–1299, 2003.

[82] G. M. Ziegler. *Lectures on Polytopes*, volume 152 of *Graduate Texts in Mathematics*. Springer, 1995.

[83] H. Zullo. *Feasibile Flows in Multicommodity Graphs*. Ph.D. thesis, University of Colorado at Denver, Mathematics Department, Denver, CO, 1995. Available at http://www-math.cudenver.edu/graduate/thesis/hzullo.pdf.

Integer Programming and Combinatorial Optimization

The general form of an *Integer Program* (IP) is the optimization (min or max) of a function over a domain such that the variables are required to have integer values. An *Integer Linear Program* (ILP) has the form of LP with $x \in \mathbb{Z}^n$. If only some of the variables must be integer, it is called a *Mixed-Integer Program* (MIP). If it has the form of LP, but with $x_j \in \mathbb{Z}$ for $j \in J \neq \emptyset$, it is a *Mixed-Integer Linear Program* (MILP). The *Linear Programming Relaxation* (LPR) of a MILP is the *Linear Programming Relaxation* of $x_j \in \mathbb{Z}$, allowing non-integer solutions for all variables.

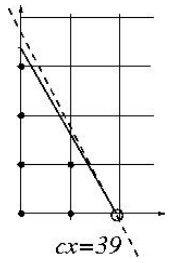
We include combinatorial optimization problems, even those that are not usually modeled with IP.

IP Myth 1. *The integer solution is a rounding of its LPR solution.*

It is possible that every MILP solution could be far from the relaxed solution. In fact, it is possible that no rounding is feasible.

Counterexample. max $21x_1 + 11x_2 : x \geq 0, 7x_1 + 4x_2 \leq 13$.

The relaxed solution is at $\left(\frac{13}{7}, 0\right)$, and the optimal integer solution is at $(0, 3)$.



Glover and Sommer[23] provide more meaningful examples, including a conditional transportation problem. Additional examples and discussion are in Glover[20] and Glover, Klingman and Phillips[22].

IP Myth 2. *If a basic solution of the LPR of a MILP is not integer-valued, no rounding is possible.*

This apparent paradox was noted by Glover and Sommer[23] with the following "proof:"

> The only fractional variables, which could be rounded, are basic, but the basic equations, $Bx_B = b$, have a unique solution. Hence, no rounding is possible!

The flaw is the assumption that all non-basic variables must remain fixed at zero (or an upper bound). In particular, slack variables may change to offset the rounding.

IP Myth 3. *The LPR solves its associated ILP if, and only if, it has an optimal basic solution that is integer-valued.*

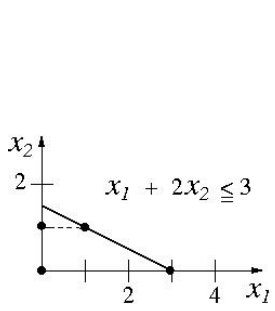
The sufficiency is always true, but the necessity of an integer-valued optimum that is basic applies to binary programs (whose LPR is in standard form) and may not hold otherwise. The following counterexample has an optimality region with non-integer extreme points but an optimal integer point in its interior.

Counterexample. max $0x : 0 \leq x \leq b\,\boldsymbol{e},\ x \in \mathbb{Z}$, where $b \notin \mathbb{Z}$ and $\boldsymbol{e}$ is a vector of ones. Dropping the integer requirement, the LP solution is any feasible point. For $b > 1$, $\boldsymbol{e}$ is feasible and hence optimal for the LP. Therefore, it is optimal for the ILP, but it is not a basic optimum for the LPR.

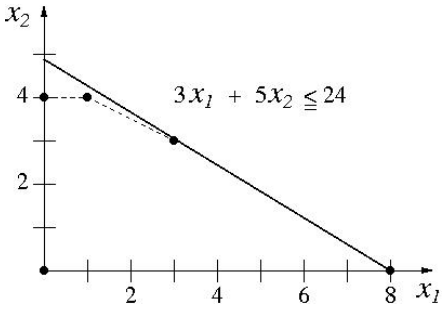
IP Myth 4. *The number of extreme points of the integer hull is bounded by some multiple of those of the LPR, where the multiple depends upon the number of variables and constraints.*

Rubin[53] provides the following counterexample that shows the integer hull can have any number of extreme points with only one constraint in $\mathbb{R}^2_+$. (Also see Jeroslow[34].)

Counterexample. Define a polytope in 2 variables and 1 constraint plus non-negativity: $P = \{x \in \mathbb{R}^2_+ : a_1x_1 + a_2x_2 \leq b\}$, where $a, b > 0$. This has three extreme points. We can choose a, b such that its integer hull, $\text{convh}(P \cap \mathbb{Z}^2)$, has N extreme points for any $N \geq 3$. The following figures show two such polytopes:



(a) $N = 4$



(b) $N = 5$

The following table shows more, and you may note a pattern that Rubin discovered.

N	a_1	a_2	b
4	1	2	3
5	3	5	24
6	8	13	168
7	21	34	1,155
8	55	89	7,920
9	144	233	54,288
10	377	610	372,099

Let F_k be the k^{th} Fibbonacci number, and $P = \{x \in \mathbb{R}^2_+ : F_{2k}x_1 + F_{2k+1}x_2 \leq F^2_{2k+1} - 1\}$. Then, $\text{convh}(P \cap \mathbb{Z}^2)$ has $k + 3$ extreme points.

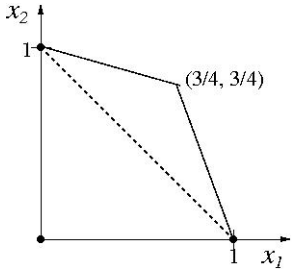
Rubin gives other ways to generate the polytope for one constraint in $\mathbb{R}^2_+$ such that it has any number of extreme points.

The same myth and counterexample applies if "facets" replaces "extreme points."

IP Myth 5. *The number of extreme points of the integer hull is at least as great as the number of extreme points of the LPR polyhedron.*

Counterexample.

The LPR polyhedron can have regions with no integer points, as illustrated to the right. The integer hull has 3 extreme points, whereas the LPR polyhedron has 4 (and could have any arbitrary number).



$$3x_1 + x_2 \leq 3$$
$$x_1 + 3x_2 \leq 3$$
$$x \geq 0$$

The same myth and counterexample applies if "facets" replaces "extreme points."

IP Myth 6. *Every integral vector of an n-dimensional integral polyhedral pointed cone C can be expressed as a non-negative integral combination of at most n elements of the Hilbert basis of C.*

It it were true, this would be an extension of Carathéodory's theorem. Let $z^1, \ldots, z^k \in \mathbb{Z}^n$ be *generators* of

$$C = \{z : z = \sum_{i=1}^{k} \lambda_i z^i \text{ for some } \lambda \in \mathbb{Z}_+^k\}.$$

Counterexample. Bruns et al.[8] provide the following:

$$\begin{array}{rclrcl} z^1 &=& (0,1,0,0,0,0), & z^6 &=& (1,0,2,1,1,2), \\ z^2 &=& (0,0,1,0,0,0), & z^7 &=& (1,2,0,2,1,1), \\ z^3 &=& (0,0,0,1,0,0), & z^8 &=& (1,1,2,0,2,1), \\ z^4 &=& (0,0,0,0,1,0), & z^9 &=& (1,1,1,2,0,2), \\ z^5 &=& (0,0,0,0,0,1), & z^{10} &=& (1,2,1,1,2,0). \end{array}$$

The generators form a Hilbert basis for C, and the myth asserts that every integral vector in C is a conical combination of only 6 of the 10 generators. Consider

$$g = (9,13,13,13,13,13) = z^1 + 3z^2 + 5z^4 + 2z^5 + z^8 + 5z^9 + 3z^{10}.$$

A minimum number of generators can be obtained by the ILP:

$$\min \textstyle\sum_{i=1}^{10} u_i : \sum_{i=1}^{10} \lambda_i z^i = g,$$
$$u_i \in \{0,\, 1\},\ 0 \le \lambda_i \le 13u_i,\ \lambda_i \in Z \text{ for } i = 1, \ldots, 10.$$

Bruns et al. solved this and found that seven generators are needed. They show how to generate more counterexamples, giving insight into why more than six are necessary.

Bruns et al. also prove that for $n \ge 6$ there exists some $C \subseteq \mathbb{Z}_+^n$ for which at least $\lfloor \frac{7}{6} n \rfloor$ vectors are needed to span its integral vectors.

IP Myth 7. **new** *Every continuous facet for the infinite group problem is piecewise linear.*
next new ▷

Let $\mathcal{H}$ denote the set of *Haar functions*:

$$\mathcal{H} = \left\{ f : [0,1] \to \mathbb{R} : \left| \{x : f(x) > 0\} \right| < \infty \right\}.$$

Because they have a finite number of non-zero values, Haar functions are summable; $\sum_{r \in [0,1]} f(r)$ restricts the summation to the support set of f. Let

$$\mathcal{F} = \{f \in \mathcal{H} : f(r) \in \mathbb{Z}_+, \forall r \in [0,1]\}.$$

The *infinite group problem* is defined as finding $f \in \mathcal{F}$ such that

$$\sum_r r f(r) = f_0, \tag{IP.5}$$

where $f_0 \in [0,1]$ and the summation is modulo 1. A function, w, is *valid* if $w \geq 0$, $w(0) = 0$, and every solution to (IP.5) satisfies

$$\sum_r w(r) f(r) \geq 1. \tag{IP.6}$$

Let W denote the class of valid functions, and let $P(w)$ denote associated solutions to (IP.6). A *facet* for (IP.5) is $w \in W$ such that $P(w') \supseteq P(w) \rightarrow w' = w$ for all $w' \in W$. (A facet is continuous if it is a continuous function on $[0,1]$.)

This myth is a conjecture posed by Gomory and Johnson[25] and resolved by Basu, Conforti, Cornuéjols, and Zambeli[4] with the following

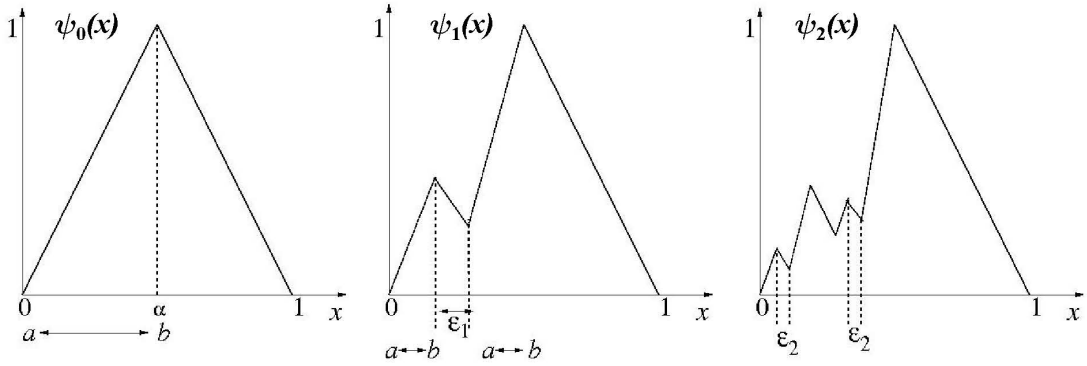
Counterexample. Define the following piecewise-linear function on $x \in [0,1]$:

$$\psi_0 = \begin{cases} \frac{1}{2}x & \text{for } x \in [0, \frac{1}{2}] \\ \frac{1}{2}(1-x) & \text{for } x \in [\frac{1}{2}, 1]. \end{cases}$$

(Note: this is a special case of the general construction given by Basu et al.) Then, construct $\{\psi_i\}_0^\infty$ by defining ψ_{i+1}, given ψ_i, by replacing the line segment from $(a, \psi_i(a))$ to $(b, \psi_i(b))$ for $[a,b] \subseteq [0, \frac{1}{2}]$ by three segments:

$$(a, \psi_i(a)) \rightarrow \left(\overline{ab} - \varepsilon_i,\ \psi_i(\overline{ab}) + 2\,\varepsilon_i\right) \rightarrow \left(\overline{ab} + \varepsilon_i,\ \psi_i(\overline{ab}) - 2\,\varepsilon_i\right) \rightarrow (b, \psi_i(b)),$$

where $\overline{ab} = \frac{1}{2}(a+b)$ and $\varepsilon_i = (\frac{1}{2})^{2(i+3)}$ for $i = 0, 1, \ldots$. The interval, $[a,b]$, is a maximal interval with positive slope. The following figure illustrates the first two steps of construction (Figure 1 in [4] for $\alpha = \frac{1}{2}$).



Define the limit function:

$$\Psi(x) = \lim_{i \to \infty} \psi_i(x),\ \forall x.$$

Basu et al. first prove that Ψ is a facet, then show that it is continuous but not piecewise linear, thus dispelling the myth. A key to showing the validity of ψ_i is that it is subadditive. The key to showing Ψ is a facet is the

Interval lemma. Let $w : \mathbb{R} \to \mathbb{R}$ be a function bounded on every bounded interval. Let $U = [u_1, u_2]$ and $V = [v_1, v_2]$ for any $u_1 < u_2$ and $v_1 < v_2$. If $w(u) + w(v) = w(u + v)$ for all $u \in U, v \in V$, there exists $c \in \mathbb{R}$ such that

$$\left.\begin{array}{ll} w(u) & = w(u_1) + c(u - u_1) \\ w(v) & = w(v_1) + c(v - v_1) \\ w(u+v) & = w(u_1 + v_1) + c(u + v - u_1 - v_1) \end{array}\right\} \quad \forall u \in U,\ v \in V.$$

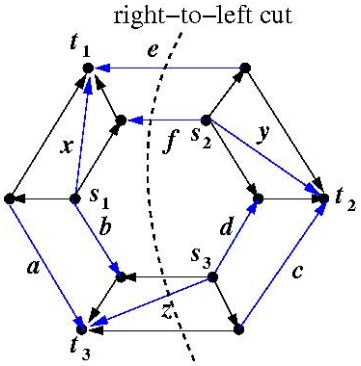
A key to establishing that Ψ is continuous is that $\{\psi_i(x)\}$ is a Cauchy sequence and thus converges. In fact, $\{\psi_i\}$ converges uniformly to Ψ, which implies Ψ is continuous.

To show that Ψ is not piecewise linear, define S_i as the subset of $[0, 1]$ for which ψ_i has negative slope. Basu et al. prove that S_i is the union of 2^i open intervals. Define $S = \cup_{i=0}^{\infty} S_i$, which is the set of points for which Ψ has negative slope. Since S is open, by showing S is dense in $[0, 1]$, Basu et al. complete the argument by applying contradiction, as follows. If Ψ is piecewise linear, there must exist $\delta > 0$ for which Ψ is linear on $(0, \delta)$. Since S is dense, it must contain a point in $(0, \delta)$. Because $\Psi(0) = 0$, we reach the contradiction that $\Psi(x) < 0$ for $x \in (0, \delta)$.

IP Myth 8. *Given a digraph and a subset of arcs that intersects each cut with k arcs, there exists a k-partition that covers the set of cuts.*

Counterexample. Schrijver[56] provides the following for $k = 2$.

The digraph on the right has three sources, s_1, s_2, s_3 and three sinks, t_1, t_2, t_3. $C = \{a, b, c, d, e, f, x, y, z\}$ intersects each (directed) cut at least twice. However, if $C = C_1 \cup C_2$, C_1 and C_2 must contain exactly one or two arcs incident with a source or sink. Further, for either $i = 1$ or $i = 2$, we must have $x, y \in C_i$ and $z \notin C_i$. Without loss in generality, assume C_1 satisfies these conditions. Then, C_1 does not intersect the cut indicated in the figure: arcs going from right to left. (Note that $e, f \in C_2$.)

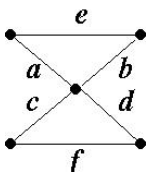


IP Myth 9. new *Every point [line] cover contains a minimum point [line] cover.* next new ▷

This illustrates that a *minim**um*** cover is not the same as a *minim**al*** cover.

Counterexample. Capobianco and Molluzzo[1] provide the following. Let G be any star with $L > 1$ leaves. The leaves comprise a point cover (which is minimal), but the only minimum cover is the center (which is a singleton).

Consider the graph shown on the right. A minimal line cover is $\{a, b, c, d\}$, but it does not contain any minimum line cover, such as $\{a, e, f\}$.



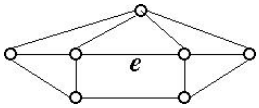
Capobianco and Molluzzo provide more counterexamples for covering properties. All pertain to the difference between minimum and minimal — equivalently, to the failure of greedy algorithms.

IP Myth 10. **new** *The chromatic number of a connected graph decreases if any vertex is removed if, and only if, it decreases if any edge is removed.* **next new** ▷

The chromatic number of graph G is denoted $\chi(G)$. A graph is χ-*minimal* if $\chi(G-e) < \chi(G)\,\forall e$; it is χ-*critical* if $\chi(G-v) < \chi(G)\,\forall v$. The myth asserts χ-minimal $\leftrightarrow$ χ-critical. It is true that χ-minimal $\rightarrow$ χ-critical, but Harary[29] shows that the converse is not true.

Counterexample.

The graph on the right has $\chi(G) = 4$. It is χ-critical because $\chi(G-v) = 3\,\forall v$, but it is not χ-minimal because $\chi(G-e) = 4$ for the particular edge labelled e.



IP Myth 11. *If there exists an optimal solution to an ILP in standard form, there exists an optimal solution with at most $\phi(m)$ positive values, where ϕ is a function of m (does not depend upon n).*

This was suggested to me by H.P. Williams, who consulted with Les Trotter.

For LP, this is the classic result that any optimal solution can be reduced to a basic solution that is optimal (so $\phi(m) = m$). The reduction process in that proof uses Carathéodory's theorem, which is shown in IP Myth 6 not to extend to integer vectors.

Counterexample. Let $p_1, \ldots, p_n$ be n distinct primes, and define $a_j = \prod_{i \neq j} p_i$. Consider the ILP:

$$\max\ cx : x \in \mathbb{Z}^n_+,\ \textstyle\sum_{j=1}^n a_j x_j = \sum_{j=1}^n a_j.$$

The only feasible solution is $x = \mathbf{1}$, so the number of positive variables is n.

IP Myth 12. *If some activities in an LP have a fixed charge, a valid MILP model is to introduce a binary variable, z, for each such activity and include constraints of the form, $0 \leq x \leq Uz$, where U is a given or derived upper bound on x. The fixed charge, K, enters the objective with the linear term Kz.*

The model is $\min\{f(x) + Kz : x \in X,\ z_j \in \{0,1\}$ and $0 \leq x_j \leq z_j U_j$ for $j \in J\}$, where J is the set of variables with fixed charge. The idea is that $z_j = 0$ forces $x_j = 0$, whereas $z_j = 1$ presents no additional constraint on x_j, and allows $x_j > 0$, in which case it incurs the fixed charge. The issue arises when $K < 0$, sometimes called a *fixed benefit*.

Counterexample. Let $K = -1$ in the following: $\min\ 5x - z : 0 \leq x \leq 10z$. The optimum sets $z = 1$, but $x = 0$, contrary to what is intended.

This is an example of the *MIP-Representable problem*, introduced by Meyer[46] and advanced by Jeroslow and Lowe[36]. For a fixed charge, the minimization renders $z = 1$ as an optimal binary value if $x > 0$ is optimal. For a fixed benefit, however, the minimization could render $z = 1$ with $x = 0$, thus not representing the problem correctly.

Ed Klotz points out another problem, using software with imperfect arithmetic. Suppose there is no a priori upper bound, and you use a "big-M" for the constraint: $0 \le x \le Mz$. If M is chosen large enough that $x \le M$ is redundant, the model is theoretically correct (for $K > 0$). However, the integrality tolerance allows $z = \tau$ to be considered integer-valued (CPLEX® uses $\tau = 10^{-5}$). Suppose you set $M = 10^9$. Then, the solver can set $x = 100$ and $z = 100/10^9 = 10^{-7} < \tau$, thus allowing $x > 0$ with a net fixed-charge of only $K \times 10^{-7}$. This suggests choosing the value of big-M with great care, taking the integrality tolerance into consideration. Other simple resolutions include computing individual bounds for x_j, perhaps using problem information.

IP Myth 13. *If an ILP has an unbounded LPR, the ILP is also unbounded.*

The following counterexample is due to Byrd, Goldman and Heller[9], based on the work of Meyer[45], who showed this cannot happen with rational data and a feasible ILP.

Counterexample. $\max x_1 : x \ge 0,\ x \in \mathbb{Z}^4,\ x_3 - \sqrt{2}\,(x_1 - x_2) = 0,\ x_2 + x_4 = 1.$

The constraint set for the LP relaxation contains the ray, $\{(t, 0, t\sqrt{2}, 1) : t \ge 0\}$. Thus, the LPR is unbounded. The integer solutions, however, must have $x_1 = x_2$ in $\{0,1\}$ and $x_3 = 0$. Thus, the only feasible solutions to the ILP are (0,0,0,1) and (1,1,0,0).

Ed Klotz points out that the IP can be bounded even with rational data if you allow type 2 SOS declarations, as in CPLEX.

Counterexample. $\max\ x_1 : x \ge 0,\ x_2 \le 1,\ x_3 - 1.41421x_1 + 1.41421x_2 = 0,$

```
SOS
S2:: x_1: 1 x_2: 2 x_3 : 3
End
```

In the LP relaxation, the ray $(t, 0, 1.41421t)$ is feasible. However, the SOS requirement allows only two consecutive variables in the SOS set to take on nonzero values, so it cuts off this unbounded direction when enforced. As a result, the MIP has a bounded, optimal solution, $x = (1, 1, 0)$.

IP Myth 14. *In a recourse model of a stochastic MILP, the duality gap tends to zero as the number of scenarios tends to infinity.*

This is true under special situations, and there is some intuition that it is true in general — that is, a "law of large numbers."

Counterexample. Carøe and Schultz[11] provide the following 2-stage recourse:

$$\min\ 3x - \frac{2}{n}\sum_{j=1}^{n} y_j : 0 \le x \le 1,\ y \in \{0,\,1\}^n,\ x - \tfrac{1}{2}y_j \ge b_j,\ \text{for } j = 1, \ldots, n,$$

where n is the number of scenarios, and y_j is the recourse variable if scenario j prevails. The probability that scenario j prevails is $\frac{1}{n}$, with associated objective coefficient of 2 and requirement:

$$b_j = \begin{cases} \frac{1}{64} & \text{if } j \text{ is even;} \\ \frac{15}{64} & \text{if } j \text{ is odd.} \end{cases}$$

The duality gap equals the difference between the optimal integer value and its LPR, which Carøe and Schultz prove is at least $\frac{5}{32}$, independent of n.

A feasible solution to the LPR is $x = \frac{1}{2} + \frac{1}{64}$ and

$$y_j = \begin{cases} 1 & \text{if } j \text{ is even;} \\ \frac{1}{2} & \text{if } j \text{ is odd.} \end{cases}$$

For convenience, henceforth assume n is even. The optimal LPR solution thus satisfies $z^* \leq 3\left(\frac{1}{2} + \frac{1}{64}\right) - 2\left(\frac{3}{4}\right) = \frac{3}{64}$.

To bound the MILP solution from below, first consider $y = 0$. This implies $x \geq \frac{15}{64}$, so $z^0 \geq \frac{45}{64}$. Second, consider $y_j = 1$ for some odd j. This implies $x \geq \frac{1}{2} + \frac{15}{64}$, which yields the bound $z^0 \geq 3\left(\frac{1}{2} + \frac{15}{64}\right) - 2 = \frac{13}{64}$. Finally, if $y_j = 0$ for all odd j and $y_j = 1$ for some even j, $x \geq \frac{1}{2} + \frac{1}{64}$. This implies $z^0 \geq \frac{3}{2} + \frac{3}{64} - 1 = \frac{35}{64}$. Putting these three cases together, we conclude

$$z^0 \geq \min\left\{\frac{45}{64}, \frac{13}{64}, \frac{35}{64}\right\} = \frac{13}{64}.$$

Hence, the duality gap satisfies:

$$z^0 - z^* \geq \frac{13}{64} - \frac{3}{64} = \frac{5}{32}.$$

Also see Sen, Higle, and Birge[57].

IP Myth 15. *If an ILP has an unbounded LPR, the ILP is feasible.*

This was motivated by a question from Marbelly Davila.

A polyhedron is unbounded if it contains a feasible half-line — that is, $\{x^0 + th : t \geq 0\} \subseteq P$, where $x^0 \in P$ and $h \neq 0$.

Counterexample. Let $P = \{(x, y) : x \geq \frac{3}{4},\ y \geq \frac{1}{2},\ x - y = \frac{1}{4}\} = \left\{(\frac{3}{4}, \frac{1}{2}) + t(1, 1) : t \geq 0\right\}$.

This does not contain any integer point since

$$(x, y) = (\tfrac{3}{4} + t, \tfrac{1}{2} + t) \to x - y = \tfrac{1}{4}.$$

IP Myth 16. *Alternative optima correspond to equally-good problem solutions.*

The myth assumes that the problem solution space is the same as the model solution space.

Counterexample. Consider the classical model for the TSP with n cities:

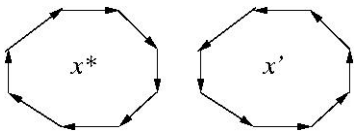
$$\min \textstyle\sum_{i,j} c_{ij} x_{ij} : x \in \{0, 1\}^{n \times n}$$
$$\textstyle\sum_i x_{ij} = 1,\ \forall j,\ \ \sum_j x_{ij} = 1,\ \forall i$$
$$\textstyle\sum_{(i,j) \in S} x_{ij} \leq |S| - 1,\ \forall S : \emptyset \neq S \subset \{1, \ldots, n\}.$$

The meaning of $x_{ij} = 1$ is that the tour visits city j right after city i. (The last constraint set is *subtour elimination*.)

Suppose c is symmetric. Then, if x^* is optimal, an alternative optimum is obtained by reversing the tour:

$$x'_{ij} = \begin{cases} 1 & \text{if } x^*_{ji} = 1 \\ 0 & \text{otherwise.} \end{cases}$$

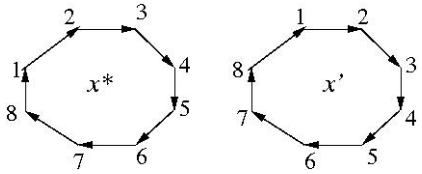
Thus, the model has alternative optima, but they do not correspond to different tours. For example, if x^* yields the tour $(1, 2, \ldots, n, 1)$, x' yields the tour $(1, n, n-2, \ldots, 2, 1)$.



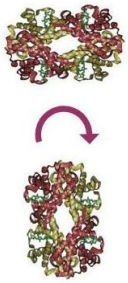
A similar model is with $x_{ik} = 1$ if i is the k^{th} city visited. If x^* is optimal, an alternative optimum is found by changing the home city:

$$x'_{ik} = \begin{cases} 1 & \text{if } x^*_{i\,(k+1) \bmod n} = 1 \\ 0 & \text{otherwise.} \end{cases}$$

The model has alternative optima, but they do not correspond to different tours. If x^* yields the tour $(1, 2, \ldots, n, 1)$, x' yields the tour $(2, 3, \ldots, n, 1, 2)$.



(Node numbers are the times visited.)



Many combinatorial optimization problems have the *symmetry problem*: graph coloring (swap colors 1 and 2 in the model solution) and protein fold prediction — in particular, over a lattice (perform rigid transformation, such as rotation). This is addressed by what is called *symmetry exclusion*. The figure on the right illustrates alternative model solutions (that is, coordinate assignments), where the second assignment is simply a 90° rotation of the first. Biologists see these as the same protein fold; changing their orientation in space does not change the protein's structure.

Opportunity Knocks

The distinction between true alternative optima and simply different representations of the same problem solution can be difficult to represent precisely, especially within one framework. There is practical benefit to doing so, such as eliminating portions of the search tree. Geometric problems, like protein folding, are amenable to symmetry exclusion[3], but not all symmetries may be eliminated a priori. Also see LP Myth 26.

IP Myth 17. *Suppose $x^*(t)$ is an optimal solution to the parametric ILP:*

$$\min\{cx : Ax \geq b + td,\ x \in \mathbb{Z}^n_+\},\ \textit{for } t \geq 0.$$

If $t' < t''$ and $x^*(t') = x^*(t'')$, then $x^*(t)$ is optimal for all $t \in [t', t'']$.

Wang and Horng[62] provide the following:

Counterexample. $\min 3x_1 + 2x_2 : x \in \mathbb{Z}^2_+$,

$$2x_2 \leq 9,\ 25x_1 + 10x_2 \geq 129 + 2t,\ 5x_1 + 20x_2 \geq 82 - 4t.$$

At $t = 0$ and $t = 1$, the optimal solution is $x^*(0) = x^*(1) = (4, 4)$; however, $x^*(\frac{1}{2}) = (4, 3)$.

IP Myth 18. *One can solve an ILP finitely by adding cuts of the form*

$$\textstyle\sum_{j \in N^k} x_j \geq 1$$

where N^k is the set of non-basic variables in the k^{th} LP relaxation having a fractional basic solution.

This is known as the *Dantzig cut*[12]. Gomory and Hoffman[24] showed that the Dantzig cuts need not converge finitely to an optimal solution with the following:

Counterexample.

$$\max\ z = 4x_1 + 3x_2 + 3x_3 : x \in \{0, 1\}^3,\ 3x_1 + 4x_2 + 4x_3 \leq 6.$$

The optimal integer solution is at $x = (1, 0, 0)$, with $z = 4$. Let s_j be the slack variable for the upper bound, $x_j + s_j = 1$, and let s_0 be the slack variable for the constraint, $3x_1 + 4x_2 + 4x_3 + s_0 = 6$. The LPR solution is at $x = (1, \frac{3}{4}, 0)$, with $z = 6\frac{1}{4}$ and $s = (0, 0, \frac{1}{4}, 1)$.

The following table gives five iterations, introducing a slack variable, t_k, when the k^{th} cut is constructed.

x_1	x_2	x_3	s_0	s_1	s_2	s_3	t_1	t_2	t_3	t_4	t_5	z	cut
1	$\frac{3}{4}$	0	0	0	$\frac{1}{4}$	1						$6\frac{1}{4}$	$x_3 + s_0 + s_1 - t_1 = 1$
$\frac{6}{7}$	0	$\frac{6}{7}$	0	$\frac{1}{7}$	1	$\frac{1}{7}$	0					6	$x_2 + s_0 + t_1 - t_2 = 1$
1	$\frac{2}{7}$	$\frac{2}{7}$	$\frac{5}{7}$	0	$\frac{5}{7}$	$\frac{2}{3}$	0	0				$5\frac{5}{7}$	$s_1 + t_1 + t_2 - t_3 = 1$
1	0	$\frac{1}{2}$	1	0	1	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	0			$5\frac{1}{2}$	$x_2 + s_1 + t_2 + t_3 - t_4 = 1$
$\frac{6}{13}$	$\frac{6}{13}$	$\frac{9}{13}$	0	$\frac{7}{13}$	$\frac{7}{13}$	$\frac{4}{13}$	$\frac{3}{13}$	$\frac{3}{13}$	0	0		$5\frac{7}{13}$	$s_0 + t_3 + t_4 - t_5 = 1$

The cuts keep going, never terminating finitely.

Bowman and Nemhauser[6] proved convergence of a modified Dantzig cut, which was improved by Rubin and Graves[54].

IP Myth 19. *A valid cut for the set covering problem is* $\sum_{i=1}^{n} x_i \geq \lfloor z^0 \rfloor + 1$*, where* x^0 *is a fractional LPR solution with objective value* z^0*.*

Counterexample. Rao[51] provides the following:

$$\begin{array}{rrrrrrrrrl} \min & x_1 & + & x_2 & + & x_3 & + & x_4 & : x \in \{0,1\}^4 \\ & x_1 & + & x_2 & & & & & \geq & 1 \\ & & & x_2 & + & x_3 & & & \geq & 1 \\ & & & & & x_3 & + & x_4 & \geq & 1 \\ & & & x_2 & & & + & x_4 & \geq & 1 \end{array}$$

An LPR solution is $x^0 = (\,1/2\,,\,1/2\,,\,1/2\,,\,1/2\,)$, and $z^0 = 2$. The cut is $x_1 + x_2 + x_3 + x_4 \geq 3$, but an optimal solution is $x^* = (0, 1, 1, 0)$, with $z^* = 2$.

IP Myth 20. *The Rudimentary Primal Algorithm (RPA) converges to an optimal solution.*

Ben-Israel and Charnes[5] introduced a variation of Gomory's algorithm, called the *Direct Algorithm*, but without a proof of finite convergence. Young[65] subsequently cited it as the *Rudimentary Primal Algorithm* (RPA).

Counterexample. Mathis[44] provides the following:

$$\begin{array}{rrrrrrrl} \max & x_1 - x_2 + 2x_3 : x \in \mathbb{Z}^3_+ & & & & & & \\ & 2x_1 & + & 4x_2 & - & x_3 & \leq & 20 \\ & -8x_1 & + & x_2 & + & 3x_3 & \leq & 10 \\ & 2x_1 & - & 9x_2 & + & 8x_3 & \leq & 6 \end{array}$$

Initial Tableau

	Column			
Row	0	1	2	3
0	0	−1	1	−2
1	0	−1	0	0
2	0	0	−1	0
3	0	0	0	−1
4	20	2	4	−1
5	10	−8	1	3
6	6	2	−9	8

Here is the sequence of tableaux:

Tableau 1

	Column			
Row	0	1	2	3
0	0	−1	−3	2
1	0	−1	0	0
2	0	0	−1	0
3	0	0	−2	1
4	20	2	2	1
5	10	−8	7	−3
6	6	2	7	−8
Cut	0	0	1	−2

Tableau 2

	Column			
Row	0	1	2	3
0	0	−1	3	−4
1	0	−1	0	0
2	0	0	1	−2
3	0	0	2	−3
4	20	2	−2	5
5	10	−8	−7	11
6	6	2	−7	6
Cut	0	−1	−1	1

Tableau 3

	Column			
Row	0	1	2	3
0	0	−5	−1	4
1	0	−1	0	0
2	0	−2	−1	2
3	0	−3	−1	3
4	20	7	3	−5
5	10	3	4	−11
6	6	8	−1	−6
Cut	0	1	−1	−1

Tableau 4

	Column			
Row	0	1	2	3
0	0	5	−6	−1
1	0	1	−1	−1
2	0	2	−3	0
3	0	3	−4	0
4	20	−7	10	2
5	10	−3	7	−8
6	6	−8	7	2
Cut	0	−2	1	0

Tableau 5

	Column			
Row	0	1	2	3
0	0	−7	6	−1
1	0	−1	1	−1
2	0	−4	3	0
3	0	−5	4	0
4	20	13	−10	2
5	10	11	−7	−8
6	6	6	−7	2
Cut	0	1	−1	−1

Tableau 6

	Column			
Row	0	1	2	3
0	0	7	−1	−8
1	0	1	0	−2
2	0	4	−1	−4
3	0	5	−1	−5
4	20	−13	3	15
5	10	−11	4	3
6	6	−6	−1	8
Cut	0	−1	−1	1

Tableau 7

	Column			
Row	0	1	2	3
0	0	−1	−9	8
1	0	−1	−2	2
2	0	0	−5	4
3	0	0	−6	5
4	20	2	18	−15
5	10	−8	7	−3
6	6	2	7	−8
Cut	0	0	1	−2

This cut is the same as tableau 1, and RPA continues to use only rows 5 and 6 as source rows for cuts. Since rows 5 and 6 are the same here as in tableau 1, RPA continues without termination. The sequence approaches $x = (1, 2, 3)$, but it does not reach it, even though the LPR objective value strictly decreases with each cut.

Valid variations of Gomory's algorithm are given by Young[64] and Glover[19].

IP Myth 21. **new** *The problem of finding $x \in \mathbb{Z}$ such that $Ax = b$, where $A \in \mathbb{Z}^{m \times n}$ and $b \in \mathbb{Z}^m$, is NP-complete.* ***next new*** ▷

My thanks to Heiko Vogel for pointing this out.

The key to this myth is that there are no bounds on x. (If we add $x \geq 0$, we have the standard form of a linear integer program, which is NP-complete.) This is the problem of solving linear *diophantine equations* for which Brown[7] provides simple, polynomial-time algorithms. Also, see Aardal, Hurkens, and Lenstra[1].

IP Myth 22. *For any 0-1 program with a single constraint, there exists a B&B algorithm that can determine if it is feasible in polynomial time.*

The following is due to Jeroslow[35]:

Counterexample. max $x_1 : x \in \{0, 1\}^n,\ 2x_1 + 2x_2 + \cdots + 2x_n = n.$

This is infeasible for n odd, but any Branch-and-Bound (B&B) algorithm (that is, with ***any*** rule for fixing values of fractional variables in the LP relaxation) must evaluate at least $2^{\lceil n/2 \rceil}$ nodes before it discovers (and certifies) that it is infeasible.

Ed Klotz points out that modern B&B algorithms are more broadly construed to include preprocessing, among other things, that would solve this example without exhaustive search. The counterexample does emphasize the need for such things. (This is another example of

how optimization software may use different conventions than in the theory — see LP Myth 16.)

IP Myth 23. *There is a strongly polynomial algorithm to maximize a separable concave function with one linear constraint.*

The mathematical program is

$$\max \sum_{j=1}^{n} f_j(x_j) : x \in \mathbb{Z}_+^n, \sum_{j=1}^{n} x_j = b,$$

where each f_j is concave on $\mathbb{R}_+$.

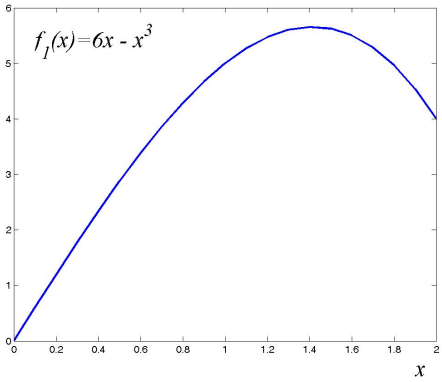
A strongly polynomial algorithm is one that is polynomial in the natural dimensions — that is, n, and not on the size of the data — in particular, b. (See LP Myth 42 for specific examples of how data values enter into the complexity.) Hochbaum[31] proves that the allocation problem has polynomial complexity, and that it is strongly polynomial when the objective function is linear. She proves the impossibility of the myth by showing that when the objective is nonlinear (and non-quadratic), the time complexity depends upon log b.

Hochbaum treats the integer and continuous cases jointly, pointing out that for the continuous case, we must define termination when $\left|\left|x^k - x^*\right|\right| \leq \varepsilon$ for some specified (but arbitrary) $\varepsilon > 0$. Otherwise, we can have no finite convergence, such as when x^* is irrational.

She provides a specific example:

$$\max\ 6x_1 - x_1^3 : x \in \mathbb{R}_+^2,\ x_1 + x_2 = 2.$$

The optimum is at $x^* = (\sqrt{2},\ 2 - \sqrt{2})$.



IP Myth 24. *An optimal schedule of jobs with deadlines on a single machine is given by the Ratio Rule.*

Thanks to Jan-Karel Lenstra for contributing this myth.

Smith[59] asserted this in the very early years, when simple rules were sought for special cases. He proposed four ideas:

1. *Shortest Processing Time (SPT) Rule*: schedule jobs in non-decreasing order of processing times.
2. If jobs are weighted, let t_j/w_j be the processing time over the (positive) weight. To

minimize total weighted completion time, schedule jobs in non-decreasing order of t_j/w_j. This is the *Ratio Rule*.

3. If each job j must be finished by a given deadline d_j, one minimizes total completion time by selecting from all jobs j that are eligible for the last position (that is, i for which $d_i \geq \sum_j t_j$) the one with largest t_i; put that job in the last position and repeat. If, at any point, there is no eligible job, there is no feasible schedule.
4. If each job j must be finished by a given deadline, one minimizes the total weighted completion time by combining the ideas of (3) and (4) — that is, by applying the Ratio Rule to the eligible job from the end of the schedule backwards.

Lenstra, Rinnooy Kan, and Brucker[43] prove that problem 4 is NP-hard, so the assertion is a myth (unless $P = NP$). Many thanks to Jan-Karel Lenstra for providing the following:

Counterexample. We have three jobs with process times: $t = (2, 1, 1)$, deadlines: $d = (4, 4, 3)$, and weights: $w = (7, 4, 1)$. The Ratio Rule yields the schedule $(2, 3, 1)$ with objective value 34. The optimal schedule is $(1, 3, 2)$ with objective value 33.

IP Myth 25. *The Dudek-Teuton job scheduling algorithm produces an optimal solution.*

Dudek and Teuton[13] presented an early algorithm purported to guarantee an optimal solution for an arbitrary problem with m machines and n jobs. That was before complexity theory, so we now know that their polynomial algorithm could not make such a guarantee (unless $P = NP$).

Counterexample. Karush[38] provides the following: Let $m = n = 3$ with duration times:

	Job		
Machine	1	2	3
1	3	22	20
2	22	20	14
3	2	20	18

The optimal sequence is 2-3-1 with total makespan 82. The Dudek-Teuton algorithm puts job 1 first, but the makespan of 1-2-3 is 83, and the makespan of 1-3-2 is 85.

IP Myth 26. *If jobs are prioritized by the slack per remaining number of operations, higher priority is given to jobs that have a greater number of remaining tasks for a given slack.*

The intent to give the greater priority as indicated matches our intuition of scheduling the jobs with many tasks remaining early in order to minimize the makespan. However, the slack value can be negative (that is, the job is late), thus reversing the intent. The slack at time τ is defined as:

$$s_i(\tau) = d_i - t_i - \tau,$$

where d_i is the due date of job i and t_i is its total processing time. The critical ratio used to prioritize the jobs is the *dynamic slack per remaining number of operations*: $s_i(\tau)/n$.

Counterexample. Adam and Surkis[2] provide the following:

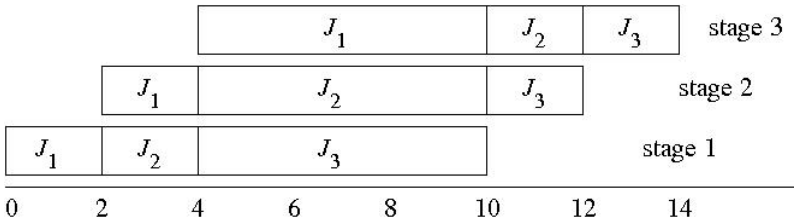
Job	Slack	# Remaining Operations	Priority Value	Priority
1	10	2	$10/2 = 5$	5
2	10	5	$10/5 = 2$	4
3	−10	2	$-10/2 = -5$	1
4	−10	5	$-10/5 = -2$	3
5	−12	4	$-12/4 = -3$	2

Kanet[37] provides clarification about the rationale for the anomaly.

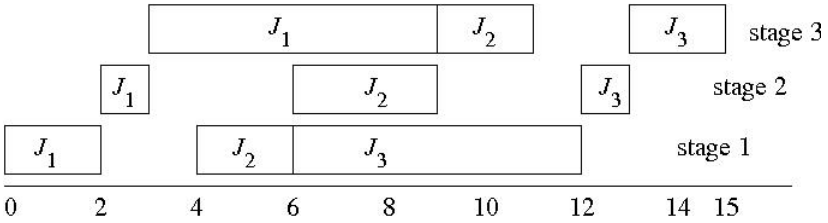
IP Myth 27. *A no-wait flow-shop's makespan cannot be worse by increasing the speed of some machines.*

Spieksma and Woeginger[60] provide the following:

Counterexample. Jobs 1, 2, and 3 are scheduled, each with three stages of operations as shown in the following figure. The minimum makespan is 14.



Now suppose the time spent on each machine in stage 2 is cut in half. Because the problem is a ***no-wait*** flow-shop, there cannot be any idle time between the processing of consecutive operations of the same job. Thus, the same job order yields a makespan of 15.



This is the minimum makespan for the new problem with speedup in stage 2. (The job orders 1-3-2 and 2-1-3 also have makespans of 15; job orders 2-3-1 and 3-1-2 have makespans of 17; and, job order 3-2-1 has a makespan of 19.)

Spieksma and Woeginger provide variations on the speedup and establish the following:

For every real number $r \geq 1$, there exists an instance of the no-wait flow-shop problem with minimum makespan C^* a speedup of processing time for some jobs and machines such that the makespan is at least $r\,C^*$.

IP Myth 28. *The worst-case solution quality of the First-Fit bin-packing algorithm equals the maximum feasible decomposition of the bin size.*

Let α denote the bin size, and let $L = \{a_1, \ldots, a_n\}$ be an ordered list of items with sizes $0 < s(a_i) \leq 1$. Let $OPT(L, \alpha)$ denote the minimum number of bins needed to pack the items, and let $FF(L, 1)$ denote the number of bins of size 1 needed for a First-Fit packing of L. The *worst-case solution quality* is the ratio:

$$R(\alpha) = \limsup_{N \to \infty} \max_{L} \left\{ \frac{FF(L,1)}{N} : OPT(L,\alpha) = N \right\}.$$

A *feasible decomposition* of α is an ordered sequence of integers $p_1 \leq p_2 \leq \ldots$, such that

$$\sum_i \frac{1}{p_i} = \alpha, \; p_1 \geq 2, \text{ and } \left|\{i : p_i > 2\}\right| \geq 2.$$

For example, for $\alpha = 1$, $p = (2, 3, 6)$ is a feasible decomposition. (Note that $p_2 > 2$ to satisfy the last condition.) Once we set $p_3 = 6$, we are done since

$$\frac{1}{p_1} + \frac{1}{p_2} + \frac{1}{p_3} = \frac{1}{2} + \frac{1}{3} + \frac{1}{6} = 1 = \alpha.$$

Let $\mathcal{P}(\alpha)$ equal the set of feasible decompositions of α, and define the *maximum feasible decomposition*:

$$W(\alpha) = \max_{p \in \mathcal{P}(\alpha)} \sum_i \frac{1}{p_i - 1}.$$

For example, $W(1) = 1 + \frac{1}{2} + \frac{1}{5} = \frac{17}{10}$.

The myth asserts $R(\alpha) = W(\alpha)$. This was conjectured by Garey, Graham, and Johnson[18], upon noticing its truth for the special case of a bin size of 1: $R(1) = W(1) = \frac{17}{10}$, a curious equation. Further, they found an efficient algorithm to compute $W(\alpha)$, so if the conjecture proved true, we could compute the worst-case solution quality without solving the worst-case.

Counterexample. Shearer[58] provides the following. Let $\alpha = \frac{1}{3} + \frac{1}{7} + \frac{1}{62} = \frac{641}{1302} = \frac{2564}{5208}$. Consider a list L of 120 items with sizes:

$$\begin{aligned} s(a_i) &= \tfrac{745}{5208}, \quad 1 \leq i \leq 30 \\ s(a_i) &= \tfrac{869}{5208}, \quad 31 \leq i \leq 60 \\ s(a_{2i-1}) &= \tfrac{1695}{5208}, \quad 31 \leq i \leq 60 \\ s(a_{2i}) &= \tfrac{1819}{5208}, \quad 31 \leq i \leq 60 \end{aligned}$$

The First-Fit algorithm packs L into 41 bins of size 1. The first five bins each contain 6 items of size $\frac{745}{5208}$; the next six bins each contain 5 items of size $\frac{869}{5208}$; and, the remaining 30 bins each contain 1 item of size $\frac{1695}{5208}$ and 1 item of size $\frac{1819}{5208}$. An optimal packing uses 60 bins of size α, so $R(\alpha) \geq \frac{41}{60}$. However, $W(\alpha) = \frac{1}{2} + \frac{1}{6} + \frac{1}{61} = \frac{2500}{3660} < \frac{41}{60}$.

IP Myth 29. **new** *Two TSP tours are adjacent only if there does not exist a distinct tour that contains their intersection and is contained in their union.* **next new** ▷

Adjacency is defined by the extreme points on the TSP polytope, which is contained in the *assignment polytope*:

$$X = \left\{ x \in \mathbb{R}_+^{n \times n} : \sum_i x_{ij} = 1, \forall j; \sum_j x_{ij} = 1, \forall i \right\}.$$

For this myth, let $x_{ij} = 1$ if city j follows city i in the tour (there are other TSP assignment models). Let $\text{ext}(X)$ denote the extreme points of X, which are the (binary-valued) permutation matrices. An extreme point is a tour if it corresponds to an arc-set of the form $t = \{(1, j_1), (j_1, j_2), \ldots, (j_{n-1}, 1)\}$ such that $j_1 j_2 \ldots j_{n-1}$ is a permutation of $\{2, \ldots, n\}$ (where the home city is fixed at 1). Then, its corresponding extreme point in X is

$$x_{ij} = \begin{cases} 1 & \text{if } (i,j) \in t; \\ 0 & \text{otherwise.} \end{cases}$$

Other assignments form *subtours* — an extreme case is the diagonal assignment, $X = I$. Restricting assignments to correspond to tours defines the *TSP polytope*:

$$T = \text{convh}\big(\{x \in \text{ext}(X) : x \leftrightarrow \text{tour.}\}\big).$$

Let t_1, t_2 be two tours that are adjacent extreme points of T. The condition of interest is the non-existence of a tour $t_3 \neq t_1, t_2$ such that $t_1 \cap t_2 \subset t_3 \subset t_1 \cup t_2$. The myth asserts that this condition is implied by the polyhedral definition of adjacency: $T = \alpha T_1 + (1-\alpha) T_2$ for $\alpha \in (0,1)$ implies T cannot be represented by any other convex combination of $\text{ext}(T)$. (Equivalently, the line segment, $[T_1, T_2]$, is an edge of T.)

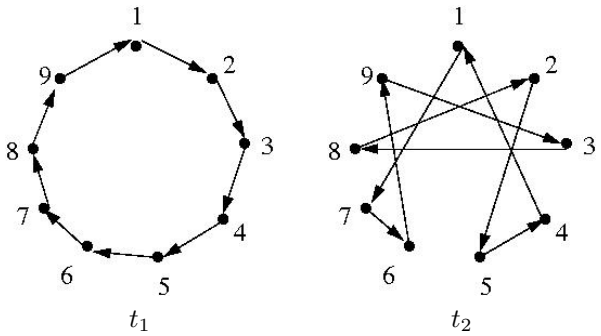
Let $T_{ij} = \{t \in \text{ext}(T) : t_1 \cap t_2 \subset t \subset t_1 \cup t_2\}$. The myth asserts that

$$(T_1 \text{ adjacent to } T_2) \rightarrow (T_{12} \setminus \{T_1, T_2\} = \emptyset). \tag{IP.7}$$

The intuition is that we can use tours in $T_{12} \setminus \{T_1, T_2\}$ to provide an alternative representation for $T \in (T_1, T_2)$, so T_1, T_2 could not be adjacent.

Murty[47] shows that the condition is sufficient, but necessity fails, as shown by Rao[52] with the following 9-city example.

Counterexample.



Rao shows that there are three tours that satisfy the condition:

$$t_3 = \{(1,7),(7,8),(8,2),(2,5),(5,6),(6,9),(9,3),(3,4),(4,1)\}$$
$$t_4 = \{(1,7),(7,8),(8,2),(2,3),(3,4),(4,5),(5,6),(6,9),(9,1)\}$$
$$t_5 = \{(1,2),(2,5),(5,6),(6,7),(7,8),(8,9),(9,3),(3,4),(4,1)\}$$

It is easy to verify that $T_{12} = \{t_3, t_4, t_5\}$. Rao notes that arcs $(3,8)$, $(7,6)$, and $(5,4)$ are in t_2, but not in t_3, t_4, t_5. Hence, each point in the open line segment, (T_1, T_2) must be uniquely determined by $\alpha T_1 + (1-\alpha)T_2$ for some $\alpha \in (0,1)$ — that is, it cannot be represented as another convex combination of the extreme points, so t_1, t_2 are adjacent extreme points that do not satisfy the condition.

Rao proceeds to derive a necessary condition and a sufficient condition for two tours to be non-adjacent. He proves that neither is both necessary and sufficient.

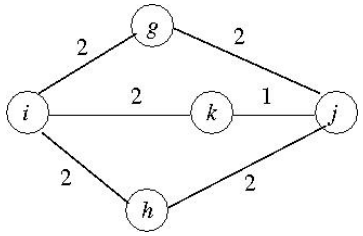
Also see Papadimtriou[50] for how the non-adjacency property renders insight into the computational complexity of TSP, and see Heller[30] for other "neighbor" relations,

IP Myth 30. *Suppose the edge weights satisfy the triangle inequality in a general routing problem. Consider required nodes i, j, k such that $(i,j) \notin E$ and $\big[(k,\ell) \in E \leftrightarrow \ell \in \{i,j\}\big]$. Then, the required nodes can be replaced by one required edge (i,j) with weight $w_{ij} = w_{ik} + w_{kj}$.*

Let $G = [N, E]$ be an undirected graph with edge weights $w \geq 0$. The *General Routing Problem* (GRP) is to find a tour with minimum total weight that contains a specified subset of nodes, $\widehat{N}$, and a specified subset of edges, $\widehat{E}$. (Note that this specializes to the TSP if $\widehat{N} = N$ and $\widehat{E} = \emptyset$ and to the Chinese Postman Problem if $\widehat{N} = \emptyset$ and $\widehat{E} = E$.) The myth assumes $w_{ij} \leq w_{ik} + w_{kj}$.

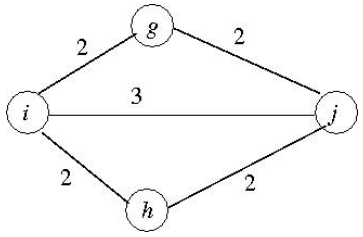
Orloff[48] introduced the reduction rule with the intuition that the added edge represents the path $i \to k \to j$. Lenstra and Rinnooy Kan[42] provide the following:

Counterexample. Let $\widehat{N} = N$ and $\widehat{E} = \emptyset$ in the following graph.



An optimal tour is
$k \to j \to g \to i \to h \to j \to k$
with total weight = 10.

Applying the reduction, the new GRP has $\widehat{N} = \{g, h\}$ and $\widehat{E} = \{(i,j)\}$ with $w_{ij} = w_{ik} + w_{kj}$.



An optimal tour is
$i \to j \to g \to i \to h \to j \to i$
with total weight = 14.

(See Orloff's rejoiner[49] and [42] for the merit of using the reduction rule as a heuristic.)

IP Myth 31. *Increasing the optimality tolerance reduces the generated search tree of B&B.*

Exact B&B closes a node when $\widehat{f} \geq z$, where $\widehat{f}$ is the estimate (lower bound) of the minimum at a node and z is the best solution obtained so far. A node would close earlier if we relax this to

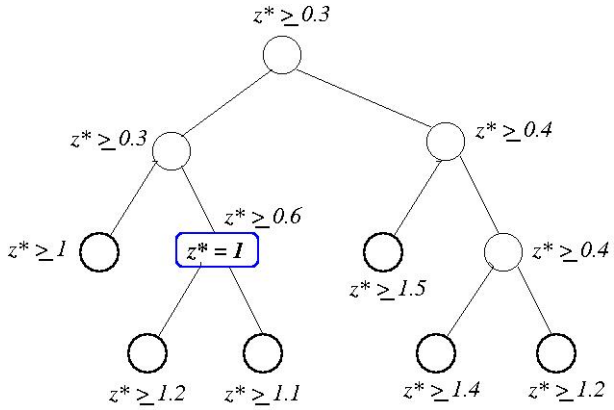
$$\widehat{f} \geq z - \varepsilon,$$

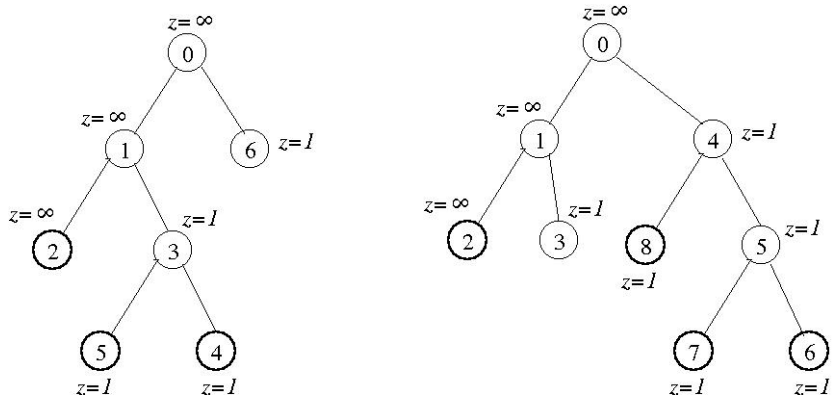
where $\varepsilon > 0$ is a cutoff tolerance. At termination, the algorithm produces a near-optimal solution (assuming an optimum exists) in the sense that

$$z \leq z^* + \varepsilon\,.$$

The myth asserts that using a positive tolerance results in a reduction in the total number of nodes generated before termination.

Counterexample. Ibaraki[33] provides the following:





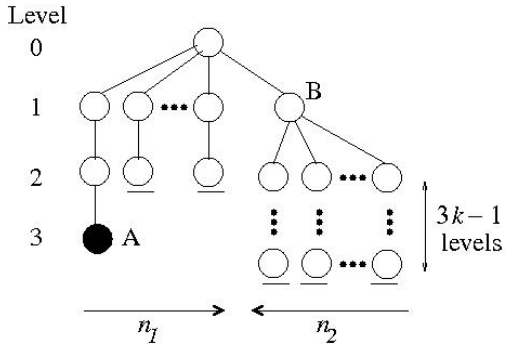
Exact ($\varepsilon = 0$) $\Rightarrow$ 6 nodes $\varepsilon = 0.5 \Rightarrow$ 8 nodes

Ibaraki points out that, as in the counterexample, exact B&B may close a node that would have generated a node that closes all other nodes. He proceeds to demonstrate that the size of the search tree could increase with an increase in ε. He also considers relative-tolerance cutoffs.

IP Myth 32. *Increasing the number of processors in a parallel B&B algorithm, reduces the generated search tree.*

Lai and Sahni[41] measure performance by the number of iterations, $I(n)$, for n processors. An iteration of an n-processor model with N open nodes expands $\min\{n, N\}$ nodes. Assuming maximization, each node is evaluated by computing an upper bound (such as with LPR). Those nodes that are feasible and have an upper bound that is greater than the current best value enter the pool; those that are infeasible or cannot have a better objective value are discarded. Lai and Sahni provide the following:

Counterexample. Assume that the bound of each node is the optimum value (but not confirmed as an optimal solution value). The following shows the state tree. The n_1-processor model selects the left portion, resulting in reaching solution node A in 3 iterations (at which point node B is closed without expansion). The n_2-processor model selects the right portion, expanding the sub-tree rooted at node B for $3k-1$ more levels before closing those leaves and finally evaluating node A.



They use this construction to prove:

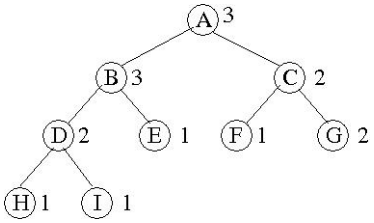
Let $n_1 < n_2$. For every $k > 0$, there exists a problem instance such that $k\,I(n_1) < I(n_2)$.

The construction in the state tree has $I(n_2) = 3k+1 = k\,I(n_1)+1 > k\,I(n_1)$. They also prove that this cannot happen if the bound is not the optimum value (which allows node B and the right-tree expansion to be candidate selections that are selected before node A).

IP Myth 33. *In B&B it is best to branch from the largest upper bound (for maximization).*

Fox et al.[16] provide the following:

Counterexample. The numbers next to each node in the following search tree are the upper bounds. Assume node G contains the maximum whose value is 2.



The *largest-upper-bound* (LUB) branching rule searches the nodes in one of the orders:

1. A, B, C, D, E, F, G (H & I not generated)
2. A, B, D, C, E, H, I, F, G

The particular order depends upon the expansion rule and how ties are broken. After B is expanded with children D & E, order 1 uses breadth-first search and chooses C; order 2 uses depth-first search and chooses D. Order 1 is better because it searches fewer nodes, but order 2 could be the one generated by the LUB branching rule.

A key to whether LUB is in some sense an optimal branching rule partly depends upon how ties are broken and the order in which the siblings are expanded. If the right-child is expanded first (among those with the same upper bound), the orders become:

1'. A, B, E, D, C, G

2'. A, B, E, D, I, H, C, G

Order 1' checks only 6 nodes, which is better than the left-child order of expansion.

In any case, the shortest path to the solution node is A, C, G, which does not follow the LUB rule. Node B must still be expanded to confirm optimality at node G, so the full sequence is 6 nodes: A, C, G, B, D, E (or E, D).

One alternative to LUB is to branch on the node with the *least ambiguity* — that is, fewest binary variables that are not fixed[27]. The rhetoric for this choice is that we can reach closure quickly, and a smart implementation computes look-ahead implications, generally arising from logical conditions in the model. For example, selecting one project may force other projects to be rejected, scheduling some job may force other schedule assignments, and so on. Thus, suppose we are given two nodes with the following properties: node A has LPR bound 100 and 75 binary variables that have not been fixed, of which 20 are fractional; node B has LPR bound 101 and 10 variables that have not been fixed, one of which is fractional. The LUB rule

expands A and ignores the other information; the least-ambiguity rule expands B and ignores the bound. (Hybrid rules use multiple criteria, of which these are two.)

Ties for node selection, whether with LUB or not, do occur in practice, partly because the underlying problem has alternative optima, and partly due to a naive modeler ignoring symmetries. For example, in graph coloring, let $x_{ij} = 1$ if we color node i with color j. For any solution, we can swap colors: x' represents an equivalent coloring as x, but in the model $x'_{i,\text{blue}} = x_{i,\text{green}}$ and $x'_{i,\text{green}} = x_{i,\text{blue}}$. Thus, in the model, these are alternative solutions since $x \neq x'$, and they have the same objective values, so if one does not add "symmetry exclusion constraints," ties are inevitable.

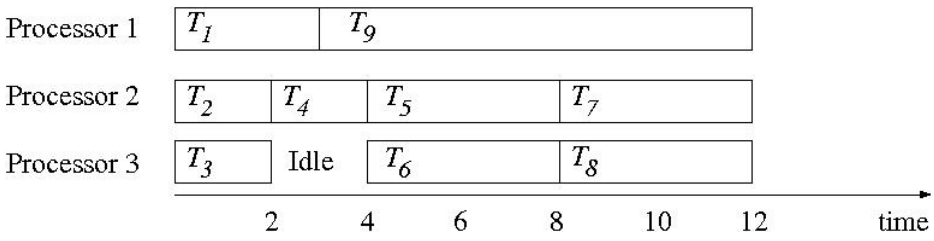
IP Background — Parallel Computation of a Schedule

In IP Myths 34–36 suppose we have n identical processors to perform computations in parallel. Tasks are presented at once with known precedence relations: $T_i \prec T_j$ means task T_i must be finished before task T_j can start. The order of the tasks is given by the list $L = \{T_{i_1}, \ldots, T_{i_r}\}$, and the rule is that a processor takes the next task in L that is ready (that is, all predecessors are finished). The processor time to perform task T_i is denoted by t_i.

To illustrate, let $L = \{T_1, \ldots, T_9\}$ with associated process times, $t = (3, 2, 2, 2, 4, 4, 4, 4, 9)$. The precedence relations are:

$$T_1 \prec T_9,\ T_4 \prec T_5, T_6, T_7, T_8.$$

Here is the time line for three processors:



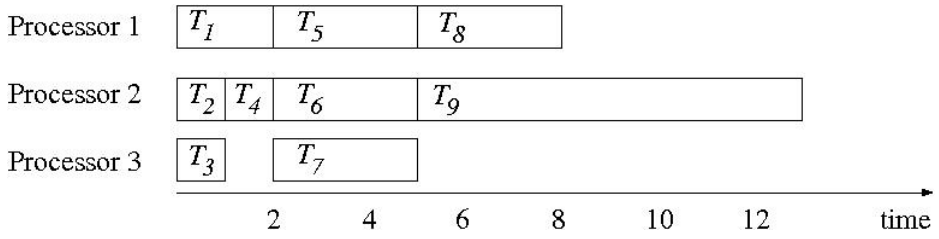
The makespan is 12.

IP Myths 34–36 are given by Graham[26], who also derives bounds on the makespan ratio for the improved system to the old, where "improved" is any combination of time reduction, added processors, precedence relaxation, and list-order rearrangement.

IP Myth 34. *If we reduce the computation time of each processor, the makespan cannot increase.*

Graham[26] provides the following:

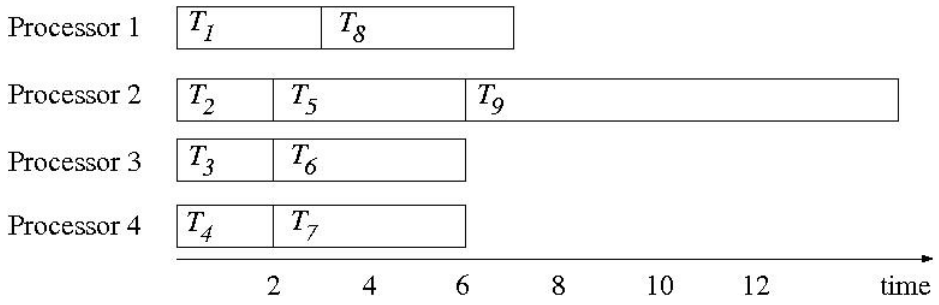
Counterexample. Change the previous example to have $t' = t - 1$. The result is a makespan of 13:



IP Myth 35. *If we add another processor, the makespan cannot increase.*

Graham[26] provides the following:

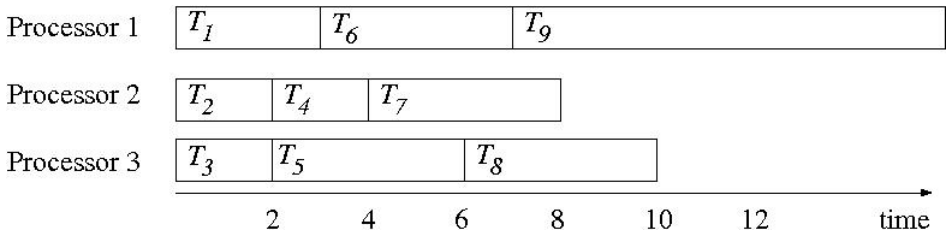
Counterexample. The fourth processor results in a makespan of 15:



IP Myth 36. *If we remove some precedence constraints, the makespan cannot increase.*

Graham[26] provides the following:

Counterexample. Remove the precedence constraints, $T_4 \prec T_5$ and $T_4 \prec T_6$. This results in a makespan of 16:



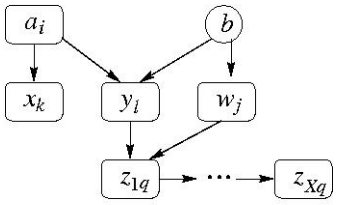
Tovey[61] extended Graham's example as follows.

Counterexample. Using Tovey's notation (nearly), the jobs are denoted: a_i, b, w_j, x_k, y_ℓ, z_{pq}, where $i = 1, \ldots, A$, $j = 1, \ldots, W$, $k = 1, \ldots, X$, $\ell = 1, \ldots, Y$, $p = 1, \ldots, X$, and $q = 1, \ldots, n+1$ (recall n = number of processors). The precedence relations are:

$$a_i \prec x_k,\ a_i \prec y_\ell \prec z_{1q},\ b \prec w_j \prec z_{1q},\ b \prec y_\ell,\ z_{pq} \prec z_{p+1,q}$$

for all i, j, k, ℓ, q. (In Graham's example, $A = 6$, $W = 1$, $Y = 6$, and $X = 4$.)

The figure on the right (taken from Tovey) shows the precedence relations.



In any optimal schedule with $n = 2$ processors, b must precede some a_i, but with $n = 3$ processors, all a_i must precede b. If all a_i must precede b, a schedule is not optimal for n processors if, and only if, $A + 1 \not\equiv 0 \mod n$. If b precedes some a_i, a schedule is not optimal for $n + 1$ processors if, and only if, $A \equiv 0 \mod (n + 1)$ and $W < n$.

In particular, consider $n = 2$ and Graham's dimensions: $a = (a_1, \ldots, a_6)$, $w = (w_1)$, $y = (y_1, \ldots, y_6)$, and $x = (x_1, \ldots, x_4)$. In this example, b does not precede any a_i in an optimal schedule. For those same dimensions, increasing n to 3, b must precede every a_i for the schedule to be optimal.

IP Myth 37. *Given jobs with unit time, an optimal schedule on n processors is obtained by assigning compatible jobs in a minimal partition.*

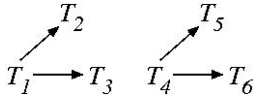
This refers to an early algorithm by Fujii, Kasami, and Ninomiya[17], which is valid for two processors and they conjectured extends to $n > 2$ processors.

Two jobs are *compatible* if neither must precede the other (that is, not adjacent in the transitive closure of the precedence graph). The algorithm is to form a minimum number of subsets of compatible jobs, such that each subset contains no more than n jobs. These are then assigned sequentially, and the minimum makespan is the number of subsets. For example, suppose $T_1 \prec T_2 \prec \cdots \prec T_N$. Then, there are no compatible jobs, and the subsets are $\{T_1\}, \ldots, \{T_N\}$, giving a minimum makespan of N, using only one processor (and having the other $n - 1$ processors idle). On the other hand, if the precedence relations are $T_1 \prec T_2 \prec \cdots \prec T_{\frac{N}{2}}$ and $T_{\frac{N}{2}+1} \prec T_2 \prec \cdots \prec T_N$ (with N even), then with two processors, the jobs can be partitioned into subsets $\{T_1, T_{\frac{N}{2}+1}\}$, $\{T_2, T_{\frac{N}{2}+2}\}, \ldots, \{T_{\frac{N}{2}}, T_N\}$. Then, the minimum makespan is $\frac{N}{2}$, obtained from the algorithm by assigning:

processor 1	T_1	T_2	$\cdots$	$T_{\frac{N}{2}}$
processor 2	$T_{\frac{N}{2}+1}$	$T_{\frac{N}{2}+2}$	$\cdots$	T_N

The issue is whether this is valid for $n > 2$ processors. In the above example, for $n = 3$ suppose $N = 3k$ and the precedence relations are $T_1 \prec \cdots \prec T_k$, $T_{k+1} \prec \cdots \prec T_{2k}$, $T_{2k+1} \prec \cdots \prec T_N$. Then, we can partition the jobs into k subsets, and assign the jobs to achieve the minimum makespan of $\frac{N}{3}$.

Counterexample. Kaufman[39] provides the following. Let $T_1 \prec T_2, T_3$ and $T_4 \prec T_5, T_6$. For 3 processors, the minimum makespan is 3.



processor 1	T_1	T_2	T_6
processor 2	T_4	T_3	
processor 3		T_5	

The algorithm, however, obtains the partition $\{\{T_1, T_5, T_6\}, \{T_4, T_2, T_3\}\}$, giving the incorrect minimum makespan of 2. The partition satisfies the properties: each subset has no more than 3 jobs, and they are compatible.

IP Background — Metaheuristics

A *metaheuristic* is a top-level general strategy that guides other heuristics to search for feasible solutions in domains where the task is NP-hard. Examples include genetic algorithms, simulated annealing, and tabu search. The *state* is a vector defined by the problem representation; often the state is a solution. A key to any metaheuristic is the definition of *neighborhood* of a state, denoted $\mathcal{N}(s)$. (Unlike the neighborhood defined in real analysis, we typically have $s \notin \mathcal{N}(s)$.) One example is removing and/or adding an object to a knapsack. Another example is replacing two arcs in a travelling salesman tour. A common neighborhood is complementing one binary value:

$$\mathcal{N}(x) = \cup_j \{x' : x'_i = x_i \text{ for } i \neq j,\ x'_j = 1 - x_j\}. \tag{IP.8}$$

Let f be the objective value (or some measure of fitness used in a metaheuristic), which we seek to maximize. The *depth* of a non-optimal feasible solution, x, is the minimum value $d(x)$, such that there exists a sequence $< x^0 = x, x^1, \ldots, x^k >$ that satisfies the following conditions:

1. x^i is feasible and $x^i \in \mathcal{N}(x^{i-1})$ for $i = 1, \ldots, k$
2. $f(x^k) > f(x^0)$.
3. $f(x^i) \geq f(x^0) + d(x)$ for $i = 1, \ldots, n$.

The depth of a problem instance P with respect to a neighborhood [and fitness function] is $d(P) = \max\{d(x) : x \in X\}$, where X is the set of feasible solutions.

IP Myth 38. *Computing the depth of a discrete optimization problem P with respect to a neighborhood is at least as hard as solving P.*

Woeginger[63] provides the following:

Counterexample. Let P be an instance of the Satisfiability Problem (SAT), which is NP-complete. Let x be a truth setting and $L(x)$ a logical expression whose truth value we seek. Let $f(x)$ be the truth value of $L(x)$ (that is, 1 if true; 0 if false), so we seek to maximize f over the 2^n binary values.

Define the neighborhood as in (IP.8). Then, the depth of any non-optimal feasible solution is 0 (with $f(x^i) = f(x^{i-1}) = 0$ for $i = 1, \ldots, k-1$ and $f(x^k) = 1$), so $d(P) = 0$. Further, the depth of x is trivial to compute.

IP Myth 39. *Computing the depth of a discrete optimization problem P with respect to a neighborhood is at most as hard as solving P.*

Woeginger[63] provides the following:

Counterexample. Let P be an instance of the Satisfiability Problem (SAT). Let x be a truth setting and $L(x)$ a logical expression whose truth value we seek. Let the state of the system be bit strings in $\{0,\,1\}^{n+2}$, where $s = (x, s_{n+1}, s_{n+2})$ and $f(s) = -c(s)$, where

$$\begin{array}{rl} s_{n+1} = s_{n+2} = 0 & \Rightarrow\ c(s) = 0 \\ s_{n+1} = s_{n+2} = 1 & \Rightarrow\ c(s) = 1 \\ s_{n+1} \neq s_{n+2},\ L(x) = 1 & \Rightarrow\ c(s) = 1 \\ s_{n+1} \neq s_{n+2},\ L(x) = 0 & \Rightarrow\ c(s) = 2 \end{array}$$

We seek to minimize c, and $\mathcal{N}(s)$ is defined to be the 1-bit flip (IP.8) plus s.

Woeginger proves: *If L is satisfiable, $d(P) = 0$. If L is not satisfiable, $d(P) = 1$.* This proves that determining the depth is NP-hard. Further, a global minimum is found simply by any output whose last two bits are zero. Thus, the myth is false (unless $P = NP$).

IP Myth 40. *In a metaheuristic search, it is better to start with a best-possible objective value, even if it is not the global optimum.*

Consider a binary IP and the neighborhood as (IP.8) plus complementing all ($x' = 1 - x$), if that is feasible.

Counterexample. Glover and Hao[21] provide the following:

$$\max\ nx_1 - \textstyle\sum_{j=2}^{n} x_j\ :\ x \in \{0,\,1\}^n,\ (n-1)x_1 - \sum_{j=2}^{n} x_j \leq 0.$$

The worst feasible solution is $x = (0, 1, 1, \ldots, 1)$. Complementing each x yields the global optimum in one iteration. Starting at some other feasible solution, such as $x = 0$, causes the search to re-visit $x = 0$ many times before reaching the worst solution (followed by the global maximum).

Opportunity Knocks

Is it possible to characterize IPs for which the Myth is true? See [21] for deeper analysis.

IP Myth 41. *For N sufficiently large, simulated annealing visits the global minimum within N iterations with probability 1.*

Fox[14, 15] provides the following:

Counterexample. Let $X = \{1, 2, 3\}$ and $f(X) = (1, 3, 0)$, so, $x^* = 3$ is the global minimum. The system state is the value of x, and the neighborhoods are: $N(1) = \{2\}$, $N(2) = \{1, 3\}$, and $N(3) = \{2\}$. The acceptance probability of an uphill move from x_1 to x_2 is given by:

$$P(X(k+1) = x_2 \,|\, X(k) = x_1) = e^{-\frac{f(x_2) - f(x_1)}{T_k}} = e^{-\frac{2}{T_k}},$$

where T_k is the temperature at iteration k. Once the state moves from x_1 to x_2, it then moves to x_3, the global minimum. So, not visiting x_3 is equivalent to remaining at x_1 forever. That probability is given by:

$$P(X(1) = X(2) = \cdots = X(k) = x_1 \,|\, X(0) = x_1) = \prod_{i=1}^{k} \left(1 - e^{-\frac{2}{T_i}}\right).$$

Thus, the system does not reach the global minimum with probability 1 within any *finite* number of iterations.

As the general theory goes, the example does converge to the global minimum asymptotically with probability 1. However, the expected number of iterations is infinite. Specifically, for a standard cooling schedule, Fox shows

$$\lim_{k \to \infty} (-k + \mathbf{E}[N \,|\, N > k]) = \infty.$$

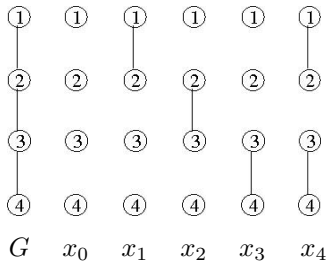
In words, the longer the search has been unsuccessful in reaching the global minimum, the longer the expected remaining time to reach it.

Fox provides variations that escape this difficulty (among others).

IP Myth 42. *In simulated annealing, it is always better to let the temperature decrease.*

Hajek and Sasaki[28] provide sufficient conditions for which no cooling temperature sequence is better than a constant temperature. They then show how the conditions apply to a matching problem, for which the following counterexample is a special case.

Counterexample. Let G be a simple path with 4 nodes, for which there are five matchings, denoted $x_0 = \emptyset,\ x_1 = \{(1,2)\},\ x_2 = \{(2,3)\},\ x_3 = \{(3,4)\},\ x_4 = \{(1,2),\ (3,4)\}$.



Let x be a matching, and let its neighborhood be any matching that differs by exactly one edge:

$$\begin{aligned}
\mathcal{N}(x_0) &= \{x_1, x_2, x_3\} \\
\mathcal{N}(x_1) &= \{x_0, x_4\} \\
\mathcal{N}(x_2) &= \{x_0\} \\
\mathcal{N}(x_3) &= \{x_0, x_4\} \\
\mathcal{N}(x_4) &= \{x_1, x_3\}.
\end{aligned}$$

The transition from x_i to x_j consists of two steps: (1) Select $x_j \in \mathcal{N}(x_i)$ with probability R_{ij}; (2) Accept x_j according to the following SA rule:

$$\mathbf{Pr}(X(k+1) = x_j \,|\, X(k) = x_i) = \begin{cases} 1 & \text{if } x_j \supset x_i; \\ e^{-\frac{1}{T_k}} & \text{if } x_j \subset x_i, \end{cases}$$

where T_k is the temperature. If x_j is not accepted, set $X(k+1) = x_i$.

Let each neighbor be equally-likely to be selected in step 1: $R_{ij} = \frac{1}{|\mathcal{N}(x_i)|}$. Then, the process is a Markov chain with the following transition probabilities:

$$Q(T_k) = \begin{array}{c} \begin{matrix} x_0 & x_1 & x_2 & x_3 & x_4 \end{matrix} \\ \begin{bmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{2}\,e^{-\frac{1}{T_k}} & \frac{1}{2}(1-e^{-\frac{1}{T_k}}) & 0 & 0 & \frac{1}{2} \\ e^{-\frac{1}{T_k}} & 0 & 1-e^{-\frac{1}{T_k}} & 0 & 0 \\ \frac{1}{2}\,e^{-\frac{1}{T_k}} & 0 & 0 & \frac{1}{2}(1-e^{-\frac{1}{T_k}}) & \frac{1}{2} \\ 0 & \frac{1}{2}\,e^{-\frac{1}{T_k}} & 0 & \frac{1}{2}\,e^{-\frac{1}{T_k}} & 1-e^{-\frac{1}{T_k}} \end{bmatrix} \end{array} \begin{matrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \end{matrix}$$

The issue is whether it is better to let $\{T_k\}$ decrease or remain constant.

For $T_k = \infty$, the search is completely random, and for $T_k = 0$ the local-maximum matching x_2 is an absorbing state.

$$Q(\infty) = \begin{bmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ 1 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix}. \quad Q(0) = \begin{bmatrix} 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Keeping the temperature constant at $T_k = \infty$ (or any value large enough to ensure acceptance), the system eventually reaches the global maximum, x_4. Whenever the system reaches the local maximum x_2, it moves to x_0. On the other hand, as $T_k \to 0$, the system could be absorbed at x_2. Thus, cooling is worse than the constant temperature.

Hajek and Sasaki conjecture the existence of other problem classes for which it is not optimal to cool the temperature.

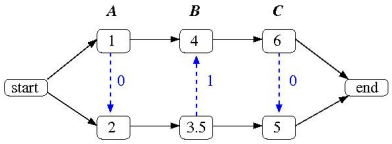
IP Myth 43. *Simulated annealing converges in probability to an optimal solution.*

Counterexample. Kolonko[40] provides the following job scheduling problem.

There are two jobs and three machines whose processing times are shown on the right. There are eight solutions, represented by the binary code of $0, \ldots, 7$. This is a 3-bit code that describes a precedence of the jobs on each machine.

	Job	
Machine	1	2
A	1	2
B	4	3.5
C	6	5

The left figure below shows how the bit values correspond to the job precedences on each machine — that is, the direction of the dotted arrows. The particular solution shown is $x^2 = 010$. The solutions are shown in the table, on the right.



i	0	1	2	3	4	5	6	7
x^i	000	001	010	011	100	101	110	111
Cost	16	19.5	21.5	17.5	18	21.5	20.5	16.5

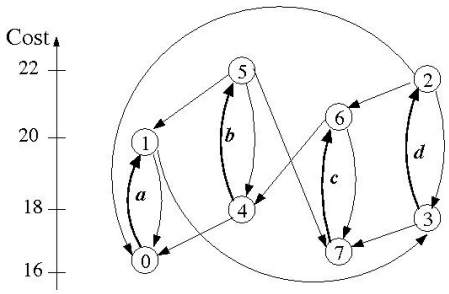
The optimal solution is x^0 with minimum cost 16.

The SA neighborhood is defined by (IP.8). Kolonko shows that the associated SA Markov process converges in probability to the suboptimal solution x^7.

The Markov state transition graph is shown on the right. Each of the eight states can move to a neighbor, which is the complement of one bit, with equal probability. The thick arcs, labelled a, b, c, d, are those leading to greater cost. They are accepted with the probabilities:

$$a = b = e^{-3.5/t},\ c = d = e^{-4/t}.$$

All downward transitions are accepted with probability 1 (because the cost decreases).



The steady-state probabilities (where $\pi P = \pi$) are given by: $\pi = \widehat{\pi}/\kappa$, where

$$\widehat{\pi}(x^0) = 3c(6 + b + 4d + bd), \qquad \widehat{\pi}(x^1) = 2ac(3 + b), (3 + 2d),$$
$$\widehat{\pi}(x^2) = 3abc(3 + d), \qquad \widehat{\pi}(x^3) = 6ac(3 + b),$$
$$\widehat{\pi}(x^4) = 6ac(3 + d), \qquad \widehat{\pi}(x^5) = 3abc(3 + d),$$
$$\widehat{\pi}(x^6) = 2ac(3 + 2b), (3 + d), \qquad \widehat{\pi}(x^7) = 3a(6 + 4b + d + bd),$$

and $\kappa = \sum_{i=0}^{7} \widehat{\pi}(x^i)$.

All values depend upon t (suppressed for notational convenience). As $t \to 0$, Kolonko shows $\lim_{t \to 0} C/a = 18$, from which he concludes

$$\lim_{t \to 0} \pi = (0, 0, 0, 0, 0, 0, 1).$$

Thus, SA converges with probability 1 to the suboptimal solution, x^7, and not to the minimum, x^0.

IP Myth 44. **new** *Simulated annealing converges more slowly than steepest descent when there is a unique optimum.* ***next new*** ▷

My thanks to Peter Salamon for suggesting this.

Conventional wisdom suggests that a purposeful algorithm, like steepest descent, is better than any form of random moves, except for the problem of converging to a local optimum. Simulated annealing (SA) is generally thought of as a way to avoid entrapment at a local optimum and move to a global optimum. Hence, the myth asserts that SA is not an algorithm of choice when there is only one minimum, thereby making local entrapment a non-issue. To dispel this myth, Hoffmann and Salamon[32] provide a family of examples.

Counterexample. This is yet another example to dispel the myth. Consider states, $X = \{0, 1, \ldots, n\}$, and objective function, $f(x) = x$. Define neighborhoods as

$$\mathcal{N}(x) = \{x - 1, n\} \text{ for } x = 1, \ldots, n-1,\ \mathcal{N}(0) = \mathcal{N}(n) = \{0\}.$$

There is only one minimum, namely at $x = 0$. Starting at $x = n - 1$, steepest descent goes through states $n - 2, n - 3, \ldots, 1$, then ends by going to 0.

Compare this to SA using a cooling temperature of $T = \infty$ (that is, accept any uphill change with probability 1) and the equally-likely neighborhood selection rule.

The underlying Markov process for $n = 5$ is given by the transition matrix on the right. The expected number of SA steps until absorption is $\sum_{i=2}^{\infty} i \left(\frac{1}{2}\right)^{i-1} = 3$.

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \\ 0 & 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Thus, although it is not certain that SA reaches the global minimum in a finite number of steps, it does get there faster on the average than steepest descent. In fact, since the expected number of steps is independent of n, and steepest descent always takes $n - 1$ steps, SA terminates relatively much faster as n gets large.

The property to be recognized is that, as in the non-monotone method in NLP Myth 36 (p. 139), a particular surface may have a landscape such that a 'shortcut' to the optimum requires the objective to worsen before it gets better. One may imagine a curved ridge or helix that attracts steepest descent to follow it in tiny steps of improvement, whereas SA has a probability of jumping through this. Such problems can be constructed that make the probability arbitrarily close to 1 so that SA reaches the solution before steepest descent. Specifically, the probability is $1 - \varepsilon$ if the helix pitch is ε. However, NLP and SA exploit the landscape structure differently. NLP specifically trades off monotonicity with curvature, *observable* from the iterates. SA does not observe a key property (like curvature) of the landscape, as it moves. In the example, we could use purely random selection: Choose x with probability $\frac{1}{n+1}$ for $x = 0, \ldots, n$. Then, the probability of visiting $x = 0$ at least once in N trials is $1 - \left(\frac{1}{n+1}\right)^N$, which rapidly approaches 1. The expected number of trials before reaching $x = 0$ is $2 + \frac{1}{n}$, which is less than the SA for all n, and it approaches 1 as n gets large. Thus, while we can demonstrate the principle of benefit for allowing an objective-worsening move, it remains to specify how this would be done without specific knowledge of the landscape. There does not appear to be observable statistics to guide this. At the very least, the neighborhood structure would have to be dynamic, using observed values to determine what states are candidates from the current one.

IP Myth 45. **new** *Using parallel processors to conduct multiple, independent random walks reduces the total computation time.* ***next new*** ▷

My thanks to Peter Salamon for suggesting this.

Following the approach by Salamon, Sibani, and Frost[55, Ch. 8], let $\{E^i(t)\}_{i=1}^N$ denote the

minimum energy of N random walks after t units of elapsed computer time. Then, define

$$E^*(t|N) = \min_{i=1,\ldots,N} E^i(t).$$

Let $F_t(\varepsilon)$ be the common Cumulative Distribution Function (cdf) of $E^i(t)$ — that is, $F_t(\varepsilon) = \mathbf{Pr}[E^i(t) \le \varepsilon]$. Since the random walks are independent, we have

$$G_N(\varepsilon, t) \stackrel{\text{def}}{=} \mathbf{Pr}[E^*(t \,|\, N) \le \varepsilon] = 1 - (1 - F_t(\varepsilon))^N.$$

For a discrete distribution, say $\varepsilon \in \{0, \ldots, E^{\max}\}$, the probability mass function is

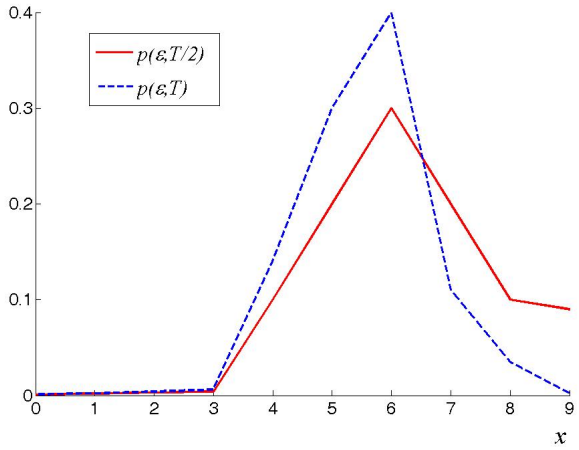
$$p_N(\varepsilon, t) = (1 - F_t(\varepsilon))^N - (1 - F_t(\varepsilon + 1))^N. \tag{IP.9}$$

Suppose we allocate T computer time. Then, we want to compare $\mathbf{E}[E^*(T \,|\, 1)]$, $\mathbf{E}[E^*(T/2 \,|\, 2)]$, $\ldots$, $\mathbf{E}[E^*(T/N \,|\, N)]$. The myth asserts $N > 1$ minimizes the expected energy found — that is, use N random walkers for T/N time units instead of just one processor for T time units. Formally, an *optimal ensemble size* is a solution to

$$\max_N \mathbf{E}[E^*(T/N \,|\, N)] : \; N \in \{1, \ldots, N^{\max}\}.$$

Counterexample. Salamon, Sibani, and Frost[55] provide the following (Example 8.1, p. 59). Consider $N^{\max} = 2$ and energy values in $\{0, 1, \ldots, 9\}$ with probabilities as follows:

ε	$p(\varepsilon, T/2)$	$p(\varepsilon, T)$
0	0.0010	0.0011
1	0.0020	0.0022
2	0.0030	0.0040
3	0.0040	0.0060
4	0.1000	0.1400
5	0.2000	0.3000
6	0.3000	0.4000
7	0.2000	0.1100
8	0.1000	0.0347
9	0.0900	0.0020



$$\mathbf{E}[E^*(T, 1)] = \sum_{\varepsilon=0}^{9} \varepsilon \, p(\varepsilon, T) = 5.554$$

$$\mathbf{E}[E^*(T/2, 2)] = \sum_{\varepsilon=0}^{9} \varepsilon \, p_2(\varepsilon, T/2) = 5.425,$$

where $p_2(\cdot, T/2)$ is the probability mass function (IP.9) for $N = 2$. Salamon et al.[55] give the numerical breakdown to derive

$$p_2(\cdot, T/2) = (0.002,\, 0.004,\, 0.006,\, 0.008,\, 0.188,\, 0.316,\, 0.324,\, 0.116,\, 0.028,\, 0.008).$$

This yields the lower expected value, 5.425, which thus shows that it is possible to do better with only one processor than to have two processors run independent random walks in half the time each.

Salamon et al. proceed to illustrate that there is eventually a law of diminishing returns with the number processors, even with other objective functions, such as minimizing the median (rather than mean) value. They provide an example where the optimal ensemble size is $N = 6 \ll N^{\max}$. In terms of general parallel-processing nomenclature, the speedup is super-linear for fewer than the optimal ensemble size, then it becomes sub-linear.

IP References

[1] K. Aardal, C. A. J. Hurkens, and A. K. Lenstra. Solving a system of diophantine equations with lower and upper bounds on the variables. *Mathematics of Operations Research*, 25(3):427–442, 2000.

[2] N. R. Adam and J. Surkis. Priority update intervals and anomalies in dynamic ratio type job shop scheduling rules. *Management Science*, 26(12):1227–1237, 1980.

[3] R. Backofen and S. Will. Excluding symmetries in constraint-based search. *Constraints*, 7(3–4):333–349, 2002.

[4] A. Basu, M. Conforti, G. Cornuéjols, and G. Zambeli. A counterexample to a conjecture of Gomory and Johnson. Technical report, Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA, 2009. To appear in *Mathematical Programming, Series A*.

[5] A. Ben-Israel and A. Charnes. On some problems of diophantine programming. *Cahiers du Centre d'Etudes de Recherche Opérationnelle*, 4:215–280, 1962.

[6] V. J. Bowman and G. L. Nemhauser. A finiteness proof for modified Dantzig cuts in integer programming. *Naval Research Logistics Quarterly*, 17:309–313, 1970.

[7] A. B. Brown. Linear diophantine equations. *Mathematics Magazine*, 31(4):215–220, 1958.

[8] W. Bruns, J. Gubaladze. M. Henk, A. Martin, and R. Weismantel. A counterexample to an integer analogue of Carathéodory's theorem. *Journal für die reine und angewandte Mathematik (Crelle's Journal)*, 510:179–185, 1999.

[9] R. H. Byrd, A. J. Goldman, and M. Heller. Recognizing unbounded integer programs. *Operations Research*, 35(1):140–142, 1987.

[10] M. Capobianco and J. C. Molluzzo. *Examples and Counterexamples in Graph Theory.* North-Holland, New York, NY, 1978.

[11] C. C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Operations Research Letters*, 24(1–2):37–45, 1999.

[12] G. B. Dantzig. Note on solving linear programs in integers. *Naval Research Logistics Quarterly*, 6(1):75–76, 1959.

[13] R. A. Dudek and O. F. Teuton, Jr. Development of m-stage decision rule for scheduling n jobs through m machines. *Operations Research*, 12(3):471–497, 1964.

[14] B. L. Fox. Faster simulated annealing. *SIAM Journal on Optimization*, 5(3):488–505, 1995.

[15] B. L. Fox. Simulated annealing: Folklore, facts, and directions. In H. Niederreiter and P. J. Shieu, editors, *Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing: Proceedings of a Conference at the University of Nevada*, volume 106 of *Lecture Notes in Statistics*, pages 17–48, Berlin, FRG, 1995. Springer.

[16] B. L. Fox, J. K. Lenstra, A. H. G. Rinnooy Kan, and L. Schrage. Branching from the largest upper bound: Folklore and fact. *European Journal of Operational Research*, 2:191–194, 1978.

[17] M. Fujii, T. Kasami, and K. Ninomiya. Optimal sequencing of two equivalent processors. *SIAM Journal on Applied Mathematics*, 17(4):784–789, 1969.

[18] M.R. Garey, R.L. Graham, and D. S. Johnson. On a number-theoretic bin packing conjecture. In *Proceedings of the 5th Hungarian Combinatorial Colloquium*, pages 377–392, Amsterdam, 1978. North-Holland. Conference was held 1976 in Budapest.

[19] F. Glover. A new foundation for a simplified primal integer programming algorithm. *Operations Research*, 16(4):727–740, 1968.

[20] F. Glover. *Netforms in Management Science and Decision Support*. University of Colorado, School of Business, Boulder, CO, 1983.

[21] F. Glover and J-K. Hao. Hard optimization problems and strategic oscillation. Working paper, Laboratoire d'Etude et de Recherche en Informatique (LERIA), Université d'Angers, Angers, France, 2008.

[22] F. Glover, D. Klingman, and N. V. Phillips. *Network Models in Optimization and Their Applications in Practice*. Wiley-Interscience, New York, NY, 1992.

[23] F. Glover and D. C. Sommer. Pitfalls of rounding in discrete management decision problems. *Decision Sciences*, 6:211–220, 1975.

[24] R. E. Gomory and A. J. Hoffman. On the convergence of an integer-programming process. *Naval Research Logistics Quarterly*, 10:121–123, 1963.

[25] R. E. Gomory and E. L. Johnson. T-space and cutting planes. *Mathematical Programming, Series B*, 96(2):341–375, 2003.

[26] R. L. Graham. Bounds on multiprocessing timing anomalies. *SIAM Journal on Applied Mathematics*, 17(2):416–429, 1969.

[27] H. J. Greenberg. MIP tactics. In R. H. F. Jackson and R. P. O'Neill, editors, *Special Issue: Mixed Integer Programming in Mathematical Programming Systems*, pages 3–11. ORSA Computer Science Technical Section and Mathematical Programming Society Committee on Algorithms, 1984.

[28] B. Hajek and G. Sasaki. Simulated annealing — to cool or not. *Systems & Control Letters*, 12:443–447, 1989.

[29] F. Harary. *Graph Theory*. Addison-Wesley, Reading, MA, 1969.

[30] I. Heller. On the travelling salesman's problem. In H. A. Antosiewtz, editor, *Proceedings of the Second Symposium in Linear Programming*, volume II, Washington, DC, 1955. National Bureau of Standards.

[31] D. S. Hochbaum. Lower and upper bounds for the allocation problem and other nonlinear optimization problems. *Mathematics of Operations Research*, 19(2):390–409, 1994.

[32] K. H. Hoffmann and P. Salamon. Simulated annealing for single minimum optimization problems. *International Journal of Computer Mathematics*, 39(3& 4):193–204, 1991.

[33] T. Ibaraki. Computational efficiency of approximate branch-and-bound algorithms. *Mathematics of Operations Research*, 1(3):287–298, 1976.

[34] R. G. Jeroslow. Comments on integer hulls of two linear constraints. *Operations Research*, 19(4):1061–1069, 1971.

[35] R. G. Jeroslow. Trivial integer programs unsolvable by branch-and-bound. *Mathematical Programming*, 6(1):105–109, 1974.

[36] R. G. Jeroslow and J. K. Lowe. Modeling with integer variables. *Mathematical Programming Study*, 22:167–184, 1984.

[37] J. J. Kanet. On anomalies in dynamic ratio type scheduling rules: A clarifying analysis. *Management Science*, 28(11):1337–1341, 1982.

[38] W. Karush. A counterexample to a proposed algorithm for optimal sequencing of jobs. *Operations Research*, 13(2):323–325, 1965.

[39] M. T. Kaufman. Counterexample to a conjecture of Fujii, Kasami and Ninomiya. *SIAM Journal on Applied Mathematics*, 26(2):370–371, 1974.

[40] M. Kolonko. Some new results on simulated annealing applied to the job shop scheduling problem. *European Journal of Operational Research*, 113(1):123–136, 1999.

[41] T-H. Lai and S. Sahni. Anomalies in parallel branch-and-bound algorithms. *Communications of the ACM*, 27(4):594–602, 1984.

[42] J. K. Lenstra and A. H. G. Rinnooy Kan. On general routing problems. *Networks*, 6(3):273–280, 1976.

[43] J. K. Lenstra, A. H. G. Rinnooy Kan, and P. Brucker. Complexity of machine scheduling problems. *Annals of Discrete Mathematics*, 1:343–362, 1977.

[44] S. J. Mathis, Jr. A counterexample to the rudimentary primal integer programming algorithm. *Operations Research*, 19(6):1518–1522, 1971.

[45] R. R. Meyer. On the existence of optimal solutions to integer and mixed-integer programming problems. *Mathematical Programming*, 7(1):223–235, 1974.

[46] R. R. Meyer. Integer and mixed-integer programming models: General properties. *Journal of Optimization Theory and Applications*, 16(3/4):191–206, 1975.

[47] K. G. Murty. On the tours of a traveling salesman. *SIAM Journal on Control*, 7(1):122–131, 1999.

[48] C. S. Orloff. A fundamental problem in vehicle routing. *Networks*, 4(1):35–64, 1974.

[49] C. S. Orloff. On general routing problems: Comments. *Networks*, 6(3):281–284, 1976.

[50] C. H. Papadimitriou. The adjacency relation on the traveling salesman polytope is NP-complete. *Mathematical Programming*, 14(1):312–324, 1978.

[51] A. Rao. Counterexamples for the location of emergency service facilities. *Operations Research*, 22(6):1259–1261, 1974.

[52] M. R. Rao. Adjacency of the traveling salesman tours and 0-1 vertices. *SIAM Journal on Applied Mathematics*, 30(2):191–198, 1976.

[53] D. S. Rubin. On the unlimited number of faces in integer hulls of linear programs with a single constraint. *Operations Research*, 18(5):940–946, 1970.

[54] D. S. Rubin and R. L. Graves. Strengthened Dantzig cuts for integer programming. *Operations Research*, 20(1):178–182, 1972.

[55] P. Salamon, P. Sibani, and R. Frost. *Facts, Conjectures, and Improvements for Simulated Annealing*. SIAM, Philadelphia, PA, 2002.

[56] A. Schrijver. A counterexample to a conjecture of Edmonds and Giles. *Discrete Mathematics*, 32:213–214, 1980.

[57] S. Sen, J. L. Higle, and J. R. Birge. Duality gaps in stochastic integer programming. *Journal of Global Optimization*, 18(2):189–194, 2000.

[58] J. P. Shearer. A counterexample to a bin packing conjecture. *SIAM Journal of Algebraic and Discrete Methods*, 2(3):309–310, 1981.

[59] W. E. Smith. Various optimizers for single-state production. *Naval Research Logistics Quarterly*, 3:59–66, 1956.

[60] F. C. R. Spieksma and G. J. Woeginger. The no-wait flow-shop paradox. *Operations Research Letters*, 33(6):603–608, 2005.

[61] C. A. Tovey. A simplified anomaly and reduction for precedence constrained multiprocessor scheduling. *SIAM Journal on Discrete Mathematics*, 3(4):582–584, 1990.

[62] H-F. Wang and J-S Horng. Structural approach to parametric analysis of an IP on the case of the right-hand side. *European Journal of Operational Research*, 92(1):148–156, 1996.

[63] G. J. Woeginger. A note on the depth function of combinatorial optimization problems. *Discrete Applied Mathematics*, 108:325–328, 2001.

[64] R. D. Young. A primal (all-integer) integer programming algorithm. *Journal of Research of the National Bureau of Standards — B*, 69B(3):213–250, 1965.

[65] R. D. Young. A simplified primal (all-integer) integer programming algorithm. *Operations Research*, 16(4):750–782, 1968.

Dynamic Programming

A dynamic program (DP) is one that can be solved as a sequence of state-dependent optimization problems. When the underlying problem is dynamic, time provides the natural ordering for sequential optimization. However, DP is also a technique used to decompose a static problem into a sequence of lower-dimensional decision problems. A classical example of this decomposition is the *knapsack problem*:

$$\max \textstyle\sum_j c_j x_j : \sum_j a_j x_j \leq b,\ x \in \mathbb{Z}^n_+,$$

where $a, c > 0$. (See Martello and Toth[21] for a more extensive introduction.) A DP formulation of this is the *forward recursion*:

$$f_k(s) = \max \{c_j x_j + f_{k-1}(s - a_j x_j) : x_j \in \mathbb{Z}_+,\ a_j x_j \leq s\} \text{ for } s = 0, 1, \ldots, b,$$

for $k = 1, \ldots, n$ and $f_0(s) = 0$ for all $s = 0, \ldots, b$. The DP algorithm starts with $k = 0$ (with $f_0(s) = 0$ for $s \geq 0$), and it proceeds forward: $k = 1, 2, \ldots, n$. The solution value to the original problem is $f_n(b)$, and x^* is computed by backtracking through the 1-dimensional optimal solutions.

At the foundation is Bellman's *Principle of Optimality*[3]:

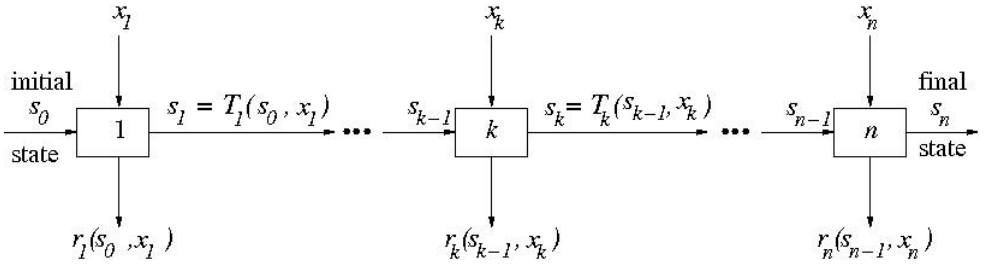
> "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."

This is what enables us to decompose an n-variable problem into a sequence of 1-variable problems. If the number of states and decisions is finite, this is equivalent to a *shortest path* through a network. The nodes are the (state, stage) pairs and the arcs are the transitions resulting from the decision.

In its natural time-ordered form, DP represents a *sequential decision process*: a discrete-time process characterized by a sequence of states, where the next state depends upon the current state and decision. (It does not upon the rest of the history of states and decisions.) At each time period, the decision yields a return and a state transition. Here is the *backward DP recursion*:

$$f_k(s) = \max_{x \in X_k(s)} \{r_k(s, x) + f_{k+1}(T_k(s, x))\},$$

where $X_k(s)$ is the set of decisions upon entering time period k in state s; $r_k(s, x)$ is the immediate return for choosing $x \in X_k(s)$; and, the last term is the total future return after transitioning to the new state, $T_k(s, x)$. This is illustrated in the following diagram:

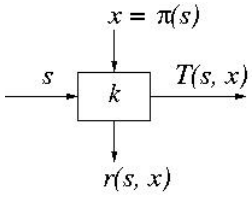


If the time periods are long enough, the present value is used with a *discount factor*, $\beta \in (0, 1]$:

$$f_k(s) = \max_{x \in X_k(s)} \{r_k(s, x) + \beta f_{k+1}(T_k(s, x))\}.$$

So, $f_0(s_0) = \sum_{k=1}^{n} \beta^{k-1} r_t(s_k, x_k^*)$, where $\{x_k^*\}$ are the decisions made at each time period, and the state sequence is given by $s_k = T_k(s_{k-1}, x_k^*)$ for $k = 1, \ldots, n$. Denote a *policy* by $\pi_k(s) =$ decision made at time k upon entering in state s. An *optimal policy* maximizes $f_0(s)$; equivalently, $\pi_k^*(s) = x_k^*$ for some $x_k^* \in \text{argmax}_{x \in X_k(s)}\{r_k(s, x) + \beta f_{k+1}(T_k(s, x))\}$.

The DP is *stationary* if the decision set and functions are independent of time: $X_k = X$, $r_k = r$, and $T_k = T$. A *stationary policy* is a function of state, but not of time: $\pi(s) \in X(s)$. It specifies the decision to be taken. This is illustrated on the right.



A *randomized policy* is one that is specified by $P_t(s, x) =$ probability that $\pi_t(s) = x$ when the system is in state s at time t. The actual decision is determined by some random selection method according to P. In general, $P_t(s, x) \in [0, 1]$ and for finite or denumerable decision sets, $\sum_{x \in X_t(s)} P_t(s, x) = 1$ for all s. ($P_t(s, x) = 0$ for $x \notin X_t(s)$.) The *non-randomized policy*, a.k.a., *pure policy*, is the special case: $P_t(s, \pi_t(s)) = 1$ for all s, t; otherwise, $P_t(s, x) < 1$ for at least one $x \in X_t(s)$ for some s, t.

DP Myth 1. *Given a separable objective function, the Principle of Optimality enables the decomposition into a series of smaller optimization problems over a state space. In particular, suppose*

$$R(x) = r_1(x_1) \oplus r_2(x_2) \oplus \cdots \oplus r_n(x_n)$$

over the separable domain, $X = X_1 \times X_2 \times \cdots \times X_n$. Further, we have a simple limit constraint, $\sum_{j=1}^{n} x_j \le b$. Then,

$$\max_{x \in X}\{R(x) : \sum_{j=1}^{n} x_j \le b\} = \max_{s \le b} f_n(s),$$

where

$$f_j(s) = \max_{x_j \in X_j} \{r_j(x_j) \oplus f_{j-1}(s - x_j) : s - x_j \le b\} \text{ for } j = 1, \ldots, n,\ s \le b.$$

with $f_0(s) = i$ = identity element for $\oplus$ (= 0 if $\oplus$ is ordinary addition; = 1 if $\oplus$ is ordinary multiplication).

The *Principle of Optimality* was originally developed for additive processes (where $\oplus$ is simple addition). Mitten[22] pointed out that non-additive processes may not decompose directly, as he developed a general framework.

Counterexample. $f(x_1, x_2) = x_1 x_2$ and $X_1 = X_2 = [-2, 1]$.

For $b \geq -4$, the 2-variable maximum value is 4. However, $f_1(s) = 1$ for all $s \leq 1$, which yields $f_2(s) = 1$ for all $s \leq 2$. The DP solution is thus $x = (1, 1)$ with $R(x) = 1$.

The problem is that $\oplus$ violates *Mitten's monotonicity condition* on this domain.

DP Myth 2. *The Principle of Optimality is a necessary condition for the optimality of a policy.*

This myth and the following counterexample are given by Porteus[24].

Counterexample. Let the state space be the interval [0, 1]. For each state there are two possible decisions: $X(s) = \{0, 1\}$. The immediate return is the same for each state: $r(s, x) = x$. Regardless of the decision and current state, the state transition is a uniform random variable. The objective is the discounted total return with discount factor $\beta < 1$.

It is optimal to set $x_n(s) = 1$ for all n and all states, s. Consider the alternative policy that sets $x_k(s) = 1$ for $s \neq 1$. This yields the same expected total discounted return, but it violates the necessity of the Principle of Optimality.

Because the probability of any one return is zero, what is done for just one decision for one state has no effect on the objective value.

DP Myth 3. *In a dynamic lot size inventory model, decreasing setup costs does not increase total inventory.*

The intuition behind this is that inventory is caused by the setup cost. In the *Economic Order Quantity* (EOQ) model, we have

$$Q = \sqrt{\frac{2Kd}{h}},$$

where Q is the min-cost order quantity, K is the setup cost, d is the demand, and h is the holding cost. Thus, reducing the setup cost does reduce the EOQ. However, this does not carry over to the dynamic lot size problem, where costs and demands occur over time, and the decision variables are how much to produce in each period to satisfy the demands.

The DP recursion is given by:

$$f_t(y) = \min_{x \geq 0} \left\{ p_t(x) + h_t(x + y - d_t) + f_{t-1}(x + y - d_t) : x + y \geq d_t \right\}, \quad \text{for } t = 1, \ldots, N$$

$$f_0(0) = 0;\ f_0(y) = \infty \text{ for } y > 0,$$

where y is the inventory level (state), starting with $y_0 = 0$, x is the production level, p_t is the production cost in period t, d_t is the demand, and h_t is the 1-period holding cost in period t for the new inventory level, $x + y - d_t$.

Zangwill[35] provides further discussion and the following:

Counterexample. A plant runs two shifts a day, a morning shift and a night shift. Consider two days of its operation which we divide into four periods. Designate period 1 as the morning shift of day 1, period 2 as the night shift of day 1, with periods 3 and 4 the day and night shifts, respectively, of day 2. Suppose the product demand during each shift is 3 units. Let the variable production cost be linear and stationary: $p_t(x_t) = px_t$ for all t. Because total production equals total demand, this form eliminates variable production cost as a factor in the objective. What remains is the setup cost, $\sum_t K_t\delta(x_t)$, where

$$\delta(x) = \begin{cases} 1 & \text{if x} > 0; \\ 0 & \text{otherwise.} \end{cases}$$

Let the holding cost be $h_t(y_t) = y_t$, for all t, where y_t is the inventory at the end of period t. Thus, the dynamic lot size model is given by:

$$\min \sum_{t=1}^{4} (K_t\delta(x_t) + y_t) : x, y \geq 0,$$
$$y_{t-1} + x_t - y_t = d_t, \text{ with } y_0 = 0.$$

Scene 1. At present the plant is quite busy during the day, and the setup costs during the day are higher than at night. In particular $K_1 = K_3 = 8$, $K_2 = K_4 = 5$. The optimal production schedule is $x^* = (3, 6, 0, 3)$ with associated inventory levels $y^* = (0, 3, 0, 0)$.

Scene 2. The engineering department undertakes to reduce setup costs and thereby move closer to a *Zero-Inventory* system. After considerable analysis, they conceive how to use the greater range of talent available during the day, which enables them to reduce setup costs more during the day than at night. After the engineering department completes its task, the setup costs are significantly reduced and become $K_1 = K_3 = 1$, $K_2 = K_4 = 4$. All other costs remain the same. The new optimal production schedule is $x' = (6, 0, 6, 0)$ with associated inventory levels $y' = (3, 0, 3, 0)$. Even though all setup costs have been cut (and no other changes made), the total inventory level has doubled.

Zangwill provides conditions under which decreasing setup costs results in decreasing inventory. The key feature of the counterexample is that the day-shift reduction is different from the night-shift reduction.

DP Myth 4. *In a dynamic lot size inventory model, decreasing setup costs does not increase minimum total cost.*

Zangwill[35] provides further discussion and the following:

Counterexample. To manufacture a particular product requires three separate operations, call them I, II, and III. Suppose also that five workstations exist, and each workstation can do the operations enumerated:

A={I}, B={II}, C={III}, D={I, II}, E={II, III}.

For example, workstation A can do only operation I whereas station D can accomplish both operation I and operation II. Since all three operations are required to complete the product, there are three possible *routings*: A→B→C, A→E, and D→C. For example, route

A→E accomplishes operation I at station A and station E does II and III. Generally, the various workstations are scheduled carefully with the work flow balanced and optimally allocated.

Upon occasion, an emergency rush order for the product arises, which is costly since it disrupts operations. The more emergency orders that occur during a day the more costly it becomes because additional disruptions cause the regular schedule to become increasingly rushed. Initially, suppose for an emergency order on a workstation that the cost for the x^{th} emergency order that day is

$$q_A(x) = q_C(x) = 10 + 10x,\ q_D(x) = q_B(x) = q_E(x) = 31 + x.$$

Here $q_D(x) = 31 + x$ means that for workstation D, the setup cost is 31 for processing an emergency order, and x is the additional cost if $x - 1$ emergency orders have already been processed at workstation D. The cost increases as more emergency orders are processed at a workstation, as mentioned, because of the increased disruption.

The expeditor is the individual who juggles the work and tries to process the emergency as inexpensively as possible by selecting the routing. The cost depends not only on which workstations are along the route but also on how many emergencies a workstation has already had to contend with that day. Given the costs, here is the minimum cost for processing if there are one or two emergencies in a day:

If one emergency occurs, an optimal route is A→E at a cost of 52.
If a second emergency occurs, an optimal route is D→C, a cost of 52.

Thus, if one emergency occurs, the total cost is 52. Should two emergencies occur, the total cost is 104.

Suppose the setup cost for an emergency on workstation B is cut from 31 to 10: $q_B(x) = 10 + x$. All other costs remain the same. Now we have:

If one emergency occurs, an optimal route is A→B→C at a cost of 51.
If a second emergency occurs, an optimal route is A→E, at a cost of 62.

Thus, the cost of one emergency during the day is 51, but if two emergencies occur, the cost is 113. If we are unfortunate enough to get two emergencies during the day, the cost is higher after the setup cost reduction, so the setup cost reduction has actually increased the minimum total cost.

DP Myth 5. *The Federgruen-Lee algorithm produces an optimal solution to the dynamic lot size model with quantity discount.*

Federgruen and Lee[7] proposed a DP algorithm, but there are special cases for which it does not necessarily produce an optimal solution.

Notation:

D_t	demand in period t
K_t	fixed setup cost in period t
c_t	unit purchase price in period t
h_t	unit holding cost
N	discount quantity
r	discount rate
x_t	amount purchased in period t

The model has $x_t \geq N \rightarrow$ purchase cost $= c_t(1-r)$ and holding cost $= h_t(1-r)$.

Counterexample. Xu and Lu[34] provide the following: $n = 4$, $D = (10, 40, 80, 20)$, $K = (150, 150, 150, 150)$, $c = (8, 8, 8, 8)$, $h = (5, 5, 5, 5)$, $N = 75$, and $r = 0.1$. The Federgruen-Lee algorithm obtains $x = D$, with a total cost of \$1,736. An optimal solution is $x^* = (10, 40, 100, 0)$, with a total cost of \$1,660.

Xu and Lu give more insights into the cause of the algorithm's failure. Another cause is given by the following:

Counterexample. $n = 3$, $D = (30, 30, 10)$, $K = (60, 60, 60)$, $c = (10, 10, 10)$, $h = (2, 2, 2)$, $N = 60$, and $r = 0.1$. The Federgruen-Lee algorithm obtains $x = (60, 0, 10)$, with a total cost of \$814. An optimal solution is $x^* = (70, 0, 0)$, with a total cost of \$762.

Xu and Lu presented a modified algorithm to overcome such counterexamples.

DP Myth 6. *Consider a dynamic lot size problem in which lead times are stochastic, and shortages are backlogged. Optimal production levels still satisfy the property that they are zero with positive incoming inventory and otherwise equal the sum of successive demands.*

The appeal of this myth is that the property holds with zero lead times. In that case, whenever there is zero inventory upon entering period t, the optimal production level is $x_t^* = \sum_{k=t}^{t'} d_k$ for some $t' \geq t$. If the entering inventory is positive, it is enough to meet the demand and $x_t^* = 0$. More generally, if the lead time of production in period t is L_t, the zero-inventory point is at period $t' = \min\{k : k \geq t + L,\ d_k > 0\}$. Thus, $x_t^* y_{t'-1}^* = 0$ is the optimality property in question.

Anderson[1] provides the following:

Counterexample. The horizon is 9 periods with $d_5 = 2$, $d_8 = 3$, and $d_t = 0$ for $t \neq 5, 8$. Setup costs are all zero, and the unit production costs are $p_1 = 2$, $p_2 = 5$, $p_4 = 30$, $p_8 = 3$, and $p_t = 1000$ for $t = 3, 6, 7, 8, 9$. The holding costs are all zero, and the 1-period unit shortage costs are nonzero for $s_5 = s_9 = 1000$.

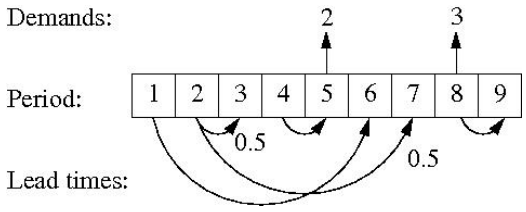


Figure taken from [1].

Production cost of 1000 is enough to render $x_3^* = x_5^* = x_6^* = x_7^* = x_9^* = 0$ in every optimal solution, so lead times for those periods are not shown. The total production cost is then $2x_1 + 5x_2 + 30x_4 + 3x_8$.

The total shortage costs for each of the two random lead times are given by:

$$L_2 = 1\colon\ -1000(x_2 + x_4 - 2)^- - 1000(x_1 + (x_2 + x_4 - 2)^+ + x_8 - 3)^-$$
$$L_2 = 5\colon\ -1000(x_4 - 2)^- - 1000(x_1 + (x_2 + x_4 - 2)^+ + x_8 - 3)^-$$

To avoid the 1000-unit costs for shortages, every optimal policy sets $x_2^* + x_4 \geq 2$, and $x_1 + x_2 + x_4 + x_8 \geq 1$.

Since this is a DP, the production levels x_4, x_8 are determined *after* the lead time from period 2 becomes known. The optimal policy is to set $x_1^* = 1$, $x_2^* = 2$, and

$$\begin{array}{llll} x_4^* = 2, & x_8^* = 0 & \text{if lead time} = 5 \\ x_4^* = 0, & x_8^* = 2 & \text{if lead time} = 1. \end{array}$$

Since the lead times in period 2 are equally-likely, the total expected cost is

$$2 + 10 + \tfrac{1}{2}(30 \times 2 + 3 \times 2) = 45.$$

Notice that the zero-inventory point for period 1 occurs at period 8 for the arrival pattern in which the lead time of x_2^* is 5 periods — that is, $x_1^* > 0$ and $y_7 > 0$ (where the zero-inventory point of period 1 is period 8), thus violating the myth's indicated property.

DP Myth 7. *In a dynamic lot size problem, a stochastically greater lead time cannot result in a lower optimal average cost.*

Let L_1, L_2 be random lead times. Then, L_1 is *stochastically greater* than L_2, denoted $L_1 \geq_{\text{st}} L_2$, if

$$\mathbf{Pr}[L_1 \geq \ell] \geq \mathbf{Pr}[L_2 \geq \ell] \text{ for all } \ell.$$

The underlying model is continuous-time, single-item, where demands form a compound Poisson process — demands occur at epochs with random batch size. Here we assume the batch size is 1. A stationary policy is optimal, where the decision variables are the target inventory levels. (This is called a *base-stock policy*, and "target" is used when the demand structure is random.)

Notation:

h	unit holding cost rate
p	unit shortage cost rate
D	lead time demand
Ψ	cdf of D
y	target inventory level for base-stock policy

Only the holding and shortage costs depend upon lead time demand, so we ignore ordering costs for purposes of policy comparisons. The expected average cost is: *ordering cost* +

$$\text{lead-time-dependent costs} = E[h(y - D)^+ + p(D - y)^+].$$

Counterexample. Song[28] provides the following: $\mathbf{Pr}[L_1 = 1] = 1$ and $\mathbf{Pr}[L_2 = 1] = 0.7$, $\mathbf{Pr}[L_2 = 0.1] = 0.3$. Note that $L_1 \geq_{\text{st}} L_2$. Let $h = 2$, $p = 9$, and the two cdfs:

$$\begin{array}{lll} \Psi_1(0) = 0.3679 & \Psi_1(1) = 0.7358 & \Psi_1(2) = 0.9197 \\ \Psi_2(0) = 0.5290 & \Psi_2(1) = 0.8137 & \Psi_2(2) = 0.9437 \end{array}$$

The optimal target inventory for each lead time is $y_1^* = y_2^* = 2$. Their costs differ by

$$E[h(y - D_1)^+ + p(D_1 - y)^+] - E[h(y - D_2)^+ + p(D_2 - y)^+] = 3.14 - 3.24 < 0.$$

Hence, the stochastically greater lead time has lower average cost, contrary to the myth.

Song also analyzes the effect of more variable lead time (with equal means).

DP Myth 8. *In a multi-item, base-stock inventory system, the total order fill rate cannot exceed the independent fill rate.*

The underlying model is a continuous-time, multi-item inventory system, where items are consumed by *demand types*. Demand rates define a Poisson process, for which a stationary base-stock policy (see p. 91) minimizes the average cost. An *order fill rate* is the probability of satisfying demand immediately.

The total order fill rate is given by:

$$FT = \sum_{k=1}^{K} q_k f_k,$$

where f_k is the type-k fill rate, and q_k is the probability that the demand is of type k. The independent fill rate, which is used to approximate FT, assumes all demands are independent of all other demands:

$$FI = \sum_{i=1}^{n} Q_i F_i,$$

where F_i is the fill rate of item i, and Q_i is the demand rate for item i. Q is determined by q:

$$Q_i = \frac{1}{\kappa} \sum_{k \in S(i)} q_k,$$

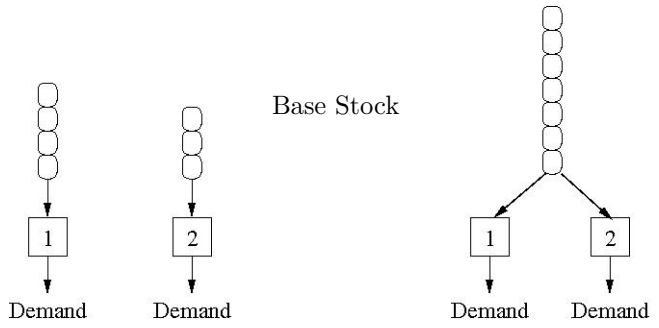
where $S(i)$ is the set of types that consume item i, and $\kappa = \sum_{i=1}^{n} \sum_{k \in S(i)} q_k$. The myth asserts $FT \leq FI$.

Counterexample. Song[29] provides the following: Let $n = 3$, all lead times are 1, and the overall demand rate is 1. Let $K = 7$ with $q = (0.01, 0.01, 0.85, 0.03, 0.01, 0.01, 0.08)$ and

$$S(1) = \{1, 4, 5, 7\},\ S(2) = \{2, 4, 6, 7\},\ S(3) = \{3, 5, 6, 7\}.$$

Let the base-stock levels be $(1, 1, 4)$. Then, $f = (0.878, 0.878, 0.984, 0.861, 0.868, 0.868, 0.858)$, so $F = (3.465, 3.465, 3.578)$, which yields the contradiction: $FT = 0.965 > FI = 0.961$.

DP Myth 9. *If inventories for substitute products are pooled by a centralized system, the optimal total base-stock does not increase.*



Separate Inventories Pooled Inventory

Let x be the vector of base-stock inventory levels, and let $f_j(x)$ denote the associated *fill rate* — that is, the probability of satisfying demand immediately. The respective mathematical programs are:

$$\begin{array}{cc} \underline{\text{Not Pooled}} & \underline{\text{Pooled}} \\ \min\ c(x_1+x_2):\ f_1(x_1)\geq\alpha,\ f_2(x_2)\geq\alpha & \min\ cy:\ f_{1,2}(y)\geq\alpha \\ x\geq 0,\ x_1+x_2\leq U & 0\leq y\leq U, \end{array}$$

where U is the total storage limit (note the same costs and fill-rate requirements). We assume a Poisson demand distribution (IID) with rate λ and a FIFO allocation rule. So,

$$f_i(x_i) = e^{-\lambda_i}\sum_{k=0}^{x_i}\frac{\lambda_i^k}{k!} \qquad \text{for } i=1,2$$

$$f_{1,2}(y) = e^{-(\lambda_1+\lambda_2)}\sum_{k=0}^{y}\frac{(\lambda_1+\lambda_2)^k}{k!}.$$

Let x^* and y^* be optimal solutions to the respective mathematical programs. The myth asserts $x_1^*+x_2^*\geq y^*$.

Counterexample. The following is based upon Song and Zhao[30]. Let $\lambda_1=\lambda_2=1$ and $\alpha=0.73$. Then, the solution to the unpooled problem is $x_1^*=x_2^*=1$ with $f_i(x_i^*)=0.735>\alpha$. The solution to the pooled problem is $y^*=3$ with $f_{1,2}(y^*)=0.857$ (but $f_{1,2}(2)=0.677<\alpha$).

Note that this also shows a key property: x_1+x_2 need not be feasible in the pooled problem even if x is feasible in the unpooled problem. For this example, we could choose any $\alpha\in(0.677,0.735)$ with $x_1=x_2=1$. This occurs with small inventory levels because for $y=\sum_i x_i$ sufficiently large

$$\min_i\ e^{-\lambda_i}\sum_{k=0}^{x_i}\frac{\lambda_i^k}{k!}\leq e^{-\sum_i\lambda_i}\sum_{k=0}^{y}\frac{\left(\sum_i\lambda_i\right)^k}{k!}.$$

Song and Zhao provide related counterexamples to propositions that intuitively seem true. Moreover, they consider a more general system with lead times that may, or may not, be equal. Also see SF Myth 8.

DP Myth 10. *A manufacturer can set a guaranteed customer-service level by knowing the supplier's service level.*

The model is a Markov chain with states equal to $(i,j)\geq 0$, where i is the supplier's level, and j is the backorder by the manufacturer. (It suffices to limit the states to those that are reachable from $(s,0)$, where s is the supplier's base-stock level.) Let $\mathrm{P}(i,j)$ denote the steady-state probability of being in state (i,j).

The manufacturer's demand is $d>0$ in each period. The supplier's maximum production capacity is $d+e$, where $e>0$, but it is possible that none of that capacity is available in a period. The actual capacity in period t is thus a random variable, κ_t, with

$$\begin{aligned}\mathbf{Pr}[\kappa_t=d+e] &= p\\ \mathbf{Pr}[\kappa_t=0] &= 1-p,\end{aligned}$$

where $0 < p < 1$. The manufacturer's capacity is $d + b$, where $b > 0$.

The manufacturer's service level is measured by the probability of a backorder:

$$\alpha_m = \sum_{i=0}^{s} \sum_{j=1}^{\infty} \mathrm{P}(i, j).$$

This says that regardless of the supplier's inventory (that is, for all i), state (i, j) has a backorder for all $j > 0$ (by definition).

The supplier's service level is measured by the probability of a shortage:

$$\alpha_s = \sum_{j=0}^{\infty} \sum_{i=0}^{d+\min\{j,b\}-1} \mathrm{P}(i, j).$$

This says that for all j) the supplier incurs a shortage when its inventory is less than the manufacturer's capacity — that is, $i < d + b$ — and less than the demand plus backorder — that is, $i < d + j$. Hence, $i \leq d + \min\{b, j\} - 1$.

The myth asserts that there exists a function ϕ such that $\alpha_m = \phi(\alpha_s)$. We'll relax that to assert that the manufacturer's service level can be approximately determined by the supplier's service level:

$$\alpha_m = \phi(\alpha_s) \pm \varepsilon \text{ for } \varepsilon < 1\%.$$

Counterexample. Choi, Dai, and Song[6] provide the following:

Table 1 Supplier's Characteristic Parameter Sets Yielding $\alpha_s \approx 5\%$

b	e	p	s	α_s (%)	α_m (%)
1	2	0.95	19	5.00	13.26
	3	0.80	55	4.98	32.39
	9	0.70	37	5.01	30.74
	15	0.70	31	5.02	34.46
2	3	0.80	63	5.09	21.30
	3	0.95	19	5.00	7.80
	5	0.79	30	4.97	20.04
	10	0.84	20	5.00	18.14
5	6	0.95	15	5.00	5.82
	9	0.80	29	5.02	7.26
	28	0.64	38	4.96	8.62
	34	0.55	44	5.01	11.17

> The supplier's service level is approximately 5% for a range of parameter values. However, the manufacturer's service level varies greatly, from about 7% to 34%. Thus, the manufacturer's service level depends upon *how* the supplier achieves its service level — that is, the parameter settings.

Choi et al. provide details and dispel other myths in their analysis of vendor-managed-inventory programs.

DP Myth 11. *A Bayesian optimal ordering level is less than a non-Bayesian optimal ordering level.*

The difference between the models is that the Bayesian updates the demand distribution as demands become known, thus affecting new decisions. The intuition is that with greater knowledge, we can maintain shortage risk while decreasing ordering and holding costs.

We shall assume the DP is stationary, and that the set of possible demands is finite: $\mathcal{D} = \{0, 1, \ldots, D\}$. The associated probabilities, $p = (p_0, \ldots, p_D)$, are unknown but have a known pdf, with parameters $\alpha = (\alpha_0, \ldots, \alpha_D)$. The estimate of the probabilities is

$$p_i(\alpha) = \frac{\alpha_i}{\sum_{j=0}^{D} \alpha_j}. \tag{DP.10}$$

Let the state variable be the inventory level, s, so the non-Bayesian DP recursion is:

$$f_n(s) = \min_{x \geq 0} cx + \sum_{i=0}^{D} p_i \left[q(i - s - x)^+ + f_{n+1}(s + x - i) \right], \qquad \text{for } n < N, \tag{DP.11a}$$

$$f_N(s) = \min_{x \geq 0} cx + \sum_{i=0}^{D} p_i \, q(i - s - x)^+, \tag{DP.11b}$$

where c is the unit ordering cost (no setup cost), and q is the unit shortage cost. (Assume no holding cost.)

The demand probabilities, p, remain equal to the initial estimate, given by (DP.10). Note that the ending inventory, $s + x - i$, which is passed to period $n+1$, could be negative. This means shortage continues to incur costs for each period that it remains unfulfilled.

Assume the pdf of the probabilities is a Dirichlet distribution of order $D+1$. Then, for demand i in period 1, the Bayesian update is to increase $\alpha_i \leftarrow \alpha_i + 1$, and keep α_j unchanged for $j \neq i$. Extending the state to include the previous period's demand, d, the Bayesian DP recursion is:

$$B_n(d, s) = \min_{X \geq 0} cX + \sum_{i=0}^{D} p'_i \left[q(i - s - X)^+ + B_{n+1}(i,\, s + X - i) \right] \quad \text{for } 1 < n < N \tag{DP.12a}$$

$$B_N(d, s) = \min_{X \geq 0} cX + \sum_{i=0}^{D} p'_i \, q(i - s - X)^+ \tag{DP.12b}$$

$$B_1(s) = \min_{X \geq 0} cX + \sum_{i=0}^{D} p_i \left[q(i - s - X)^+ + B_2(i,\, s + X - i) \right] \tag{DP.12c}$$

where α is updated to be $\alpha + \boldsymbol{e}_d$, which yields the probability update: $p' = p(\alpha + \boldsymbol{e}_d)$, for $n > 1$. For $n = 1$, the original α is used to estimate p, and demand is not part of the state.

Counterexample. Azoury and Miller[2] provide the following 2-period instance: $D = 2$, $c = 1$, and $q = 1.00999$. We assume that the initial inventory is zero, and $\alpha = (1, 97, 1)$. The initial probability estimates for both models is $p = (1, 97, 1)/99$.

For the non-Bayesian model, the costs for period 2 are:

x	$s=-2$	$s=-1$	$s=0$	$s=1$	$s=2$
0	3.029970	2.019980	1.009990	0.010202	0.000000
1	3.019980	2.009992	1.010202	1.000000	1.000000
2	3.009990	2.010202	2.000000	2.000000	2.000000
3	3.010202	3.000000	3.000000	3.000000	3.000000
4	4.000000	4.000000	4.000000	4.000000	4.000000
$f_2(s)$	3.009990	2.009990	1.009990	0.010201	0.000000
$x_2^*(s)$	2	1	0	0	0

Using the non-Bayesian DP recursion (DP.11), we compute $f_1(0) = 2.0102$ with $x_1^*(0) = 1$. Here are some details:

x	$x + q\mathbf{E}_i[(i-x)^+]$	$\mathbf{E}_i[f_2(x-i)]$	Total	
0	1.009990	2.009990	3.019980	
1	1.010202	1.009990	2.020194	$\leftarrow$ min
2	2.000000	0.020198	2.020198	
3	3.000000	0.000103	3.000103	
4	4.000000	0	4.000000	

Now consider the Bayesian model, using (DP.12), starting with the last stage. The following table entries are $B_2(d, s)$.

d	$s=-2$	$s=-1$	$s=0$	$s=1$	$s=2$
0	2.99989	1.99989	0.99989	0.01010	0
1	3.00999	2.00999	1.00999	0.01010	0
2	3.02009	2.02009	1.02009	0.02020	0

The Bayesian update of the probabilities are:

d	p'_0	p'_1	p'_2
0	$\frac{2}{100}$	$\frac{97}{100}$	$\frac{1}{100}$
1	$\frac{1}{100}$	$\frac{98}{100}$	$\frac{1}{100}$
2	$\frac{1}{100}$	$\frac{97}{100}$	$\frac{2}{100}$

Here are the total costs for each ordering level in period 2 for some states ($X^*(s,d)$ is in bold):

s	d	$X=0$	$X=1$	$X=2$	$X=3$	$X=4$	$B_2(s,d)$
-1	0	2.00988	**1.99989**	2.01010	3.00000	4.00000	1.99989
	1	2.01998	**2.00999**	2.01010	3.00000	4.00000	2.00999
	2	2.03008	**2.02009**	2.02020	3.00000	4.00000	2.02009
0	0	**0.99989**	1.01010	2.00000	3.00000	4.00000	0.99989
	1	**1.00999**	1.01010	2.00000	3.00000	4.00000	1.00999
	2	**1.02009**	1.02020	2.00000	3.00000	4.00000	1.02009
1	0	**0.01010**	1.00000	2.00000	3.00000	4.00000	0.01010
	1	**0.01010**	1.00000	2.00000	3.00000	4.00000	0.01010
	2	**0.02020**	1.00000	2.00000	3.00000	4.00000	0.02020

Using B_2 and the Bayesian update of α (and hence p), we obtain $B_1(0) = 2.0202$, with $X_1^*(0) = 2$. Here are some details:

x	$x + q\mathbf{E}_i[(i-x)^+]$	$\mathbf{E}_i[B_2(i, x-i)]$	Total	
0	1.009990	2.009990	3.019980	
1	1.010202	1.010093	2.020295	
2	2.000000	0.020201	2.020201	← min
3	3.000000	0.000204	3.000204	
4	4.000000	0	4.000000	

The Bayesian ordering level is greater than the non-Bayesian, thus contradicting the myth.

Caveate: I used a different DP recursion, so my shortage and cost calculations may not seem consistent with those of Azoury and Miller. The end result is the same: $x_1^*(0) = 1 < X_1^*(0) = 2$.

Opportunity Knocks

Azoury and Miller identified a class of inventory models for which the myth's statement is true. More generally, what are necessary and/or sufficient conditions for this?

DP Myth 12. *When entering a parallel queueing system, it is optimal to join the shortest queue.*

Assume the arrivals into the system follow a Poisson process, with rate λ. Each new arrival knows the queue lengths and must decide which queue to join to minimize expected time in the system.

One reason this myth is not true is that "shortest" need not be the least wait time. For example, in a supermarket checkout line people have different amounts of groceries. Let us assume, however, that our queueing system serves indistinguishable customers. (Each customer may be in one of several classes, requiring different service times, but the new arrival cannot determine to which class each customer belongs.)

Counterexample. Whitt[33] provides the following: Let there be two queues with independent service times. Let the common service-time distribution be given by the mass function:

$$\mathbf{Pr}(t = \tau) = 1 - \varepsilon \quad \text{and} \quad \mathbf{Pr}(t = 2) = \varepsilon,$$

where $0 \leq \tau \ll \varepsilon \ll 1$. We may consider this to be a 2-class population, but unlike the supermarket example, an arrival cannot determine the other customers' class. An example is a bank (with separate lines) such that a customer may have a very quick transaction, like a deposit, or may require a lot of time, like complicated transfers.

Define the system state (s_1, s_2) = queue sizes. The shortest-queue rule is optimal for states: (1) $s_1 = 0$ or $s_2 = 0$, and (2) $|s_1 - s_2| \leq 1$. However, if $s_1, s_2 > 0$ and $|s_1 - s_2| \geq 2$, Whitt proves that it is optimal to join the *longer* queue for τ, ε sufficiently small.

DP Myth 13. *Given continuous control over arrival and service rates in a network of queues, the optimal arrival rate into a queue decreases with its size.*

Weber and Stidham[32] consider a network of m queues in which customers arrive at queue i in a Poisson stream with rate λ_i and complete service at a rate μ_i. The completion may be rejected, so the customer may remain at the same queue; otherwise, the customer may leave the system or move to another queue.

The arrival and service rates are subject to continuous control over intervals, $[\underline{\lambda}, \overline{\lambda}]$ and $[\underline{\mu}, \overline{\mu}]$, respectively. Service rates incur costs, $c_i(\mu_i)$, and arrival rates bring rewards, $r_i(\lambda_i)$. The state of the system is $s = (s_1, \ldots, s_m) \geq 0$, where s_i is the number of customers in queue i. There is a holding (or waiting) cost, $\sum_{i=1}^{m} h_i(s_i)$.

Here is the DP recursion for minimizing the total expected cost:

$$f_{n+1}(s) = \sum_{i=1}^{m} \min_{\substack{\lambda_i \in [\underline{\lambda}_i, \overline{\lambda}_i] \\ \mu_i \in [\underline{\mu}_i, \overline{\mu}_i]}} \left\{ h_i(s_i) + c_i(\mu_i) - r_i(\lambda_i) + \lambda_i f_n(s + \boldsymbol{e}_i) + \mu_i \mathbf{E}[f_n(T_i(s, \mu_i))] \right\},$$

where $\boldsymbol{e}_i$ is the i^{th} unit vector, and $T_i(s, \mu_i)$ is the new state resulting from completion events at queue i:

$$\begin{array}{ll} T_i(s, \mu_i) = s - \boldsymbol{e}_i & \text{if customer leaves system;} \\ T_i(s, \mu_i) = s - \boldsymbol{e}_i + \boldsymbol{e}_k & \text{if customer joins queue } k; \\ T_i(s, \mu_i) = s & \text{if customer remains in queue } i. \end{array}$$

The myth says that an optimal rate satisfies:

$$\lambda_i^*(s + \boldsymbol{e}_i) \leq \lambda_i^*(s).$$

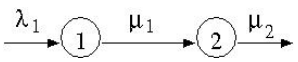
Weber and Stidham[32] call this property *transition-monotone*, and they prove that it holds under certain assumptions. The intuition is that it is less costly to slow down the entrance of new arrivals if the queue grows.

Counterexample. Weber and Stidham[32] provide the following:

$\lambda_1 \in [0, 0.01], \lambda_2 = 0,\ \mu_1 = 1, \mu_2 \in [0, 2]$,

$c_1 = c_2 = 0$, $r_1(\lambda_1) = 0.944\lambda_1$,

$$h_1(s_1) = \begin{cases} 0 & \text{if } s_1 = 0, 1, 2 \\ \infty & \text{if } s_1 > 2, \end{cases}$$



$$h_2(s_2) = \begin{cases} 0 & \text{if } s_2 = 0, 1, 2, 3, 4 \\ \infty & \text{if } s_2 > 4, \end{cases}$$

An optimal solution has $\mu_2^*(s) = 2$ for all s, and

$$\lambda^*(s) = \begin{cases} 0.01 & \text{if } s \in S \\ 0 & \text{otherwise,} \end{cases}$$

where $S = \{(0,0), (1,0), (0,1), (1,1), (1,2)\}$. This violates the myth with $s = (0, 2)$ since

$$\lambda^*(1, 2) \not\leq \lambda^*(0, 2).$$

DP Myth 14. *In a 2-server queueing system, when the system is empty, it is optimal to have the faster server take the first arrival.*

Consider Poisson arrivals with rate λ, and they balk (that is, leave the system) if there is a queue. The objective is to maximize the number of customers served.

Counterexample. Seth[27] provides the following: Servers 1 and 2 have mixed distributions:

$$\begin{array}{lll} \text{Server 1:} & \mathbf{Pr}[t=0]=\frac{1}{2}, & \mathbf{Pr}[t\sim exp(\mu/2)]=\frac{1}{2}; \\ \text{Server 2:} & \mathbf{Pr}[t\sim exp(\mu)]=\frac{1}{2}, & \mathbf{Pr}[t\sim exp(\mu/2)]=\frac{1}{2}. \end{array}$$

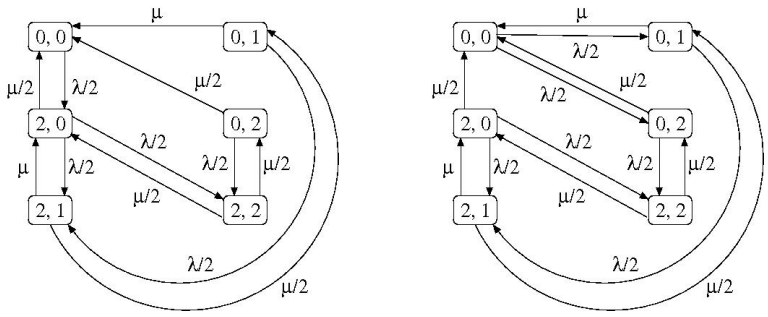
Server 1's time is stochastically less than Server 2, so the myth asserts that Server 1 will take the next arrival in an empty system.

Let policy i have Server i take the first arrival in an empty system, for $i = 1, 2$. The states are (i, j), where

$i = 0$ means Server 1 is free;
$i = 2$ means Server 1 is serving with time $\sim exp(\mu/2)$;

$j = 0$ means Server 2 is free;
$j = 1$ means Server 2 is serving with time $\sim exp(\mu)$;
$j = 2$ means Server 2 is serving with time $\sim exp(\mu/2)$.

Ergodic Transition Probabilities



Policy 1 Policy 2

Seth solves the balance equations for each policy, from which he derives the proportion of customers served. The difference in this proportion of policy 2 minus policy 1 is

$$D = (3\rho^3 + 16\rho^2 + 26\rho + 12)\rho^2/ab,$$

where

$$\begin{aligned} \rho &= \tfrac{\lambda}{\mu} \\ a &= 3\rho^4 + 17\rho^3 + 34\rho^2 + 32\rho + 12 \\ b &= 3\rho^4 + 14\rho^3 + 23\rho^2 + 19\rho + 6. \end{aligned}$$

Since $D > 0$ for all λ, μ, this dispels the myth.

Seth gives some intuition why it is optimal for Server 2 to take the next arrival in an empty system. Suppose λ is small, so that the probability of more than three customers arriving during a busy period is very small. When three customers arrive, four different situations could exist with equal probability, as follows:

	Server 1	Server 2	Optimal Policy
situation 1	0	$exp(\mu)$	Both equally good
situation 2	0	$exp(\mu/2)$	Both equally good
situation 3	$exp(\mu/2)$	$exp(\mu)$	Both equally good
situation 4	$exp(\mu/2)$	$exp(\mu/2)$	Send to Server 2

DP Background — Infinite Horizon

The *infinite horizon* DP has two basic models:

$$\text{Discounted:} \quad V_\beta(s,\pi) = \lim_{n\to\infty} V_\beta^n(s,\pi) \qquad \text{for } \beta < 1$$

$$\text{Average:} \quad A(s,\pi) = \lim_{n\to\infty} \frac{V_1^n(s,\pi)}{n},$$

where

$V_\beta^n(s,\pi) = \sum_{t=1}^n \beta^{t-1} r_t(s_t, \pi_t(s_t))$

π is a policy — that is, a decision rule with $\pi_t(s_t) \in X_t(s_t)$

β is the discount factor $\in [0,1]$.

For the deterministic model, the state transition is given by $s_{t+1} = T_t(s_t, \pi_t(s_t))$. The stochastic transition is given by:

$$\mathbf{Pr}[s_{t+1} = s | s_t, x] = q_t(s_t, s_{t+1}; x) \text{ for } x \in X_t(s_t).$$

The stochastic models use the expected returns in the objective, and the current state, s_t, is known at the time of the decision, $x = \pi_t(s_t) \in X_t(s_t)$. This is a *Markov decision process* with discrete time.

In words, V_β is the total discounted return when starting in state s and using policy π; A is the longterm, undiscounted average return. The former has an optimal solution under mild assumptions (but see DP Myth 16); the latter is approached under certain circumstances by letting $\beta \to 1$ from below.

The DP recursion for the discounted model is given by:

$$f_t(s) = \max_{x\in X_t(s)} \left\{ r_t(s,x) + \beta \textstyle\sum_{s'} q_t(s,s';x) f_{t+1}(s') \right\}.$$

The deterministic model is included with

$$q_t(s,s';x) = \begin{cases} 1 & \text{if } s' = T_t(s,x); \\ 0 & \text{otherwise.} \end{cases}$$

If the maximum exists for each state, an optimal policy is to let

$$\pi_t^*(s) \in \operatorname*{argmax}_{x\in X_t(s)} \left\{ r_t(s,x) + \beta \textstyle\sum_{s'} q_t(s,s';x) f_{t+1}(s') \right\}.$$

In a stationary DP, the recursion may be regarded as *value iteration* whose limiting function is the solution to

$$f(s) = \max_{x\in X(s)} \left\{ r(s,x) + \beta \textstyle\sum_{s'} q(s,s';x) f(s') \right\}.$$

The value function, f, is a fixed-point of the mapping, where $\beta < 1$ makes it a contractor.

Let V_β^* and A^* denote suprema values of the discounted and average-return models, respectively. The existence of optimal policies is not guaranteed, especially for the average-return model, where the limit may not exist. There are several variations of the DP objective for π^* to be an optimal policy:

B-opt:	$\exists \bar{\beta} \in (0,1) : V_\beta(s,\pi^*) \geq V_\beta^*(s)$	$\forall \beta \in (\bar{\beta}, 1)$;
nearly optimal:	$\lim_{\beta \to 1^-} \big(V_\beta(s,\pi^*) - V_\beta^*(s)\big) = 0$;	
1-optimal:	$\liminf_{\beta \to 1^-} \big(V_\beta(s,\pi^*) - V_\beta(s,\pi)\big) \geq 0$	$\forall \pi$;
discount ε-optimal:	$V_\beta(s,\pi^*) \geq V_\beta^*(s) - \varepsilon$	for $\varepsilon > 0$;
average ε-optimal:	$A(s,\pi^*) \geq A^*(s) - \varepsilon$	for $\varepsilon > 0$;
liminf average optimal:	$\liminf_{n \to \infty} \frac{1}{n} V_1^n(s,\pi^*) \geq \liminf_{n \to \infty} \frac{1}{n} V_1^n(s,\pi)$	$\forall \pi$;
limsup average optimal:	$\limsup_{n \to \infty} \frac{1}{n} V_1^n(s,\pi^*) \geq \limsup_{n \to \infty} \frac{1}{n} V_1^n(s,\pi)$	$\forall \pi$;
average-overtaking:	$\liminf_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} \big(V_1^n(s,\pi^*) - V_1^n(s,\pi)\big) \geq 0$	$\forall \pi$.

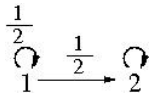
Each defining condition applies to all states, s. (See Lippman[18] and Flynn[11] for succinct introductions and how the objectives relate to each other.) Also see LP Myth 9 (p. 8).

The remaining DP Myths pertain to infinite horizon, stationary DPs.

DP Myth 15. *If there is a nearly optimal solution to a discounted DP with a finite number of states and decisions, it is an optimal policy.*

Blackwell[4] provides the following:

Counterexample. The state space is $S = \{1, 2\}$, and the decision sets are $X(s) = \{1, 2\}$ for each $s \in S$. The return functions are $r(1, x) = x$ and $r(2, x) = 0$. The state transition probabilities are: $q(1, 1; 1) = q(1, 2; 1) = \frac{1}{2}$, $q(1, 2; 2) = 1$, and $q(2, 2; x) = 1$.



State transitions for $x = 1$

State transitions for $x = 2$

Let π^x be any policy for which $\pi^x(1) = x$ for $x = 1, 2$ Then, starting in state 1:

$$V_\beta(\pi^1) = 1 + \tfrac{1}{2}\beta + \tfrac{1}{4}\beta^2 + \cdots = \frac{2}{2-\beta}$$

$$V_\beta(\pi^2) = 2.$$

Thus, π^2 is an optimal policy for each β, and $\lim_{\beta \to 1^-} V_\beta(\pi^1) = 2$. So, π^1 is nearly optimal but not optimal for any $\beta < 1$.

Also see Hordijk and Spieksma[15].

DP Myth 16. *For the discounted-return DP, there exists an ε-optimal policy for all $\varepsilon > 0$.*

Counterexample. Blackwell[5] provides the following. Let the state space be the unit interval: $S = [0, 1]$. For each $s \in S$, the decision set is $X(s) = [0, 1]$, and the state remains unchanged: $T(s, x) = s$ for all $s \in S, x \in X(s)$. Let B be a Borel subset of $[0, 1]^2$, and let D be the projection of B on S. Choose B such that D is not a Borel set, and define the return function:

$$r(s, x) = \begin{cases} 1 & \text{if } (s, x) \in B; \\ 0 & \text{otherwise.} \end{cases}$$

An optimal policy, π^*, is such that $(s, \pi^*(s)) \in B$, so the optimal value is $V_\beta(s, \pi^*) = \frac{1}{1-\beta}$. For any other policy, π, there exists $s \in D$ such that $r(s, \pi(s)) = 0$, so

$$V_\beta(s, \pi) \leq \beta + \beta^2 + \cdots = \frac{\beta}{1 - \beta}.$$

Hence, there is no ε-optimal policy for $0 < \varepsilon < 1$.

DP Myth 17. *There exists a stationary policy that is B-opt.*

While this is true for finite decision sets and state space, it fails for non-finite decision sets. Maitra[19] proved that finiteness is not necessary if the objective is the discounted model — that is, V_β is well defined for any particular $\beta \in [0, 1)$. He provided the following for this myth:

Counterexample. Let the state space be $\{1, 2 \dots\}$, and let the decision set be binary: $X = \{0, 1\}$, independent of the state. The returns are $r(s, 0) = c_s$ and $r(s, 1) = 0$; and, the state transitions are $T(s, 0) = s$ and $T(s, 1) = s + 1$. Choose $c_s > 0$ such that $\{c_s\} \uparrow c < \infty$ (for example, $c_s = c - \frac{1}{s}$).

Note that an advantage of choosing decision 0 is the positive immediate return, and the advantage of choosing decision 1 is the transition to the next state, which has a greater immediate return for the next decision. To illustrate, suppose $\pi^x(s) = x$ for all s. Then, $V_\beta(1, \pi^1) = 0$ and

$$V_\beta(1, \pi^0) = \sum_{t=1}^{\infty} \beta^{t-1} c_1 = \frac{c_1}{1 - \beta}.$$

More generally, suppose $\pi(s) = 0$ for all $s \in S \neq \emptyset$ and $\pi(s) = 1$ for $s \notin S$. Then, with $S = \{s_1, s_2, \dots\}$,

$$V_\beta(1, \pi) = \sum_{t=1}^{\infty} \beta^{t-1} c_{s_1} = \frac{1 - \beta^{s_1}}{1 - \beta} c_{s_1}.$$

Here is an optimal (stationary) policy for any fixed β:

$$\pi(s) = \begin{cases} 0 & \text{if } \beta^k c_{s+k} < c_s \text{ for all } k \geq 1; \\ 1 & \text{otherwise.} \end{cases}$$

This defines s_1 in the above equation for $V_\beta(1, \pi)$ as

$$s_1 = \min\{s : \beta^k c_{s+k} < c_s \text{ for all } k \geq 1\}.$$

There is thus an optimal policy for each fixed $\beta \in (0,1)$. However, there is no B-opt policy, as Maitra proves by contradiction. Suppose π^* is B-opt. He first proves

$$\lim_{\beta \to 1^-} (1-\beta) V_\beta(1, \pi^*) = c.$$

An implication of this equation is that $\pi^*(s)$ selects decision 0 a finite number of times when in state s in order to advance to the greater immediate returns that converge to c.

Maitra constructs another policy, π', that contradicts the optimality of π^* by showing $V_\beta(1, \pi') > V_\beta(1, \pi^*)$ for all β sufficiently close to 1.

Maitra[20] later provided the following:

Counterexample. Let there be just one state, and let the decision set at each time period be given by $X = \{1, 2, \dots\}$. Let the return function be $r(x) = 1 - \frac{1}{x}$, so there is no optimal policy.

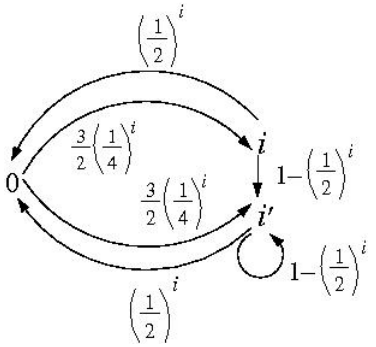
Thus, non-finite state or decision sets can result in there being no B-opt policy.

DP Myth 18. *If the average-return DP has an optimal policy, there exists a stationary policy that is optimal.*

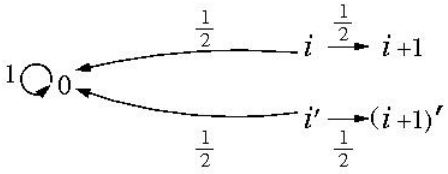
Fisher and Ross[8] provide the following:

Counterexample. Let $S = \{0, 1, 1', 2, 2', 3, 3', \dots, \}$, $X(s) = \{1, 2\}$ for $s = 1, 2, 3 \dots$, and $X(s) = \{1\}$ for $s = 0, 1', 2', 3', \dots$. The return values are $r(0, x) = -1$ and $r(s, x) = 0$ for $s \neq 0$. The state transition probabilities are:

For $s = i > 0$:	$q(0, i; 1) = q(0, i'; 1) = \frac{3}{2}\left(\frac{1}{4}\right)^i$
For $s = i$:	$q(i, 0; 1) = q(i', 0, 1) = \left(\frac{1}{2}\right)^i = 1 - q(i, i'; 1) = 1 - q(i', i'; 1)$
For $s = i$:	$q(i, 0; 2) = q(i, i+1; 2) = \frac{1}{2}$



State transitions for $x = 1$



State transitions for $x = 2$

Let $M_{ij}(\pi)$ denote the expected number of periods to reach state j, starting in state i and following policy π. For example, suppose π always selects decision 2. Then,

$$M_{00}(\pi) = 1; \quad \text{for } s \neq 0\text{: } M_{0s}(\pi) = \infty \text{ and } M_{s0}(\pi) = \textstyle\sum_{j=1}^{\infty} j(\frac{1}{2})^j = 2.$$

Let π^m be the policy that selects decision 2 at states $0 < i < m$ and decision 1 otherwise. Then,

$$M_{00}(\pi^m) = 1 + \sum_{i=1}^{\infty} \tfrac{3}{2} \left(\tfrac{1}{4}\right)^i M_{i0}(\pi^m) + \sum_{i=1}^{\infty} \tfrac{3}{2} \left(\tfrac{1}{4}\right)^i 2^i.$$

Fisher and Ross derive the fact that $M_{00}(\pi^m) < 5$ for all m, and $\lim_{m \to \infty} M_{00}(\pi^m) = 5$.

Let π be any stationary policy, and let $p_i = \mathbf{Pr}(x = 1 | s = i)$ define a randomized policy. Then,

$$\begin{aligned} M_{00}(\pi; p) &= 1 + \sum_{j=1}^{\infty} \tfrac{3}{2} \left(\tfrac{1}{4}\right)^j M_{j0}(\pi^m; p) + \sum_{j=1}^{\infty} \tfrac{3}{2} \left(\tfrac{1}{4}\right)^j 2^j. \\ M_{j0}(\pi; p) &= \sum_{m=j}^{\infty} p_m \prod_{k=j}^{m-1} (1 - p_k) M_{j0}(\pi^m; p) + 2 \prod_{k=j}^{\infty} (1 - p_k) < (2 + 2^j). \end{aligned}$$

Hence, $M_{00}(\pi; p) < 5$ for all stationary policies randomized by p. They proceed to construct the following non-stationary policy and prove that it has the optimal value of 5 for some $\{N_i\}$:

$$\pi_t(s) = \begin{cases} \pi^1(s) & \text{for} \quad 0 < t \le N_1 \\ \pi^2(s) & \text{for} \quad N_1 < t \le N_1 + N_2 \\ \vdots & \\ \pi^m(s) & \text{for} \quad \sum_{i=1}^{m-1} N_i < t \le \sum_{i=1}^{m} N_i \\ \vdots & \end{cases}$$

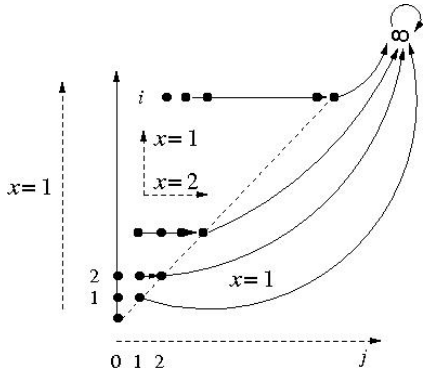
Ross[25] provides the theory needed to establish this, with application to the replacement process.

DP Myth 19. *There exists an average ε-optimal policy for $\varepsilon > 0$ that is optimal for the discounted-return model for β sufficiently close to 1.*

The intuition is that $\liminf_{\beta \to 1^-} V_\beta(s, \pi) = A(s, \pi)$, but Ross[25] provides the following:

Counterexample. Let $S = \{(i, j) : 0 \le j \le i,\ i \ge 1\} \cup \{\infty\}$ and $X(s) = \{1, 2\}$ for $s = (i, 0)$, $X(s) = \{1\}$ for $s = (i, j > 0), \infty$. The state transitions are deterministic:

$$\begin{aligned} T((i,0),1) &= (i+1, 0) \\ T((i,0),2) &= (i, 1) \\ T((i,j),1) &= (i, j+1) \quad \text{for } 0 < j < i \\ T((i,i),1) &= \infty = T(\infty, 1). \end{aligned}$$



Starting at state $(i, 0)$, one can move upward ($x = 1$) to some point, say $(i + h, 0)$, then go right ($x = 2$). There is no choice from that point; after reaching the 45° line (where $j = i + h$), one jumps to ∞ and stays there.

The immediate returns are $r(s, x) = 0$, except $r((i, 0), x) = 2$ and $r(\infty, x) = 1$. Suppose we start at state $(1, 0)$. Letting π^1 be the policy that always selects decision 1, there is no right turn, so the average return is 2. Otherwise, the average return is 1 (reaching ∞ and staying there forever).

Letting π be the policy that selects action 2 at state $(n, 0)$ and decision 1 otherwise, we have:

$$\begin{aligned} V_\beta((1,0),\pi) &= \sum_{k=0}^{n-1} \beta^k + 2 \sum_{k=2n}^{\infty} \beta^k = \frac{1 - \beta^n + 2\beta^{2n}}{1-\beta} \\ &< \frac{1}{1-\beta} \text{ for } n \text{ sufficiently large} \\ &= V_\beta((1,0),\pi^1). \end{aligned}$$

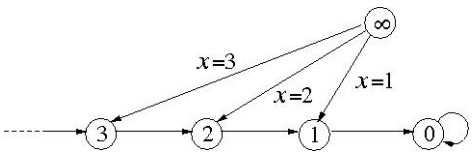
Hence, for $\beta \in (0, 1)$, $V_\beta((1, 0), \pi^1) \neq V^*_\beta((1, 0))$. This implies that the discounted-return model optimum does not approach the optimal average return, so it cannot become average ε-optimal.

DP Myth 20. *The successive value sequence converges to the infinite-horizon solution.*

The value sequence is $\{f_t(s)\}_{t=1,2,\ldots}$ for each $s \in S$. The myth asserts that this converges to the infinite-horizon solution, as $t \to \infty$.

Counterexample. Kreps and Porteus[17] provide the following. Let $S = \{0, 1, 2 \ldots\} \cup \{\infty\}$ and only one decision at each state, except $X(\infty) = \{1, 2, \ldots\}$. All returns are zero, except $r(x, 1) = -1$ (x is the only decision at $s = 1$).

The state transitions are deterministic, shown on the right. State 0 is absorbing, so the system eventually reaches it with a return of -1. Thus, $f_t(s) = -1$ for $s > 0$, but the infinite-horizon model never has to pay that cost, so $f_\infty(s) = 0$ for all $s \in S$.



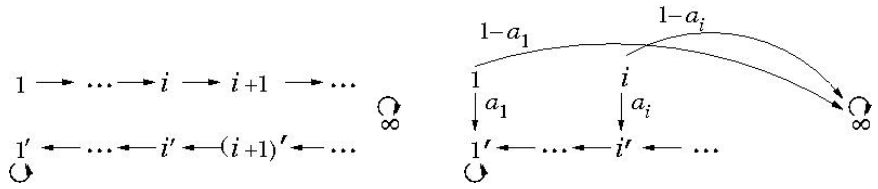
DP Myth 21. *If the optimal average-return is finite, there is an ε-optimal stationary policy for any $\varepsilon > 0$.*

Ross[26] provides the following:

Counterexample. Let the state space be $S = \{1, 1', 2, 2', \ldots, i, i', \ldots, \infty\}$. For each $i \in S$, the decision set is binary: $X(i) = \{0, 1\}$. For each $i' \in S$, the decision set is a singleton: $X(i') = \{0\}$. The state-transition probabilities are:

$$\begin{aligned} &\text{For } s = i: && q(i, i+1; 0) = 1,\ q(i, i'; 1) = a_i = 1 - q(i, \infty; 1), \\ &\text{For } s = i': && q(i', (i-1)'; 0) = 1 \text{ for } i \geq 2,\ q(1', 1; 0) = 1, \\ &\text{For } s = \infty: && q(\infty; \infty, s) = 1 \text{ for all } s \in S, \end{aligned}$$

where a_i satisfy: $0 < a_i < 1$ and $\prod_{i=1}^{\infty} a_i = \frac{3}{4}$.



State transitions for $x = 0$ State transitions for $x = 1$

The returns are $r(i, x) = 2$ and $r(i', x) = 0$ for all $i, i' \in S$ and all x.

Let the initial state be $s = 1$. Then, every stationary policy has a return of 2 in all but a finite number of time periods. This implies (by the bounded convergence theorem) that the average expected return is 2.

Let π be a (non-stationary) policy such that:

$$\begin{aligned} \pi_1(1) &= 1 \\ \pi_t(1') &= 0 \text{ for } t = 2, \ldots, T \\ \pi_{T+1}(1') &= 1. \end{aligned}$$

The average return equals:

$$\begin{array}{lll} 2 \text{ with probability} & 1 - \prod_{i=1}^{\infty} a_i & = \frac{1}{4} \\ 1 \text{ with probability} & \prod_{i=1}^{\infty} a_i & = \frac{3}{4}. \end{array}$$

Hence, the expected average return is $\frac{1}{2} + \frac{3}{4} = \frac{5}{4}$, so there is no ε-optimal stationary policy for $\varepsilon < \frac{3}{4}$.

DP Myth 22. *If a policy is B-opt, it optimizes the average return.*

Flynn[9] proved this for finite state spaces and provides the following for a non-finite state space:

Counterexample. Let $\{s_j\}_{j=1}^{\infty}$ be a real sequence such that

$$s^* \stackrel{\text{def}}{=} \liminf_{\beta \to 1^-} (1-\beta) \sum_{j=1}^{\infty} \beta^{j-1} s_j \; > \; \liminf_{n \to \infty} \frac{\sum_{j=1}^{n} s_j}{n} \stackrel{\text{def}}{=} s_*.$$

(Flynn establishes existence by an earlier theorem.) Let the state space be $\{0, s_1, s_2, \ldots\}$. The decision sets are binary, independent of the state: $X(s) = \{0, 1\}$ for all s. The state transition functions are deterministic: $T(s_j, x) = s_{j+1}$, $T(0, 0) = 0$, and $T(0, 1) = s_1$. The immediate returns are independent of the decision: $r(s_j, x) = s_j$ and $r(0, x) = \frac{1}{3}(s^* + 2s_*)$.

Let π^x denote a policy that always selects $x \in \{0, 1\}$. We have $V_\beta(s, \pi^1) = V_\beta(s, \pi)$ for $s \neq 0$. For $\pi(0) = 0$,

$$(1-\beta) V_\beta(0, \pi) = \tfrac{1}{3}(s^* + 2s_*) < s^* = (1-\beta) \sum_{j=1}^{\infty} \beta^{j-1} s_j = (1-\beta) V_\beta(0, \pi^1).$$

Hence, $V_\beta(0,\pi) < V_\beta(0,\pi^1)$, so π^1 is B-opt. However, we also have

$$A(0,\pi^0) = \tfrac{1}{3}(s^* + 2s_*) > s_* = A(0,\pi^1),$$

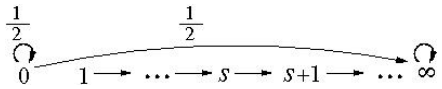
so π^1 does not maximize the average return, starting in state 0.

Also see Flynn[10]. *added*

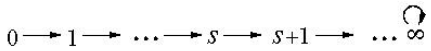
DP Myth 23. *If a stationary policy is B-opt, it is average-overtaking.*

This is true for finite state spaces, and Flynn[11] provides the following:

Counterexample. Let $S = \{0, 1, \ldots, \infty\}$ and $X(s) = \{0, 1\}$ for all $s \in S$. State transitions are stochastic only for $s = 0$ and $x = 0$: $q(0, 0; 0) = q(\infty, 0; 0) = \frac{1}{2}$. Otherwise, the transition is deterministic: $T(0, 1) = 1$ and $T(s, x) = s + 1$ for all $s > 0$, $x \in X(s)$ (note: $T(\infty, x) = \infty$).



State transitions for $x = 0$



State transitions for $x = 1$

Flynn establishes the existence of a sequence $\{a_j\}_{j=1}^{\infty}$ that satisfies:

$$\liminf_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{n} a_j = -1 \tag{DP.13}$$

$$\liminf_{\beta\to 1^-} \sum_{j=1}^{\infty} \beta^{j-1} a_j = \limsup_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{n} a_j = 0 \tag{DP.14}$$

Using this sequence, the returns are defined as: $r(s, x) = a_{s+1}$ for $s > 0$; $r(0, 1) = a_1$, $r(0, 0) = -\frac{1}{4}$, and $r(\infty, x) = 0$.

Let π^x denote the policy that always selects x. A B-opt policy is π^1 because $V^*_\beta(s) = V_\beta(s, \pi^1)$ for all s. (For $s = 0$, $V_\beta(0, \pi^1) = 0$ from (DP.14), whereas if $\pi^*(0) = 0$, $V_\beta(0, \pi^*) = \frac{-1/4}{1-\beta}$.) However, π^1 is not average-overtaking because the defining inequality fails for $s = 0$:

$$\liminf_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{n} \left(V_1^n(0,\pi^1) - V_1^n(0,\pi^0)\right) = \liminf_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \sum_{j=1}^{n} a_j + \tfrac{1}{2} = -\tfrac{1}{2}.$$

The last step uses (DP.13).

DP Myth 24. *Every stationary, 1-optimal policy is average-overtaking.*

Flynn[11] establishes this for finite state spaces and provides the following:

Counterexample. Let $S = \{0, 1, 2, \ldots\}$ and $X(s) = \{0, 1\}$ for all $s \in S$. The state transitions are deterministic: $T(0, 0) = 0$, $T(0, 1) = 1$, $T(s, x) = s + 1$ for $s > 0$.

State transitions for $x = 0$ State transitions for $x = 1$

The immediate returns are $r(0,0) = 0$, $r(0,1) = a_1$, and $r(s,x) = a_{s+1}$ for $s > 0$.

Let π^x denote the policy that selects x each time. Flynn proves that π^0 is both 1-optimal and average-overtaking, whereas π^1 is 1-optimal but not average-overtaking.

For $s > 0$, $V_\beta(s,\pi^0) = V_\beta(s,\pi^1) = V_\beta(s,\pi)\,\forall\pi$, so the defining inequality for 1-optimal is valid. Now consider $s = 0$. If $\pi^*(0) = 0$, $V_\beta(s,\pi^0) = V_\beta(s,\pi), \forall\pi$ and π^0 is optimal. Otherwise, applying (DP.14), we have:

$$\lim_{\beta\to 1^-}\big(V_\beta(0,\pi^0) - V_\beta(0,\pi)\big) = -\lim_{\beta\to 1^-}\sum_{j=1}^{\infty}\beta^{j-1}a_j = 0.$$

Hence, π^0 is 1-optimal. Similarly, if $\pi^*(0) = 1$, $V_\beta(s,\pi^1) = V_\beta(s,\pi)$ and π^1 is optimal. Otherwise, we have:

$$\lim_{\beta\to 1^-}\big(V_\beta(0,\pi^1) - V_\beta(0,\pi)\big) = \lim_{\beta\to 1^-}\sum_{j=1}^{\infty}\beta^{j-1}a_j = 0.$$

Hence, π^1 is 1-optimal.

For any policy, π, $V_1^n(s,\pi) = V_1^n(s,\pi^0) = V_1^n(s,\pi^1)$ for $s > 0$. Consider $s = 0$. If $\pi(0) = 0$, π^1 is not average-overtaking because (DP.13) yields:

$$\liminf_{N\to\infty}\frac{1}{N}\sum_{n=1}^{N}\big(V_1^n(0,\pi^1) - V_1^n(0,\pi)\big) = -1.$$

Whereas, if $\pi(0) = 1$,

$$\liminf_{N\to\infty}\frac{1}{N}\sum_{n=1}^{N}\big(V_1^n(0,\pi^0) - V_1^n(0,\pi)\big) = 1,$$

so π^0 is average-overtaking.

DP Myth 25. *If a policy is both B-opt and average-overtaking, it is liminf average optimal.*

The assertion is true for finite state spaces, and Flynn[11] provides the following:

Counterexample. Let the state space, decision sets, and state transition functions be the same as in DP Myth 24, but with returns: $r(0,x) = 0$ and $r(s,x) = v_s$ for $s > 0$, where Flynn establishes the existence of $\{v_j\}_{j=1}^{\infty}$ that satisfies:

$$\lim_{N\to\infty}\frac{1}{N}\sum_{n=1}^{N}\sum_{j=1}^{n}v_j = \infty, \qquad \liminf_{n\to\infty}\frac{1}{n}\sum_{j=1}^{n}v_j < 0. \tag{DP.15}$$

Let π^x denote the policy that always selects decision x. π^1 is average-overtaking because $\forall\pi$:

$$\begin{aligned} s > 0 &\rightarrow V_1^n(s,\pi^1) = V_1^n(s,\pi) \\ s = 0, \pi(0) = 1 &\rightarrow V_1^n(0,\pi^1) = V_1^n(0,\pi) \\ s = 0, \pi(0) = 0 &\rightarrow V_1^n(0,\pi) = 0 \text{ and } V_1^n(0,\pi^1) = \sum_{j=1}^{n} v_j > 0 \text{ for } n \text{ sufficiently large.} \end{aligned}$$

Further, π^1 is B-opt because $\forall\pi$:

$$\liminf_{N\to\infty} \frac{1}{N}\sum_{n=1}^{N} V_1^n(s,\pi) \le \liminf_{\beta\to 1^-} V_\beta(s,\pi) \le \limsup_{\beta\to 1^-} V_\beta(s,\pi) \le \limsup_{N\to\infty} \frac{1}{N}\sum_{n=1}^{N} V_1^n(s,\pi).$$

However, π^1 is not liminf average optimal because (DP.15) yields:

$$\liminf_{n\to\infty} \frac{1}{n} V_1^n(0,\pi^1) = \liminf_{n\to\infty} \frac{1}{n}\sum_{j=1}^{n} v_j < 0 = \liminf_{n\to\infty} \frac{1}{n} V_1^n(0,\pi^0).$$

DP Myth 26. *If a policy is both B-opt and average-overtaking, it is limsup average optimal.*

The assertion is true for finite state spaces, and Flynn[11] provides the following:

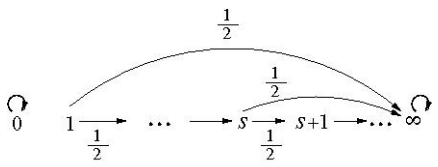
Counterexample. Let the state space, decision sets, and state transition functions be the same as in DP Myth 25, but with returns: $r(0,x) = 0$ and $r(s,x) = -v_s$ for $s > 0$, satisfying (DP.15). Using a similar proof, π^0 is both B-opt and average-overtaking, but it is not limsup average optimal because:

$$\limsup_{n\to\infty} \frac{1}{n} V_1^n(0,\pi^0) = 0 < -\liminf_{n\to\infty} \frac{1}{n}\sum_{j=1}^{n} v_j = \limsup_{n\to\infty} \frac{1}{n}\sum_{j=1}^{n} -v_j = \limsup_{n\to\infty} \frac{1}{n} V_1^n(0,\pi^1).$$

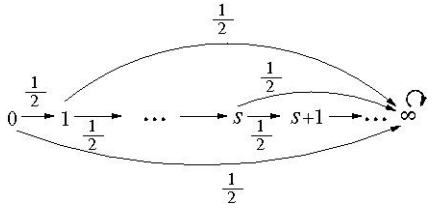
DP Myth 27. *If a policy is B-opt among stationary policies, it optimizes the average return among stationary policies.*

Flynn[11] establishes this for finite state spaces and provides the following:

Counterexample. Let $S = \{0, 1, \ldots, \infty\}$ and $X(s) = \{0, 1\}$ for all $s \in S$. State transitions are: $q(\infty,\infty;x) = q(0,0;1) = 1$, $q(0,1;1) = q(0,\infty;1) = \frac{1}{2}$, and $q(s,s+1;x) = q(s,\infty;x) = \frac{1}{2}$ for $s > 0$.



State transitions for $x = 0$



State transitions for $x = 1$

Letting $\{v_j\}$ satisfy (DP.15), define the returns independent of the decisions: $r(0,x) = r(\infty,x) = 0$ and $r(s,x) = -2^s v_s$ for $s > 0$ ($s \neq \infty$).

Under any stationary policy, the system is absorbed in state ∞ with probability 1. Following the same arguments as in the counterexample to DP Myth 25, Flynn proves π^0 is B-opt, but not limsup average optimal.

DP Myth 28. *If a policy is average-overtaking among stationary policies, it optimizes the average return among stationary policies.*

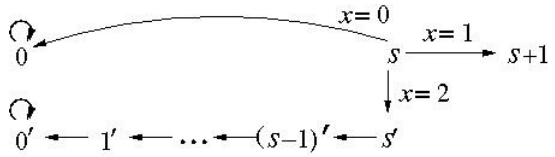
Flynn[11] establishes this for finite state spaces and provides the following:

Counterexample. Let $S = \{0, 1, \ldots, \infty\}$ and $X(s) = \{0, 1\}$ for all $s \in S$. State transition functions are as in DP Myth 27, but the immediate returns are $r(s,x) = 2^{s+1}a_{s+1}$, $r(\infty,x) = 0$, $r(0,0) = -\frac{1}{4}$, and $r(0,1) = 2a_1$, where $\{a_j\}$ satisfies (DP.13). Using the same arguments as in DP Myth 23, π^1 is B-opt, but not average-overtaking.

DP Myth 29. *We can approximate an infinite horizon Markov decision process with a sufficiently long, finite horizon.*

Hinderer[14] first raised this issue for both discounted and average return models. Flynn[12] provides the following:

Counterexample. Let $S = \{0, 0', 1, 1', 2, 2', \ldots\}$ and $X(s) = \{0, 1, 2\}$ for all $s \in S$. The state transitions are deterministic: $T(s,0) = 0$, $T(s,1) = s+1$, $T(s,2) = s'$, $T(s',x) = (s-1)'$ for $s = 1, 2, \ldots$.

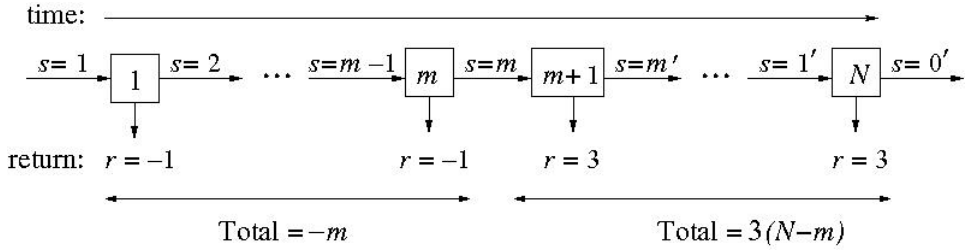


The immediate returns are $r(0,x)$, $r(s,x) = -1$ for $s = 0', 1, 2, \ldots$, and $r(s,x) = 3$ for $s = 1', 2', \ldots$.

Let π^0 be the policy of always selecting $x = 0$, and note that the infinite horizon solution is π^0 with $A^*(s) = 0$ for all s. Let x^N be an optimal policy for N time periods, and let $m = \lfloor \frac{N}{2} \rfloor$. Then,

$$x^N(s) = \begin{cases} 1 & \text{if } s \leq m; \\ 2 & \text{if } s > m. \end{cases}$$

So, starting in state 1, we have $V_1^N(1, x^N) = N$ if N is even, and $V_1^N(1, x^N) = N + 2$ if N is odd.



Optimal State and Return Sequences for N-period Horizon

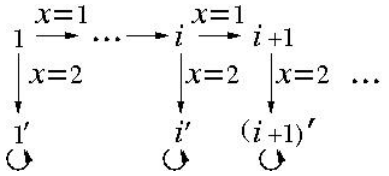
Consider x^N as a finite approximation for the infinite horizon. As N becomes large, $\{V_1^N(1,x^N) - V_1^N(1,\pi^0)\} \uparrow \infty$, so x^N is a poor approximation. From the other view, π^0 becomes increasingly less desirable as an approximation to the N-horizon DP as $N \to \infty$. Moreover, the average return for the N-horizon approaches 1, whereas the average return for the infinite horizon DP is 0.

DP Myth 30. *A discounted-return stationary DP with finite decision space for each state has a pure, stationary policy that is optimal.*

Hordijk and Tijms[16] provide the following:

Counterexample. Let the state space be given by the denumerable set, $S = \{1, 1', 2, 2', \dots\}$.

Let the decision sets be $X(s) = \{1, 2\}$ for $s = 1, 2, \dots$ and $X(s) = \{1\}$ for $s = 1', 2', \dots$ Let the state transitions be $T(i,1) = i+1$ and $T(i,2) = T(i',1) = i'$ for $i = 1, 2, \dots$ The immediate returns are $r(s,1) = 0$ for all s and $r(i,2) = \beta^{-i}\left(1 - \frac{1}{i}\right)$ for $i = 1, 2, \dots$



State Transitions

There are two pure, stationary policies:

$$\pi^1(s) = 1 \text{ and } \pi^1(s') = 1$$
$$\pi^2(s) = 2 \text{ and } \pi^2(s') = 1$$

Then,

$$\begin{array}{lll} V_\beta(s,\pi^1) & = 0 & \text{for all } s \\ V_\beta(i,\pi^2) & = \beta^{-i}\left(1 - \frac{1}{i}\right) \text{ and } V_\beta(i',\pi^2) = 0 & \text{for } i = 1, 2, \dots \end{array}$$

π^2 is optimal among pure, stationary policies, and $V_\beta(i,\pi^2) < \beta^{-i}$ for all $i = 1, 2 \dots$

Now consider the following randomized policy: $P_i(t) =$ probability that $\pi_t(i) = 1$ when the system is in state i at time t. Suppose $P_i(t) < 1$ for at least one t. Then,

$$V_\beta(i,\pi) = \sum_{t=0}^{\infty} \beta^t (1 - P_i(t)) \beta^{-(i+t)} \prod_{n=0}^{t-1} P_i(n) \left(1 - \frac{1}{i+t}\right).$$

This yields

$$V_\beta(i,\pi) = \beta^{-i} \sum_{t=0}^{\infty} \beta^t (1 - P_i(t)) \prod_{n=0}^{t-1} P_i(n) \left(1 - \frac{1}{i+t}\right).$$

Using the identity

$$\sum_{t=0}^{\infty} \beta^t (1 - P_i(t)) \prod_{n=0}^{t-1} P_i(n) = 1 - \prod_{t=0}^{\infty} P_i(t),$$

we obtain

$$V_\beta(i,\pi) < \beta^{-i}.$$

Consider the policy with

$$\pi(i) = 1 \text{ for } i = 1, \dots, m-1 \text{ and } \pi(i) = 2 \text{ for } i = m, m+1, \dots$$

Then, $V_\beta(i,\pi) = \beta^{-i}\left(1-\frac{1}{m}\right)$ for all $i \geq 1$. Letting $m \to \infty$, we see that

$$\sup_\pi V_\beta(i,\pi) = \beta^{-i}.$$

Thus, the supremum cannot be achieved (finitely), so there is no optimal policy.

DP Myth 31. *There exists an optimal policy that is stationary for the dynamic portfolio problem of minimizing variance subject to an expected return requirement.*

This is known as Tobin's theorem, one of the first results in the theory of optimal multi-period portfolio selection. The myth was accepted as true by Samuelson, among other leading economists, until Stevens[31] provided the following:

Counterexample. Let there be two periods with expected returns denoted by E_1, E_2. Further, let the return requirement be 4. Tobin's problem is thus:

$$\min\ \left(\sigma_1^2 + (1+E_1)^2\right)\left(\sigma_2^2 + (1+E_2)^2\right) :\ (1+E_1)(1+E_2) = 4.$$

Consistent with Tobin's development, Stevens relates variance with expected return as: $\sigma_i^2 = E_i^2$ for $i = 1, 2$. The objective thus becomes:

$$\min\ \left(E_1^2 + (1+E_1)^2\right)\left(E_2^2 + (1+E_2)^2\right) :\ (1+E_1)(1+E_2) = 4.$$

Among three solutions to the first-order conditions, Tobin's theorem chooses $E_1 = E_2 = 1$, with an objective value of 25. A better solution is

$$E_1 = 1\tfrac{1}{4} + \tfrac{1}{8}\sqrt{68} \text{ and } E_2 = 1\tfrac{1}{4} - \tfrac{1}{8}\sqrt{68}.$$

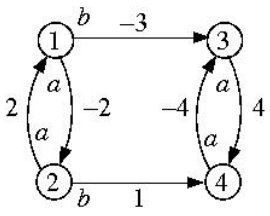
This has an objective value of 24.5.

Stevens suggested that Tobin may have been thinking that the functional symmetry and some form of convexity are sufficient to conclude $E_1 = E_2$. (See [13] for elaboration.)

DP Myth 32. *Denardo's policy improvement algorithm computes a 1-optimal policy.*

Counterexample. O'Sullivan[23, Appendix A] provides the following: Let $S = \{1,2,3,4\}$, $X(1) = X(2) = \{a,b\}$, and $X(3) = X(4) = \{a\}$. The state transitions are deterministic:

$T(1,a) = 2$	$r(1,a) = -2$
$T(1,b) = 3$	$r(1,b) = -3$
$T(2,a) = 1$	$r(2,a) = 2$
$T(2,b) = 4$	$r(2,b) = 1$
$T(3,a) = 4$	$r(3,a) = 4$
$T(4,a) = 3$	$r(4,a) = -4$



The only 1-optimal policy is $\pi^a = (a,a,a,a)$ because

$$\liminf_{\beta \to 1^-}\left(V_\beta(s,\pi^a) - V_\beta(s,\pi)\right) = \begin{cases} \frac{2}{1+\beta} - 1 > 0 & \text{if } \pi(s) = b \text{ and } s = 1,2 \\ 0 & \text{otherwise.} \end{cases}$$

Starting with $\pi^0 = (b, b, a, a)$, the algorithm computes a solution to $Q_{\pi^0} v^1 = v^0_{\pi^0}$, where $Q_{\pi^0} = P(\pi^0) - I$ ($P(\pi^0)$ is the state transition probability matrix using policy π^0), and

$$v^0_{\pi^0} = \lim_{\beta \to 1^-} V_\beta(\bullet, \pi^0) = \lim_{\beta \to 1^-} \begin{pmatrix} -3 + \frac{4\beta}{1+\beta} \\ 1 + \frac{4\beta}{1+\beta} \\ \frac{4\beta}{1+\beta} \\ -\frac{4\beta}{1+\beta} \end{pmatrix} = \begin{pmatrix} -1 \\ 3 \\ 2 \\ -2 \end{pmatrix}.$$

Such a solution is given by $v^1 = (1, -1, 0, 2)^\intercal$:

$$\begin{bmatrix} -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{pmatrix} 1 \\ -1 \\ 0 \\ 2 \end{pmatrix} = \begin{pmatrix} -1 \\ 3 \\ 2 \\ -2 \end{pmatrix}.$$

Then, π^1 is the solution to a maximum-reward-rate problem among all policies π with $Q_\pi v^1 = v^0_\pi$. However, only π^0 has $Q_{\pi^0} v^1 = v^0_{\pi^0}$, so $\pi^1 = \pi^0$. Finally, the algorithm seeks a 1-optimal policy π^2 that is transient on the same states as $\pi^1 (= \pi^0)$. Thus, π^2 is transient on states 1 and 2 so $\pi^2 \neq \pi^a = (a, a, a, a)$ as this policy is recurrent on states 1 and 2. Thus, π^2 is not 1-optimal.

DP References

[1] E. J. Anderson. A note on the dynamic lot-size model with uncertainty in demand and supply processes. *Management Science*, 35(5):635–640, 1989.

[2] K. S. Azoury and B. L. Miller. A comparison of the optimal ordering levels of Bayesian and non-Bayesian inventory models. *Management Science*, 30(8):993–1003, 1984.

[3] R. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, NJ, 1957.

[4] D. Blackwell. Discrete dynamic programming. *The Annals of Mathematical Statistics*, 33(2):719–726, 1962.

[5] D. Blackwell. Discounted dynamic programming. *The Annals of Mathematical Statistics*, 36(1):226–235, 1965.

[6] K-S. Choi, J. G. Dai, and J-S. Song. On measuring supplier performance under vendor-managed-inventory programs in capacitated supply chains. *Manufacturing & Service Operations Management*, 6(1):53–72, 2004.

[7] A. Federgruen and C. Lee. The dynamic lot size model with quantity discount. *Naval Research Logistics*, 37:707–713, 1990.

[8] L. Fisher and S. M. Ross. An example in denumerable decision processes. *The Annals of Mathematical Statistics*, 39(2):674–675, 1968.

[9] J. Flynn. Averaging versus discounting in dynamic programming: A counterexample. *Annals of Statistics*, 2(2):411–413, 1974.

[10] J. Flynn. Steady state policies for a class of deterministic dynamic programming models. *SIAM Journal on Applied Mathematics*, 28(1):87–99, 1975.

[11] J. Flynn. Conditions for the equivalence of optimality criteria in dynamic programming. *Annals of Statistics*, 4(5):936–953, 1976.

[12] J. Flynn. On optimality criteria for dynamic programs with long finite horizons. *Journal of Mathematical Analysis and Applications*, 76(1):202–208, 1980.

[13] H. J. Greenberg and W. P. Pierskalla. Symmetric mathematical programs. *Management Science*, 16(5):309–312, 1970.

[14] K. Hinderer. Estimates for finite-stage dynamic programs. *Journal of Mathematical Analysis and Applications*, 55(1):207–238, 1976.

[15] A. Hordijk and F. Spieksma. Are limits of α-discounted optimal policies Blackwell optimal? A counterexample. *Systems & Control Letters*, 13(1):31–41, 1989.

[16] A. Hordijk and H. C. Tijms. A counterexample in discounted dynamic programming. *Journal of Mathematical Analysis and Applications*, 39(2):455–457, 1972.

[17] D. M. Kreps and E. L. Porteus. On the optimality of structured policies in countable stage decision processes. II: Positive and negative problems. *SIAM Journal on Applied Mathematics*, 32(2):457–466, 1977.

[18] S. A. Lippman. Criterion equivalence in discrete dynamic programming. *Operations Research*, 17(5):920–923, 1969.

[19] A. Maitra. Dynamic programming for countable state systems. *Sankhyā, The Indian Journal of Statistics, Series A*, 27:241–248, 1965.

[20] A. Maitra. A note on undiscounted dynamic programming. *The Annals of Mathematical Statistics*, 37(4):1042–1044, 1966.

[21] S. Martello and P. Toth. *Knapsack Problems: Algorithms and Computer Implementations*. John Wiley & Sons, New York, NY, 1990.

[22] L. G. Mitten. Composition principles for synthesis of optimal multistage processes. *Operations Research*, 12(4):610–619, 1964.

[23] M. J. O'Sullivan. *New Methods for Dynamic Programming Over an Infinite Time Horizon*. Ph. D. thesis, Stanford University, Stanford, CA, 2002.

[24] E. Porteus. An informal look at the principle of optimality. *Management Science*, 21(11):1346–1348, 1975.

[25] S. M. Ross. Non-discounted denumerable Markovian decision models. *The Annals of Mathematical Statistics*, 39(2):412–423, 1968.

[26] S. M. Ross. On the nonexistence of ε-optimal randomized stationary policies in average cost Markov decision models. *The Annals of Mathematical Statistics*, 42(5):1767–1768, 1971.

[27] K. Seth. Optimal service policies, just after idle periods, in two-server heterogeneous queuing systems. *Operations Research*, 25(2):356–360, 1977.

[28] J-S. Song. The effect of leadtime uncertainty in a simple stochastic inventory model. *Management Science*, 40(5):603–613, 1994.

[29] J-S. Song. On the order fill rate in a multi-item, base-stock inventory system. *Operations Research*, 46(6):831–845, 1998.

[30] J-S. Song and Y. Zhao. The value of component commonality in a dynamic inventory system with lead times. *Manufacturing & Service Operations Management*, (to appear), 2009.

[31] G. V. G. Stevens. On Tobin's multiperiod portfolio theorem. *Review of Economic Studies*, 39(4):461–468, 1972.

[32] R. R. Weber and S. Stidham, Jr. Optimal control of service rates in networks of queues. *Advances in Applied Probability*, 19(1):202–218, 1987.

[33] W. Whitt. Deciding which queue to join: Some counterexamples. *Operations Research*, 34(1):55–62, 1986.

[34] J. Xu and L. L. Lu. The dynamic lot size model with quantity discount: Counterexamples and correction. *Naval Research Logistics*, 45:419–422, 1998.

[35] W. I. Zangwill. From EOQ towards ZI. *Management Science*, 33(10):1209–1223, 1987.

Nonlinear Programming

A nonlinear program (NLP) has the general form:

$$\max\ f(x) :\ x \in X,\ g(x) \leq 0,\ h(x) = 0,$$

where $\emptyset \neq X \subseteq \mathbb{R}^n$, $f : X \to \mathbb{R}$, $g : X \to \mathbb{R}^m$, $h : X \to \mathbb{R}^M$. This is the most general form with no restrictions on the domain (X) or the nature of the functions, except to have at least one of the functions be nonlinear. Historically, the domain is a simple convex set, like all of $\mathbb{R}^n$ or $\mathbb{R}^n_+$. If the functions are differentiable, methods of calculus are used to establish optimality conditions and provide a foundation for algorithm design.

We refer to some special NLPs:

Convex (CP). X is closed and convex, f is concave, g is convex, and h is affine.

Quadratic (QP). $\max\ x^\intercal Q x + cx :\ Ax \leq b$.

Typically, Q is assumed to be symmetric, but this is no restriction because the same quadratic form results upon replacing Q with $\frac{1}{2}(Q + Q^\intercal)$.

NLP Myth 1. *If f has continuous n^{th}-order derivatives, local behavior of f can be approximated by Taylor's series:*

$$f(x + th) = f(x) + t\nabla f(x)h + \tfrac{1}{2}t^2 h^\intercal \nabla^2 f(x)h + \ldots,$$

where h is a vector with $||h|| = 1$, t is a scalar, $\nabla f(x)$ is the gradient of f, and $\nabla^2 f(x)$ is the hessian of f.

The reason this is not correct is that, although Taylor's series might converge under the stated assumptions, it need not be to the correct value.

Counterexample. Let $f(x) = e^{-\frac{1}{x^2}}$ for $x \neq 0$, and $f(0) = 0$ (x is a scalar). It can be shown that f is infinitely differentiable everywhere. At $x = 0$, the n^{th} derivative is 0 for all n. Thus, the Taylor series converges to 0, which gives the approximation, $f(t) = 0$ (with $h = 1$) for all t. This is incorrect for $t \neq 0$.

This myth is used all too often in textbooks. The correct assumption is that f is *analytic*. Then, by definition, the Taylor's series does converge to the correct value of the function, so it can be used for approximation when proving theorems — viz., that necessary conditions for x to be an unconstrained minimum are: $\nabla f(x) = 0$ and $h^\intercal \nabla^2 f(x) h \geq 0$ for all h.

NLP Myth 2. *Given differentiable functions, an optimal point must satisfy the Lagrange Multiplier Rule.*

We are given

$$\text{NLP: } \max\ f(x) :\ x \in \mathbb{R}^n,\ g(x) \leq 0,\ h(x) = 0,$$

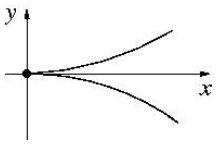
where f, g, h are differentiable functions on $\mathbb{R}^n$. For just equality constraints (g vacuous), the

Lagrange Multiplier Rule (LMR) states: x^* is optimal only if there exists λ such that:

$$\nabla f(x^*) - \lambda \nabla h(x^*) = 0.$$

Counterexample. Consider max $-x : x^3 - y^2 = 0$. The optimum is at $(x^*, y^*) = (0, 0)$. The LMR requires $-1 - \lambda\, 0 = 0$ for some λ, which is impossible.

cusp: $h(x, y) = x^3 - y^2$



The LMR for equality constraints is valid with the *constraint qualification*: $\nabla h(x^*)$ has full row rank. This is what Lagrange assumed, using the Implicit Function Theorem to prove the necessity of the LMR. (Affine functions need no constraint qualification.)

One extension of the LMR to include inequality constraints is simple: there exists λ, μ such that:

$$\mu \geq 0,\ \mu_i > 0 \rightarrow g_i(x^*) = 0 \tag{NLP.16}$$

$$\nabla f(x^*) - \mu \nabla g(x^*) - \lambda \nabla h(x^*) = 0 \tag{NLP.17}$$

The extended Lagrange constraint qualification is simply

$$\text{rank}\left(\begin{bmatrix} \nabla g_A(x^*) \\ \nabla h(x^*) \end{bmatrix}\right) = |A| + M,$$

where A is the set of *active constraints* among the inequalities — that is, $A = \{i : g_i(x^*) = 0\}$ — and M is the number of equality constraints.

The classical extension and deeper meaning into saddlepoint equivalence by Kuhn and Tucker[39] gave a weaker constraint qualification, but it is violated by the following:

Counterexample. max $x : x \geq 0,\ y - (1 - x)^3 \leq 0,\ -y \leq 0$. The solution is at $(1, 0)$. The LMR requires $(\mu_1, \mu_2) \geq 0$ to satisfy:

$$\begin{aligned} 1 - \mu_1\, 3(1 - x)^2 &= 0 \\ -\mu_1 + \mu_2 &= 0 \end{aligned}$$

The first equation is impossible at $(1, 0)$.

Here is another counterexample with g convex: max $x : x^2 \leq 0$. The problem here is that the strict interior is empty — that is, $\{x : g(x) < 0\} = \emptyset$.

Since linear constraints require no constraint qualification for the LMR to be valid, another myth was given by Mond[49]:

> *The LMR is valid if the constraints are linear, except one is convex with a non-empty strict interior.*

Counterexample. Mond provides the following:

$$\min\ x_1 : x_1 \geq 0,\ x_2 \geq 1,\ x_1 + x_3 \geq 1,\ x_2^2 + x_3^2 \leq 1.$$

We have $\{x : x_2^2 + x_3^2 < 1\} \neq \emptyset$, so the constraint qualification holds. However, the optimal solution is $x = (1, 1, 0)$, and the LMR requires that there exist $\lambda \geq 0$ such that $\lambda_1 = 0$ and

$$1 - \lambda_3 = 0,\ -\lambda_2 + 2 = 0,\ -\lambda_3 + 0\lambda_4 = 0.$$

The first and last equations are inconsistent, so the LMR fails.

The true statement is that $\{x : Ax \geq b,\ g(x) < 0\} \neq \emptyset$ — that is, the strict interior is defined over the entire feasible region.

NLP Myth 3. *A point that satisfies the Lagrange multiplier necessary conditions is a local optimum.*

Most people know this is a myth because the Lagrange (a.k.a., Kuhn-Tucker-Karush) conditions hold at stationary points that are not minima or maxima (for example, at a saddle point). This is included here, however, because it appears too often in textbooks and even some research articles. Those not expert in mathematical programming are told that an algorithm converges to a local optimum when, in fact, it converges to a point that satisfies the Lagrange multiplier conditions. (Methods of descent can rule out converging to a "pessimal" solution — that is, to a max when seeking a min — if it moves from its initial point.)

Counterexample. min $x^2 - y^2 : -1 <= x, y <= 1$. A Lagrange point is at $(x, y) = (0, 0)$ with all four multipliers = 0, but this is not a local min (or max) of the NLP. It is a saddlepoint.

NLP Myth 4. *Suppose f is analytic and x is a minimum of f. Then, $\nabla f(x) = 0$, and if $h^\intercal \nabla^2 f(x) h = 0$ for all h, it is necessary that all terms of the third derivative shall vanish. In that case, if the fourth-order term is positive, the point is a minimum.*

This is a classical error, made by the great Lagrange. A complete discussion is given by Hancock[34]. (Qi[59] provides a qualification that makes an "extended Lagrange claim" valid.)

The proposition is a natural extension of the (correct) result for one variable: *the first non-vanishing derivative must be of even order; and, it is positive for a minimum and negative for a maximum.* For two variables, however, we have a problem with the ambiguous case.

For notational convenience, translate the solution to the origin, and suppose $f(0) = 0$. Then, Taylor's expansion is:

$$f(h, k) = \tfrac{1}{2}(Ah^2 + 2Bhk + Ck^2) + 3^{\text{rd}}\text{-order terms},$$

where the quadratic form coefficients (A, B, C) are the associated second partial derivatives of f, evaluated at (0,0). When $B^2 - 4AC > 0$, the origin is a proper (local) minimum; when $B^2 - 4AC < 0$, the origin is a proper (local) maximum. The ambiguous case, when $B^2 - 4AC = 0$, is at issue. Here is where Lagrange claimed the 3^{rd}-order term must vanish, and that the sign of the 4^{th}-order term (if it does not vanish) can then determine whether the point is a (local) minimum or a maximum. The following is a special case of a counterexample found by Peano.

Counterexample. $f(x,y) = (y-x^2)(y-2x^2)$. We have first derivatives: $f_x = -6xy + 8x^3$ and $f_y = 2y - 3x^2$. These vanish at $(0,0)$, so we proceed to the second derivatives:

$$\nabla^2 f(x,y) = \begin{bmatrix} -6y+24x^2 & -6x \\ -6x & 2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 2 \end{bmatrix} \text{ at } (0,0).$$

This is the ambiguous case, where the hessian is positive semi-definite at the origin. Let the change in the y-direction be zero, and let the change in x be t, so the quadratic form is $(t,0)\nabla^2 f(0)(t,0)^\intercal = 0$ for all t. We proceed to third derivatives, but since we maintain no change in the y-direction, we need to compute derivatives of only x:

$$f_{xxx} = 48x \text{ and } f_{xxxx} = 48,$$

so $f(t,0) = 48t^4 > 0$ for all t. According to the myth, this implies f achieves a minimum at $(0,0)$; however, consider $y = {}^3\!/_2\, x^2$. Along this parabola, $f(x,y) = -{}^1\!/_4\, x^4$, which is negative for $x \neq 0$. Thus, $(0,0)$ is not a local minimum of f.

NLP Myth 5. *Given* $\min\{f(x,y) = g(x) + h(y) : ay = g(x)\}$*, we can equivalently solve* $\min\{ay + h(y)\}$.

Counterexample. The following is given by Bloom[6]. Determine the shortest distance from the point $(0,5)$ to the parabola defined by $16y = x^2$. Using the square distance as the objective function, our problem is:

$$\min\ x^2 + (y-5)^2 : 16y = x^2.$$

Substituting x^2, the unconstrained "equivalent" is given by:

$$\min\ 16y + (y-5)^2.$$

The only critical point (where $f' = 0$) is at $y = -3$. However, this produces an imaginary value of x, so the minimum does not exist. The problem is that we cannot simply replace x^2 with $16y$; we must divide the problem into the cases: $x \geq 0$ and $x \leq 0$.

The Lagrange Multiplier rule does not run into any problem. The Lagrange conditions for the original problem are:

$$2x + \lambda 2x = 0,\ 2(y-5) - 16\lambda = 0,\ \text{and } 16y = x^2.$$

With $x = 0$ we obtain $y = 0 \Rightarrow \lambda = -\frac{5}{8}$. With $x \neq 0$, we obtain $\lambda = -1 \Rightarrow y = -3 \Rightarrow$ contradiction. Hence, the minimum is at $(0,0)$.

NLP Myth 6. *A smooth surface with one critical point that is a local, but not a global, minimum must have a second critical point.*

Let the surface be given by $(x,y,f(x,y))$ for $x,y \in \mathbb{R}$. Define "smooth" as f is infinitely differentiable. The intuition stems from the fact that the statement is true in one variable.

Counterexample. Ash and Sexton[4] provide the following:

$$f(x,y) = -\frac{1}{1+x^2} + (2y^2 - y^4)\left(e^x + \frac{1}{1+x^2}\right).$$

The origin is a local, but not a global, minimum with

$$\nabla f(0,0) = (0,0),\ \nabla^2 f(0,0) = \begin{bmatrix} 2 & 0 \\ 0 & 8 \end{bmatrix},\ f(0,0) = -1 > f(0,2) = -17.$$

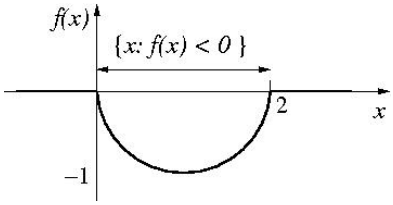
There are no other critical points.

NLP Myth 7. *If f is continuous, the closure of its strict interior equals its level set. That is,* $\mathrm{cl}\{x : f(x) < 0\} = \{x : f(x) \leq 0\}$.

One importance of this in stability — see NLP Myth 8.

Counterexample. Let f be the following function on $\mathbb{R}$:

$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ (x-1)^2 - 1 & \text{if } 0 \leq x < 2 \\ 0 & \text{if } 2 \leq x \end{cases}$$



f is continuous (and quasiconvex). However, the strict interior of the 0-level set is $(0,2)$, so its closure is only $[0,2]$. We lose the flat portions in the tails.

NLP Background — Semi-continuity

Some myths involve continuity properties of the optimal value as a function of the right-hand side. This requires us to consider the feasibility region a *point-to-set map*, as follows. Let $X(b) = \{x \in X : g(x) \leq b\}$ denote the feasible region, and let $B = \{b : X(b) \neq \emptyset\}$. The optimal value function is $f^*(b) = \sup\{f(x) : x \in X(b)\}$, and the optimality region is $X^*(b) = \{x \in X(b) : f(x) = f^*(b)\}$. Unless stated otherwise, we are interested in continuity properties at $b = 0$, and we assume $0 \in B$.

The optimal value function is *lower semi-continuous* (lsc) at $b = 0$ if

$$\liminf_{b \to 0} f^*(b) \geq f^*(0).$$

The optimal value function is *upper semi-continuous* (usc) at $b = 0$ if

$$\limsup_{b \to 0} f^*(b) \leq f^*(0).$$

The optimal value function is *continuous* if it is both lsc and usc.

The neighborhood of a set $S \subseteq \mathbb{R}^n$ is given by:

$$\mathcal{N}_\varepsilon(S) = \{y \in \mathbb{R}^n : ||y - x|| \leq \varepsilon \text{ for some } x \in S\},$$

where $\varepsilon > 0$ and $||\bullet||$ is any norm of interest.

A point-to-set map $S(b)$ is *lower semi-continuous* (lsc) at $b = 0$ if for $b^k \to 0$ and $\varepsilon > 0$

$$\exists K \ni S(0) \subset \mathcal{N}_\varepsilon(S(b^k)) \text{ for } k > K.$$

$S(b)$ is *upper semi-continuous* (usc) at $b = 0$ if for $b^k \to 0$ and $\varepsilon > 0$

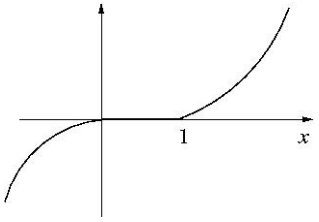
$$\exists K \ni S(b^k) \subset \mathcal{N}_\varepsilon(S(0)) \text{ for } k > K.$$

NLP Myth 8. *Given the objective is continuous and the feasible region is non-empty and compact at $b = 0$, the optimal value function is lsc at $b = 0$.*

Evans and Gould[18] provide the following:

Counterexample. max $x : g(x) \leq 0$, where g is given by:

$$g(x) = \begin{cases} x^3 & \text{if } x \leq 0 \\ 0 & \text{if } 0 \leq x \leq 1 \\ (x-1)^3 & \text{if } x \geq 1. \end{cases}$$

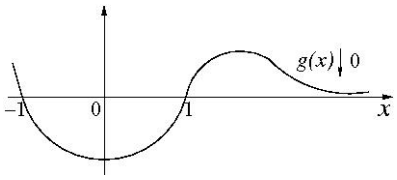


Then, for $b^k = -\frac{1}{k^3}$, we have $f^*(b^k) = -\frac{1}{k} \to 0$, but $f^*(0) = 1$. The key to this discontinuity is that $\text{cl}\{x : g(x) < 0\} \neq \{x : g(x) \leq 0\}$.

NLP Myth 9. *Given the objective is continuous and the feasible region is non-empty and compact at $b = 0$, the optimal value function is usc at $b = 0$.*

Evans and Gould[18] provide the following:

Counterexample. max $x : g(x) \leq 0$, where g has the following shape:



Then, for $b^k = g(k)$, we have $f^*(b^k) = k \to \infty$, but $f^*(0) = 1$. The key to this discontinuity is that $\{x : g(x) \leq b\}$ is unbounded for all $b > 0$ (even though $\{x : g(x) \leq 0\}$ is bounded).

NLP Myth 10. *If the feasibility region and optimal value function are lsc at $b = 0$, so is the optimality region.*

Greenberg and Pierskalla[30] provide the following:

Counterexample. Consider max $x : g(x) \le 0$, where g is given in NLP Myth 9. Specifically, let $g(x) = \min\{x^2 - 1, e^{-x^2}\}$ (they cross at about $x = \pm 1.3$). We have

$$X(b) = \begin{cases} [-\sqrt{b+1}, \sqrt{b+1}\,] & \text{if } -1 < b < 0; \\ [-1, 1\,] & \text{if } b = 0; \\ [-\sqrt{b+1}, \sqrt{b+1}\,] \cup [\sqrt{-\ln b}, \infty) & \text{if } \quad 1 > b > 0. \end{cases}$$

We have

$$\big(-1 \ge -\sqrt{b+1} - \varepsilon \text{ and } 1 \le \sqrt{b+1} + \varepsilon \big) \leftrightarrow 1 - \varepsilon \le \sqrt{b+1} \leftrightarrow b \ge \varepsilon^2 - 2\,\varepsilon\,.$$

Hence, for any $\varepsilon > 0$, let $b \ge \varepsilon^2 - 2\,\varepsilon$ to have $X(0) \subset \mathcal{N}_\varepsilon(X(b))$. This proves that X is lsc at $b = 0$. Further,

$$f^*(b) = \begin{cases} \sqrt{b+1} & \text{if } -1 < b < 0; \\ 1 & \text{if } b = 0; \\ \infty & \text{if } \quad 1 > b > 0. \end{cases}$$

Hence, $\liminf_{b \to 0} f^*(b) = 1 = f^*(0)$, so f^* is lsc at $b = 0$.

Now consider the optimality region:

$$X^*(b) = \begin{cases} \{\sqrt{b+1}\} & \text{if } -1 < b < 0; \\ \{1\} & \text{if } b = 0; \\ \emptyset & \text{if } \quad 1 > b > 0. \end{cases}$$

Let $b^k = e^{-k}$, so $X^*(b^k) = \emptyset$ for all k. Then,

$$X^*(0) = \{1\} \not\subset \mathcal{N}_\varepsilon(X^*(b^k)) = \emptyset,$$

so X^* is not lsc at $b = 0$.

Also see Dantzig, Folkman, and Shapiro[15] and Gauvin and Dubeau[23].

NLP Myth 11. *If the feasibility region and optimal value function are usc at $b = 0$, so is the optimality region.*

Greenberg and Pierskalla[30] provide the following:

Counterexample. Consider max $x : g(x) \le 0$, where g is given in NLP Myth 8. We have

$$X(b) = \begin{cases} (-\infty, \sqrt[3]{b}\,] & \text{if } b < 0; \\ (-\infty, 1] & \text{if } b = 0; \\ (-\infty, 1 + \sqrt[3]{b}\,] & \text{if } b > 0. \end{cases}$$

Since $\mathcal{N}_\varepsilon(X(0)) = (-\infty, 1 + \varepsilon]$, we have

$$X(b) \subset \mathcal{N}_\varepsilon(X(0)) \quad \text{for } b \le \varepsilon^3\,.$$

Hence, X is usc at $b = 0$. Further,

$$f^*(b) = \begin{cases} \sqrt[3]{b} & \text{if } b < 0; \\ 1 & \text{if } b = 0; \\ 1 + \sqrt[3]{b} & \text{if } b > 0. \end{cases}$$

Hence,

$$\limsup_{b \to 0} f^*(b) = 1 = f^*(0),$$

so f^* is usc at $b = 0$.

Now consider the optimality region:

$$X^*(b) = \begin{cases} \{\sqrt[3]{b}\} & \text{if } b < 0; \\ \{1\} & \text{if } b = 0; \\ \{1 + \sqrt[3]{b}\} & \text{if } b > 0. \end{cases}$$

We have $\mathcal{N}_\varepsilon(X^*(0)) = [1 - \varepsilon, 1 + \varepsilon]$. Let $b \uparrow 0$, so $X^*(b) = \{\sqrt[3]{b}\} \not\subset \mathcal{N}_\varepsilon(X^*(0))$ for $\varepsilon < 1$. Hence, X^* is not usc at $b = 0$.

Also see Dantzig, Folkman, and Shapiro[15] and Gauvin and Dubeau[23].

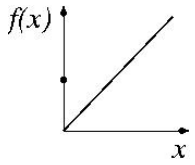
NLP Myth 12. *A set is convex if it contains the midpoint of any pair of its points.*

Counterexample. The set of rational values.

NLP Myth 13. *A convex function is continuous.*

This is true in the interior of its effective domain, but not necessarily on its boundary.

Counterexample. Let $f : \mathbb{R}_+ \to \mathbb{R}$, with $f(0) = 1$ and $f(x) = x$ if $x > 0$.



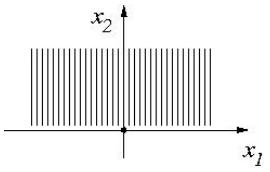
NLP Myth 14. *A convex function is upper semi-continuous on its boundary.*

Fenchel[20, 21] shows that $\liminf_{y \to x} f(y) \leq f(x)$, and his example, which follows, shows f need not be upper semi-continuous on the boundary.

Counterexample. Consider the following:

$$X = \{x \in \mathbb{R}^2 : x_2 > 0 \vee x = (0,0)\}$$

$$f(x) = \begin{cases} \dfrac{x_1^2 + x_2^2}{2x_2} & \text{if } x_2 > 0; \\ 1 & \text{if } x = (0,0). \end{cases}$$



No points on x_1-axis, except $(0,0)$.

f is convex on X and $\liminf_{y \to (0,0)} f(y) = 0$. By letting x^k take the nonlinear path such that $x_2^k = (x_1^k)^3 = \frac{1}{k}$,

$$\limsup_{y \to (0,0)} f(y) = \lim_{k \to \infty} f(x^k) = \lim_{k \to \infty} \frac{(x_1^k)^2 + (x_1^k)^6)}{2(x_1^k)^3} = \lim_{k \to \infty} \left(\frac{k}{2} + \frac{1}{2k^3} \right) = \infty.$$

NLP Myth 15. *A strictly quasiconvex function is quasiconvex.*

The definition of strictly quasiconvex says that f is defined on a convex set X, and $f(\alpha x + (1-\alpha)y) < \max\{f(x), f(y)\}$ for $x, y \in X$ such that $f(x) \neq f(y)$ and $\alpha \in (0,1)$. (Note that the definition imposes no restriction if $f(x) = f(y)$.) Karamardian[37] found the following:

Counterexample. $X = \mathbb{R}$ and $f(x) = 0$ for $x \neq 0$, $f(0) = 1$. It can be shown that f is strictly quasiconvex, but the level set, $\{x : f(x) \leq 0\}$, is not convex, so f is not a quasiconvex function.

This is what led to the definition of an *explicitly quasiconvex* function by Martos[43]. Details and further properties are given by Greenberg and Pierskalla[29].

NLP Myth 16. *The LMR is sufficient for a differentiable, quasiconvex program.*

We are given

$$\min\ f(x) : g(x) \leq 0,\ x \in \mathbb{R}^n,$$

where f and g are differentiable, quasiconvex functions on $\mathbb{R}^n$. The myth asserts that x^* is optimal if $g(x^*) \leq 0$ and there exists $\lambda \geq 0$ such that

$$\nabla f(x^*) + \lambda \nabla g(x^*) = 0 \ \text{ and } \ \lambda g(x^*) = 0.$$

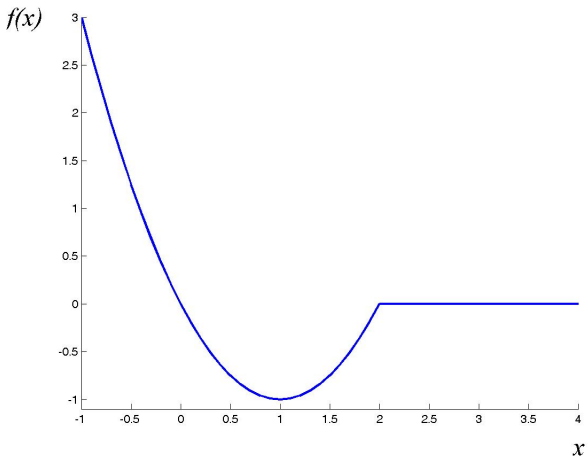
That is, the LMR is sufficient for optimality, which is true for convex programs.

Counterexample. Majumdar[41] provides the following:

$$\min\ f(x) : -1 \le x \le 4,$$

where

$$f(x) = \begin{cases} x(x-2) & \text{if } x \le 2 \\ 0 & \text{if } x > 2 \end{cases}$$



At $x = 3$, which is feasible, the LMR is satisfied with $\lambda = 0$. However, this is not optimal since $f(1) < 0 = f(3)$.

NLP Myth 17. *Let f be convex on $X \neq \emptyset$, where $X \subseteq \mathbb{R}^n$, and the range of f is in $\mathbb{R}^m$. Then, either there exists $x \in X$ such that $f(x) \le 0$, or there exists $y \in \mathbb{R}^m$ such that $y^\intercal f(x) > 0$ for all $x \in X$. Further, the two alternatives exclude each other.*

The reason this seems reasonable is due to the theorem by Fan, Glicksburg and Hoffman[19], where the first system is $f(x) < 0$, and the alternative is $y \in \mathbb{R}^m \setminus \{0\}$ such that $y^\intercal f(x) \ge 0$ for all $x \in X$. The myth "seems reasonable," considering related transposition theorems in linear systems.

Counterexample. Let $X = \{(x_1, x_2) : x_2 > 0 \vee (x_2 = 0 \text{ and } x_1 > 0)\}$ and $f(x) = x^\intercal$. Then, $f(x) \le 0$ has no solution in X. The (fallacious) alternative is $y \ge 0$ and $y^\intercal f(x) = y_1 x_1 + y_2 x_2 > 0$ for all $x \in X$. If $y_1 > 0$, let $x_2 = 1$ and $x_1 \le -\frac{y_2}{y_1}$, so $x \in X$, but $y^\intercal x \le 0$. If $y_1 = 0$, let $x_2 = 0$ and $x_1 > 0$, so $y^\intercal x = 0$. Thus, the alternative system also has no solution.

NLP Myth 18. *Suppose $x^* \in \operatorname{argmax}\{f(x) : x \in X, g(x) \le 0\}$ and $g(x^*) < 0$. Then, $x^* \in \operatorname{argmax}\{f(x) : x \in X, g(x) \le b\}$ for any $b > 0$.*

This is true if f is concave and g is convex on X, in which case x^* is the (unconstrained) maximum of f on X.

Counterexample. $\max x_1 + 2x_2 : x_1, x_2 \in \{x : x(1-x) = 1\}$, $2x_1 + 4x_2 - 3 \le b$.

At $b = 0$, $x^* = (1, 0)$ and $g(x*) < 0$; however, at $b = 4$, the optimal solution is $x^* = (0, 1)$, which is not optimal for $b = 0$.

Other examples, which are not integer-valued, include the case where x^0 is a global maximum for $b = 0$, but it is only a local maximum for $b > 0$. The objective function (f) decreases for a while, but then it turns back upward to a maximum at $x^* > x^0$ with $f(x^*) > f(x^0)$.

NLP Myth 19. *Cauchy's steepest ascent either diverges or converges to a relative maximum.*

We seek to maximize $f(x)$, and the iterations are:

$$x^{k+1} = x^k + s_k \nabla f(x^k), \text{ where } s_k > 0.$$

Wolfe[69] presented an insightful analysis, but he later corrected some statements that seemed intuitive at first, such as this myth.

Counterexample. Wolfe[70] provides the following: $f(x,y) = -\frac{1}{3}x^3 - \frac{1}{2}y^2$, which is concave for $x > 0$. Starting at (x^0, y^0) such that $0 < x^0 < 1$, the sequence satisfies $0 < x^k < 1$ for all k. Moreover, $\{(x^k, y^k)\} \to (0,0)$, which is not a relative maximum. Wolfe discusses this further, giving more insight into underlying behavior when studying the differential equation $\dot{x} = \nabla f(x)$.

NLP Myth 20. *If f is concave on $[a,b]$, the truncated gradient algorithm converges to an optimal solution. That is, $x' = x + sd$ yields a sequence for which $d = 0$ in the limit, where d is the projected steepest ascent direction:*

$$d_j = \begin{cases} \dfrac{\partial f(x)}{\partial x_j} & \text{if} \quad a_j < x_j < b_j \\ \max\left\{0, \dfrac{\partial f(x)}{\partial x_j}\right\} & \text{if} \quad a_j = x_j \\ \min\left\{0, \dfrac{\partial f(x)}{\partial x_j}\right\} & \text{if} \quad x_j = b_j. \end{cases}$$

Note: $d = 0$ if, and only if, x satisfies the first-order optimality conditions, which is equivalent to x being optimal in the case of a concave maximand. If $d \neq 0$, $s \in \operatorname{argmax}_{t>0}\{f(x+td)\}$.

Wolfe[71] provided the following:

Counterexample. Let $f(x,y,z) = -\frac{4}{3}(x^2 - xy + y^2)^{\frac{3}{4}} + z$, on the cube, $[0,100]^3$. It can be shown (non-trivially) that f is concave, and that the truncated gradient algorithm converges to the non-optimal point, $(0,0,c)$, where $c < 100$, depending on the starting point (in particular, $c = 0$ for $z^0 = 0.1$). See Dussault and Fournier[17] and Greenberg[28] for some details.

The basic problem is that the zig-zagging can cause non-finite convergence on some face, but the optimum lies on another face.

NLP Myth 21. *Rosen's projected gradient algorithm with linear constraints and inexact line search converges to a Kuhn-Tucker point.*

The NLP is max $f(x) : Ax \leq b$, where f is continuously differentiable. The *active set* of constraints is denoted $I(x) = \{i : A_{i\bullet}x = b_i\}$, and $A_{I(x)}$ is the submatrix whose rows are $I(x)$. At a general iteration, Rosen's projected gradient method[60] is to set the (feasible) direction: $d(x) = P(x)\nabla f(x)$, where $P(x)$ is the projection matrix onto the active face:

$$P(x) = I - A_{I(x)}^{\mathsf{T}} \left[A_{I(x)} A_{I(x)}^{\mathsf{T}}\right]^{-1} A_{I(x)}.$$

If $d(x^k) = 0$, the first-order (Lagrangian) conditions are satisfied, and the algorithm terminates. If $d(x^k) \neq 0$ and $A_{i\bullet}d(x^k) \leq 0$ for all $i \notin I(x^k)$, the problem is unbounded, and the algorithm terminates. Otherwise, let $\mathcal{I}(x) = \{i \notin I(x^k) : A_{i\bullet}d(x^k) > 0\}$ $(\neq \emptyset)$, and $\overline{s} = \min_{i \in \mathcal{I}(x^k)} \dfrac{b_i - A_{i\bullet}x^k}{A_{i\bullet}d(x^k)}$. Then, the iteration is given by:

$$x^{k+1} = x^k + s_k d(x^k),$$

where s_k is the step size, limited by $0 \leq s_k \leq \overline{s}$. An *inexact line search* is specifying s_k without optimizing along the direction, such as using Armijo's rule[2] (See NLP Myth 36.) Also, successive directions are not orthogonal, so the zig-zag phenomenon does not apply, as in NLP Myth 20.

Counterexample. Hu[36] provides the following:

$$\max\ -x_1^2 + x_2^2 :\ x \geq 0,\ x_1 - x_2 \leq 1.$$

The step size is determined by a near-optimal line search:

$$f(x^k + s_k d(x^k)) \geq \max_{0<s\leq\overline{s}} \{f(x^k + s d(x^k))\} - \varepsilon_k,$$

where $\varepsilon_k = 2^{1-k}$. Starting at $x^0 = (2, 1)$, Hu's inexact line search generates the sequence $\{x^k\} = \{(1 + 2^{-k},\ 2^{-k})\} \to (1, 0)$ (with $I(x^k) = \{3\}$, $d(x^k) = (-x_1^k + x_2^k, -x_1^k + x_2^k) = (-1, -1)$, and $s_k = 2^{1-k}$). The optimal step size is $s^* = \overline{s} = 2^{-k}$, so

$$f(x^k + s_k d(x^k)) - f(x^k + s^* d(x^k)) = -(1 + 2^{1-k}) - (-1) = -2^{1-k} = -\varepsilon_k\,.$$

This dispels the myth because the only Kuhn-Tucker point is at $x = 0$.

NLP Myth 22. *Rosen's initial-point method converges to a feasible solution or ascertains that there is no feasible solution.*

This myth is concerned with Rosen's method[60] for obtaining an initial feasible point to the system, $Ax \geq b$.

Notation: A_i is the i^{th} row of A.

The test condition at each iteration is:

$$A_j + \sum_{i \in I(x^k)} \lambda_i A_i = 0 \text{ for some } \lambda \geq 0,$$

where I is the active set (see NLP Myth 21) and $j \in \text{argmin}_i\{A_i x^k - b_i\}$. If the test condition holds, the system is not feasible. Otherwise, a member of the active set is selected for which its multiplier is negative:

$$q \in I(x^k) :\ \lambda_q < 0.$$

Then, x^{k+1} is obtained by solving:

$$A_i x^{k+1} = b_i \text{ for } i \in I(x^k) \setminus q \cup \{j\}.$$

This results in $A_q x^{k+1} > b_q$ and $A_i x^{k+1} \geq b_i$ for all $i \in I(x^k)$.

Counterexample. Powell[56] provides the following:

$$\begin{array}{lll} 10^{-\frac{1}{2}}\,(x_1 - 3x_3 + 2\delta) \geq 0, & 5^{-\frac{1}{2}}\,(-x_1 - 2x_2 + 2) \geq 0, & \\ 10^{-\frac{1}{2}}\,(-x_2 - 3x_3 + 2\delta) \geq 0, & 5^{-\frac{1}{2}}\,(-2x_1 + x_2 + 2) \geq 0, & \\ 10^{-\frac{1}{2}}\,(-x_1 - 3x_3 + 2\delta) \geq 0, & 5^{-\frac{1}{2}}\,(x_1 + 2x_2 + 2) \geq 0, & \\ 10^{-\frac{1}{2}}\,(x_2 - 3x_3 + 2\delta) \geq 0, & 5^{-\frac{1}{2}}\,(2x_1 - x_2 + 2) \geq 0, & x_3 \geq 0, \end{array}$$

where $0 < |\delta| \leq 0.1$. For $\delta < 0$, the system is infeasible; for $\delta > 0$, the system has a feasible solution, $x = 0$. The counterexample, therefore, shows non-convergence for both situations.

For $\delta = 0.1$ and $x_3 = 0$, constraints 1, 3, 5, and 7 are simply $-0.2 \leq x_1, x_2 \leq 0.2$, and constraints 2, 4, 6, and 8 are redundant: $-2 \leq x + 2y,\ 2x - y \leq 2$.

Rosen's algorithm cycles among 8 points:

$$\begin{array}{lll} x^1 = (-2\delta, 2 - 4\delta, 0), & x^2 = (-2\delta, 1 + \delta, 0), & x^3 = (2 - 4\delta, 2\delta, 0), \\ x^4 = (1 + \delta, 2\delta, 0), & x^5 = (2\delta, -2 + 4\delta, 0), & x^6 = (2\delta, -1 - \delta, 0), \\ x^7 = (-2 + 4\delta, -2\delta, 0), & x^8 = (-1 - \delta, -2\delta, 0). & \end{array}$$

For $k > 8$, $x^k = x^{k-8}$.

The values of $A_i x^k - b_i$ for $i = 1, \ldots, 9$ are as follows:

i	x^1	x^2	x^3	x^4	x^5	x^6	x^7	x^8
$1 \times 10^{\frac{1}{2}}$	**0**	**0**	$2 - 2\delta$	$1 + 3/d$	4δ	4δ	$-2 + 6\delta$	$-1 + \delta^{\dagger}$
$2 \times 5^{\frac{1}{2}}$	$-2 + 10\delta^{\dagger}$	**0**	**0**	$1 - 5\delta$	$6 - 10\delta$	4	4	$3 + 5\delta$
$3 \times 10^{\frac{1}{2}}$	$-2 + 6\delta$	$-1 + \delta^{\dagger}$	**0**	**0**	$2 - 2\delta$	$1 + 3\delta$	4δ	4δ
$4 \times 5^{\frac{1}{2}}$	4	$3 + 5\delta$	$-2 + 10\delta^{\dagger}$	**0**	**0**	$1 - 5\delta$	$6 - 10\delta$	4
$5 \times 10^{\frac{1}{2}}$	4δ	4δ	$-2 + 6\delta$	$-1 + \delta^{\dagger}$	**0**	**0**	$2 - 2\delta$	$1 + 3\delta$
$6 \times 5^{\frac{1}{2}}$	$6 - 10\delta$	4	4	$3 + 5\delta$	$10 - 2\delta$	**0**	**0**	$1 - 5\delta$
$7 \times 10^{\frac{1}{2}}$	$2 - 2\delta$	$1 + 3\delta$	4δ	4δ	$-2 + 6\delta^{\dagger}$	$-1 + \delta^{\dagger}$	**0**	**0**
$8 \times 5^{\frac{1}{2}}$	**0**	$1 - 5\delta$	$6 - 10\delta$	4	4	$3 + 5\delta$	$-2 + 10\delta^{\dagger}$	**0**
9	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

†Most violated

Note the active sets — in particular, $I(x^1) = \{1, 8, 9\}$. The only violated constraints at x^1 are 2 and 3. The most violated constraint is 2, rather than 3, because of the scale factors:

$$5^{-\frac{1}{2}}|10\delta - 2| > 10^{-\frac{1}{2}}|6\delta - 2| \quad (\text{for } \delta \in [-0.1,\, 0.1]).$$

Thus, $j = 2$, and the test condition is:

$$\begin{array}{rl} A_2 + \lambda_1 A_1 + \lambda_8 A_8 + \lambda_9 A_9 & = 0 \\ 5^{-\frac{1}{2}}(-1, -2, 0) + \lambda_1 10^{-\frac{1}{2}}(1, 0, -3) + \lambda_8 5^{-\frac{1}{2}}(2, -1, 0) + \lambda_9(0, 0, 1) & = 0 \\ \Rightarrow \lambda_1 = 5\sqrt{2},\ \lambda_8 = -2,\ \lambda_9 = 3\sqrt{5}. & \end{array}$$

Thus, $q = 8$, yielding x^2 as the solution to

$$A_i x^2 = b_i \quad \text{for } i \in \{1, 2, 9\}.$$

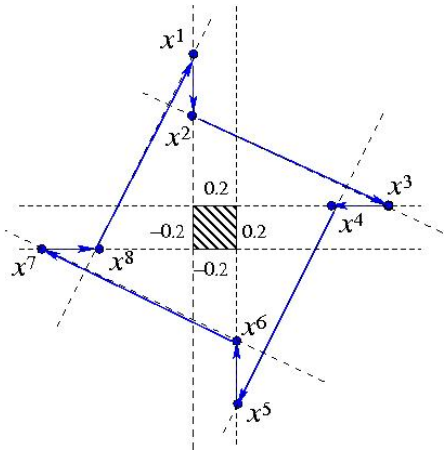
The remaining sequence is easily verified, and at x^8, the most violated constraint is $j = 1$. The test condition is:

$$\begin{aligned} A_1 + \lambda_7 A_7 + \lambda_8 A_8 + \lambda_9 A_9 &= 0 \\ 10^{-\frac{1}{2}}(1,0,-3) + \lambda_7 10^{-\frac{1}{2}}(1,0,-3) + \lambda_8 5^{-\frac{1}{2}}(2,-1,0) + \lambda_9(0,0,1) &= 0 \\ \Rightarrow \lambda_7 = -1,\ \lambda_8 = 0,\ \lambda_9 = 0. \end{aligned}$$

Thus, $q = 7$, which brings us back to x^1 by solving

$$A_i x^8 = b_i \quad \text{for } i \in \{1, 8, 9\}.$$

The figure on the right shows the feasible region in the x_1-x_2 plane with $x_3 = 0$ and $\delta = 0.1$. The cycle is shown to move from the intersection of a redundant constraint with one of the (non-redundant) bounds. The intersections correspond to the active sets (with constraint 9 in every active set.) The redundancies keep the method from reaching the feasible region.



Opportunity Knocks

Does Rosen's method work when there are no redundant constraints?

NLP Myth 23. *Newton's method has converged when the change in the iterate value is less than some specified, small tolerance.*

Let $f : \mathbb{R} \to \mathbb{R}$ be a function in C^1, for which we seek a root, $f(x) = 0$. Let $\{x^k\}$ be generated by Newton's method:

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)}.$$

The stopping criterion in the statement says that we terminate when

$$|x^{k+1} - x^k| < \tau,$$

where τ is the tolerance. Donovan, Miller and Moreland[16] provided the following:

Counterexample. $f(x) = \sqrt[3]{x}\, e^{-x^2}$. The generated sequence satisfies the iteration equation:

$$x^{k+1} = x^k - \frac{3x^k}{1 - 6(x^k)^2},$$

which does not converge. Yet, $|x^{k+1} - x^k| < \tau$ is equivalent to: $\left| \frac{3x^k}{1 - 6(x^k)^2} \right| < \tau$, which is eventually satisfied since the authors prove that $\{x^k\} \to \infty$.

They also derive properties of f and insight into its construction for the counterexample. In particular, they note that the first part, $\sqrt[3]{x}$, fails Newton's method on its own ($x^{k+1} = -2x^k$ implies $x = 0$ is a repelling fixed point). The second part, e^{-x^2}, gives the "false convergence" property.

NLP Myth 24. *Newton's method converges to a stationary point if the starting point is sufficiently close.*

Newton's method is applied to the root-finding problem, $g(x) = 0$, with the iterations:

$$x^{k+1} = x^k - [\nabla g(x^k)]^{-1} g(x^k),$$

where $\nabla g(x^k)$ is assumed to be non-singular when $g(x^k) \neq 0$. The algorithm stops if $g(x^k) = 0$. (In optimization, this is applied to the derivative of the objective function, where $g(x) = \nabla f(x)$.) In one dimension, Newton's method simplifies to:

$$x^{k+1} = x^k - \frac{g(x^k)}{g'(x^k)} \text{ if } g'(x^k) \neq 0.$$

A simple counterexample to the myth is any quadratic with two distinct roots, a, b, and $x^0 = \frac{a+b}{2}$. In that case $g'(x^0) = 0$, so Newton's method is undefined. (We can make a, b arbitrarily close to each other to satisfy the condition of the myth.)

A more interesting example is analyzed by Ascher[3], where Newton's method cycles — that is, it generates x^0 after a finite number of iterations. (The analysis goes beyond this simple example.)

Counterexample. Let $g(x) = x^2 + 3$, so $x^{k+1} = \frac{1}{2}x^k - \dfrac{3}{2\,x^k}$. Then, for $x^0 = \pm 1$, the iterates cycle in two iterations.

NLP Myth 25. *Newton's method has converged when the change in the iterate value is less than some specified, small tolerance.*

Let $f : \mathbb{R} \to \mathbb{R}$ be a function in C^1, for which we seek a root, $f(x) = 0$. Let $\{x^k\}$ be generated by Newton's method:

$$x^{k+1} = x^k - \frac{f(x^k)}{f'(x^k)}.$$

The stopping criterion in the statement says that we terminate when

$$|x^{k+1} - x^k| < \tau,$$

where τ is the tolerance. Donovan, Miller and Moreland[16] provided the following:

Counterexample. $f(x) = \sqrt[3]{x}\, e^{-x^2}$. The generated sequence satisfies the iteration equation:

$$x^{k+1} = x^k - \frac{3x^k}{1 - 6(x^k)^2},$$

which does not converge. Yet, $|x^{k+1} - x^k| < \tau$ is equivalent to: $\left|\frac{3x^k}{1-6(x^k)^2}\right| < \tau$, which is eventually satisfied since the authors prove that $\{x^k\} \to \infty$.

They also derive properties of f and insight into its construction for the counterexample. In particular, they note that the first part, $\sqrt[3]{x}$, fails Newton's method on its own ($x^{k+1} = -2x^k$ implies $x = 0$ is a repelling fixed point). The second part, e^{-x^2}, gives the "false convergence" property.

NLP Background — Sequence construction

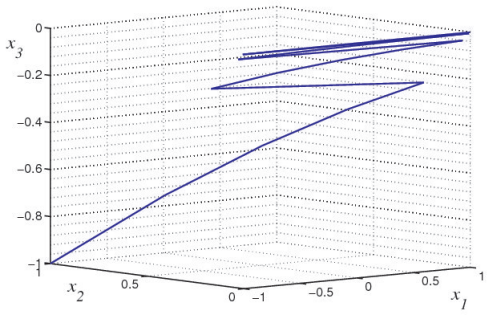
In the next several myths, convergence to an optimum is called into question. Unlike several previous myths that address this, an approach is taken to construct a sequence, $\{x^k\}$, that either has no limit, or $\{x^k\} \to x^\infty$ but $\nabla f(x^\infty) \neq 0$ — that is, the limit is not even a stationary point. Powell[54] introduced the notion of identifying such sequences that could be generated by some variable metric algorithms. He raised the question whether there exists parameters p for a family of functions, $f(x;p)$, such that $\nabla^2 f(x;p)$ is continuous, $\{x^k\}$ is generated from the algorithm, and there exists $c > 0$ such that $\left|\left|\nabla f(x^k;p)\right|\right| \geq c$ for $k = 1, 2, \ldots$

This is how he constructed his counterexample in NLP Myth 28, and how Thompson[62] approached his counterexample construction for quasi-Newton methods. More recently, Mascarenhas[46] focused on constructing counterexamples for line search methods, and Dai[14] focused on counterexample construction that satisfies Wolfe's step size conditions (NLP.18):

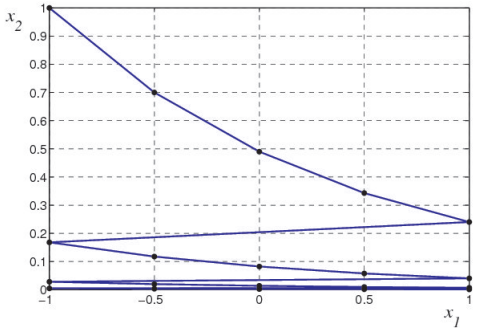
$$\begin{aligned} f(x^{k+1}) - f(x^k) &\leq A\, s_k \nabla f(x^k)^\intercal d^k \\ \nabla f(x^{k+1})^\intercal\, d^k &\geq B\, \nabla f(x^k)^\intercal d^k \end{aligned} \qquad \text{(NLP.18)}$$

for some constants, $0 < A < B < 1$.

Non-convergent sequences include *cycling* (or *spiraling*), where one coordinate repeats its value. The other coordinates could be convergent. The constructions by Powell and Dai have $\{x_1^k\}$ cycling while $\{x_j^k\} \to 0$ for $j > 1$. The following is a generic plot where $\{x_1^k\}$ cycles around $\{-1, -1/2, 0, 1/2, 1\}$, $x_2^k = 0.7^k$, and $x_3^k = -0.7^k$.



Plot of x^k



Top view (x_1-x_2 plane)

These constructions are designed to establish a *proof* of non-convergence for the algorithm in question (under stated conditions). Other examples are not proofs, but rather offer empirical

evidence of *stalling* — that is, the algorithm terminates at a non-solution point, but it is not *proven* to be non-convergent. An example of this is given in NLP Myth 27.

NLP Myth 26. *Newton's method converges to a stationary point if Wolfe's conditions are satisfied.*

We consider unconstrained minimization, $\min_{x\in\mathbb{R}^n} f(x)$, where f has continuous second derivatives. As usual, let s_k and d^k denote the step size and direction at iteration k, respectively (so $x^{k+1} = x^k + s_k d^k$).

Counterexample. Mascarenhas[46] provides the following:

$$\begin{aligned} f(x) &= \psi(x) + \psi(x)^2 && \text{for } x \in \mathbb{R}^2 \\ \psi(x) &= x_1 + x_2^2 - 28x_1x_2^3 + 24x_1^2 + 3(105x_2^3 - 2)u(x_2) \\ u(y) &= \tfrac{y^3(\ln 2)^2}{8\pi^2} \sin^2\left(\pi \tfrac{\ln y^2}{\ln 2}\right) && \text{for } y \neq 0 \\ u(0) &= 0. \end{aligned}$$

Let $x^k = \begin{pmatrix} 8^{-k} \\ 2^{-k} \end{pmatrix}$. With some abuse in notation, let $y = 2^{-k}$, so $x^k = \begin{pmatrix} y^3 \\ y \end{pmatrix}$ and

$$\begin{aligned} \nabla f(x^k) &= \nabla\psi(x^k)\left(1 + 2\psi(x^k)\right) \\ &= \begin{pmatrix} 1 + 20y^3 \\ 3y^2(1 - 28y^3) \end{pmatrix}. \end{aligned}$$

We have $\{x^k\} \to 0$, but $\{\nabla f(x^k)\} \to (1, 0)^\intercal$, so this sequence converges to a non-stationary point. It remains to prove that this sequence can be generated by Newton's method with a step size rule that satisfies Wolfe's conditions.

The hessian of the objective function is:

$$\begin{aligned} \nabla^2 f(x^k) &= \nabla^2\psi(x^k) + 2\left(\nabla\psi(x^k)\nabla\psi(x^k)^\intercal + \psi(x^k)\nabla^2\psi(x^k)\right) \\ &= \begin{bmatrix} 48 & -84y^3 \\ -84y^3 & 147y^4 \end{bmatrix}. \end{aligned}$$

The hessian of the iterates is positive definite (note $\det\left(\nabla^2 f(x^k)\right) = 7056y^4 - 7056y^6 > 0$).

Mascarenhas sets the step size to

$$s_k = \frac{2}{1 + 2\psi(x^k)} \nabla\psi(x^k)^\intercal d^k.$$

The sequence is generated by Newton's method if $\nabla^2 f(x^k) d^k = -s_k \nabla f(x^k)$. Because $\nabla^2\psi(x^k)d^k = 0$, this is equivalent to

$$\nabla^2 f(x^k) d^k = 2\nabla\psi(x^k)\nabla\psi(x^k)^\intercal d^k.$$

Thus, the task is to show

$$2\nabla\psi(x^k)\nabla\psi(x^k)^\intercal d^k = \frac{2}{1+2\psi(x^k)}\nabla\psi(x^k)^\intercal d^k \nabla\psi(x^k)(1+2\psi(x^k)).$$

Upon canceling $1+2\psi(x^k)$ on the right and 2 on both sides of the equation, we obtain the identity:

$$\nabla\psi(x^k)\left(\nabla\psi(x^k)^\intercal d^k\right) = \left(\nabla\psi(x^k)^\intercal d^k\right)\nabla\psi(x^k).$$

Hence, $\{x^k\}$ is generated by Newton's method with the specified step size. It remains to show that Wolfe's conditions are satisfied. Mascarenhas shows this with $A = \frac{1}{2}$ and $B = \frac{3}{4}$.

$$\begin{aligned}
f(x^{k+1}) - f(x^k) &= \left(\psi(x^{k+1}) - \psi(x^k)\right)\left(1+2\psi(x^k)\right) \\
&= -3y^3\left(1 - 8y^6 - 32y^9\right). \\
s_k\nabla f(x^{k+1})^\intercal d^k &= \left(1+2\psi(x^{k+1})\right)\nabla\psi(x^{k+1})^\intercal(x^{k+1}-x^k) \\
&= \left(1+\tfrac{5}{2}y^3,\ \tfrac{3}{4}y^2\left(1-\tfrac{7}{2}y^3\right)\right)\begin{pmatrix} -\frac{7}{8}y^3 \\ -\frac{1}{2}y \end{pmatrix} \\
&= -\tfrac{1}{4}y^3\left(5+\tfrac{7}{2}y^3\right).
\end{aligned}$$

For Wolfe's first condition, we need to show

$$-3\left(1-8y^6-32y^9\right) \le -\tfrac{1}{8}\left(5+\tfrac{7}{2}y^3\right),$$

which clearly holds. For Wolfe's second condition, we need to show

$$-\tfrac{1}{4}y^3\left(5+\tfrac{7}{2}y^3\right) \ge \tfrac{3}{4}\nabla f(x^k)^\intercal\left(x^{k+1}-x^k\right) = -\tfrac{3}{4}y^3\left(\tfrac{19}{8}+\tfrac{51}{2}y^3\right).$$

Equivalently, we must show

$$5+\tfrac{7}{2}y^3 \le 3\left(\tfrac{19}{8}+\tfrac{51}{2}y^3\right) = 7\tfrac{1}{8}+\tfrac{153}{2}y^3,$$

which holds. Here are some values to compare the left-hand side (LHS) with the right-hand side (RHS):

k	y^3	LHS	RHS
1	0.1250	5.4375	16.6875
2	0.0156	5.0547	8.3203
3	0.0020	5.0068	7.2744
⋮			
8	0.0000	5	7.125

Mascarenhas provides a second example and a more general analysis of the behavior of the iterates. Also, see [45] for a generalization and deeper insights.

NLP Myth 27. *The BFGS method converges to a stationary point.*

Counterexample. Shanno[61] provides the following:

$$\min\ f(x_1,x_2) = \left(1.5 - x_1(1-x_2)\right)^2 + \left(2.25 - x_1(1-x_2)^2\right)^2 + \left(2.625 - x_1(1-x_2^3)\right)^2.$$

(This is known as *Beale's function*, which is used to test pathological behavior in NLP.) Shanno reports that BFGS terminates after 37 iterations with the termination condition: $\left|\left|x^{k+1} - x^k\right|\right| < 10^{-20}$. Starting from $x^0 = (100, 100)$, the computed solution is $x^{33} = (74.50580319350425, 0.9865597565555755)$ with objective value 0.43146925. There is little change from iteration 33 to termination. The optimal solution is at $x^* = (3, 0.5)$ with objective value 0. There is also a saddle point at $(0, 1)$, but the sequence does not converge to that stationary point either.

A possible problem is with the finite precision of the computer, causing a premature termination despite the very small tolerance. Shanno gives more details and runs additional experiments to test if inexact line search is the cause. The conclusion is that a more accurate line search does not result in convergence to a stationary point, though a more precise line search did improve the terminal solution a little. Also, Shanno's counterexample is an example of stalling, based on observed numerical behavior; it is not *proven* to be non-convergent.

Mascarenhas[44] carried this further and *proved* that BFGS (and other variable metric algorithms in the Broyden class) can fail with ***exact*** line search. His counterexample to convergence is a (non-convex) cubic spline. Also see [45] for perceptive insights stemming from the symmetry associated with orthogonal transformations of variables.

See Dai[14] for another counterexample construction that satisfies Wolfe's step size conditions, but BFGS fails to converge. Also see NLP Myth 28.

NLP Myth 28. *The Polak-Ribiére conjugate gradient algorithm converges to a stationary point.*

A general iteration ($k > 1$) of the algorithm computes the direction vector as:

$$d^k = -g^k + \beta_k d^{k-1},$$

where $g^k = \nabla f(x^k) \neq 0$ and $\beta_k = (g^k)^\intercal (g^k - g^{k-1}) / \left|\left|g^{k-1}\right|\right|^2$. For $k = 1$, $d^1 = -g^1$. The step size, s_k, is the first local minimum of $f(x^k + sd^k)$ for $s > 0$. As usual, $x^{k+1} = x^k + s_k d^k$.

The Myth asserts $\lim_{k \to \infty} g^k = 0$. Powell[57] points out that $\left\{\left|\left|g^k\right|\right|\right\}$ is bounded away from zero only if $\left\{\left|\left|d^k\right|\right|\right\} \to \infty$. From this, he notes that in order to construct a counterexample, we must have no limit for $\{x^k\}$. Instead, the iterates must have a coordinate that cycles. He derives a family of 8-cycle examples with three variables, defined by parameters whose values are limited by the conditions we seek:

1. The sequence, $\{x^k\}$, is generated by the Polak-Ribiére algorithm.
2. $x_1^k = x_1^{k-8}$ for $k > 8$.
3. The objective function (f) is twice continuously differentiable.

Counterexample. Powell[57] provides a spline function of the form:

$$\min\ f(x) = \lambda_i(x_1)x_2 + \mu_i(x_1)x_3 \text{ for } \tau_{i-1} \leq x_1 \leq \tau_i,$$

where $0 = \tau_0 < \tau_1 < \cdots < \tau_b$ are the break-points. The functions λ_i and μ_i are twice continuously differentiable with the symmetry property: $\lambda_i(v) = \lambda(\tau_i - (v - \tau_{i-1}))$ and

$\mu_i(v) = \mu(\tau_i - (v - \tau_{i-1}))$ for $\tau_{i-1} \le v \le \tau_i$. (In particular, $\lambda(\tau_{i-1}) = \lambda(\tau_i)$ and $\mu(\tau_{i-1}) = \mu(\tau_i)$.)

The algorithm's sequence has x_1^k cycling around eight values: 0, 336, 1200, 1564, 1565, 1229, 365, 1. The iterates are, for $k > 8$:

$$x_1^k = x_1^{k-8} \text{ and } x_j^k = \theta x_j^{k-8} \quad \text{for } j = 1, 2,$$

where $\theta \in (0, 1)$. Powell shows that $\nabla^2 f$ is continuous and that $\{f(x^k)\}$ is increasing.

Powell derives parameter relations that yield the non-convergence. His function also applies to show BFGS need not converge to a stationary point.

Dai[14] provides another example of spiraling (where $\{x_1^k\}$ cycles while $\{x_j^k\} \to 0$ for $j > 1$), with only two variables and a cycle length of six. His construction is different, and he provides further analysis of convergence properties for a class of conjugate gradient methods, which includes Polak-Ribiére and BFGS. His focus is on constructing pathological sequences that satisfy Wolfe's step size conditions, (NLP.18).

My thanks to Professor Dai for providing this specialization of his counterexamples, including specified parameter values.

Counterexample.

$$f(x) = (g(x_1; u, p) + g(x_1; v, p)x_1)x_2,$$

where g is a univariate function depending upon two vector-parameters. The parameters p, u (and v) are each m-vectors, where m is the length of the cycle. The parameters $p_1, \ldots, p_m$ define endpoints of intervals in $\mathbb{R}$; u and v define the value of g for $|x_1 - p_i| \le 0.1$. The remaining values of g are determined by the function:

$$\psi(x_1; p_\ell, p_r, u_\ell, u_r) = u_\ell + \frac{6(x_1 - p_\ell)^5 - 15(u_r - u_\ell)(x_1 - p_\ell)^4 + 10(u_r - u_\ell)^2(x_1 - p_\ell)^3}{(u_r - u_\ell)^4}.$$

Specific parameter values for this counterexample are:

$$\begin{aligned} p &= (-87.5,\ -86.5,\ -73.5,\ 73.5,\ 86.5,\ 87.5); \\ u &= \left(\frac{8251}{458}, -\frac{6981}{212}, -\frac{2847}{387}, -\frac{2847}{387}, -\frac{6981}{212}, \frac{8251}{458}\right); \\ v &= \left(\frac{55}{229}, -\frac{33}{106}, -\frac{44}{387}, \frac{44}{387}, \frac{33}{106}, -\frac{55}{229}\right). \end{aligned}$$

The g-function values are thus:

$$g(x_1; u, p) = \begin{cases} u_1 & \text{for } x_1 \in (-\infty, -87.4] \\ \psi(x_1, p_1, p_2, u_1, u_2) & \text{for } x_1 \in (-87.4, -86.6) \\ u_2 & \text{for } x_1 \in [-86.6, -86.4] \\ \psi(x_1, p_1, p_2, u_2, u_3) & \text{for } x_1 \in (-86.4, -73.6) \\ u_3 & \text{for } x_1 \in [-73.6, 73.6] \\ \psi(x_1, p_1, p_2, u_4, u_5) & \text{for } x_1 \in (73.6, 86.4) \\ u_5 & \text{for } x_1 \in [86.4, 86.6] \\ \psi(x_1, p_1, p_2, u_5, u_6) & \text{for } x_1 \in (86.6, 87.4) \\ u_6 & \text{for } x_1 \in [87.4, \infty). \end{cases}$$

It is similar for $g(x_1; v, p)$, using the same endpoints but different values, $v \neq u$.

This construction renders f twice continuously differentiable. An infinitely differentiable counterexample is given as follows. Let

$$\phi(x_1; p_\ell, p_r) = \begin{cases} e^{-\frac{1}{(x_1 - p_\ell)(p_r - x_1)}} & \text{for } p_\ell < x_1 < p_r \\ 0 & \text{otherwise.} \end{cases}$$

Dai proves this generates the 6-cycle:

$$x_1^1 = p_1,\ x_1^2 = p_3,\ x_1^3 = p_5,\ x_1^4 = p_6,\ x_1^5 = p_4,\ x_1^6 = p_2,\ x_1^k = x_1^{k-6} \text{ for } k > 6.$$

He further proves that the step size rule satisfies Wolfe's conditions.

Opportunity Knocks

Both Powell's and Dai's counterexamples could benefit from the construction of specific numerical examples with an implementation (for example, with MATLAB®). If a cycle is unstable, numerical error can cause the sequence to "jump out" of the cycle and converge. It would be useful to have [a family of] counterexamples that are both theoretically and numerically non-convergent.

NLP Myth 29. *Cyclic descent produces a local minimum.*

We are given the mathematical program: min $f(x) : x \in X$, where X is a non-empty, closed subset of $\mathbb{R}^n$. Cyclic descent proceeds as follows (where $\boldsymbol{e}_i$ is the i^{th} unit vector):

Set $y = x$
for i=1:n **do**
  find $t^* \in \text{argmin}\{f(y + t\boldsymbol{e}_i) : (y + t\boldsymbol{e}_i) \in X\}$
  Set $y \leftarrow y + t^*\boldsymbol{e}_i$
end for
if $||y - x|| \leq \tau$, exit; **else** set $x \leftarrow y$ and repeat.

The problem is that while f may not be increased by a change in any one variable, it could increase with simultaneous changes in more than one variable.

Counterexample. $f(x) = (x_2 - x_1^2)(x_2 - 2x_1^2)$; start at $x = (0, 0)$.

$\min_t f(t, 0) = \min_t 2t^4 = 0$, so there is no change after $i = 1$. Similarly, $\min_t f(0, t) = \min_t t^2 = 0$, so there is no change after $i = 2$. Hence, cyclic descent terminates after one iteration with the same point with which it started, $x = (0, 0)$. This is not a minimum, even locally, because we can let $x_2 = \frac{3}{2}x_1^2$. Then, for x_1 arbitrarily close to 0, but $x_1 \neq 0$, $f(x) = -\frac{1}{4}x_1^2 < 0$.

Also see Powell[55].

NLP Myth 30. *If one algorithm has a higher order of convergence than another, it is better.*

The reason that this is wrong is that the goodness of a solution (for example, how close it is to optimal) cannot be accurately described by one number. Greenberg[26] provides the following:

Counterexample. Let $\{x^k\}$ be a sequence of solutions converging to x^*, and let $f(x^*)$ be the optimal objective value. Define the deviations, $\{e^k = f(x^k) - f(x^*)\}$. For definiteness, suppose $e^k > 0$ for all k and we are minimizing (so $\{f(x^k)\}$ is approaching from above, as in a primal algorithm). Define the "goodness" of x^k to be e^k — that is, how close $f(x^k)$ is to the optimal objective value. Now suppose another algorithm generates the sequence $\{X^k\}$ whose associated goodness is $\{E^k\}$, where

$$E^k = \begin{cases} \min\{e^k, e^{k+1}\}/k & \text{if } k \text{ is odd;} \\ E^{k-1} & \text{if } k \text{ is even.} \end{cases}$$

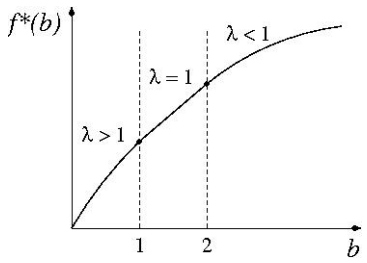
The result is that the second sequence is sublinear (the worst possible for a monotonically decreasing sequence), but X^k is always better since $E^k < e^k$ for all k.

An algorithm that has plateaux exhibits this behavior — no improvement for an iteration, then a sharp improvement. Some measures of the order of convergence take constant plateaux into account, but the example can be revised to have a plateau of length k at iteration k, so the order of convergence is still sublinear.

NLP Myth 31. *For a convex program, the Generalized Lagrange Multiplier Method converges to an optimal solution.*

Counterexample. Let $f^*(b) = \max\{f(x) : 0 \le x \le b\}$ for $b \ge 0$, where

$$f(x) = \begin{cases} 2\sqrt{x} & \text{if} \quad x \le 1 \\ x + 1 & \text{if} \quad 1 \le x \le 2 \\ 4 - e^{-(x-2)} & \text{if} \quad x \ge 2 \end{cases}$$



(Note: $f^*(b) = f(b)$.) Using any interval-reduction method[27] that does not terminate finitely, the left endpoint converges to 1 and the right endpoint converges to 2.

Finite termination occurs when the two endpoints equal the linearity portion, so the next iteration chooses the multiplier equal to the slope ($\lambda = 1$). Then, the set of optimal solutions is the interval $[1, 2]$, so that any b in this interval is generated by searching the set of alternative optima. Without finite termination, no b in (1,2) is a limit point. There is thus a *pseudo-gap*[25] for $b \in (1, 2)$ in that the algorithm cannot reach the solution, but there is no Lagrangian duality gap.

NLP Myth 32. *Ritter's method to solve a QP converges to a global optimal solution.*

The counterexample was found by Zwart[72]. The problem is that the sequence of feasible regions (with a cut added each iteration) does not approach the optimality region. A non-global optimum point persists in the sequence of optima.

Counterexample.

$$\max\ 2x_1^2 + x_1x_2 + 2x_2 :\ -x_1 \le 0,\ x_1 + x_2 \le 1,\ 1.5x_1 + x_2 \le 1.4,\ -x_2 \le 10.$$

Each cut has the form $\frac{1}{2^k}x_1 + x_2 \le \frac{1}{2^k}$, and the optimal point is at the extreme point, $(0, \frac{1}{2^k})$. Ritter's method does not eliminate (0,0), so it cannot converge to the global optimum, which is at $(7.6, -10)$.

NLP Myth 33. *Tui's method to maximize a convex function subject to linear constraints converges to a global optimal solution.*

This counterexample was found by Zwart[72]. The problem is that Tui's algorithm can cycle — that is, repeat the generated subproblems.

Counterexample.

$$\begin{aligned} \max\ x_1^2 + x_2^2 + (x_3 - 1)^2 :\ x_2 &\ge 0 \\ x_1 + x_2 - x_3 &\le 0 \\ -x_1 + x_2 - x_3 &\le 0 \\ 12x_1 + 5x_2 + 12x_3 &\le 22.8 \\ 12x_1 + 12x_2 + 7x_3 &\le 17.1 \\ -6x_1 + x_2 + x_3 &\le 1.9 \end{aligned}$$

Zwart gives the following generated sequence, starting at $x = (0, 0, 0)$.

THE CYCLING OF TUI'S METHOD

q	k	y^{q,k_1}	y^{q,k_2}	y^{q,k_3}	$x_{\text{opt}}^{q,k}$	$fx_{\text{opt}}^{q,k}$
1	1	(1, 0, 1)	(0, 1, 1)	(−1, 0, 1)	(0, 0, 1.90)	−0.81
2	1	(0, 0, 2)	(0, 1, 1)	(−1, 0, 1)	(0.05, 0.60, 1.60)	−0.7225
2	2	Not required because $\lambda_2^{1,1}=0$				
2	3	(1, 0, 1)	(0, 1, 1)	(0, 0, 2)	(0.05, 0.60, 1.60)	−0.7225
3	1	(0.055, 0.66, 1.75)	(0, 1, 1)	(−1, 0, 1)	*	
3	2	(0, 0, 2)	(0.055, 0.66, 1.75)	(−1, 0, 1)	*	
3	3	(0, 0, 2)	(0, 1, 1)	(0.055, 0.66, 1.75)	(−0.95, 0, 0.95)	−0.905
3	4	(0.055, 0.66, 1.75)	(0, 1, 1)	(0, 0, 2)	(−0.95, 0, 0.95)	−0.905
3	5	(1, 0, 1)	(0.055, 0.66, 1.75)	(0, 0, 2)	*	
3	6	(1, 0, 1)	(0, 1, 1)	(0.055, 0.66, 1.75)	*	
4	1	(−1, 0, 1)	(0, 1, 1)	(0.055, 0.66, 1.75)	*	
4	2	(0, 0, 2)	(−1, 0, 1)	(0.055, 0.66, 1.75)	*	
4	3	(0, 0, 2)	(0, 1, 1)	(−1, 0, 1)	(0.05, 0.60, 1.60)	−0.7225

* Optimal LP objective has value ≦1.

Notation: q indexes auxiliary problem; k_q indexes solution generated for q^{th} auxiliary problem; y^{qk_q} is Tui's search direction. See [72] for details.

Also see Porembski[53].

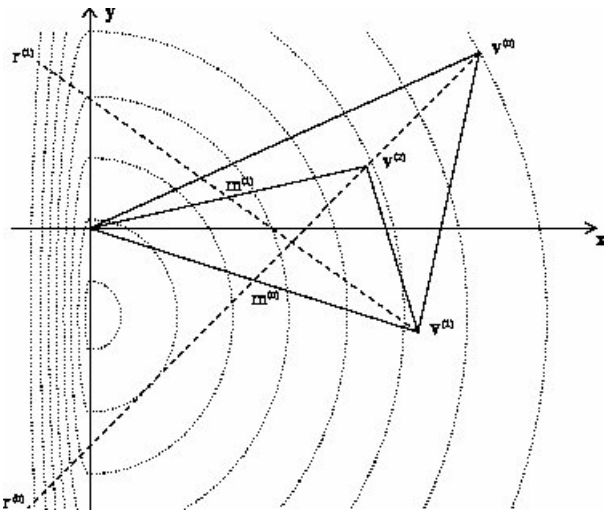
NLP Myth 34. *The Nelder-Mead method converges to a local optimum.*

The Nelder-Mead method is a very good heuristic that does well in many hard nonlinear problems. For a long time after its publication in 1965, many thought it converges to a local optimum, but McKinnon[47] provided the following:

Counterexample. Let

$$f(x,y) = \begin{cases} AB|x|^c + y + y^2 & \text{if } x \le 0 \\ B\,x^c + y + y^2 & \text{if } x \ge 0, \end{cases}$$

where A, B, c are positive constants. Also, f is convex and has continuous first derivatives for $c > 1$.



Contours of McKinnon's function for $A = 6$, $B = 60$, $c = 2$, taken from [47]. Note: $(0,-1)$ is a descent direction from $(0,0)$.

McKinnon proves (nontrivially) that for certain choices of these constants, the algorithm repeats the inside contraction step with the best vertex remaining fixed. In particular, with $A = 6$ and $B = 60$, the counterexample works for $0 \le c \le \overline{c}$, and it does not work for $c > \overline{c}$, where $\overline{c} \approx 3.06$, from McKinnon's derivation.

NLP Myth 35. *Rosen's decomposition method converges to an optimal solution for convex programs.*

We are given the convex program:

$$\min_{(x,y)} \; cx : \; Ax \ge b(y),$$

where b is a convex function. Rosen's decomposition is to separate x and y problems and proceed as follows. For any fixed $y = \bar{y}$, we obtain $\bar{x}$ by solving:

$$\text{LP}(\bar{y}): \min_x cx : Ax \geq b(\bar{y}).$$

Partition $[A\ b]$ into the tight and surplus constraints at the solution:

$$B\bar{x} = b_B(\bar{y}),\ N\bar{x} > b_N(\bar{y}).$$

(So $A = \binom{B}{N}$ and $b = \binom{b_B}{b_N}$.) We suppose B is nonsingular and use the tight constraints to eliminate $x = B^{-1}b_B(y)$ for any choice of y. To maintain feasibility of the surplus constraints, we require

$$b_N(y) - N^\intercal B^{-1} b_B(y) \leq 0.$$

Using the Taylor expansion at $\bar{y}$ to linearize the constraints, Rosen's method solves the nonlinear program:

$$\begin{aligned}\text{NLP(B}, \bar{y}): \quad & \min_y\ cB^{-1}b_B(y) : \\ & b_N(\bar{y}) + \nabla b_N(\bar{y})(y - \bar{y}) - N^\intercal B^{-1}\big(b_B(\bar{y}) + \nabla b_B(\bar{y})(y - \bar{y})\big) \leq 0.\end{aligned}$$

Rosen's method is to start with y^0, then solve iteratively:

1. Solve LP(y^k) and obtain B
2. Solve NLP(B, y^k) and obtain y^{k+1}.

Subject to some details about step 2, the idea is to solve a sequence of problems that are decomposed, rather than tackle the whole nonlinear problem.

Grossmann[32] provided the following:

Counterexample. min $x : x \geq y^2,\ x \geq y$. Let the starting value satisfy $y^0 > 1$. Grossmann proves $y^k > 1$ for all $k = 0, 1, \ldots$, but the optimum is at $(x^*, y^*) = (0, 0)$.

Proceeding inductively, suppose $y^k > 1$. Then, $x^k = (y^k)^2$, and the optimal basis has the slack variable $s = x - y$. Therefore, y^{k+1} is determined by

$$y^{k+1} \in \text{argmin}\{y^2 : (2y^k - 1)y \geq (y^k)^2\}.$$

Since $y^k > 1$, the solution is $y^{k+1} = \frac{(y^k)^2}{2y^k - 1} > 1$, and that completes the induction proof.

NLP Myth 36. *In methods of feasible directions, it is better to improve the objective function each iteration than allow it to worsen.*

Counterexample. Grippo, Lampariello, and Lucidi[31] illustrated the use of their *nonmonotone method*

$$x^{k+1} = x^k - s_k[\nabla^2 f(x^k)]^{-1}\nabla f(x^k),$$

where $s_k = \text{sign}\big(\nabla f(x^k)^\intercal[\nabla^2 f(x^k)]^{-1}\nabla f(x^k)\big)$.

The counterexample applies this to an unconstrained minimization in $\mathbb{R}^2$ using Rosenbrock's function: $f(x_1, x_2) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2$. The minimum is at $x^* = (1, 1)$,

and the starting point is $x^0 = (-1.2, 1)$. Their nonmonotone method converges in 7 iterations with $f(x^4) > f(x^3)$:

k	x_1^k	x_2^k	$f(x^k)$
0	−1.2000000000	1.0000000000	24.200
1	−1.1752808989	1.3806741573	4.732
2	0.7631148712	−3.1750338547	1.412
3	0.7634296789	0.5828247755	0.056
4	0.9999953111	0.9440273239	0.3132
5	0.9999956957	0.9999913913	1.853×10^{-11}
6	1.0000000000	1.0000000000	3.433×10^{-20}
7	1.0000000000	1.0000000000	$< 10^{-38}$

Steepest descent using optimal step size takes 33 iterations to get as close. Armijo's rule[2] takes 22 iterations. This illustrates that improving the objective function every iteration is not necessarily a most effective way to reach the optimum.

Generally, Rosenbrock's function is used to illustrate profuse zig-zagging in Cauchy's steepest descent with $s_k \in \operatorname{argmin}\{f(x^k + s\nabla f(x^k)) : s \geq 0\}$. The nonmonotone method in this counterexample highlights the need to capture curvature information, as does Newton's method. (Also see MacMillan[40] for mixing steepest descent with Newton's method.)

NLP Myth 37. *Sequential quadratic programming (SQP) is quadratically convergent when it is sufficiently close to the solution.*

Given $\max\{f(x) : x \in \mathbb{R}^n,\ g(x) = 0\}$, where f, g have continuous second derivatives, theSQP subproblem is given by the iteration subproblem:

$$\max\ f(x^k) + \nabla f(x^k)(x - x^k) + \tfrac{1}{2}(x - x^k)^\intercal H(x^k)(x - x^k) : \\ g(x^k) + \nabla g(x^k)(x - x^k) = 0,$$

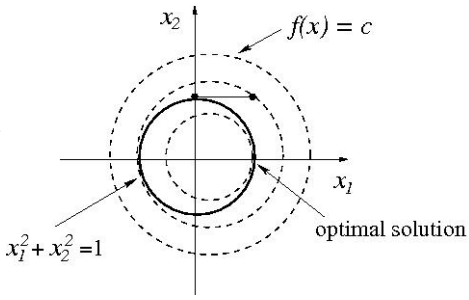
where $H(x^k)$ is the hessian of the Lagrangian: $H(x^k) = \nabla^2 f(x^k) - \lambda^k \nabla^2 g(x^k)$. The subproblem thus requires both x^k and λ^k as input. The constraint is replaced by its linear approximation from the Taylor series, so the iterates need not be feasible.

The intuition is thatSQP behaves like Newton's method when it is close to the solution. The problem is that the step size can become so close to zero that it slows the convergence. This is known as the *Maratos effect*, and it can prevent convergence entirely — see Maratos[42], Panier and Tits[51], Bonnans et al.[7], and Vanden Berghen[63].

Counterexample. Vanden Berghen[63] provides the following:

$$\min\ 2(x_1^2 + x_2^2 - 1) - x_1 : x_1^2 + x_2^2 = 1.$$

The optimum is at $x^* = (1, 0)$, and the starting point is $x^0 = (0, 1)$. TheSQP algorithm finds $x^1 = (1, 1)$, and the next step size is zero.



Taken from Vanden Berghen[63].

Also see Powell[58] for an example of the Maratos effect on convergence of Lagrangian Quadratic Approximation for the NLP:

$$\min\ f(x) = -x_1 + 2(x_1^2 + x_2^2) :\ x_1^2 + x_2^2 = 1.$$

(The optimal solution is at $x^* = (1, 0)$, and the NLP is Lagrange regular.)

The Maratos effect can be overcome by *Second Order Correction* and sometimes by *filtering* — see Fletcher, Leyffer, and Toint[22] — and sometimes by non-monotone methods — see Grippo, Lampariello, and Lucidi[31].

NLP Myth 38. **new** *If an NLP has a unique optimum that satisfies the Mangasaarian-Fromovitz constraint qualification and the quadratic growth condition, there is a locally convex augmented Lagrangian.* ***next new*** ▷

Consider the NLP:

$$\min\ f(x) :\ x \in \mathbb{R}^n,\ g(x) \leq 0,$$

where f, g are once continuously differentiable. Let $I(x^*)$ denote the set of active constraints at x^*, and let $G(x) = g_{I(x^*)}$ be the subvector of g, restricted to $I(x^*)$. The Mangasarian-Fromovitz constraint qualification (MFCQ)† requires the existence of $d \in \mathbb{R}^n$ such that:

$$\nabla G(x^*) d < 0. \tag{NLP.19}$$

(See NLP Myth 2, p. 115.)

The quadratic growth condition is that there exists $\kappa, \varepsilon > 0$ such that

$$f(x^* + d) \geq f(x^*) + \kappa \left|\left|d\right|\right|^2,\ \forall d \in \mathbb{R}^2 :\ \left|\left|d\right|\right| < \varepsilon\,. \tag{NLP.20}$$

The myth asserts that (NLP.19) and (NLP.20) are sufficient to ensure the existence of an augmented Lagrangian,

$$L_a(x, \lambda, \alpha) = f(x) + \lambda g(x) + \alpha G(x)^\intercal G(x),$$

that is convex in some neighborhood of x^* (where $\lambda, \alpha \geq 0$).

†Olvi Mangasarian points out that with only inequality constraints, the qualification is due to Arrow, Hurwicz, and Uzawa; the MFCQ is the extension to *both* inequality and equality constraints. Most people refer to MFCQ even when there are only inequality constraints, and the Myth uses the statement by Anitescu[1].

Anitescu[1] provides the following

Counterexample. Let $Q = \begin{bmatrix} 1 & 0 \\ 0 & -2 \end{bmatrix}$, and define the rotation matrices:

$$U_k = \begin{bmatrix} \cos\left(\frac{k\pi}{4}\right) & \sin\left(\frac{k\pi}{4}\right) \\ -\sin\left(\frac{k\pi}{4}\right) & \cos\left(\frac{k\pi}{4}\right) \end{bmatrix} \qquad \text{for } k = 0, 1, 2, 3.$$

Define $Q_k = {U_k}^{\mathsf{T}} Q U_k$ for the NLP:

$$\min\ x_3 : (x_1,\, x_2) Q_k \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - x_3 \leq 0 \qquad \text{for } k = 0, 1, 2, 3.$$

For any $v \in \mathbb{R}^2$, there exists k such that $v^{\mathsf{T}} Q_k v \geq \frac{1}{4} ||v||^2$, so $x_3 \geq 0$. It follows that the minimum is at $x^* = 0$.

All constraints are active, and the MFCQ is satisfied with $d = (0, 0, -1)$. The quadratic growth condition (NLP.20) is satisfied with constant $\kappa = \frac{1}{8}$.

Consider the hessian of the Lagrangian with multipliers $\lambda = (\lambda_0, \lambda_1, \lambda_2, \lambda_3) \geq 0$:

$$\begin{bmatrix} \sum_{k=0}^{3} \lambda_k Q_k & 0 \\ 0 & 0 \end{bmatrix}.$$

This is positive semi-definite on the critical cone (where $x_3 = 0$) if, and only if,

$$\sum_{k=0}^{3} \lambda_k Q_k \succeq 0.$$

Anitescu proves this is impossible for any multiplier. By a rotation argument, if there is one multiplier for which it holds, it has to hold for the rotated multipliers, and then for their simple average, which is $\lambda = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$. Then,

$$\sum_{k=0}^{3} \lambda_k Q_k = \tfrac{1}{4} \sum_{k=0}^{3} Q_k = -\tfrac{1}{4} I.$$

In forming the augmented Lagrangian, convert the inequality constraints to equality:

$$q_k(x) + s_k = 0,$$

where $q_k(x) = (x_1,\, x_2) Q_k \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - x_3$ and $s \geq 0$. Then, the augmented Lagrangian for $\lambda = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$ is

$$\min\ L_a = x_3 + \sum_{k=0}^{3} \left(\tfrac{1}{4}(q_k(x) + s_k) + \alpha (q_k(x) + s_k)^2 \right) : s \geq 0.$$

The hessian (in $\mathbb{R}^3$) is

$$\nabla^2 L_a = \begin{bmatrix} \frac{1}{4} \sum_{k=0}^{3} Q_k & 0 \\ 0 & 8\alpha \end{bmatrix} = \begin{bmatrix} -\frac{1}{4} & 0 & 0 \\ 0 & -\frac{1}{4} & 0 \\ 0 & 0 & 8\alpha \end{bmatrix},$$

which is not positive semi-definite for any α.

The issue is that second-order conditions may not be satisfied with just (NLP.19) and (NLP.20). Anitescu considers the effects on using the augmented Lagrangian approach of LANCELOT and the linear convergence of SQP.

NLP Myth 39. *A barrier algorithm is globally convergent if the functions are smooth and there is a unique stationary point that is the global optimum.*

Counterexample. Wächter and Biegler[65] provide the following:

$$\min\ x_1 :\ x_1^2 - x_2 = 1,\ x_1 - x_3 = 0.5,\ x_2, x_3 \geq 0.$$

The barrier problem is

$$\min\ x_1 - \mu\big(\ln(x_2) + \ln(x_3)\big) :\ x_1^2 - x_2 = 1,\ x_1 - x_3 = 0.5,\ x_2, x_3 > 0.$$

Note that the non-negativity constraints are replaced by positivity constraints. That is the formal statement of the barrier NLP, but the positivity constraints are ignored (and sometimes omitted, with risk of confusion) because we must use a *continuous-trajectory algorithm*, starting with a feasible point. This excludes, for example, applying the Nelder-Mead algorithm.

Starting with $x^0 = (-2, 3, 1)$, Wächter[64] gives the iterates for a particular barrier algorithm:

k	x_1^k	x_2^k	x_3^k	s_k
0	-2	3	1	–
1	-1.8077165354330709	2.2308661	0.01	0.338
2	-1.1941467232510745	0.0223087	0.0115512	0.264
3	-1.1827848033469259	0.223E-03	0.157E-02	0.125E-01
4	-1.1825077076013453	0.223E-05	0.160E-04	0.109E-02
5	-1.1825049104784042	0.223E-07	0.160E-06	0.111E-04
6	-1.1825048825066040	0.223E-09	0.160E-08	0.111E-06
7	-1.1825048822268860	0.223E-11	0.160E-10	0.111E-08
8	-1.1825048822240889	0.223E-13	0.160E-12	0.111E-10
9	-1.1825048822240609	0.223E-15	0.160E-14	0.111E-12

The algorithm aborts because the step size becomes too small. (see NLP Myth 37 for the Maratos effect.)

Wächter and Biegler note that the example has no degeneracy hidden in the equality constraints, the Jacobian is nonsingular everywhere, and the minimum satisfies second-order sufficient conditions and is strictly complementary. Hence, the counterexample is well posed, not some esoteric pathology.

Wächter's thesis[64] provides a deeper analysis of the above counterexample and shows that seemingly reasonable barrier algorithms from a generic class cannot be globally convergent under mild assumptions. The root of the problem is that those methods compute search directions that satisfy the linearization of the constraints (for example, line-search methods) and are later cut short to keep the iterates positive. Only algorithms that deviate from this paradigm (such as certain trust-region or filter methods) can be shown to have good

convergence properties. See Benson, Shanno, and Vanderbei[5] for further analysis of this example.

Larry Biegler adds the following points.

1. Failure of this example occurs for barrier methods where the search direction satisfies linearization of the equality constraints, followed by a line search, using any merit function (for example, a line-search based Newton method). Because of this restriction and the need to remain feasible to the bounds, the algorithm eventually terminates because it is too constrained to find a search direction to reduce the infeasibility of the equalities. [This is the Maratos effect, which affects SQP and Newton-based methods — see NLP Myth 37.]
2. Wächter's thesis mentions that convergence proofs for Newton-based line search barrier methods (from earlier studies) require boundedness of the multipliers (or similar regularity assumption). This assumption turns out to be violated for this example.
3. There are other barrier methods that can solve this counterexample. For instance, the trust region method (as in KNITRO) generates search directions that are not restricted by the constraint linearization, generate search directions that improve the constraint infeasibility, and avoid this failure of the counterexample.

NLP Myth 40. *Suppose NLP is Lagrange regular with solution at x^*, and the quadratic Lagrangian-approximation algorithm is applied to obtain the direction vector, d^k. Then, convergence is superlinear if* $\lim_{k\to\infty} \frac{||x^k+d^k-x^*||}{||x^k-x^*||} = 0$.

The NLP is of the form $\min\{f(x) : x \in \mathbb{R}^n,\ g(x) = 0\}$, where f and g are twice continuously differentiable and $\nabla g(x^*)$ has full row rank. The algorithm defines the direction vector by the approximation:

$$\min\ d^\intercal \nabla f(x^k) + \tfrac{1}{2} d^\intercal \nabla_x^2 L(x^k, \lambda^k) d :\ g(x^k) + d^\intercal \nabla g(x^k) = 0,$$

where $\nabla_x^2 L(x, \lambda) = \nabla^2 f(x) - \lambda \nabla^2 g(x)$.

Counterexample. Powell[58] provides the following:

$$f(x) = -x_1 + 2(x_1^2 + x_2^2 - 1) :\ x_1^2 + x_2^2 = 1.$$

The optimal solution is at $x^* = (1, 0)$ with Lagrange multiplier, $\lambda^* = \frac{3}{2}$. Hence,

$$\nabla_x^2 L(x^*, \lambda^*) = \begin{bmatrix} 4 & 0 \\ 0 & 4 \end{bmatrix} - \tfrac{3}{2} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

For x close to x^*, $\nabla_x^2 L(x, \lambda)$ is positive definite. Further,

$$x^k = \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} \Rightarrow d^k = \begin{pmatrix} \sin^2\theta \\ -\sin\theta\cos\theta \end{pmatrix}.$$

Choosing θ sufficiently small gives $\left|\left|x^k - x^*\right|\right| \approx |\theta|$ and $\left|\left|x^k + d^k - x^*\right|\right| \approx \frac{1}{2}\theta^2$. Choosing x^0 sufficiently close to x^* establishes the condition:

$$\lim_{k\to\infty} \frac{\left|\left|x^k + d^k - x^*\right|\right|}{||x^k - x^*||} = 0.$$

Further, we have

$$\begin{array}{lll} f(x^k) & = -\cos\theta, & f(x^k + d^k) & = -\cos\theta + \sin^2\theta; \\ g(x^k) & = 0, & g(x^k + d^k) & = \sin^2\theta. \end{array}$$

This rules out the step size $s_k = 1$ because it worsens both the objective value and the constraint violation.

The essence of the counterexample is that a unit step size may not be possible even though it yields superlinear convergence. One would have to accept worsening the objective value and the constraint violation.

NLP Myth 41. *If every optimum in the trajectory of a barrier function satisfies strong second-order necessary conditions, so does its limit.*

Philippe Toint brought this to my attention. The intuition behind the myth is that many unconstrained algorithms can guarantee convergence to a stationary point that satisfies weak second-order conditions if each point in the trajectory does. The issue is whether the same could be said of the strong second-order conditions.

The NLP is min $f(x) : g(x) \geq 0$, for which we consider the trajectory of the logarithmic barrier function:

$$x^*(\mu) \in \operatorname{argmin} \left\{ f(x) - \mu \textstyle\sum_i \log\, g_i(x) : g(x) > 0 \right\} \text{ for } \mu > 0.$$

The myth assumes that $x^*(\mu)$ satisfies strong second-order conditions:

$$h^\intercal \left[\nabla^2 f(x^*(\mu)) - \mu \sum_i \nabla^2 g_i(x^*(\mu)) \right] h > 0 \text{ for all } h \neq 0.$$

Counterexample. Gould and Toint[24] consider the following:

$$\min_{x \in \mathbb{R}^n_+} \ \tfrac{1}{2} x^\intercal Q x,$$

where Q is symmetric and indefinite. The *strong second-order conditions* are that the hessian of the Lagrangian be positive semi-definite over the space of feasible directions, strengthened by disallowing change in $x_i^* = 0$ when its associated Lagrange multiplier, λ_i, is positive:

$$h^\intercal Q h \geq 0 \text{ for all } h : \begin{cases} h_i = 0 \text{ for } x_i^* = 0,\ \lambda_i > 0 \\ h_i \geq 0 \text{ for } x_i^* = 0,\ \lambda_i = 0. \end{cases} \tag{NLP.21}$$

We apply the logarithmic barrier (which is a special case of [24]). For $x > 0$:

$$b(x;\mu) = \tfrac{1}{2} x^\intercal Q x - \mu \textstyle\sum_i \log\, x_i \tag{NLP.22}$$

$$\nabla b(x;\mu) = \quad Q x - \mu X^{-1} \boldsymbol{e} \tag{NLP.23}$$

$$\nabla^2 b(x;\mu) = \quad Q \ + \mu X^{-2} \boldsymbol{e}, \tag{NLP.24}$$

where $\boldsymbol{e}$ is a vector of ones and $X = \operatorname{diag}\{x_i\}$.

Define $Q = I - \frac{3}{2}\frac{z \otimes z^{\mathsf{T}}}{||z||^2}$, where $z = \boldsymbol{e} - ne_1$ and $||\bullet||$ is the Euclidean norm. Then, $x^*(\mu) = \sqrt{\mu}\,\boldsymbol{e}$. This follows from $Q\boldsymbol{e} = \boldsymbol{e}$:

$$\nabla b(x;\mu) = 0 \Rightarrow \sqrt{\mu}\boldsymbol{e} = \mu\,\mathrm{diag}\left(\frac{1}{\sqrt{\mu}}\right)\boldsymbol{e}.$$

To show that the strong second-order conditions hold, substitute in (NLP.24):

$$\nabla^2 b(x;\mu) = \tfrac{1}{2}I + \tfrac{3}{2}\left(I - \frac{z \otimes z^{\mathsf{T}}}{||z||^2}\right).$$

This is positive definite, thus satisfying the condition in the myth. However, $\lim_{\mu \to 0} x^*(\mu) = 0$, and e_1 is an admissible direction for the strong second-order conditions. Hence, for the myth to be true, we require:

$$0 \leq {e_1}^{\mathsf{T}} Q e_1 = 1 - \tfrac{1}{2}\frac{({e_1}^{\mathsf{T}} \otimes z)^2}{||z||^2} = \frac{n - \frac{3}{2}(n-1)}{n}.$$

This is violated for $n \geq 4$.

NLP Myth 42. *The central path converges to the analytic center of the optimality region of a semidefinite program.*

This is true for a linear program, but an attempted extension failed to assume a strictly complementary solution. Halická, de Klerk, and Roos[33] provide the following:

Counterexample.

$$\min\ x_{44}:$$
$$X = \begin{bmatrix} 1 - x_{22} & x_{12} & x_{13} & x_{14} \\ x_{12} & x_{22} & -\frac{1}{2}x_{44} & -\frac{1}{2}x_{33} \\ x_{13} & -\frac{1}{2}x_{44} & x_{33} & 0 \\ x_{14} & -\frac{1}{2}x_{33} & 0 & x_{44} \end{bmatrix} \succeq 0.$$

The optimality region consists of all positive semidefinite matrices of the form:

$$X^* = \begin{bmatrix} 1 - x_{22} & x_{12} & 0 & 0 \\ x_{12} & x_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

In particular, a positive definite optimum is given by setting $x_{22} = \frac{1}{2}$, $x_{33} = x_{44} = \frac{1}{4}$, and $x_{ij} = 0$ for $i \neq j$. Its analytic center is

$$X^* = \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

Halická et al. prove that the central path satisfies

$$\lim_{\mu \downarrow 0} X(\mu) = \begin{bmatrix} 0.4 & 0 & 0 & 0 \\ 0 & 0.6 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

NLP Myth 43. *If an NLP is infeasible, one can sequentially test for the deletion of constraints: if its deletion renders the system feasible, keep it; if its deletion maintains the system infeasible, remove it. What remains is an IIS.*

The indicated method is *deletion filtering*, introduced by Chinneck, to compute an *Irreducible Infeasible Subsystem* (IIS). His recent book[13] provides all background analysis, including the following:

Counterexample. $y - \sqrt{x} = 0,\ x \geq 0,\ y \leq -1.$

If the algorithm drops $x \geq 0$, the solver issues an error message and the algorithm cannot proceed.

Unlike LP, such logical constraints may be needed in NLP. Another source of failure is the inability of the NLP solver to determine whether a nonlinear system is feasible. This is more difficult than for an LP.

NLP Background — Facility Location

The *facility location problem* has the basic formulation:

$$\min \sum_{i=1}^{m} w_i \left|\left|x - a^i\right|\right|,$$

where $a^1, \ldots, a^m$ are distinct points in $\mathbb{R}^n$ and $w \geq 0$.

The *Fermat-Weber problem* is this basic location problem with a Euclidean norm in $\mathbb{R}^2$. It is also of interest to use $\left|\left|x - a^i\right|\right|^2$, which is a strictly convex, quadratic program. The L_1 and L_∞ norms are also of interest and result linear programs.

One extension is to consider p facilities. Let $x^k \in \mathbb{R}^n$ be the coordinates of the k^{th} facility. Then, the basic NLP extends to:

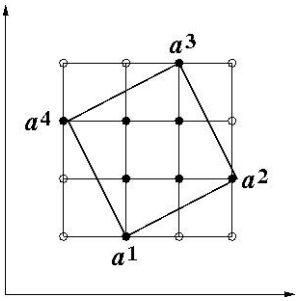
$$\min \sum_{i=1}^{m} w_i d(x, a^i) : \ x \in \mathbb{R}^{p \times n},$$

where $d(x, a^i) = \min_{k=1,\ldots,p} \left|\left|x^k - a^i\right|\right|$ — that is, customers at a^i go to the nearest facility. This is also called the *p-median problem*

One may add a cost (or benefit) of placing the facilities far from each other:

$$\min \sum_{i=1}^{m} w_i d(x, a^i) + \sum_{i=1}^{m} \sum_{k=1}^{p-1} \sum_{j=k+1}^{p} v_{kj} \left|\left|x^k - x^j\right|\right| : \ x \in \mathbb{R}^{p \times n}.$$

Some location problems are defined on a grid, determined by the convex hull of the demand points. The grid points are called *intersection points*, introduced by Wendell and Hurter[68]. They proved that optimal locations for the L_1 norm are at intersection points in the convex hull. The figure at the right is taken from [68]. The filled dots are the intersection points (including the vertices).

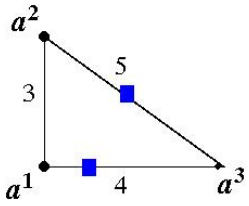


Some location problems are defined on a graph, called the *network location problem*, where the edges have distances between vertices, and a non-empty subset of vertices is specified as the set of demand points. Some of this literature is concerned with the *p-center problem*, where there are p facilities, and the objective is to minimize the maximum distance:

$$\min_{x=(x^1,\ldots,x^p)} \max_i \{d(x,a^i)\}.$$

The edge values are assumed to satisfy the elementary distance properties: $d(a^i,a^j) = d(a^j,a^i) \geq 0$ and $d(a^i,a^j) = 0 \leftrightarrow i = j$. In some cases, it is also assumed they satisfy the triangle inequality: $d(a^i,a^j) \leq d(a^i,a^k) + d(a^k,a^j)$. In this model, the facilities must be located on edges. The distance from a point to the closest facility is the shortest path from the point to a facility.

For example, the graph to the right has three points, whose pair-wise distances are shown. With two facilities, located a fourth of the way along edge $[a^1,a^3]$ and halfway along $[a^2,a^3]$,



$$d(x,a^1) = 1,\ d(x,a^2) = d(x,a^3) = 2.5,$$

so $\max_i\{d(x,a^i)\} = 2.5$. An optimal solution is to place one facility halfway along $[a^1,a^2]$ and the other at a^3. Then,

$$d(x,a^1) = d(x,a^2) = 1.5,\ d(x,a^3) = 0,$$

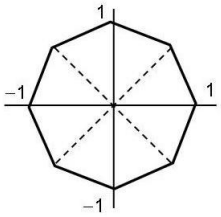
so $\max_i\{d(x,a^i)\} = 1.5$.

NLP Myth 44. *For any polyhedral norm, the optimal location of each facility is in the finite set of intersection points.*

The myth is true for the L_1 norm, but not necessarily for any polyhedral norm, introduced by Ward and Wendell[66].

Counterexample. Michelot[48] provides the following, for a 3-facility problem with the objective equal to the sum of distances and w, v indicated below.

Let $B(0,1)$ be a regular octagon inscribed in a circle centered at the origin with radius 1, as shown on the right. The associated polyhedral norm is defined by:



$$||x|| = \inf\{\lambda > 0 : (x/\lambda) \in B(0,1)\}.$$

To compute $||x||$ we solve the inequalities associated with the faces of the octagon:

$$\begin{array}{l}\min\ \lambda: \\ \begin{array}{rcrcr} \frac{1}{\sqrt{2}}\lambda & \geq & -\frac{1}{\sqrt{2}}x_1 & + & \left(1-\frac{1}{\sqrt{2}}\right)x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & \left(\frac{1}{\sqrt{2}}-1\right)x_1 & + & \frac{1}{\sqrt{2}}x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & \left(1-\frac{1}{\sqrt{2}}\right)x_1 & + & \frac{1}{\sqrt{2}}x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & \frac{1}{\sqrt{2}}x_1 & + & \left(1-\frac{1}{\sqrt{2}}\right)x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & \frac{1}{\sqrt{2}}x_1 & + & \left(\frac{1}{\sqrt{2}}-1\right)x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & \left(1-\frac{1}{\sqrt{2}}\right)x_1 & - & \frac{1}{\sqrt{2}}x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & \left(\frac{1}{\sqrt{2}}-1\right)x_1 & - & \frac{1}{\sqrt{2}}x_2 \\ \frac{1}{\sqrt{2}}\lambda & \geq & -\frac{1}{\sqrt{2}}x_1 & + & \left(\frac{1}{\sqrt{2}}-1\right)x_2 \end{array} \end{array}$$

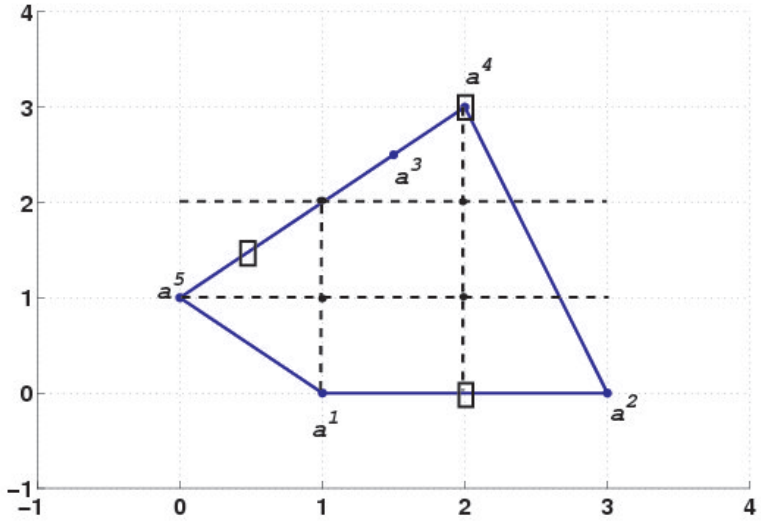
Thus,

$$\begin{aligned} \lambda &= \max \left\{ \begin{array}{rcr} -x_1 & + & (\sqrt{2}-1)\,x_2 \\ (1-\sqrt{2})\,x_1 & + & x_2 \\ (\sqrt{2}-1)\,x_1 & + & x_2 \\ x_1 & + & (\sqrt{2}-1)\,x_2 \\ x_1 & + & (1-\sqrt{2})\,x_2 \\ (\sqrt{2}-1)\,x_1 & - & x_2 \\ (1-\sqrt{2})\,x_1 & - & x_2 \\ -x_1 & + & (1-\sqrt{2})\,x_2 \end{array} \right. \\ &= \max\left\{\{|x_1| + (\sqrt{2}-1)\,|x_2|,\ (\sqrt{2}-1)\,|x_1| + |x_2|\right\} \\ &= \max\{|x_1|,|x_2|\} + (\sqrt{2}-1)\min\{|x_1|,|x_2|\}. \end{aligned}$$

Let $a^1 = (1,0)$, $a^2 = (3,0)$, $a^3 = \frac{1}{2}(3,5)$, $a^4 = (2,3)$, and $a^5 = (0,1)$. Then, Michelot's NLP is:

$$\begin{array}{l} \min\ \frac{3}{\sqrt{2}}\left|\left|x^1 - a^1\right|\right| + 2\sqrt{2}\left|\left|x^1 - a^2\right|\right| \ +\ \left|\left|x^2 - a^3\right|\right| + (1+\frac{1}{\sqrt{2}})\left|\left|x^2 - a^4\right|\right| \\ +\ (1+\sqrt{2})\left|\left|x^3 - a^3\right|\right| + (1+\sqrt{2})\left|\left|x^3 - a^5\right|\right| \ +\ \left|\left|x^1 - x^2\right|\right| + \left|\left|x^1 - x^3\right|\right|. \end{array}$$

The solution is $x^1 = (2,0)$, $x^2 = (2,3)$, and $x^3 = (0.5, 1.5)$, with objective value $f(x) = 18.8995$.



The best solution at intersection points is $x^1 = (2, 0)$, $x^2 = (2, 3)$, and $x^3 = (1, 1)$, with $f(x) = 27.8492$.

NLP Myth 45. *A location model with non-closest facility service has an optimal solution that locates all facilities at the given points.*

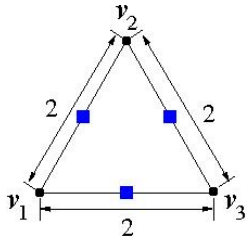
Weaver and Church[67] posed the problem of assigning demand points to multiple facilities. Let d_{ij} denote the distance from point i to facility j, for $i = 1, \ldots, n$ and $j = 1, \ldots, p$. Let $\pi(i, j) = k$ mean that j is the k^{th} closest facility to i. Let a_i denote the population at point i, and let $b_{\pi(i,j)}$ denote the fraction of time point i must be serviced by facility j. The problem is to locate p facilities to minimize

$$\sum_{i=1}^{n} a_i \sum_{j=1}^{p} b_{i\pi(i,j)} d_{ij}.$$

Counterexample. Hooker and Garfinkel[35] provide the following. Let $a_i = 1$, $b_{i1} = b_{i3} = 0$, and $b_{i2} = 1$, for $i = 1, 2, 3$. In words, this says that each point is serviced by its second-closest facility.

Let the coordinates of the given points define an equilateral triangle with the length of each side equal to 2. Placing the facilities at the points yields a total cost of 6 — that is,

$$\sum_{i=1}^{n} a_i \sum_{j=1}^{p} b_{i\pi(i,j)} d_{ij} = d_{12} + d_{23} + d_{31} = 6.$$



Placing the facilities on the mid-point of each side, as shown, yields a total distance of 3.

NLP Myth 46. *In the p-center problem, for $p < n$, it is optimal to locate the facilities at local centers of degree $p + 1$ or less.*

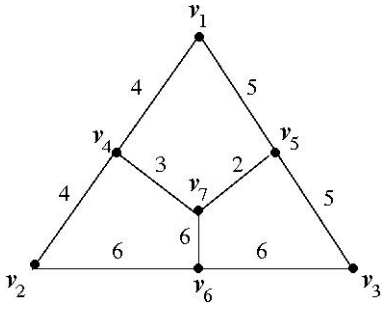
A *local center* of degree k is any set of points x^* that minimizes the maximum distance to the demand points:

$$\{x^* : x^* \in \operatorname*{argmin}_x \max_{i \in I}\{d(x, a^i)\}\},$$

where $I \subseteq \{a^1, \ldots, a^n\}$ such that $|I| = k$. In particular, the local center of an edge (where $|I| = 2$) is its midpoint.

Counterexample. Moreno[50] provides the following. Let $p = 1$, so the myth asserts that the optimal location of the facility is at one of the edge local centers.

The graph to the right has seven points, and vertex v_7 is the optimal location for $A = \{v_1, v_2, v_3\}$. However, it is not a local center for any pair of vertices.



NLP Myth 47. *The set of vertices and local centers is a finite dominating set for the p-facility centdian network problem.*

A *dominating set* for a location problem is a set of points that contains an optimal solution for all instances of the problem. For network location problems, the set of vertices is a finite dominating set for the p-median problem, and the set of local centers (as defined by Moreno's correction[50] — see NLP Myth 46) is a finite dominating set for the p-center problem. The p-facility centdian network problem consists of finding the p points that minimize a convex combination of the p-center and p-median objective functions. The myth asserts that the union of their dominating sets is a finite dominating set for any convex combination of objective functions. Indeed, this is true for $p = 1$.

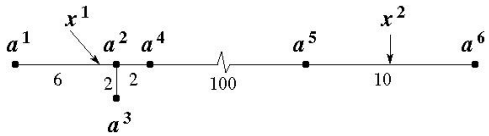
The model is to minimize

$$f(x) = \lambda \max_i d(x, a^i) + (1 - \lambda) \sum_i d(x, a^i),$$

where $\lambda \in [0, 1]$. This is a compromise between minimizing the maximum distance ($\lambda = 1$) and the total distance ($\lambda = 0$).

Counterexample. Pérez-Brito, Moreno-Pérez, and Rodríguez-Martín[52] provide the following.

There are six points, shown on the right, with the distances shown at each edge. For example, $||a^1 - a^2|| = 6$.



The optimal locations for $\lambda = 0.8$ are $x^1 = 5 \in [a^1, a^2]$ and $x^2 = 113 \in [a^5, a^6]$. The optimal objective value is

$$\begin{aligned} f(x) &= 0.8\left|\left|x^1 - a^1\right|\right| + 0.2\left(\sum_{i=1}^{4}\left|\left|x^1 - a^i\right|\right| + \sum i = 4^5\left|\left|x^2 - a^i\right|\right|\right) \\ &= 0.8 \times 5 + 0.2(12 + 10) = 8.4. \end{aligned}$$

Note that x^1 is not a vertex or local center, thus violating the myth.

Pérez-Brito et al. propose a new finite dominating set and prove its validity.

NLP Myth 48. *The minimum total cost is a (discrete) convex function of the number of facilities.*

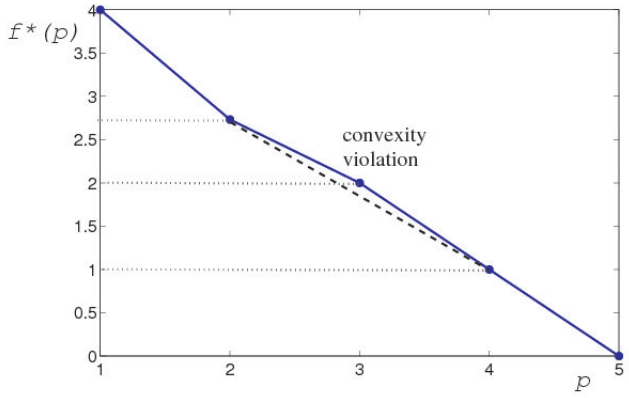
The model is the basic form:

$$f^*(p) = \min\left\{\sum_{i=1}^{m} w_i d(x, a^i) : x \in \mathbb{R}^{p \times n}\right\}.$$

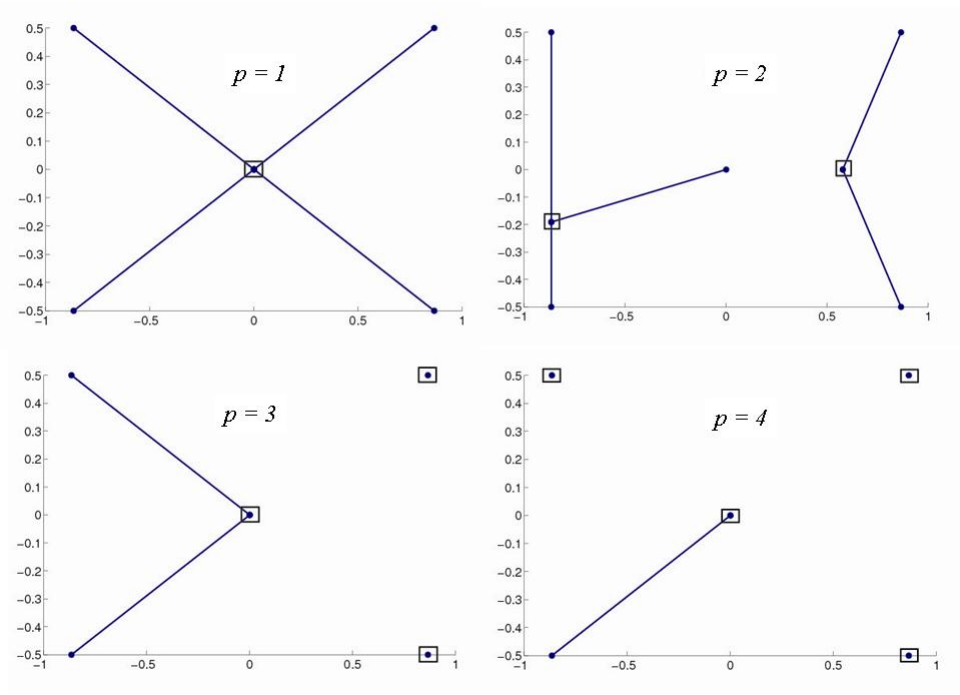
Brimberg, Mladenović, and Salhi[10] provide the following:

Counterexample. Let $a^1 = (0, 0)$, $a^2 = (\frac{1}{2}\sqrt{3}, -\frac{1}{2})$, $a^3 = (\frac{1}{2}\sqrt{3}, \frac{1}{2})$, $a^4 = (-\frac{1}{2}\sqrt{3}, \frac{1}{2})$, and $a^5 = (-\frac{1}{2}\sqrt{3}, -\frac{1}{2})$. Here are the optimal values as a function of p for $w = \mathbf{1}$:

p	$f^*(p)$
1	4.00
2	2.73
3	2.00
4	1.00
5	0



Here are the optimal facility locations:



NLP Myth 49. *Weiszfeld's algorithm converges to the optimal solution of the Fermat-Weber location problem for all but a denumerable number of starting points.*

Thanks to Adi Ben-Israel for suggesting this.

Weiszfeld's algorithm computes $x^{k+1} = T(x^k)$, where

$$T(x) = \begin{cases} a^i & \text{if } x = a^i \text{ for some } i; \\ \dfrac{\sum_{i=1}^m w_i \left|\left|x - a^i\right|\right|^{-1} a^i}{\sum_{i=1}^m w_i \left|\left|x - a^i\right|\right|^{-1}} & \text{otherwise.} \end{cases}$$

Kuhn[38] proposed the myth at a time when Weiszfeld's algorithm was very popular because it is so easy to implement.

Note that once x equals one of the given points, the iterates remain at that point — that is, $T(a^i) = a^i$ for all i. Thus, $x^0 = a^i$ is not convergent to an optimum unless a^i is optimal. More generally, we could have $T(x) = a^i$ for a non-denumerably infinite number of x. As an example[12], consider $a^1 = (1, 0, 0)$, $a^2 = (0, 1, 0)$, $a^3 = (-1, -1, 0)$, and $a^4 = (0, 0, 0)$. Then, for $w = \mathbf{1}$, $T(x) = a^4$ for any $x \in \{(-1/6, -1/6, x_3) : x_3 \in \mathbb{R}\}$. a^4 is optimal for $w = \mathbf{1}$, so Wieszfeld's algorithm is successful in this case.

Counterexample. Chandrasekaran and Tamir[12] provide the following: $a^1 = (1, 0, 0)$, $a^2 = (-1, 0, 0)$, $a^3 = (0, 0, 0)$, $a^4 = (0, 2, 0)$, and $a^5 = (0, -2, 0)$. Let $w_1 = w_2 = w_3 = w_5 = 1$ and $w_4 = 3$. Wieszfeld's algorithm converges to a^3 for $x^0 \in X^0$, where

$$X^0 = \left\{(0, x_2, x_3) : (x_2 + \tfrac{5}{2})^2 + x_3^2 = \tfrac{9}{4}\right\}.$$

However, $f(a^3) = 10 > f(0,1,0) = 7 + 2\sqrt{2}$. To prove $T(x^0) = a^3$ for $x^0 \in X^0$, we have

$$\begin{aligned} T(x) = a^3 \quad &\iff \frac{3a^4}{||x - a^4||} + \frac{a^5}{||x - a^5||} = 0 \\ &\iff 6\sqrt{(x_2+2)^2 + x_3^2} = 2\sqrt{(x_2-2)^2 + x_3^2} \\ &\iff \left(x_2 + \tfrac{5}{2}\right)^2 + x_3^2 = \tfrac{9}{4}. \end{aligned}$$

Chandrasekaran and Tamir conjectured that Kuhn's convergence criterion holds if the convex hull of the vertices is of full dimension — that is,

> *If* $\dim\left(\text{convh}\{a^i\}\right) = n$, $\{x : T(x) = a^i\}$ *is denumerable.*

In an effort to prove this, Brimberg[8] extended it to:

> *Let* $X^0 = \{x^0 : T(x^k) = a^i$ *for some* $k\}$. *Then,* X^0 *is denumerable if, and only if,* $\dim\left(\text{convh}\{a^i\}\right) = n$.

This is dispelled by Cánovas, Cañavate, and Marín[11] with the following:

Counterexample. Let $a^1 = (1,0,0)$, $a^2 = (0,1,0)$, and $a^3 = (0,0,0)$. The solution to $T(x) = a^i$ is uniquely $x = a^i$, so $|X^0| = 1$. However, $\dim\left(\text{convh}\{a^i\}\right) = 2 < n = 3$.

Cánovas et al. provide deeper insights into the flaws.

Note: Brimberg and Love[9] proved convergence of Weiszfeld's algorithm for the L_∞ norm.

NLP References

[1] M. Anitescu. Degenerate nonlinear programming with a quadratic growth condition. *SIAM Journal on Optimization*, 10(4):1116–1135, 2000.

[2] L. Armijo. Minimization of functions having Lipschitz continuous first partial derivatives. *Pacific Journal of Mathematics*, 16(1):1–3, 1966.

[3] M. Ascher. Cycling in the Newton-Raphson algorithm. *International Journal of Mathematical Education in Science and Technology*, 5:229–235, 1974.

[4] J. M. Ash and H. Sexton. A surface with one local minimum. *Mathematics Magazine*, 58(3):147–149, 1985.

[5] H. Y. Benson, D. F. Shanno, and R. J. Vanderbei. Interior-point methods for nonconvex nonlinear programming: Jamming and numerical testing. *Mathematical Programming, Series A*, 99(1):35–48, 2004.

[6] D. M. Bloom. FFF #34. The shortest distance from a point to a parabola. *The College Mathematics Journal*, 22(2):131, 1991.

[7] J. F. Bonnans, E. R. Panier, André Tits, and J. L. Zhou. Avoiding the Maratos effect by means of a nonmonotone line search II: Inequality constrained problems — feasible iterates. *SIAM Journal on Numerical Analysis*, 29(4):1187–1202, 1992.

[8] J. Brimberg. The Fermat-Weber location problem revisited. *Mathematical Programming*, 71(1):71–76, 1995.

[9] J. Brimberg and R. F. Love. Global convergence of a generalized iterative procedure for the minisum location problem with ℓp distances. *Operations Research*, 41(6):1153–1163, 1003.

[10] J. Brimberg, N. Mladenović, and S. Salhi. The multi-source Weber problem with constant opening cost. *Journal of the Operational Research Society*, 55(6):640–646, 2004.

[11] L. Cánovas, R. Cañavate, and A. Marín. On the convergence of the Weiszfeld algorithm. *Mathematical Programming, Series A*, 93:327–330, 2002.

[12] R. Chandrasekaran and A. Tamir. Open quesions concerning Weiszfeld's algorithm for the Fermat-Weber problem. *Mathematical Programming*, 44(1–3):293–295, 1989.

[13] J. W. Chinneck. *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*, volume 118 of *International Series in Operations Research & Management Science*. Springer, New York, NY, 2008.

[14] Y-H. Dai. Convergence properties of the BFGS algorithm. *SIAM Journal on Optimization*, 13(3):693–701, 2002.

[15] G. B. Dantzig, J. Folkman, and N. Shapiro. On the continuity of the minimum set of a continuous function. *Journal of Mathematical Analysis and Applications*, 17(3):519–548, 1967.

[16] G. C. Donovan, A. R. Miller, and T. J. Moreland. Pathological functions for Newton's method. *Teaching Numerical Analysis*, 100(1):53–58, 1993.

[17] J. P. Dussault and G. Fournier. On the convergence of the projected gradient method'. *Journal of Optimization Theory and Applications*, 77(1):197–208, 1993.

[18] J. P. Evans and F. J. Gould. Stability in nonlinear programming. *Operations Research*, 18(1):107–118, 1970.

[19] K. Fan, I. Glicksburg, and A. J. Hoffman. Systems of inequalities involving convex functions. *American Mathematical Society Proceedings*, 8(3):617–622, 1957.

[20] W. Fenchel. On conjugate convex functions. *Canadian Journal of Mathematics*, 1(1):73–77, 1949.

[21] W. Fenchel. Convex cones, sets and functions. Lecture notes, Princeton University, Department of Mathematics, Princeton, NJ, 1953.

[22] R. Fletcher, S. Leyffer, and Ph. Toint. A brief history of filter methods. Preprint ANL/MCS-P1372-0906, Argonne National Laboratory, Mathematics and Computer Science Division, Argonne, IL, 2006.

[23] J. Gauvin and F. Dubeau. Some examples and counterexamples for the stability analysis of nonlinear programming problems. *Mathematical Programming Study*, 21:69–78, 1983.

[24] N. I. M. Gould and Ph. L. Toint. A note on the convergence of barrier algorithms to second-order necessary points. *Mathematical Programming, Series A*, 85(2):433–438, 1999.

[25] H. J. Greenberg. Lagrangian duality gaps: Their source and resolution. Technical Report CP-69005, CS/OR Center, Southern Methodist University, Dallas, Texas, 1969.

[26] H. J. Greenberg. Interpreting rate analysis. *SIGMAP Bulletin*, 17:33–37, 1974.

[27] H. J. Greenberg. The one dimensional generalized Lagrange multiplier problem. *Operations Research*, 25(2):338–345, 1977.

[28] H. J. Greenberg. Wolfe's example and the zigzag phenomenon. In A. G. Holder, editor, *Mathematical Programming Glossary*, http://glossary.computing.society.informs.org/, 1996. INFORMS Computing Society.

[29] H. J. Greenberg and W. P. Pierskalla. A review of quasi-convex functions. *Operations Research*, 19(7):1553–1570, 1971.

[30] H. J. Greenberg and W. P. Pierskalla. Extensions of the Evans-Gould stability theorems for mathematical programs. *Operations Research*, 20(1):143–153, 1972.

[31] L. Grippo, F. Lampariello, and S. Lucidi. Nonmonotone line search technique for Newton's method. *SIAM Journal on Numerical Analysis*, 23(4):707–716, 1986.

[32] Ch. Grossmann. A counterexample to Rosen's decomposition method. *Optimization*, 13(4):511–514, 1982.

[33] M. Halická, E. de Klerk, and C. Roos. On the convergence of the central path in semidefinite optimization. *SIAM Journal on Optimization*, 12(4):1090–1099, 2002.

[34] H. Hancock. *Theory of Maxima and Minima*. Dover, New York, NY, 1960.

[35] J. N. Hooker and R. S. Garfinkel. On the vector assignment p-median problem. *Transportation Science*, 23(2):139–140, 1989.

[36] X.-D. Hu. A counterexample of the convergence of Rosen's algorithm with inexact line search. *Operations Research Letters*, 13(2):95–97, 1993.

[37] S. Karamardian. *Duality in Mathematical Programming*. Ph.D. thesis, University of California, Berkely, CA, 1965.

[38] H. W. Kuhn. A note on Fermat's problem. *Mathematical Programming*, 4(1):98–107, 1973.

[39] H. W. Kuhn and A. W. Tucker. Nonlinear programming. In J. Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, Berkeley, CA, 1951. University of California Press.

[40] D. MacMillan. *Relaxing Convergence Conditions to improve the Convergence Rate*. Ph. D. thesis, University of Colorado Denver, Department of Mathematical Sciences and Statistics, Denver, CO, 1999. available at http://www-math.cudenver.edu/graduate/thesis/macmillan.pdf.

[41] A. A. K. Majumdar. Optimality conditions in differentiable multiobjective programming. *Journal of Optimization Theory and Applications*, 92(2):419–427, 1997.

[42] N. Maratos. *Exact Penalty Function Algorithms for Finite Dimensional and Control Optimization Problems*. Ph. D. thesis, University of London, London, UK, 1978.

[43] B. Martos. Quasi-convexity and quasi-monotonicity in non-linear programming. *Studia Scientiarum Mathematicarum Hungarica*, 2:265–273, 1967.

[44] W. F. Mascarenhas. The BFGS method with exact line searches fails for non-convex objective functions. *Mathematical Programming, Series A*, 99(1):49–61, 2004.

[45] W. F. Mascarenhas. On the divergence of line search methods. *Matemática Aplicada e Computacional*, 26(1):1–29, 2007.

[46] W. F. Mascarenhas. Newton's iterates can converge to non-stationary points. *Mathematical Programming, Series A*, 112(2):327–334, 2008.

[47] K. I. M. McKinnon. Convergence of the nelder-mead simplex method to a nonstationary point. *SIAM Journal on Optimization*, 9(1):148–158, 1998.

[48] C. Michelot. Localization in multifacility location theory. *European Journal of Operational Research*, 31(2):177–184, 1987.

[49] B. Mond. On converse duality for a nondifferentiable programming problem. *Operations Research*, 25(4):713–715, 1977.

[50] J. A. Moreno. A correction to the definition of local center. *European Journal of Operational Research*, 20(3):382–386, 1985.

[51] E. R. Panier and André Tits. Avoiding the Maratos effect by means of a nonmonotone line search I: General constrained problems. *SIAM Journal on Numerical Analysis*, 28(4):1183–1195, 1991.

[52] D. Pérez-Brito, J. A. Moreno-Pérez, and I. Rodríguez-Martín. The 2-facility centdian network problem. *Location Science*, 6(1–4):369–381, 1998.

[53] M. Porembski. Finitely convergent cutting planes for concave minimization. *Journal of Global Optimization*, 20(2):113–136, 2001.

[54] M. J. D. Powell. Some properties of the variable metric algorithm. In F. A. Lootsma, editor, *Numerical Methods for Non-linear Optimization*, pages 1–18, London, UK, 1972. Academic Press. This is the proceedings of the Conference held in Dundee, Scotland, 1971.

[55] M. J. D. Powell. On search directions for minimization algorithms. *Mathematical Programming*, 4(1):193–201, 1973.

[56] M. J. D. Powell. An example of cycling in a feasible point algorithm. *Mathematical Programming*, 20(1):353–357, 1981.

[57] M. J. D. Powell. Nonconvex minimization calculations and the conjugate gradient method. In D. F. Griffiths, editor, *Numerical Analysis Proceedings*, volume 1066 of *Lecture Notes in Mathematics*, pages 122–141, Berlin, FRG, 1984. Springer-Verlag.

[58] M. J. D. Powell. Convergence properties of algorithms for nonlinear optimization. *SIAM Review*, 28(4):487–500, 1986.

[59] L. Qi. On an extended Lagrange claim. *Journal of Optimization Theory and Applications*, 108(3):685–688, 2001.

[60] J. B. Rosen. The gradient projection method for nonlinear programming. part I. linear constraints. *Journal of the Society for Industrial and Applied Mathematics*, 8(1):181–217, 1960.

[61] D. F. Shanno. An example of numerical nonconvergence of a variable-metric method. *Journal of Optimization Theory and Applications*, 46(1):87–94, 1985.

[62] J. R. Thompson. Examples of non-convergence of conjugate descent algorithms with exact line-searches. *Mathematical Programming*, 12(1):356–360, 1977.

[63] F. Vanden Berghen. *CONDOR: a constrained, non-linear, derivative-free parallel optimizer for continuous, high computing load, noisy objective functions*. Ph. D. thesis, Université Libre de Bruxelles, Faculté des Sciences Appliquées, Brussels, BE, 2004.

[64] A. Wächter. *An Interior Point Algorithm for Large-Scale Nonlinear Optimization with Applications in Process Engineering*. Ph. D. thesis, Carnegie Mellon University, Department of Chemical Engineering, Pittsburgh, PA, 2002.

[65] A. Wächter and L. T. Biegler. Failure of global convergence for a class of interior point methods for nonlinear programming. *Mathematical Programming*, A 88(3):565–574, 2000.

[66] J. E. Ward and R. E. Wendell. A new norm for measuring distance which yields linear location problems. *Operations Research*, 28(3):836–844, 1980.

[67] J. R. Weaver and R. L. Church. A median location model with nonclosest facility service. *Transportation Science*, 19(1):58–74, 1985.

[68] R. E. Wendell and A. P. Hurter, Jr. Location theory, dominance, and convexity. *Operations Research*, 21(1):314–320, 1973.

[69] P. Wolfe. Convergence conditions for ascent methods. *SIAM Review*, 11(2):226–235, 1969.

[70] P. Wolfe. Convergence conditions for ascent methods. II: Some corrections. *SIAM Review*, 13(2):185–188, 1971.

[71] P. Wolfe. On the convergence of gradient methods under constraint. *IBM Journal of Research and Development*, 16(4):407–411, 1972.

[72] P. B. Zwart. Nonlinear programming: Counterexamples to two global optimization algorithms. *Operations Research*, 21(6):1260–1266, 1973.

Multiple-Objective Programming

The multiple-objective mathematical program has the range of the objective function in $\mathbb{R}^N$ with $N > 1$. An optimal solution is defined as follows. A point $x \in X$ is *dominated* by $x' \in X$ if $f(x') \geq f(x)$ and $f_i(x') > f_i(x)$ for some i. (Reverse the inequalities for minimization.) A *Pareto-optimum* is a feasible solution that is not dominated. This is denoted:

$$\text{Pareto-max } f(x) : x \in X,\ g(x) \leq 0,\ h(x) = 0.$$

The *Pareto frontier* is the set of Pareto-optima, denoted

$$\text{argPareto-max}\{f(x) : x \in X,\ g(x) \le 0,\ h(x) = 0\}.$$

One way to generate a Pareto-optimal solution is by taking a strictly-positive combination of the objective functions and solving:

$$\max \textstyle\sum_{i=1}^{N} w_i f_i(x) : x \in X,\ g(x) \le 0,\ h(x) = 0,$$

where $w > 0$. This is sometimes called the *weighted-objective* model, and each optimal solution is Pareto-optimal. Typically, but not always, the weights are normalized by $\sum_{i=1}^{N} w_i = 1$.

For a multiple-objective standard *Linear Program* (LP), the form is given by:

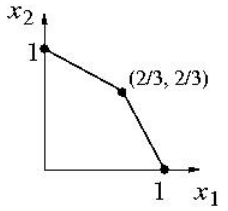
$$\text{Pareto-max } Cx : Ax = b,\ x \ge 0,$$

where C is $N \times n$.

See Zeleny[32] and Steuer[28] for MOP basics.

MOP Myth 1. *For LP, the Pareto frontier is a convex set.*

Counterexample. The following has its Pareto frontier along the two edges defined by $2x_1 + x_2 = 2$ and $x_1 + 2x_2 = 2$, respectively. In particular, points $(1, 0)$ and $(0, 1)$ are Pareto-optimal, but their midpoint, $(\frac{1}{2}, \frac{1}{2})$, is dominated by $(\frac{2}{3}, \frac{2}{3})$



$$\text{Pareto-max } \left\{ \begin{matrix} x_1 \\ x_2 \end{matrix} \right\} : x \ge 0$$
$$2x_1 + x_2 \le 2$$
$$x_1 + 2x_2 \le 2$$

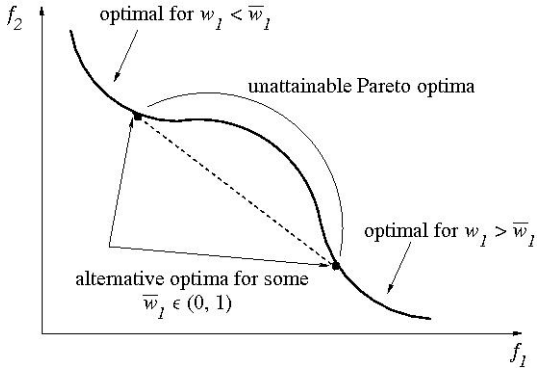
.

More interesting examples are given by Dybvig and Ross[7].

MOP Myth 2. *Varying the weights of a convex combination of objectives generates the Pareto frontier.*

Although it is trivial to show that a solution to the weighted-objective (with $w > 0$) is a Pareto-optimum, the myth asserts the converse: each Pareto-optimum can be generated by some positive weight.

The problem is the same as the duality gap. In particular, the Lagrangian has a duality gap when the optimal response function is not convex (for minimization). That is what happens when the Pareto frontier does not produce a convex function in f_1-f_2 space. (See figure on right.)



Counterexample. Das and Dennis[6] provide the following:

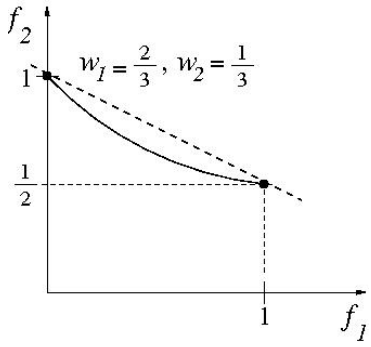
$$\text{Pareto-min} \left\{ \begin{array}{l} f_1(x) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2, \\ f_2(x) = 3x_1 + 2x_2 - \frac{1}{3}x_3 + 0.01(x_4 - x_5)^3\} \end{array} \right\} :$$
$$x_1 + 2x_2 - x_3 - 0.5x_4 + x_5 = 2$$
$$4x_1 - 2x_2 + 0.8x_3 + 0.6x_4 + 0.5x_5^2 = 0$$
$$x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 \le 10.$$

The essential features of the counterexample are:

1. the objectives are in conflict — that is, f_2 is a decreasing function of f_1 in the range space;
2. the Pareto frontier (f_1–f_2 curve) lacks a linear support in some region.

A simpler counterexample is thus:

$$\text{Pareto-max} \begin{pmatrix} x \\ \dfrac{1}{1+x} \end{pmatrix} : 0 \le x \le 1.$$



The weighted-objective problem is

$$\max\ w_1 x + w_2 \frac{1}{1+x} : 0 \le x \le 1,$$

where $w > 0$ and $w_1 + w_2 = 1$.

Every feasible point is Pareto-optimal, but only the extreme points, $\{0, 1\}$, can be generated by varying the weights. For $w_1 > \frac{2}{3}$, the weighted-objective solution is at $x = 1$; for $w_1 < \frac{2}{3}$, it is at $x = 0$. For $w_1 = \frac{2}{3}$, both extreme points are optimal, but no point in the interior, $(0, 1)$, is ever optimal in the weighted-objective model.

This is the same property as the Lagrangian duality gap, viewed in response space. Specifically, consider the parametric program:

$$\max\ f_2(x) : x \in X,\ f_1(x) \ge b,$$

where both f_1 and f_2 are increasing and continuous on X, which is compact. This yields the Pareto frontier as b is varied in $\left[\min_{x \in X} f_1(x), \max_{x \in X} f_1(x)\right]$. The associated Lagrangian problem is

$$L^*(\lambda) = \max\{f_2(x) + \lambda f_1(x) : x \in X\},$$

for $\lambda \ge 0$. Since f is convex, the solution occurs at an extreme point:

$$L^*(\lambda) = \max_{0 \le x \le 1} \left\{ \tfrac{1}{1+x} + \lambda x \right\} = \max\{1,\ \tfrac{1}{2} + \lambda\};$$

$$x^*(\lambda) = \left\{ \begin{array}{ll} 0 & \text{if } \lambda < \frac{1}{2} \\ 1 & \text{if } \lambda > \frac{1}{2} \\ 0,\ 1 & \text{if } \lambda = \frac{1}{2}. \end{array} \right.$$

The Lagrangian dual is to minimize $L^*(\lambda) - \lambda b$ on $\mathbb{R}_+$. In response space, the Lagrangian is a linear function, so there is a duality gap whenever there is no linear support for the optimal value function:

$$f^*(b) = \max\ f_2(x) : x \in X,\ f_1(x) \geq b$$

on $B = \{b : f_x(x) \geq b \text{ for some } x \in X\}$. In particular, $\nexists \lambda \in \mathbb{R}_+$ such that $x^* \in \text{argmax}\{f_2(x) + \lambda f_1(x) : x \in X\}$ for any $x^* \in (0, 1)$.

The Lagrangian problem corresponds to the weighted objective with $w_1 = \frac{1}{1+\lambda}$ and $w_2 = \frac{\lambda}{1+\lambda}$. The inability to generate a region of the Pareto frontier is precisely the Lagrangian duality gap.

MOP Myth 3. *Points obtained by computing individual-objective maxima bound the range of objective values over the Pareto frontier.*

My thanks to Wlodzimierz Ogryczak for suggesting this.

The idea is that we first compute

$$f_i^{\max} = \max\{f_i(x) : x \in X\} \text{ for } i = 1, \ldots, N \Rightarrow X^{\max} = \bigcup_{i=1}^{N} \text{argmax}\{f_i(x) : x \in X\}.$$

This yields

$$f_i^{\min} = \min\{f_i(x) : x \in X^{\max}\} \text{ for } i = 1, \ldots, N.$$

The myth asserts $f^{\min} \leq f(x) \leq f^{\max}$ for all $x \in X^*$. The upper bound is valid, but $f^{\min}$ need not be a valid lower bound. Ogryczak provided the following

Counterexample. Pareto-max $\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} : x \in X$, where

$$X = \{(10, 3, 3), (3, 10, 3), (3, 3, 10), (1, 8, 8), (8, 1, 8), (8, 8, 1)\}.$$

Since no $x \in X$ is dominated, the Pareto frontier, X^*, equals X. The individual-objective maxima are computed:

$$f_i^{\max} = \max\{x_i : x \in X^*\} = 10 \text{ for } i = 1, 2, 3 \Rightarrow X^{\max} = \{(10, 3, 3), (3, 10, 3), (3, 3, 10)\}.$$

This yields

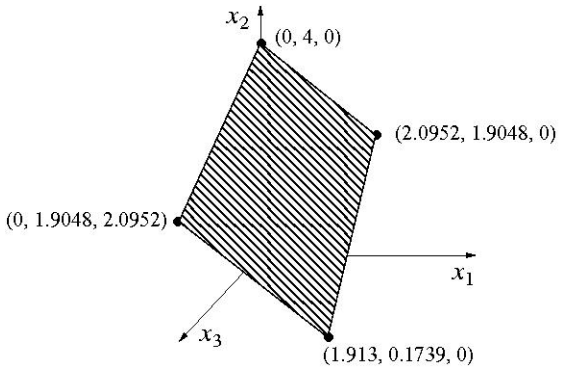
$$f_i^{\min} = \min\{x_i : x \in X^{\max}\} = 3 \text{ for } i = 1, 2, 3.$$

The myth asserts $3 \leq f_i(x) \leq 10$ for all $x \in X^*$ and all i. However, $f_1(x^4) = f_2(x^5) = f_3(x^6) = 1$.

Counterexample. Korhonen and Wallenius[17] provide an LP counterexample:

$$\text{Pareto-max} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} : x \geq 0$$

$$\begin{array}{rcl} x_1 + x_2 + x_3 & \leq & 4 \\ 0.05x_1 - 0.05x_2 + x_3 & \leq & 2 \\ x_1 - 0.05x_2 + 0.05x_3 & \leq & 2 \end{array}$$



The Pareto frontier (shaded diamond) is the convex hull of the four points indicated:

$$X^* = \text{convh}\{(0, 1.9048, 2.0952),\ (2.0952, 1.9048, 0), (0, 4, 0),\ (1.913, 0.1739, 1.913)\}.$$

Computing $f^{\max}$, we obtain

$$\begin{array}{rll} f_1^{\max} & = 2.0952 & = f_1(2.0952,\ 1.9048,\ 0) \\ f_2^{\max} & = 4 & = f_2(0,\ 4,\ 0) \\ f_3^{\max} & = 2.0952 & = f_3(0,\ 1.9048,\ 2.0952) \\ \Rightarrow f^{\min} & = (0,\ 1.9048,\ 0) & \end{array}$$

However, $x = (1.913, 0.1739, 1.913) \in X^*$ violates $f_2(x) \geq f_2^{\min}$.

MOP Myth 4. *Adding an objective function expands the set of Pareto-optima.*

Let X_N^* be the set of Pareto-optima for $f = (f_1, \ldots, f_N)^{\mathsf{T}}$. The Myth asserts $X_N^* \subseteq X_{N+1}^*$.

Counterexample. Lowe, Thisse, Ward, and Wendell[20] provide the following. Let f_1 have two maxima, x^1 and x^2, such that $f_2(x^1) < f_2(x^2)$. Then, $x^1 \in X_1^*$ but $x^1 \notin X_2^*$.

MOP Myth 5. *Consider the multiple-objective convex program:*

$$\text{Pareto-min}\ \ f(x) : x \in \mathbb{R}^n,\ g(x) \leq 0,$$

where f and g are convex and differentiable. Then, x^* is a Pareto-optimal if $g(x^*) \leq 0$, and there exists $w, \lambda \geq 0$ such that $\lambda g(x^*) = 0$ and

$$w\nabla f(x^*) + \lambda \nabla g(x^*) = 0.$$

The basis for this is that these are the Lagrangian (Kuhn-Tucker-Karush) conditions for the weighted model. (The sufficiency is due to the convexity assumptions.)

Counterexample. Kim et al.[16] provide the following:

$$\text{Pareto-min}\ \ \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} : x_1 \geq 0,\ x_1(x_1 - 1) \leq x_2.$$

Consider $x^* = (0,1)$. This is not a Pareto-optimum because it is dominated by $(0,0)$. Let $w = \lambda = (1,0)$, so the conditions stated in the myth are satisfied with:

$$(1,0)\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} + (1,0)\begin{bmatrix}-1 & 0\\ -1 & -1\end{bmatrix} = \begin{pmatrix}0\\ 0\end{pmatrix}.$$

MOP Myth 6. *Consider a multiple-objective LP in standard form. A Pareto-maximum can be obtained from a weighted objective, where the weights (w) are obtained from a solution to:*

$$\min b^{\mathsf{T}}u : u^{\mathsf{T}}A - wC \geq 0,\ w \geq 1.$$

Isermann[14] proposed this with the intuition that this is a sort-of dual to the original Pareto-maximum in the sense that the weighted objective wCx yields this LP (but with w fixed). The goal here is to obtain some initial Pareto-maximum, then find others.

Counterexample. Ecker and Hegner[8] provide the following:

$$\text{Pareto-max}\begin{pmatrix}-x_1\\ x_4\end{pmatrix} : x \geq 0,$$

$$\begin{array}{rcrcrcl} x_1 & - & x_2 & & & = & 1\\ x_1 & & & + & x_3 & = & 2\\ & & & & x_4 & = & 1 \end{array}$$

A Pareto-maximum is $x = (1,0,1,1)$. The LP to obtain the weights is given by:

$$\min\ u_1 + 2u_2 + u_3 : w_1, w_2 \geq 1$$

$$\begin{array}{rcrcrcl} u_1 & + & u_2 & + & w_1 & \geq & 0\\ -u_1 & & & & & \geq & 0\\ & & u_2 & & & \geq & 0\\ & & & u_3 - & w_2 & \geq & 0 \end{array}$$

This is unbounded because we can let $(u,w) = (-2t,0,t,2t,t)$, which is feasible for all $t \geq 1$. The minimand is $-t$, which diverges to $-\infty$ as $t \to \infty$.

Ecker and Kouada[9] give the correct result as follows. Suppose $Cx^0 \neq 0$ for some feasible x^0. Then, there exists a Pareto-maximum if, and only if, the following LP has an optimum:

$$\max \textstyle\sum_{i=1}^{N} s_i : Cx = s + Cx^0,\ Ax \leq b,\ x, s \geq 0.$$

(Omitting Cx^0, the dual is Isermann's LP. The counterexample shows that Cx^0 cannot be omitted.)

In the counterexample, x^0 does not exist because $Cx \neq 0$ for any feasible x. See Benson[4] for additional discussion and another way to get an initial Pareto-maximum that is an extreme point of the feasible polyhedron.

MOP Myth 7. *Let U be an increasing utility function on the range of f on X, and*

$$\max\ U(f(x)) : x \in X.$$

Then, an extreme point with greatest utility value is Pareto-optimal.

Steuer[28, p. 157] provides the following:

Counterexample.

$$\text{Pareto-max} \left\{ \begin{matrix} x_1 \\ x_2 \end{matrix} \right\} :$$

$$\begin{matrix} x_1 & + & x_2 & \leq & 18 \\ 8x_1 & + & 6x_2 & \geq & 112 \\ 5x_1 & + & 7x_2 & \geq & 96 \end{matrix}$$

$$x \geq 0.$$

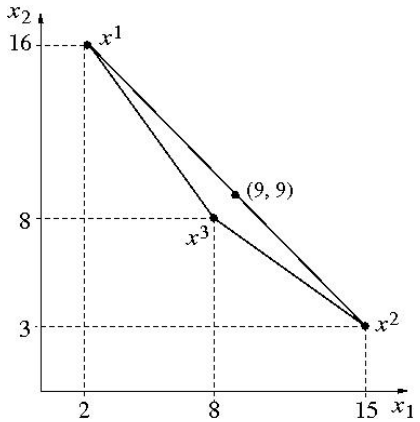


Figure taken from [28].

This has three extreme points:

$$x^1 = (2, 16),\ x^2 = (15, 3),\ x^3 = (8, 8).$$

x^3 is not Pareto-optimal because it is dominated by the non-extreme point $(9, 9)$. However, for $U(f_1, f_2) = f_1 f_2$, x^3 has the greatest utility value (64); the two Pareto-optimal extreme points have lower utility values:

$$U(f(x^1)) = 32,\ U(f(x^2)) = 45.$$

MOP Myth 8. *In a multiple-objective LP, one should put the greatest weight on the most important objective.*

Steuer[28, p. 198–9] provides the following:

Counterexample.

$$\text{Pareto-max} \begin{pmatrix} 2x_1 + 4x_2 \\ 3x_1 - 3x_2 \\ 5x_2 \end{pmatrix} : \begin{matrix} x \geq 0,\ x_2 \leq 12 \\ 3x_1 + 5x_2 \leq 72 \\ 3x_1 - 5x_2 \leq 12 \end{matrix}$$

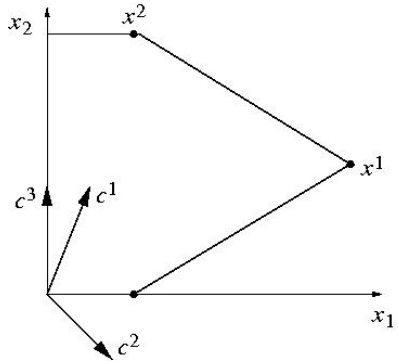


Figure taken from [28].

Assume the objectives are in order of importance, and consider the following two weights: $w^1 = (0.7, 0.2, 0.1)$ and $w^2 = (0, 0.1, 0.9)$. The first weight reflects the relative importance of the objectives and generates the Pareto-optimum point $x^1 = (14, 6)$, with objective values $(52, 33, 30)$. The second one is contrary to the relative importance and generates the Pareto-optimum point $x^2 = (4, 12)$, with objective values $(56, -24, 60)$.

These are counter-intuitive results because x^2 better reflects the objectives' relative importance. The outcome, particularly the latter, where x^1 is the solution, is due to the *correlation* between c^1 and c^3. By placing a large weight on c^3, it is not necessary to place a high weight on c^1.

MOP Myth 9. *All Pareto-optimal solutions are equal.*

The issue is that of *value trade-off*, say between two objectives that are in conflict. In particular, suppose f_1 is cost and f_2 is risk. Keeney[15] provides the following:

Counterexample. Suppose it costs \$3 billion annually if carbon monoxide concentrations are limited to 3 parts per million, and suppose that it costs \$6 billion if concentrations are held to 2 parts per million. We must ask, "Is it worth \$3 billion to lower concentrations from 3 parts per million to 2 parts per million?" The appropriate way to address this is to model the causal relationship between pollutant concentrations and potential health effects. Then, one could deal directly with the value trade-offs between cost of the national air quality standard and the health effects averted.

The two Pareto-optima solutions are not equal, but the model focused attention on them for further analysis.

By its definition, points on the Pareto frontier are indifferent to each other as far as the model is concerned. Applying multiple-objective programming to decision-making, however, we must go beyond finding points on the Pareto frontier. Also, MOP Myth 8 demonstrates that using weights may not resolve the issue. It is important that Pareto-optima help to focus what can be achieved, but ultimately there is a trade-off in the value of one Pareto-optimum versus another. That trade-off could be subjective or with refined analysis. Keeney gives details on this, citing 12 common mistakes in making value trade-offs.

Also see Zeleny[33] and his references.

MOP Background — Pareto-optimum Graph

The following benefitted from comments by Jochen Gorski.

Define a *Pareto-optimum graph* whose nodes correspond to some finite set of Pareto-optimum solutions and whose edges correspond to their *adjacency*. The notion of the finite set of solutions and of their adjacency are not defined in general. For LP, it is natural to define Pareto-optimal adjacency the same as in LP: the nodes are basic optimal solutions, and their adjacency is that of their bases. For combinatorial problems, it is natural to use underlying combinatorial structures. For example, two spanning trees are adjacent if they differ by one edge (having $n-2$ edges in common).

The significance of a Pareto-optimum graph is its connectedness, raising the question if one can traverse the nodes without having to compute a solution that is not Pareto-optimal. If so, this enables neighborhood search to produce them. (See Gorski, Klamroth, and Ruzika[12] for a substantive description of this concept and an up-to-date review of results.)

MOP Myth 10. *The Pareto-minimum graph for spanning trees is connected.*

Ehrgott and Klamroth[10] provide the following:

Counterexample. The edge numbers are the costs of two objectives. In particular, $(0,0)$ is an edge with zero cost in both objectives, as the edge (s_1, s_{11}).

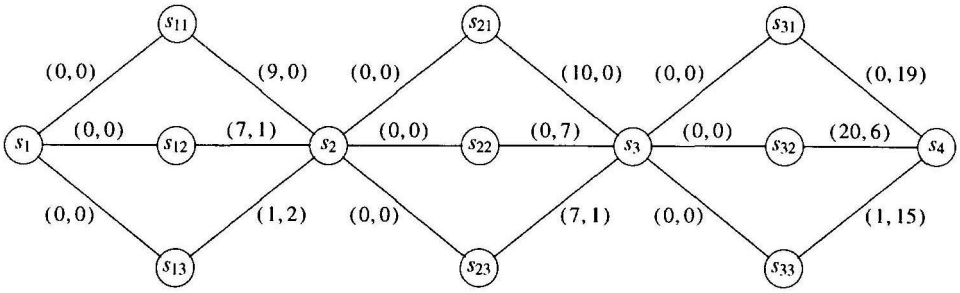


Figure taken from [10].

The following table lists the 12 Pareto-minimum spanning trees, showing the edges with positive costs (all edges with zero cost are in each tree).

Pareto-minimum Tree	Edges with Non-zero Cost	Objective Values
T_1	(s_{13}, s_2), (s_{22}, s_3), (s_{31}, s_4)	$(1, 28)$
T_2	(s_{13}, s_2), (s_{22}, s_3), (s_{33}, s_4)	$(2, 24)$
T_3	(s_{13}, s_2), (s_{23}, s_3), (s_{31}, s_4)	$(8, 22)$
T_4	(s_{13}, s_2), (s_{23}, s_3), (s_{33}, s_4)	$(9, 18)$
T_5	(s_{13}, s_2), (s_{21}, s_3), (s_{33}, s_4)	$(12, 17)$
T_6	(s_{11}, s_2), (s_{23}, s_3), (s_{33}, s_4)	$(12, 17)$
T_7	(s_{11}, s_2), (s_{21}, s_3), (s_{33}, s_4)	$(17, 16)$
T_8	(s_{12}, s_2), (s_{22}, s_3), (s_{32}, s_4)	$(20, 15)$
T_9	(s_{13}, s_2), (s_{23}, s_3), (s_{32}, s_4)	$(27, 14)$
T_{10}	(s_{13}, s_2), (s_{21}, s_3), (s_{32}, s_4)	$(28, 9)$
T_{11}	(s_{11}, s_2), (s_{23}, s_3), (s_{32}, s_4)	$(36, 7)$
T_{12}	(s_{11}, s_2), (s_{21}, s_3), (s_{32}, s_4)	$(39, 6)$

Tree T_8 is not adjacent to any other Pareto-minimum spanning tree.

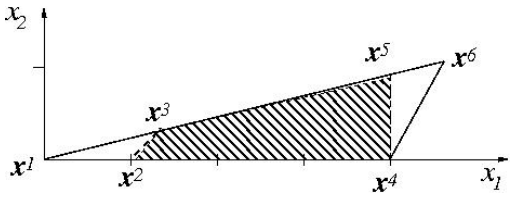
An implication is that to visit each Pareto-minimum spanning tree, we may need to visit a non-optimal spanning tree during the pivoting process. See Przybylski, Gandibleux, and Ehrgott[26] for how this invalidates a class of algorithms that seek to generate Pareto-optimal spanning trees and shortest paths.

MOP Myth 11. *The Pareto frontier is closed.*

The result is true for LP, but Kornbluth and Steuer[18] provide the following for a fractional program:

Counterexample.

$$\text{Pareto-max} \begin{pmatrix} \dfrac{x_1 - 4}{3 - x_2} \\ \dfrac{-x_1 + 4}{x_2 + 1} \\ -x_1 + x_2 \end{pmatrix} : \quad \begin{array}{c} -x_1 + 4x_2 \leq 0 \\ x_1 - \frac{1}{2}x_2 \leq 4 \\ x \geq 0 \end{array}$$



$x^1 = (0, 0)$ $\quad x^2 = (1, 0)$
$x^3 = (\,{}^4\!/_3\,,\, {}^1\!/_3\,)$ $\quad x^4 = (4, 0)$
$x^5 = (4, 1)$ $\quad x^6 = (\,{}^{32}\!/_7\,,\, {}^8\!/_7\,)$

The feasible region is the convex hull of the extreme points, denoted $\text{convh}\{x^1, x^4, x^6\}$. The objective values for each of the points are:

$$f(x^1) = \begin{pmatrix} -4/3 \\ 4 \\ 0 \end{pmatrix} \qquad f(x^2) = \begin{pmatrix} -1 \\ 3 \\ -1 \end{pmatrix}$$

$$f(x^3) = \begin{pmatrix} -1 \\ 2 \\ -1 \end{pmatrix} \qquad f(x^4) = \begin{pmatrix} 0 \\ 0 \\ -4 \end{pmatrix}$$

$$f(x^5) = \begin{pmatrix} 0 \\ 0 \\ -3 \end{pmatrix} \qquad f(x^6) = \begin{pmatrix} 4/13 \\ -4/15 \\ -24/7 \end{pmatrix}.$$

The Pareto frontier is given by the union of convex hulls minus two half-open line segments:

$$X^* = \text{convh}\{x^2, x^3, x^4, x^5\} \cup [x^1, x^2] \cup [x^5, x^6] - (x^2, x^3] - [x^4, x^5),$$

where $-$ denotes the set-minus. Points x^3 and x^4 are not Pareto-optimal, but they are in the closure of X^*. In particular, x^3 is dominated by x^2, but all feasible points on $\{(x^3, x^3 + (\varepsilon, 0)) : \varepsilon > 0\}$ are not dominated — they are Pareto-optimal. Thus, x^3 is a cluster point of X^*, so X^* is not closed.

MOP Myth 12. *If the Pareto frontier contains an interior point, it must contain all interior points.*

The result is true for LP, but the fractional program given by Kornbluth and Steuer[18] in MOP Myth 11 provides the following:

Counterexample. Interior points in $\text{convh}\{x^2, x^3, x^4, x^5\}$ are Pareto-optimal, but those in $\text{convh}\{x^1, x^2, x^3\} \cup \text{convh}\{x^4, x^5, x^6\}$ are not.

MOP Myth 13. *The Pareto frontier is edge-connected.*

The result is true for LP, but the fractional program given by Kornbluth and Steuer[18] in MOP Myth 11 provides the following:

Counterexample. Points x^1 and x^6 are Pareto-optimal, but they are not edge-connected because the edge $(x^2, x^3]$ is not in X^* (neither is $[x^4, x^5)$).

MOP Background — Lexico-optima and Bilevel Programs

Another way to generate Pareto-optimal solutions is by ordering the objective functions and solving sequentially. Suppose $f_1 \succ f_2 \succ \cdots \succ f_N$. Then, the *lexico-optimum* model is given by:

$$\begin{aligned} X^1 &= \text{argmax}\{f_1(x) : x \in X\} \\ X^2 &= \text{argmax}\{f_2(x) : x \in X^1\} \\ &\vdots \\ X^N &= \text{argmax}\{f_N(x) : x \in X^{N-1}\} \end{aligned}$$

(The sequence stops if it is initially infeasible or encounters an unbounded solution.) The points in X^N are Pareto-optimal, with f lexio-graphically ordered: $f(x^*) \succeq f(x)$ for $x^* \in X^N$ and $x \in X$. (See Sherali and Soyster[27].)

As a practical matter, the sequence is modified by tolerances that allow a small amount of sub-optimality, which often results in X^k containing more than one point — that is, near-optimal solutions. Given $\tau = (\tau_1, \ldots, \tau_N) \geq 0$,

$$\begin{array}{ll} z_1 = \max\{f_1(x) : x \in X\} & X^1 = \{x \in X \quad : f_1(x) \geq z_1 - \tau_1\} \\ z_2 = \max\{f_2(x) : x \in X^1\} & X^2 = \{x \in X^1 \quad : f_2(x) \geq z_2 - \tau_2\} \\ \vdots & \vdots \\ z_N = \max\{f_N(x) : x \in X^{N-1}\} & X^N = \{x \in X^{N-1} \quad : f_N(x) \geq z_N - \tau_N\} \end{array}$$

Related to $N = 2$, we have the *bilevel* mathematical program:

$$\max\ f_2(x, y^*) : x \in X,\ y^* \in \operatorname{argmax}\{f_1(x, y) : y \in Y(x)\}.$$

This also represents the ordered preference $f_1 \succ f_2$, but the inner optimality constraint is a restriction that y^* be optimal in the priority objective, whereas the lexico-optimum second problem would include x as:

$$\max\ f_2(x^*, y^*) : (x^*, y^*) \in \operatorname{argmax}\{f_1(x, y) : x \in X,\ y \in Y(x)\}.$$

See MOP Myth 16 to avoid thinking the bilevel solution is the Pareto-optimum:

$$\text{Pareto-max} \begin{Bmatrix} f_1(x, y) \\ f_2(x, y) \end{Bmatrix} : (x, y) \in \mathcal{X},$$

where $\mathcal{X} = \{(x, y) : x \in X,\ y \in Y(x)\}$. Also, see Fliege and Vicente[11] for a recent analysis of how bilevel optima relate to Pareto-optima.

MOP Myth 14. *Every Pareto-optimum is a solution to the lexico-optimum of some lexico-ordering.*

Counterexample. Consider

$$\text{Pareto-max} \begin{pmatrix} x \\ y \end{pmatrix} : x, y \geq 0,\ 2x + y \leq 2,\ x + 2y \leq 2.$$

Using the given order, we first maximize x and obtain $X^1 = \{(1, 0)\}$. Since this is unique, the generated Pareto-optimum is $(1, 0)$. Reversing the order, we maximize y and obtain $X^1 = \{(0, 1)\}$. Again, since this is unique, the generated Pareto-optimum is $(0, 1)$. Thus, the generated solutions are two points, but the Pareto frontier contains the point $(\,{}^2\!/_3\,,\,{}^2\!/_3\,)$ (generated by weights $w = (\,{}^1\!/_2\,,\,{}^1\!/_2\,)$).

MOP Myth 15. *If the inner optimization of a bilevel program has an optimal solution for each x, and the outer maximization objective is continuous and bounded, the bilevel program has an optimal solution.*

Counterexample. Bard and Falk[2] provide the following:

$$\min\ x_1(2y_1^* + 3y_2^*) + x_2(4y_1^* + y_2^*) :\ x \geq 0,\ x_1 + x_2 = 1$$
$$y^* \in \operatorname{argmax}\big\{(x_1 + 3x_2)y_1 + (4x_1 + 2x_2)y_2 :\ y \geq 0,\ y_1 + y_2 = 1\big\}.$$

Solving the inner maximization for $x = (v, 1 - v) \geq 0$,

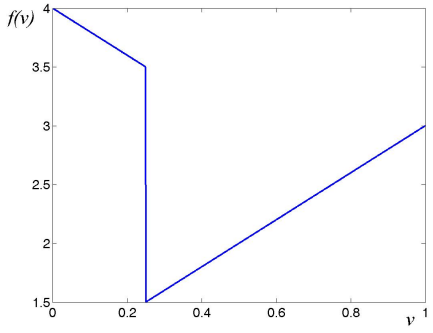
$$y^*(v) = \begin{cases} (1,0) & \text{if } v < \frac{1}{4} \\ \{(\zeta, 1-\zeta) :\ 0 \leq \zeta \leq 1\} & \text{if } v = \frac{1}{4} \\ (0,1) & \text{if } v > \frac{1}{4} \end{cases}$$

Thus, the bilevel program reduces to:

$$\min\ f(v) :\ 0 \leq v \leq 1,$$

where

$$f(v) = \begin{cases} 4 - 2v & \text{if } v < \frac{1}{4} \\ [1.5, 3.5] & \text{if } v = \frac{1}{4} \\ 1 + 2v & \text{if } v > \frac{1}{4} \end{cases}$$



The mapping, f, is not a function; it is vertical at $v = \frac{1}{4}$. If the inner optimization is viewed adversarial (and x, y represent mixed strategy solutions in a game), the value of y^* can be any member of $\{(\zeta, 1-\zeta) :\ 0 \leq \zeta \leq 1\}$, in which case $f(\frac{1}{4}) = 3.5$, preventing achievement of the minimum value of 1.5. As a practical matter, Bard and Falk suggest that x could be chosen with some perturbation: $x = \big(\frac{1}{4} + \varepsilon, \frac{3}{4} - \varepsilon\big)$, and let $\varepsilon \downarrow 0$. Then, the objective value approaches the minimum value (with $y^* = (0,1)$), but it cannot attain this value for any particular x.

The real lesson is to be careful in the interpretation of the two levels. Alternative optima to the inner optimization poses a problem for the game-theoretic interpretation. However, taken literally, y^* is a variable in the outer optimization, constrained by the inner maximization. With that interpretation, the optimal solution is $x^* = (\frac{1}{4}, \frac{3}{4})$, $y^* = (0,1)$.

Bard and Falk offer more insights into bilevel programming and its pitfalls.

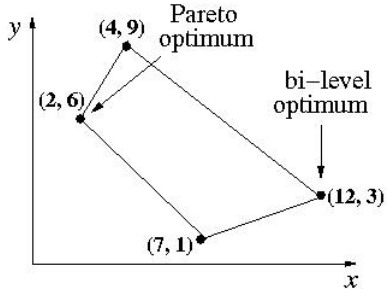
MOP Myth 16. *A solution to the bilevel mathematical program can be found with some weighted objective solution.*

The myth says there exists $w \in [0,1]$ such that a solution to the bilevel mathematical program is found by solving:

$$\max\ wf_1(x,y) + (1-w)f_2(x,y) :\ x \in X,\ y \in Y(x).$$

Counterexample. Haurie, Savard, and White[13] provide the following:

$$\begin{aligned} \max\ & -x - 5y^* :\ x \geq 0, \\ y^* \in Y(x) = \operatorname{argmax}\{ & y :\ y \geq 0 \\ x & + y \geq 8 \\ -3x & + 2y \leq 6 \\ 3x & + 4y \leq 48 \\ 2x & - 5y \leq 9\}. \end{aligned}$$



The optimal bilevel solution is at $(x^*, y^*) = (12, 3)$, but a weighted-objective solution is given by other extreme points of the polyhedron. Here are the weighted-objective solutions for ranges of w:

w-range	optimal extreme point
$0 \leq w \leq 0.15$	$(2,6)$
$0.15 \leq w \leq 0.20$	$(4,9)$
$0.20 \leq w \leq 1$	$(7,1)$

The only extreme point with a greater value of the first-level objective is $(7,1)$, but $1 \notin Y(7)$, so it is not feasible in the bilevel model. The bilevel solution $(12,3)$ is not a solution to any of the weighted-objective models. If it were, it would be Pareto-optimal; Haurie et al. point out that bi-criteria solutions are generally not Pareto-optimal.

Also see the counterexamples by Candler[5] , Wen and Hsu[31]. Further, the counterexample by Ben-Ayed and Blair[3] is for the Grid Search Algorithm, which rests on this myth. See Marcotte[21] for a counterexample to a solution for the equilibrium network design problem based on the same myth.

MOP Myth 17. *An optimal solution to the linear bilevel program is either Pareto-optimal, or it solves the outer LP.*

Counterexample. Wen and Hsu[31] provide the following:

$$\begin{aligned} \max\ & f_2(x,y) = -2x + 11y :\ x \geq 0, \\ y^* \in \operatorname{argmax}\{ & f_1(y) = -3y :\ y \geq 0, \\ x & - 2y \leq 4 \\ 2x & - y \leq 24 \\ 3x & + 4y \leq 96 \\ x & + 7y \leq 126 \\ -4x & + 5y \leq 65 \\ x & + 4y \geq 8\}. \end{aligned}$$

The bilevel solution is at $(x^*, y^*) = \frac{1}{11}(192, 120)$. It is not Pareto-optimal because it is dominated by $(x, y) = \frac{1}{11}(19, 108)$ with the objective values:

$$f(x^*, y^*) = (-32.727, 85.091)^\intercal < f(x, y) = (-29.45, 104.55)^\intercal.$$

The myth asserts that (x^*, y^*) must solve the "outer LP:"

$$\begin{array}{rcrcr} \max & -2x & + 11y : & x, y \geq 0 & \\ x & - & 2y & \leq & 4 \\ 2x & - & y & \leq & 24 \\ 3x & + & 4y & \leq & 96 \\ x & + & 7y & \leq & 126 \\ -4x & + & 5y & \leq & 65 \\ x & + & 4y & \geq & 8\}. \end{array}$$

The optimal solution is $(x, y) = (5.333, 0.667)$ with $f_2(x, y) = -8.667 > f_2(x^*, y^*) = -32.727$.

MOP Myth 18. *A linear bilevel optimum is Pareto-optimal if the coefficient vectors of the inner variable forms an acute angle.*

The linear bilevel program is:

$$\max\ cx + dy^* :\ x \geq 0,\ Ax \leq b,\ y^* \in \operatorname{argmax}\{fy :\ y \geq 0,\ Fx + Gy \leq g\}.$$

The myth asserts that if (x^*, y^*) is a bilevel optimum, it is Pareto-optimal if $df^\intercal > 0$.

The intuition behind this is as follows. From 1983[1]–1988[5, 21] it was believed that the linear bilevel program is Pareto-optimal for:

$$\max \begin{pmatrix} cx + dy \\ fy \end{pmatrix} :\ x, y \geq 0,\ Ax \leq b,\ Fx + Gy \leq g.$$

The rationale is that weights can purportedly be established using the Lagrange (Karush-Kuhn-Tucker) conditions, so that the bilevel program must solve

$$\max\ \lambda(cx + dy) + (1 - \lambda)fy :\ x, y \geq 0,\ Ax \leq b,\ Fx + Gy \leq g$$

for some $\lambda \in [0, 1]$. Before that myth was dispelled (see MOP Myth 16), Ünlü[30] proceeded to correct this by pointing out that Pareto-optimality is not ensured for $\lambda = 1$, resulting in MOP Myth 17. That case is the outer LP — Wen and Hsu[31] gave a counterexample to show that this is not always true. Applying the Kuhn-Tucker conditions, they proposed that Ünlü's theorem is true if $df \geq 0$.

Counterexample. Marcotte and Savard[22] provide the following:

$$\begin{array}{c} \max\ f_2(x, y) = -x - 2y_1 - 2y_2 :\ x \geq 0,\ x \leq 1, \\ y^* \in \operatorname{argmax}\{f_1(y) = y_1 - 2y_2 :\ y \geq 0, x + y_1 \leq 2\}. \end{array}$$

We have $df^\intercal = (2, 2)(-1, 2)^\intercal > 0$. The bilevel optimum is at $(1, 1, 0)$ This is not Pareto-optimal because it is dominated by $(x, y) = (0.5, 1.1, 0)$ with $f_1(x, y) = -2.7 > f_1(x^*, y^*) = -3$ and $f_2(x, y) = 1.1 > f_2(x^*, y^*) = 1$.

MOP Myth 19. *A solution to the minimax goal programming model, with individual optima as an objective reference point, is Pareto-optimal.*

Let $f_i^* = \max\{f_i(x) : x \in X\}$. The associated minimax goal program is:

$$\min_{x \in X} \max_i \{w_i s_i\} : f(x) + s = f^*,\ s \geq 0,$$

where w is the weight vector.

Counterexample. Ogryczak[25] provides the following:

$$\text{Pareto-max} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} : x \in X = \{x \in \mathbb{R}^3_+ : x_1 + x_2 \leq 2,\ 1.5 \leq x_3 \leq 2\}.$$

The Pareto frontier is $X^* = \{x \in \mathbb{R}^3_+ : x_1 + x_2 = 2,\ x_3 = 2\}$, and $f^* = (2, 2, 2)$. The corresponding minimax goal program with $w_i = 1$ for all i is:

$$\begin{gathered} \min\ z : x, s \geq 0,\ x_1 + x_2 \leq 2,\ 1.5 \leq x_3 \leq 2 \\ z \geq s_i \text{ and } x_i + s_i = 2 \text{ for } i = 1, 2, 3. \end{gathered}$$

The optimality region for this is $X^{\text{GP}} = \{(1, 1, x_3) : 1.5 \leq x_3 \leq 2\}$. The only Pareto-optimum point is $(1, 1, 2)$.

To ensure that the minimax goal program generates a Pareto-optimum, add the weighted-sum[24]:

$$\text{lexico-min} \begin{pmatrix} \max_i \{w_i s_i\} \\ \sum_i w_i s_i \end{pmatrix} : x \in X,\ f(x) + s = f^*.$$

This yields the unique solution, $(1, 1, 2)$, which is Pareto-optimal (and solves the minimax goal program).

The GP optimality region generally contains dominated points (non-Pareto-optimal), such as $\{(1, 1, 2 - \varepsilon)\}$ in the example. Be sure to see Tamiz, Jones, and Romero[29] for a clarification of the conditions under which the myth's statement is valid.

MOP Myth 20. *The Parametric Complementary Pivot algorithm obtains a bilevel solution.*

Counterexample. Ben-Ayed and Blair[3] provide the following:

$$\begin{gathered} \max\ 1.5x + 6y_1^* + y_2^* : 0 \leq x \leq 1 \\ y^* \in \text{argmax}\{y_1 + 5y_2 : y \geq 0,\ x + 3y_1 + y_2 \leq 5,\ 2x + y_1 + 3y_2 \leq 5\}. \end{gathered}$$

The PCP algorithm searches for a solution to the feasibility and complementary slackness conditions:

$$\begin{array}{ccc} x + 3y_1 + y_2 + s_1 = 5 & 0.01y_1 + 3u_1 + u_2 - t_1 = 1 & 1.5x + 6y_1 + y_2 - z = 2 \\ 2x + y_1 + 3y_2 + s_2 = 5 & 0.01y_1 + u_1 + 3u_2 - t_2 = 5 & yt = u(s_1, s_2)^{\intercal} = 0 \\ x + s_3 = 1 & x, y, s, t, u, z \geq 0. & \end{array}$$

The algorithm starts by ignoring the inner maximization (but does satisfy the constraints). That solution is $x = 0$ and $y = (1.667, 0)$, with $s = (0, 3.333, 1)$. The complementary slackness conditions require $t_1 = 0$ and $u = (0.328, 0)$. The middle equation then yields $t_2 = .01667 + 0.328 - 5 < 0$, so an artificial variable, w, is introduced:

$$0.01y_1 + u_1 + 3u_2 - t_2 + w = 5.$$

Entering x, y_2, u_2, s_1, or t_1 decreases w. However, u_2 cannot enter because $s_2 > 0$; similarly, neither s_1 nor t_1 can enter. If we choose y_2 to enter, s_2 leaves. At the next step, we may have u_2 enter (u_1 leaves), then s_1 enters to produce the system:

$$y_1 + 0.147x - 0.059s_2 - 0.176z = 0.059.$$

At this point, the PCP algorithm stops with the conclusion that the system has no solution. However, a solution is: $x = 1$, $y = (0, 1)$, $s = (3, 0, 0)$, $u = (0, 1.663)$, $t = (0.663, 0)$, $z = 0.5$.

Ben-Ayed and Blair prove that the bilevel LP is NP-hard, so no polynomial algorithm can ensure optimality (unless $P = NP$).

MOP Myth 21. *A Kuhn-Tucker point of the weighted-objective bilinear program associated with a linear complementarity problem (LCP) is a solution to the LCP.*

Given an $n \times n$ matrix M and n-vector q, the linear complementarity problem is

$$\min\ x'y : (x, y) \in S \stackrel{\text{def}}{=} \{(x, y) \in \mathbb{R}^{2n}_+ : y = Mx + q\}.$$

A complementary solution is when $x'y = 0$. Kostreva and Wiecek[19] approached this as a multiple-objective program:

$$\text{Pareto-min} \begin{pmatrix} x_1y_1 \\ \vdots \\ x_ny_n \end{pmatrix} : (x, y) \in S.$$

The MOP is equivalent to the LCP when it has a complementary solution.

Now consider the weighted-objective bilinear program:

$$P(w, M, q) : \min \sum_{i=1}^{n} w_i x_i y_i : (x, y) \in S,$$

for $w \geq 0$. For $w \not> 0$, it is easy to generate solutions that are not complementary (that is, letting $x_iy_i > 0$ for $w_i = 0$). The myth asserts that a Kuhn-Tucker point for $w > 0$ is a solution to the LCP.

Counterexample. Mohan, Neogy, and Das[23] provide the following:

$$M = \begin{bmatrix} 1 & -3 & 0 \\ -3 & 5 & 2 \\ 2 & -5 & 0 \end{bmatrix} \text{ and } q = \begin{pmatrix} 0 \\ 1 \\ -3 \end{pmatrix}.$$

The $LCP(M, q)$ solution is $x = (9, 3, 5.5)^{\intercal}$, $y = 0$, which is complementary. For $w_1 = w_2 = w_3 = 1$, a Kuhn-Tucker point of $P(w, M, q)$ is $x = (1.5, 0, 1.8)$ and $y = (1.5, 0.1, 0)$ (with multipliers for the linear constraints: $\lambda = (1.5, 0, 1.8)$.) Since $x_1y_1 > 0$, this is not complementary and hence not a solution to $LCP(M, q)$.

Many algorithms converge to a Kuhn-Tucker point and cannot guarantee it is a global minimum. Since the objective in $P(w, M, q)$ is not convex, it can have multiple Kuhn-Tucker points that are not minima — that is, not complementary.

MOP References

[1] J. F. Bard. An efficient point algorithm for a linear two-stage optimization problem. *Operations Research*, 31(4):670–684, 1983.

[2] J. F. Bard and J. E. Falk. An explicit solution to the multi-level programming problem. *Computers & Operations Research*, 9(1):77–100, 1982.

[3] O. Ben-Ayed and C. E. Blair. Computational difficulties of bilevel linear programming. *Operations Research*, 38(3):56–560, 1990.

[4] H. P. Benson. Finding an initial efficient extreme point for a linear multiple objective program. *Journal of the Operational Research Society*, 32(6):495–498, 1981.

[5] W. Candler. A linear bilevel programming algorithm: A comment. *Computers & Operations Research*, 15(3):297–298, 1988.

[6] J. Das and J. E. Dennis. A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, 14(4):63–69, 1997.

[7] P. H. Dybvig and S. A. Ross. Portfolio efficient sets. *Econometrica*, 50(6):1525–1546, 1982.

[8] J. G. Ecker and N. S. Hegner. On computing an initial efficient extreme point. *Journal of the Operational Research Society*, 29(10):1005–1007, 1978.

[9] J. G. Ecker and I. A. Kouada. Finding efficient points for linear multiple objective programs. *Mathematical Programming*, 8(1):375–377, 1975.

[10] M. Ehrgott and K. Klamroth. Connectedness of efficient solutions in multiple criteria combinatorial optimization. *European Journal of Operational Research*, 97(1):159–166, 1997.

[11] J. Fliege and L. N. Vicente. Multicriteria approach to bilevel optimization. *Journal of Optimization Theory and Applications*, 131(2):209–225, 2006.

[12] J. Gorski, K. Klamroth, and S. Ruzika. Connectedness of efficient solutions in multiple objective combinatorial optimization. Preprint 310, Institute of Applied Mathematics, University of Erlangen-Nuremberg, Erlangen, Germany, 2006.

[13] A. Haurie, G. Savard, and D. J. White. A note on: An efficient point algorithm for a linear two-stage optimization problem. *Operations Research*, 38(3):553–555, 1990.

[14] H. Isermann. The enumeration of the set of all efficient solutions for a linear multiple objective program. *Operational Research Quarterly*, 28(3, Part2):711–725, 1977.

[15] R. L. Keeney. Common mistakes in making value trade-offs. *Operations Research*, 50(6):925–925, 2002.

[16] D. S. Kim, G. M. Lee, B. S. Lee, and S. J. Cho. Counterexample and optimality conditions in differentiable multiobjective programming. *Journal of Optimization Theory and Applications*, 109(1):187–192, 2001.

[17] P. Korhonen and J. Wallenius. A careful look at efficiency in multiple objective linear programming. In A. G. Lockett and G. Islei, editors, *Improving Decision Making in Organisations: Proceedings of the Eighth International Conference on Multiple Criteria Decision Making*, volume 335 of *Lecture Notes in Economics and Mathematical Systems*, pages 334–344, Berlin, FRG, 1988. Springer-Verlag.

[18] S. H. Kornbluth and R. E. Steuer. Multiple objective linear fractional programming. *Management Science*, 27(9):1024–1039, 1981.

[19] M. M. Kostreva and M. M. Wiecek. Linear complementarity problems and multiple objective programming. *Mathematical Programming*, 60(1–3):349–359, 1993.

[20] T. J. Lowe, J.-F Thisse, J. E. Ward, and R. E. Wendell. On efficient solutions to multiple objective mathematical programs. *Management Science*, 30(11):1346–1349, 1984.

[21] P. Marcotte. A note on a bilevel programming algorithm by Leblanc and Boyce. *Transportation Research*, 22B(3):233–237, 1988.

[22] P. Marcotte and G. Savard. A note on the pareto optimality of solutions to the linear bilevel programming problem. *Computers & Operations Research*, 18(4):355–359, 1991.

[23] S. R. Mohan, S. K. Neogy, and A. K. Das. A note on linear complementarity problems and multiple objective programming. *Mathematical Programming, Series A*, 100(2):339–344, 2004.

[24] W. Ogryczak. A goal programming model of the reference point method. *Annals of Operations Research*, 51:33–44, 1994.

[25] W. Ogryczak. Comments on properties of minmax solutions in goal programming. *European Journal of Operational Research*, 132(1):17–21, 2001.

[26] A. Przybylski, X. Gandibleux, and M. Ehrgott. The biobjective integer minimum cost flow problem — incorrectness of Sedeño-Noda and González-Martin's algorithm. *Computers & Operations Research*, 33(5):1459–1463, 2006.

[27] H. D. Sherali and A. L. Soyster. Preemptive and nonpreemptive multi-objective programming: Relationships and counterexamples. *Journal of Optimization Theory and Applications*, 39(2):173–186, 1983.

[28] R. E. Steuer. *Multiple criteria optimization: Theory, computation, and application.* John Wiley & Sons, New York, NY, 1986.

[29] M. Tamiz, D. Jones, and C. Romero. Comments on properties of the minimax solutions in goal programming — a reply. *European Journal of Operational Research*, 131(3):685–686, 2001.

[30] G. Ünlü. A linear bilevel programming algorithm based on bicritera programming. *Computers & Operations Research*, 14(2):173–178, 1987.

[31] U-P. Wen and S-T. Hsu. A note on a linear bilevel programming algorithm based on bicriteria programming. *Computers & Operations Research*, 16(1):79–83, 89.

[32] M. Zeleny. *Multiple criteria decision making.* McGraw-Hill, New York, NY, 1982.

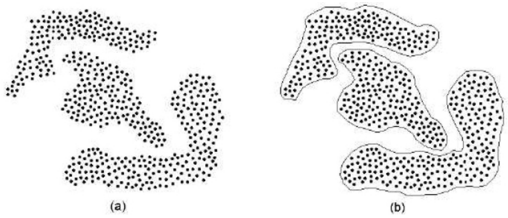
[33] M. Zeleny. The KM-MCDM interface in decision design: tradeoffs-free conflict dissolution. *International Journal of Applied Decision Sciences*, 1(1):3–23, 2008.

Special Forms of Mathematical Programs

This section presents some myths and counterexamples for mathematical programs that do not fit easily into one of the other sections, notably some particular applications.

SF Myth 1. *A good cluster results from maximizing inter-cluster distances and minimizing cluster diameters.*

Counterexample. The following is taken from Climer and Zhang[8].



Using Euclidean distance as the similarity measure for (a), the "intuitive cluster" is (b). The myth fails because many points are closer to a different cluster than their own (so the diameters are not minimized), and the distance between clusters is less than maximal.

SF Myth 2. *A chance-constraint stochastic program has a solution if its certainty equivalent has a solution.*

The model is given by:

$$\max\ \mathbf{E}[f(x;\theta)] : x \in X,\ \mathbf{Pr}[g(x;\theta) \le 0] \ge \alpha,$$

where θ is a vector of uncertain parameters and $\alpha \in (0,1)$. In words, this seeks a policy to maximize the expected value of the objective, subject to it being feasible with probability at least α. The probability and expected value operators are generally taken with respect to θ, and x is a *pure strategy* solution.

Greenberg[15] pointed out that one could allow *mixed-strategy* solutions, in which case the chance constraint could be violated a certain percentage of time. The model becomes:

$$\max_H \int_{x\in X}\int_\theta f(x;\theta)dF(\theta)dH(x) : \mathbf{Pr}[g(x;\theta) \le 0] \ge \alpha,$$

where H is a distribution function on X, subject to choice. The chance constraint is now a joint probability with respect to (x,θ). (Since x is selected before θ is known, the events are independent; hence, we see the product of their distributions.)

Counterexample. Suppose $\int_\theta f(x;\theta)dF(\theta)$ is unbounded over X: $\exists\{x^k\} \subseteq X$ such that $\int_\theta f(x^k;\theta)dF(\theta) \uparrow \infty$. Let x^0 be any *always-feasible solution* — that is, $g(x^0,\theta) \le 0$ for all θ, and choose H such that $\mathbf{Pr}[x = x^0] = \alpha$, thus satisfying the chance constraint regardless of how we assign the remaining probability, $1-\alpha$. Let $\{\phi_k\}_{k\ge1}$ be any series such that $\phi_k \downarrow 0$, $\sum_k \phi_k = 1-\alpha$, and

$$\sum_k \phi_k \int_\theta f(x^k;\theta)dF(\theta) \to \infty.$$

(Could be $\phi_k = \frac{K}{\int_\theta f(x^k;\theta)dF(\theta)}$ for the appropriate constant, $K > 0$.) Then, the objective is unbounded though the chance constraint is satisfied by the randomization of selecting the one feasible solution α portion of the time.

As a specific example, consider

$$\max\ x : x \ge 0,\ P(x \le b) \ge \alpha,$$

where $\mathbf{Pr}[b = i] = (\frac{1}{2})^{i+1}$ for $i = 0, 1, 2, \ldots$ and $0 < \alpha < 1$. The optimal chance-constraint solution is $x^* = 1 - \log_2 \alpha$. However, the problem is unbounded for $\alpha = 0$, so we can randomize between $x = 0$ and any arbitrarily large value of x, say $x = i$, where

$$\mathbf{Pr}[x = 0] = \alpha \text{ and } \mathbf{Pr}[x = i] = 1 - \alpha.$$

The expected value is $(1 - \alpha)i$, which diverges as $i \to \infty$.

A realistic application of this model is with a government prohibition constraint. Suppose a chemical plant must limit the emissions of some toxic chemical, but it is not possible to have zero emissions (except by shutting down the plant). The regulation could be stipulated in the form of a chance constraint, and the plant could choose a randomized strategy to improve its expected value, even though that is not what the government had in mind. (See LP Myth 23.)

Bounding $x \leq 1$, the same example illustrates the result by LaValle[23]:

> *A chance-constraint problem is* Bayesable *if, and only if, no chance constraint is binding.*

LaValle's meaning of "Bayesable" is that the (pure) strategy of the chance-constraint program also maximizes some utility function. (He confined attention to linear forms, but the result clearly generalizes to this possibly nonlinear and integer chance-constraint program.)

Also see Eisner, Kaplan, and Soden[13].

SF Myth 3. *In a chance-constraint stochastic program, the expected value of additional information is non-negative.*

The stochastic model is

$$Z = \min\ cx :\ x \geq 0,\ Ax \geq b,$$

where A, b, c (and hence Z) are random variables. The chance-constraint equivalent is:

$$z = \min\ \mathbf{E}[c]x :\ x \geq 0,\ \mathbf{Pr}[Ax \geq b] \geq \alpha,$$

where $\alpha \in (0, 1)$. The *expected value with perfect information* is given by:

$$EVPI = z - \mathbf{E}[Z].$$

Suppose additional information about the random values can be obtained by an experiment, leading to a conditional chance-constraint model:

$$Z(W) = \min\ \mathbf{E}[c \,|\, W]x :\ x \geq 0,\ \mathbf{Pr}[Ax \geq b \,|\, W] \geq \alpha.$$

This yields the *expected value with sample information*:

$$EVSI = z - \mathbf{E}_W[Z(W)].$$

The value of the sample information is $EVSI - EVPI$, which the myth asserts is not negative.

Counterexample. Blau[3] provides the following:

$$\min\ x :\ x \geq 0,\ x \geq b.$$

This represents a simple inventory model, where x is the level of inventory whose cost is known to be \$1 per unit. The constraint $x \geq b$ is the demand requirement, where b has the following distribution:

$$\mathbf{Pr}[b = 0] = 0.9, \ \mathbf{Pr}[b = 1] = 0.1.$$

Then, $EVPI = 0 - (0 \times 0.9 + 1 \times 0.1) = -0.1$.

Now suppose a marketing expert could perform an experiment (for example, a survey) with an outcome $W \in \{0, 1\}$ such that:

$$\begin{aligned} \mathbf{Pr}[W = 0 \,|\, b = 0] = 0.99 \quad &= 1 - \mathbf{Pr}[W = 1 \,|\, b = 0]; \\ \mathbf{Pr}[W = 0 \,|\, b = 1] = 0.1 \quad &= 1 - \mathbf{Pr}[W = 1 \,|\, b = 1]. \end{aligned}$$

Using Bayes Rule, the posterior distribution of b is given by:

$$\begin{aligned} \mathbf{Pr}[b = 0 \,|\, W = 0] = \frac{(0.99)(0.9)}{0.901} \quad &= 0.9889 \quad = 1 - \mathbf{Pr}[b = 1 \,|\, W = 0]; \\ \mathbf{Pr}[b = 0 \,|\, W = 1] = \frac{(0.99)(0.01)}{0.099} \quad &= 0.0909 \quad = 1 - \mathbf{Pr}[b = 1 \,|\, W = 1]. \end{aligned}$$

For each outcome, the chance-constraint programs are, for $\alpha = 0.9$:

$$\begin{aligned} Z(0) \quad &= \min\{x : \ x \geq 0, \ \mathbf{Pr}[x \geq b \,|\, W = 0] \geq 0.9\} \\ &= \min\{x : \ x \geq 0, \ x \geq 0\} \\ &= 0; \\ Z(1) \quad &= \min\{x : \ x \geq 0, \ \mathbf{Pr}[x \geq b \,|\, W = 1] \geq 0.9\} \\ &= \min\{x : \ x \geq 0, \ x \geq 1\} \\ &= 1. \end{aligned}$$

The marginal distribution of W is

$$\begin{aligned} \mathbf{Pr}[W = 0] &= 0.99 \times 0.9 + 0.1 \times 0.1 = 0.901 \\ \mathbf{Pr}[W = 1] &= 0.01 \times 0.9 + 0.9 \times 0.1 = 0.099. \end{aligned}$$

Hence, $EVSI = z - (0 \times 0.901 + 1 \times 0.099) = -0.099$, so $EVPI < EVSI$. This appears to say that we are better off not performing the experiment to gain more knowledge, even if the cost of the experiment is zero.

Blau attributes this "dilemma" to the structure of the chance-constraint model. Also see Charnes and Cooper[5], Hogan, Morris, and Thompson[18], Jagannathan[19, 20], LaValle[22, 23, 24] and Nau[27].

All that said, there is a more elementary explanation. With the original probability distribution, $x = 1$ is feasible, but the new information changes the probabilities such that this is infeasible. Thus, the value of x^* drops from 1 to 0 in the presence of the new information.

This simpler explanation supports Blau's claim that the chance-constraint must be used with care. Combined with SF Myth 2, the effects of infeasibility, even if only $1 - \alpha$ of the time, must be thought through in the decision process to determine if the chance-constraint model is appropriate. (Just as other stochastic programming models must be considered for their appropriateness. No one model is the right one for all situations!)

SF Myth 4. **updated** *A recourse MILP with interval data uncertainty has its minimax regret solution with scenarios composed of only the endpoints.*

My thanks to Tiravat Assavapokee and Matthew J. Realff for correcting my previous version.

Counterexample. Assavapokee, Realff, and Ammons[2] provide the following. Let θ be uncertain on the interval $[0, 10]$ in the MILP:

$$z^*(\theta) = \max\ f(x|\theta) :\ x \in \{0,\ 1\}^3$$
$$f(x|\theta) = \max\{2y_1 + y_2 + y_3 - x_2 :\ y \geq 0,$$
$$y_1 + y_2 \leq 10x_1,\ y_1 \leq 5x_2,\ y_2 \leq \theta,\ y_1 \leq y_2,\ y_1 + y_3 \leq 5x_3.\}$$

For known value of θ, the MILP optimal value is:

$$z^*(\theta) = \begin{cases} \theta + 5 & \text{for } \theta \in [0, 1] \\ 2\theta + 4 & \text{for } \theta \in [1, 5] \\ 14 & \text{for } \theta \in [5, 9] \\ \theta + 5 & \text{for } \theta \in [9, 10]. \end{cases}$$

With θ unknown at the outset, the maximum regret for x is

$$R(x) = \max_{\theta \in [0,10]} \{z^*(\theta) - f(x|\theta)\}.$$

The myth asserts that we can minimize the maximum regret by restricting θ to its end points:

$$\min_{x \in \{0,\ 1\}^3}\ \max\{z^*(0) - f(x|0),\ z^*(10) - f(x|10)\}.$$

There is clearly no disadvantage to setting $x_1 = x_3 = 1$, so the issue is whether $x_2 = 0$ or $x_2 = 1$. In the endpoint model, we have

$$\begin{aligned} x_2 = 0 \quad &\Rightarrow \quad f(x|0) = 5, \quad f(x|10) = 15 \\ &\Rightarrow \quad \max\{z^*(0) - f(x|0),\ z^*(10) - f(x|10)\} = 0 \\ x_2 = 1 \quad &\Rightarrow \quad f(x|0) = 4, \quad f(x|10) = 14 \\ &\Rightarrow \quad \max\{z^*(0) - f(x|0),\ z^*(10) - f(x|10)\} = 1\ . \end{aligned}$$

Therefore, the endpoint model's optimal solution is $x = (1, 0, 1)$ with its perceived regret value of zero; however, the actual regret value is $R(x) = 4$, with $\theta = 5$ ($z^*(5) - f(x|5) = 14 - 10 = 4$). For the full model, with $\theta \in [0, 10]$, the optimal solution is $x = (1, 1, 1)$ with regret value

$$R(x) = \max_{0 \leq \theta \leq 10} \{z^*(\theta) - f(x|\theta)\} = 1,$$

thus contradicting the myth.

SF Myth 5. *Someone with constant risk aversion always selects the less risky of two investments.*

This can fail for small wealth, as shown by Lippman, McCall, and Winston[25].

Counterexample. Let s be the wealth of an investor. There are two possible investments with r_i = random return for the i^{th} investment. Letting $V(s)$ denote the maximum expected return for a wealth of s, the expected discounted return model is:

$$V(s) = \max_i \{1 - \mathbf{E}[e^{-\lambda(s+r_i)}] + \beta\, \mathbf{E}[V(s+r_i)]\},$$

where $\beta \in (0,1)$ and λ is the constant risk aversion factor — that is, independent of the wealth.

Assume wealth and returns are integer-valued, and that investment continues indefinitely unless the investor becomes bankrupt — that is, $s = 0$, in which case $V(0) = 0$ and the process stops. Further, assume each investment produces either one positive return, $\overline{r}_i$, or a loss of 1 with probabilities:

$$\mathbf{Pr}[r_i = \overline{r}_i] = p_i > 0 \text{ and } \mathbf{Pr}[r_i = -1] = 1 - p_i > 0.$$

Set the parameter values as follows:

$$\lambda = 1,\ \beta = 0.9,\ \overline{r}_1 = 2,\ \overline{r}_2 = 1,\ p_1 = 0.5,\ p_2 = 0.6.$$

Then, $\mu_1 = 1.4268$ and $\mu_2 = 1.3080$, so we have

$$\begin{aligned} V(s) \quad = \max \big\{ & 1 - e^{-s}1.4268 + 0.9\,(0.5V(s+2) + 0.5V(s-1)), \\ & 1 - e^{-s}1.3080 + 0.9\,(0.6V(s+1) + 0.4V(s-1)) \big\} \end{aligned}$$

for $s = 1, 2, \ldots$

Note that $\mathbf{E}[r_1] = 0.5$, $\mathrm{Var}[r_1] = 5.625$, $\mathbf{E}[r_2] = 0.2$, and $\mathrm{Var}[r_2] = 0.544$. Hence, investment 1 has both the greater expected return and the greater risk. Lippman et al. showed that the investor chooses the risky investment when $s = 1$ — that is,

$$-e^{-1}1.4268 + 0.45V(3) > -e^{-1}1.3080 + 0.54V(2).$$

Thus, the optimality of risk aversion depends upon the level of wealth; it is possible for a risk-averse investor to choose a risky investment for a low level of wealth. Lippman et al. note that V is concave in their counterexample, so the counter-intuitive property is not due to any lack of convexity structure. They also prove that the myth remains if we allow no investment as a decision.

SF Myth 6. *There is always a maximum likelihood estimator.*

Wise and Hall[9] provide a counterexample such that the likelihood function is unbounded.

Counterexample. Let the density function be

$$f(x) = \frac{1-\varepsilon}{\sigma}\, h\left(\frac{x-\mu}{\sigma}\right) + \varepsilon\, h(x-\mu),$$

where μ = mean, σ^2 = variance, $h(\cdot)$ is the standard Gaussian density function, and ε is a (fixed) value in $(0, 1)$. We want to estimate (μ, σ^2).

The likelihood function for independent samples $\{x_1, \ldots, x_n\}$ is

$$L(\mu, \sigma^2) = \prod_{i=1}^{n} \left[\frac{1-\varepsilon}{\sigma} h\left(\frac{x_i - \mu}{\sigma} \right) + \varepsilon\, h(x_i - \mu) \right].$$

This is lower-bounded by

$$L(\mu, \sigma^2) = \frac{1-\varepsilon}{\sigma} h\left(\frac{x_1 - \mu}{\sigma} \right) \prod_{i=2}^{n} \varepsilon\, h(x_i - \mu).$$

Consider $\mu = x_1$, so that $h\left(\frac{x_1-\mu}{\sigma}\right) = h(0) > 0$, independent of σ. Let $K = h(0) \prod_{i=2}^{n} \varepsilon\, h(x_i - x_1) > 0$, so the likelihood function is unbounded:

$$\limsup_{\sigma \to 0} L(x_1, \sigma^2) = \limsup_{\sigma \to 0} \frac{1-\varepsilon}{\sigma} K = \infty.$$

Thus, this distribution has no maximum likelihood estimator.

SF Myth 7. *If the demand density function in the newsvendor model is unimodal, so is the profit function.*

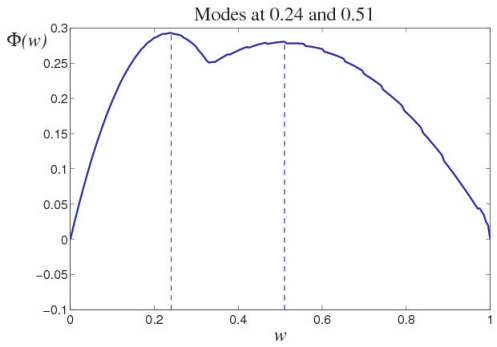
The newsvendor problem is a single-period inventory problem. A manufacturer produces a product at a unit cost c and sells it to a retailer at unit cost $w > c$, who sells it to consumers at unit price $r > w$. Demand is a random variable with density function $f(x)$ and distribution function $F(x)$. The optimal order quantity is $x^*(w) = F^{-1}\big((r-w)/r\big)$. The profit function is $\Phi(w) = (w-c)F^{-1}\big((r-w)/r\big)$.

Counterexample. The following is based on Paul[29]: Let $c = 0$ and $r = 1$ with

$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ 20x & \text{if } 0 \le x \le 0.05 \\ 1.25 - 5x & \text{if } 0.05 \le x \le 0.1 \\ 0.751 - .01x & \text{if } 0.1 \le x \le \frac{29}{40} \\ 7.9285 - 10x & \text{if } \frac{29}{40} \le x \le \frac{31}{40} \\ 0.256 - .01x & \text{if } \frac{31}{40} \le x \le 2.525 \\ 0 & \text{if } x > 2.525 \end{cases}$$

(A slight slope is added to Paul's flat regions to satisfy strict unimodality.)

Then, f is unimodal (with mode at $x = 0.05$). For $0 \le w \le 1$, $\Phi(w) = wF^{-1}(1-w)$, which has two modes, at $w = 0.24$ and $w = 0.51$, thus violating the myth.



Paul also showed that a non-increasing density function can result in a multi-model profit function, and that shifting the demand function may destroy the unimodality of the profit function.

SF Myth 8. *If inventories for substitute products are pooled by a centralized system, the optimal total base-stock cannot increase.*

We have n products with random demands, $D_1, \ldots, D_n$. Optimal inventories for each (ignoring the others) may use a simple model, such as the newsboy problem. Letting F_i denote the cumulative distribution function of D_i, its optimal inventory level is

$$F_i^{-1}(R) = \inf\{d : \, R \leq F_i(d)\},$$

where $R = \dfrac{c_i^s}{c_i^s + c_i^e}$ (called the *newsboy ratio*), c_i^s = unit cost of shortage, and c_i^e = unit cost of excess.

Here we suppose the products could substitute for each other — that is, if there is a shortage in one, some portion of consumers are willing to buy any other. Then, the inventory model pools the inventories, and optimal levels depend upon various assumptions about the substitution. Letting $\mathcal{F}$ denote the c.d.f. of the sum, $D_1 + \cdots + D_n$, the myth asserts that $\mathcal{F}^{-1}(R) \leq \sum_{i=1}^n F_i^{-1}(R)$.

Gerchak and Mossman[14] provide the following:

Counterexample. Let $n = 2$ and let $F_1 = F_2 = F$ be the exponential distribution with common parameter λ. Then, $\mathcal{F}$ is the exponential distribution with parameter 2λ. Thus,

$$F(d) = 1 - e^{-\lambda d} \text{ and } \mathcal{F}(2d) = 1 - e^{-4\lambda d}.$$

Here are some distribution values for $\lambda = 1$:

d	$F(d)$	$\mathcal{F}(d)$
0	0.6321	0.3935
1	0.8647	0.6321
2	0.9502	0.7769
3	0.9807	0.8647

For $c^s = 4c^e$, the newsboy ratio is $R = 0.8$, which implies that the optimal level for each product without any substitution is $F^{-1}(0.8) = 1$, for a total inventory of 2. With full substitution, the optimal (pooled) level is $\mathcal{F}^{-1}(0.8) = 3$. Hence, the pooled inventory is greater than the total of the separate inventories, which violates the myth.

Another counterexample is with the Poisson distribution and $\lambda = 1$. This is perhaps more realistic for a demand distribution, giving some skewness to the right and limiting the demand to integer values. In this case, we have the following distribution values:

d	$F(d)$	$\mathcal{F}(d)$
0	0.3679	0.1353
1	0.7358	0.4060
2	0.9197	0.6767
3	0.9810	0.8571

For $c^s = 2.5c^e$, the newsboy ratio is $R = 0.7143$, which implies that the optimal level for each product without any substitution is $F^{-1}(0.7143) = 1$, for a total inventory of 2. With full substitution, the optimal (pooled) level is $\mathcal{F}^{-1}(0.7143) = 3$. Hence, the pooled inventory is greater than the total of the separate inventories, which violates the myth.

In both cases we choose R to satisfy $\underline{R} < R < \overline{R}$. Equivalently, we have a range on the cost ratio:

$$\frac{\underline{R}}{1-\underline{R}} < \frac{c^s}{c^e} < \frac{\overline{R}}{1-\overline{R}}.$$

For the two cases, these ratios are:

$$\text{Exponential:} \quad 3.482 = \frac{0.7769}{0.2231} < \frac{c^s}{c^e} < \frac{0.8647}{0.1353} = 6.391 \qquad \text{(I chose 4.)}$$

$$\text{Poisson:} \quad 2.093 = \frac{0.6767}{0.3233} < \frac{c^s}{c^e} < \frac{0.7358}{0.2642} = 5.998. \qquad \text{(I chose 2.5.)}$$

Yang and Schrage[38] establish a sufficient condition for the myth to be violated:

Suppose $D_1, \ldots, D_n$ are i.i.d. with common cumulative distribution function, F, and costs, c^s, c^e. Then, $\mathcal{F}^{-1}(R) > nF^{-1}(R)$ if there exists d such that

$$F(d) > \mathcal{F}(nd) \text{ and } F(d) \geq R \geq 0.5.$$

One can verify that the counterexamples satisfy this condition for the indicated range of cost ratios.

Yang and Schrage provide more analysis of the full substitution model, identifying skewness of the probability distribution as the key property that creates a counterexample. They extend the model to include *partial substitution*, which limits the percentages of substitution for each product.

Also see Netessine and Rudi[28], who show that the optimal centralized inventory may exceed the corresponding competitive demand. Their example specifies demands in two categories: (1) demands are first-choice, rarely from substitution; and, (2) demands serve primarily as substitutes. Also, see DP Myth 9 for a dynamic version of this anomaly.

SF Myth 9. *It is not possible to increase both throughput and the fairness of an optimal bandwidth allocation.*

A capacitated network with M links is shared by N sources. The $M \times N$ routing matrix is denoted by R. Letting x_i denote the (controllable) transmission rate of source i, these rates are limited by the constraint $Rx \leq c$, where c_j is the capacity of link j. The allocation model uses the objective function:

$$f(x;\alpha) = \sum_{i=1}^{N} U(x_i, \alpha),$$

where

$$U(v;\alpha) = \begin{cases} \dfrac{v^{1-\alpha}}{1-\alpha} & \text{if } \alpha \neq 1 \\ \log v & \text{if } \alpha = 1. \end{cases}$$

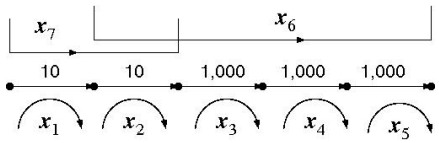
α is a parameter that imposes greater "fairness" as it increases. Here are some special cases:

$$\begin{array}{lll} \alpha = 0 & \Rightarrow f = \sum_i x_i & = \text{throughput} \\ \alpha = 1 & \Rightarrow f = \sum_i \log\, x_i & = \text{proportional fairness} \\ \alpha = 2 & \Rightarrow f = -\sum_i \frac{1}{x_i} & = \text{potential delay} \\ \alpha = \infty & \Rightarrow f \leftrightarrow \min\{x_i\} & = \text{fairness} \end{array}$$

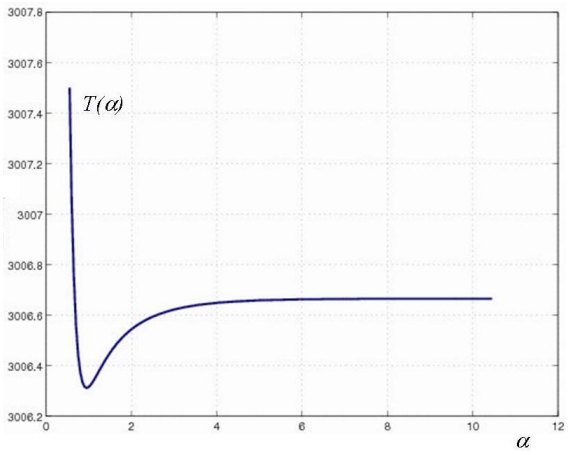
The problem is to maximize $f(x;\alpha)$ subject to $x > 0,\ Rx \leq c$, for a specified value of $\alpha \geq 0$.

Counterexample. Tang, Wang, and Low[36] provide the following example.

There are 5 links and 7 sources, shown on the right. Let $T(\alpha)$ denote the throughput for an α-optimal solution. The first two links have capacity 10 and the last three have capacity 1,000.



The myth asserts T decreases as α increases. However, Tang, Wang, and Low demonstrate that this is not true, as shown on the right.



SF Myth 10. *If a team has the greatest win percentage before and after some specified date, it also has the greatest overall win percentage.*

This is an instance of *Simpson's Paradox*, for which there is a vast literature. Cochran[9] used the baseball players' strike of 1981 as the demarcating date, and he used Simpson's Paradox to teach some elements of integer programming modeling, particularly the formation of objective functions.

Counterexample. Consider the following win-loss records:

	Pre-Strike				Post-Strike				Total			
Team	w	ℓ	$w+\ell$	$\frac{w}{w+\ell}$	w	ℓ	$w+\ell$	$\frac{w}{w+\ell}$	w	ℓ	$w+\ell$	$\frac{w}{w+\ell}$
A	17	18	35	**0.4857**	18	15	33	**0.5455**	35	33	68	0.5147
B	15	16	31	0.4839	19	16	35	0.5429	34	32	66	**0.5152**

(Teams A and B play other teams too.)

Team A stands above Team B in both the pre-strike and post-strike games, but Team B stands above Team A overall.

See the Wikipedia entry at http://en.wikipedia.org/wiki/Simpson's_paradox for more examples and further explanation.

SF Myth 11. *In revenue management, it is always better to re-solve dynamic allocations than use the planned allocations.*

The key to this myth is the definitions of *planned allocation* and *reallocation*. Cooper[10] provides the following:

Counterexample. Consider two fare classes in a flight with one leg and two seats available. Class 1 pays \$1,000 and class 2 pays \$200. The LP to plan allocations during the planning horizon, $[0, T]$ time periods is:

$$\max\ 10x_1 + 2x_2 :\ x_1 + x_2 \le 2,\ 0 \le x \le T.$$

Assume demands, $D_i(t)$, for classes $i = 1, 2$, are independent Poisson processes, each with rate 1. For $T = 2$, an optimal solution to the above LP is $x^* = (2, 0)$. This is the planned solution: reserve 2 seats for class 1 and none for class 2.

The issue is whether to reallocate after one period, having had demand $d = (d_1, d_2)$:

$$\max\ 10y_1 + 2y_2 :\ y_1 + y_2 \le 2 - \big(\min\{d_1, x_1^*\} + \min\{d_2, x_2^*\}\big),\ 0 \le y \le \mathbf{E}[D(2)],$$

where $\mathbf{E}[D(2)]$ is the expected demand in period 2 for each class.

Here are optimal re-allocations for each possible value of d_1 (the value of d_2 is irrelevant since $x_2^* = 0$):

d_1	y_1^*	y_2^*	$\mathbf{Pr}(D_1(1) = d_1)$
0	1	1	0.3679
1	1	0	0.3679
≥ 2	0	0	0.2642

The expected revenue for this reallocation policy satisfies:

$$\begin{aligned}
&\mathbf{E}[10\ \min\{D_2(1), 1\} + 2\ \min\{D_2(2) - D_1(1), 1\} \mid D_1(1) = 0] \\
&\qquad = 10\,\mathbf{E}[\min\{X, 1\}] + 2\,\mathbf{E}[\min\{X, 1\}] = 7.59 \\
&< \mathbf{E}[10\ \min\{D_2(1), 2\} \mid D_1(1) = 0] = 10\,\mathbf{E}[\min\{X, 2\}] = 8.96,
\end{aligned}$$

where X is a random variable with Poisson distribution having rate 1.

Therefore, the expected remaining revenue is less by reallocation than by staying with the planned allocations, given the demand in the first period satisfies $D_1(1) = 0$.

SF Myth 12. *Among no-memory rules to order list items, the move-to-front rule minimizes the average cost.*

The cost to access an item is its position in the list. For example, if the order does not change and P_i is the probability that item i is requested, the average cost for accessing n items is

$\sum_{i=1}^{n} i\,P_i$. We assume that the probabilities are not known a priori. A *no-memory rule* is one that does not use any information about the history of requests. (This includes relative frequencies, so their probabilities cannot be estimated.) Rivest[30] introduced the *move-to-front rule*: replace the requested item with the one at the front of the line. For example, if the items are in their natural order, $1, 2, \ldots, n$, and there is a request for item m, the new order is $m, 1, 2 \ldots, m-1, m+1, \ldots, n$. The myth asserts that this has the least average cost among all no-memory rules. Rivest conjectured the myth using examples for some intuition.

Counterexample. Anderson, Nash, and Weber[1] provide the following. The request probabilities for six items are $P = (0.851, 0.146, 0.001, 0.001, 0.001)$. Consider the no-memory rule defined by six permutations, where Π_{ij} is the position of the j^{th} item after receiving a request for the i^{th} item.

$$\Pi = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 4 & 5 & 6 \\ 1 & 2 & 4 & 3 & 5 & 6 \\ 1 & 2 & 3 & 5 & 4 & 6 \\ 3 & 4 & 1 & 2 & 6 & 5 \end{bmatrix}.$$

For example, if the request is for item 1 or 2, the order does not change. If the request is for item 3, the new order is 231456.

The average cost for the move-to-front rule is approximately 1.26, whereas the average cost of this transposition rule is approximately 1.22. (Both calculations are done by forming the associated Markov chain — see Rivest[30].) See Sleator and Tarjan[34] for insight as to why the move-to-front rule is "approximately optimal" in practice.

SF Background — Data Envelopment Analysis

We are given features of each *Decision-Making Unit* (DMU), which we partition into inputs, denoted I, and outputs, denoted O. The *production possibility set*, P, is the convex hull of $\{(I_k, O_k)\}$, and we wish to evaluate the k^{th} DMU using the features of the others. There are several ways to approach this.

Assume, for our limited purposes here, that (I_k, O_k) is in the interior of the convex hull. Let c denote the cost vector associated with the DMUs, and our goal is to compare c_k with the other DMUs. To do so, we solve the *cost-comparison* LP:

$$\min \sum_{i \neq k} c_i x_i : \; x \in F_k(I_k, O_k), \tag{SF.25}$$

where

$$F_k(I_k, O_k) = \{x : \; x \geq 0, \; \textstyle\sum_{i \neq k} x_i = 1, \; \sum_{i \neq k} I_i x_i \leq I_k, \; \sum_{i \neq k} O_i x_i \geq O_k\}. \tag{SF.26}$$

In words, we find a point in the production possibility set such that each of its inputs does not exceed the input for the k^{th} DMU, and each of its outputs is at least as great as that of the k^{th} DMU. We find the least costly point and compare that with c_k to evaluate how well the k^{th} DMU performs.

Two other LPs are used to evaluate a DMU:

$$\text{Input-oriented: } \min\ \theta : x \in C_k(\theta I_k, O_k) \tag{SF.27}$$

$$\text{Output-oriented: } \max\ \theta : x \in C_k(I_k, \theta O_k), \tag{SF.28}$$

where C_k is the set of conical combinations of DMU features, except the k^{th} DMU:

$$C_k(I_k, O_k) = \{x : x \geq 0,\ \textstyle\sum_{i \neq k} I_i x_i \leq I_k,\ \sum_{i \neq k} O_i x_i \geq O_k\}.$$

The Input-oriented LP asks for the minimum proportionate input for which the output could be satisfied. The Output-oriented LP asks for the maximum proportionate output within the input limit. (See Cooper, Gu, and Li[11] for alternative DEA models.)

Consider the Input-oriented model with $O_k > 0$ (so $x = 0$ is not feasible). Let θ^* be the minimum proportionate change with optimal weight set X^* for (SF.27). The *return to scale* exhibited by the k^{th} DMU is classified by the total of the solution weights in X^*:

Constant return to scale (CRS): $\sum_i x_i^* = 1$ for some $x^* \in X^*$;

Decreasing return to scale (DRS): $\sum_i x_i^* > 1$ for all $x^* \in X^*$;

Increasing return to scale (IRS): $\sum_i x_i^* < 1$ for all $x^* \in X^*$.

For example, an increasing return to scale means that the same output requirements can be satisfied with proportionally less input from the k^{th} DMU. The maximum total weight, among the optima that yield θ^*, is given by:

$$\sigma^{-1} = \max_{x \in X^*} \left\{ \textstyle\sum_{i \neq k} x_k \right\}. \tag{SF.29}$$

(See Seiford and Zhu[32] for details and additional models.)

DEA Pitfalls and Protocols

Dyson et al.[12] describe pitfalls to avoid in using DEA. Here are some examples:

Homogeneity. Use clustering, if necessary, to have the DMUs comparable. For example, do not compare a science department with a language department.

Correlation. Reduce features to be as uncorrelated as possible. For example, if staff size is one input, total staff budget is correlated, so they should not be used as though they are two independent inputs.

Feature measurement. The inputs and outputs may be subject to measurement errors, and some may be qualitative. Several approaches have been considered (cited by Dyson et al.).

Linearity. The DEA models assume that we can represent a feature by taking a (non-negative) linear combination of the features of the DMUs.

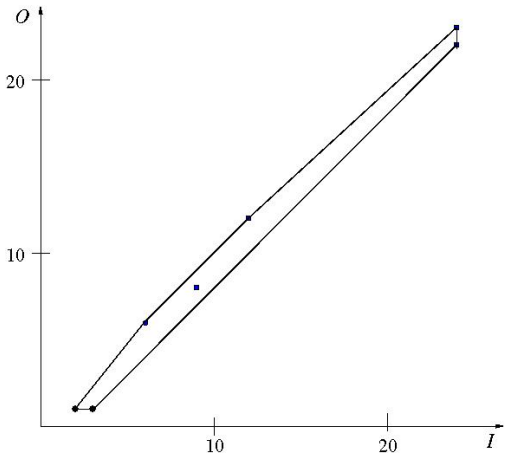
Weight restrictions. There may be restrictions, such as simple bounds, $\underline{x} \leq x \leq \overline{x}$. These may depend upon the DMUs in the database. Removing or adding a DMU could change the weights, the manner of which needs explanation.

SF Myth 13. *A DMU that exhibits increasing return to scale continues to do so up to a proportional increase of all outputs equal to* $\alpha \in [1, \sigma)$.

This is one of the results by Seiford and Zhu[32] (also see [33] for further discussions and placing such errors in context).

Counterexample. Jahanshahloo, Lofti, and Zohrehbandian[21] provide the following:

	DMU						
	1	2	3	4	5	6	7
Input	2	3	6	12	24	24	9
Output	1	1	6	12	22	23	8



Let $k = 2$, so the Input-oriented LP is:

$$\begin{gathered} \min\ \theta : x \geq 0 \\ 2x_1 + 6x_3 + 12x_4 + 24x_5 + 24x_6 + 9x_7 \leq 3\,\theta \\ x_1 + 6x_3 + 12x_4 + 22x_5 + 23x_6 + 8x_7 \geq 1. \end{gathered}$$

An optimal solution is $\theta^* = \frac{1}{3}$ with $x^* = (0, \cdot, 0.0911, 0.0378, 0, 0, 0)$. This means that DMU_2 can produce its output with a combination of inputs from DMU_3 and DMU_4 that use only $\frac{1}{3}$ of the DMU_2 input. That combination sums to less than one, so DMU_2 exhibits an increasing return to scale.

Using (SF.29)[32], $\sigma = 6$, so the myth asserts that the IRS remains in effect if the output is increased to αO for $\alpha \in [1, \sigma)$. However, for $\alpha = 5 < 6$, we get $(I_2, O_2) = (3, 5) \notin P$. Thus, $(3, 5)$ does not exhibit IRS because it is not in the production possibility set.

SF Myth 14. *When using LP for DEA, it does not matter if you obtain a basic or interior optimum.*

Given a solution, x^*, to (SF.25) the associated *peer group* is $\sigma(x^*) = \{i : x_i > 0\}$. If there are alternative optima, different peer groups can be generated, depending upon which solution is obtained. That raises the issue stated in the myth, given by Greenberg[16].

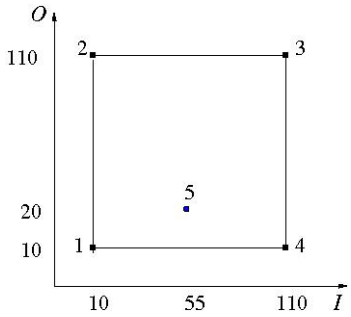
Any interior solution gives the union of all peer groups. Let x^0 denote an interior solution, and let $\{x^\ell\}_{\ell=1}^L$ denote the basic solutions. Then,

$$\sigma(x^0) = \bigcup_{\ell=1}^{L} \sigma(x^\ell).$$

Because of the potential sensitivity of the evaluation process to the choice of peer group, the interior solution better serves the interests of full disclosure. If only one peer group is used in the evaluation, one may question whether another peer group should have been used.

Counterexample.

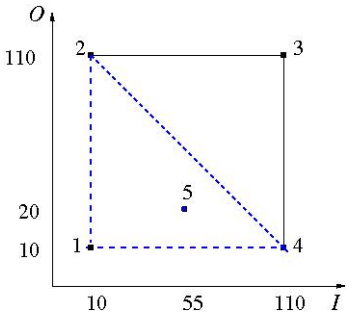
Consider just one input and one output, so the feature space is in the plane. There are five DMUs in the database, and we are evaluating number 5, illustrated on the right.



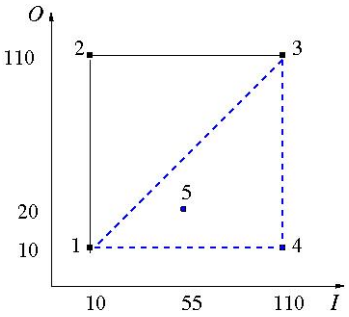
Let the cost vector be $c = (8, 6, 5, 7)$, so the LP is:

$$\begin{array}{rrrrrrrrrr} \min & 8x_1 & + & 6x_2 & + & 5x_3 & + & 7x_4 & : x \geq 0 \\ & x_1 & + & x_2 & + & x_3 & + & x_4 & = & 1 \\ & 10x_1 & + & 10x_2 & + & 110x_3 & + & 110x_4 & \leq & 55 \\ & 10x_1 & + & 110x_2 & + & 110x_3 & + & 10x_4 & \geq & 20 \end{array}$$

There are two optimal basic solutions: $x^1 = (0.45, 0.10, 0, 0.45)$ and $x^2 = (0.55, 0, 0.10, 0.35)$, with associated peer groups $\{1, 2, 4\}$ and $\{1, 3, 4\}$.



Basic Solution x^1



Basic Solution x^2

An interior solution is $x^0 = (0.5, .05, 0.05, 0.4)$ with peer group $\sigma(x^0) = \{1, 2, 3, 4\}$.

One may note, from the example, that the interior solution by itself does not provide all of the useful information. In particular, DMUs 1 and 4 must be in the peer group, whereas the third member could be either DMU 2 or 3 to form a basic feasible solution. Thus, the essential inclusion of DMUs 1 and 4 is lost if only one solution is obtained, regardless of whether it is basic or interior.

The bottom line is that the interior solution is preferred. Its peer group better fits the need for disclosure, and with a modest amount of additional computation, each *essential* member of the peer group can be identified (that is, fix $x_i = 0$ for each $i \in \sigma(x^0)$; i is essential if the LP becomes infeasible).

SF Myth 15. *The input-oriented and output-oriented LPs are infeasible if, and only if, the k^{th} DMU is dominant.*

Chen[7] (Y.) proposed this in her analysis of "super-efficiency" — that is, exclusion of the the k^{th} DMU. Here, we use the exclusion in our basic definitions, (SF.27) and (SF.28). The k^{th} DMU is dominant if it uses no more input and produces no less output than any other DMU — that is,

$$\begin{array}{lll} I_k \leq I_i & \text{and} & O_k \geq O_i \text{ for all } i \neq k, \text{ and} \\ I_k < I_i & \text{or} & O_k > O_i \text{ for some } i \neq k. \end{array}$$

Counterexample. Soleimani-dmaneh, Jahanshahloo, and Foroughi[35] provide the following. In each example, let $k = A$ (first DMU) in (SF.27) and (SF.28).

DMU A does not dominate any other DMU. However, both LPs are infeasible because A has the strict minimum of input 1 and the strict maximum of output 1. Thus, infeasibility does not imply dominance (except for one input and one output).

	DMU				
	A	B	C	D	E
Input 1	1	5	3	3	$^3/_2$
Input 2	1	1	2	4	$^1/_2$
Output 1	10	9	1	1	$^1/_5$
Output 2	3	5	17	1	$^1/_5$
Output 3	2	1	1	2	$^1/_5$

DMU A dominates B, but the output-oriented LP is feasible. Thus, dominance does not imply infeasibility.

	DMU	
	A	B
Input	1	2
Output	2	2

SF Myth 16. *An acceleration in technology improvement results in a more rapid introduction of new technology.*

Cheevaprawatdomrong and Smith[6] present this myth as a paradox in equipment replacement. Their model is as follows. Let C_i denote the undiscounted cost of acquiring, operating, maintaining, and salvaging a machine kept for i periods, for $i = 1, \ldots$ The incremental costs of keeping a machine is

$$c_i = C_i - C_{i-1},$$

where $C_0 \stackrel{\text{def}}{=} 0$. The total discounted cost for i periods is

$$TC_t(\beta) = \sum_{i=1}^{t} \beta^{i-1} c_i,$$

where $\beta \in (0, 1)$. The tacit assumptions are:

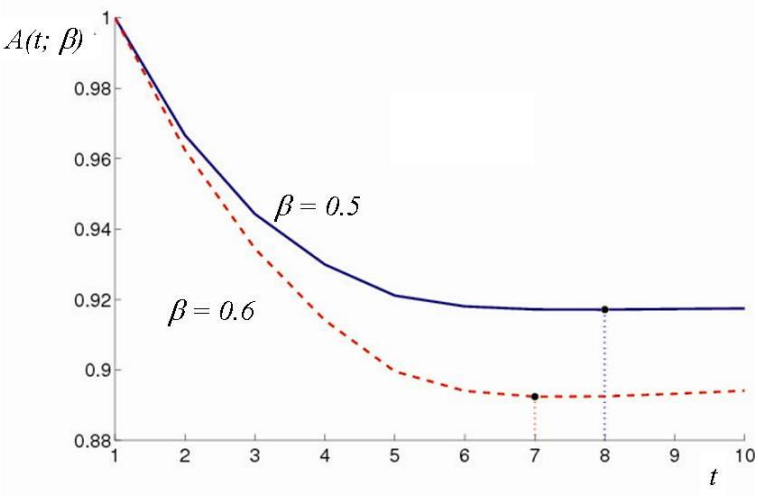
1. Costs are stationary and $c_i \geq 0$ for all i.
2. The effect of technological improvement is to reduce the discount factor.

The decision variable is the cycle time, $t^*(\beta)$, that minimizes annual charge:

$$t^*(\beta) \in \operatorname*{argmin}_{t=1,2,\ldots} A(t;\beta) = \frac{TC_t(\beta)}{1-\beta^t}.$$

The myth asserts $\beta' < \beta \to t^*(\beta') < t^*(\beta)$.

Counterexample. Cheevaprawatdomrong and Smith establish a family of counterexamples that includes the following instance. Let $c_i = 0.9^i$ for $i = 1, \ldots, 5$ and $c_i = c_{i-1}/0.9$ for $i > 5$. Then, $t^*(0.5) = 8 > t^*(0.6) = 7$.



The key to this is that there exists $m \in \mathbb{Z}_+$ such that $c_i \geq c_{i+1}$ for $1 \leq i < m$ and $c_i \leq c_{i+1}$ for $i \geq m$. In the example, $m = 5$.

SF Myth 17. *The least-cost path between two points in the presence of congested regions coincides with segments of the grid obtained by passing horizontal and vertical lines through the vertices of the congested region and existing facility locations.*

My thanks to Avijit Sarkar for providing this myth.

The problem is to find a path that minimizes the total rectilinear distance travelled, with a penalty, called the *congestion factor*, when traveling in a congested region defined by a polytope. Butt and Cavalier[4] proposed a model of the form:

$$\begin{gathered}\min\ |x_s - x_{\text{in}}| + |y_s - y_{\text{in}}| + (1+\alpha)\big(\,|x_{\text{out}} - x_{\text{in}}| + |y_{\text{out}} - y_{\text{in}}|\,\big) + |x_{\text{out}} - x_t| + |y_{\text{out}} - y_t| : \\ \underline{x}_{\text{in}} \leq x_{\text{in}} \leq \overline{x}_{\text{in}}, \quad \underline{x}_{\text{out}} \leq x_{\text{out}} \leq \overline{x}_{\text{out}}, \\ \underline{y}_{\text{in}} \leq y_{\text{in}} \leq \overline{y}_{\text{in}}, \quad \underline{y}_{\text{out}} \leq y_{\text{out}} \leq \overline{y}_{\text{out}} \\ (x_{\text{in}}, y_{\text{in}}) \in \mathcal{G}_{\text{in}},\ (x_{\text{out}}, y_{\text{out}}) \in \mathcal{G}_{\text{out}},\end{gathered}$$

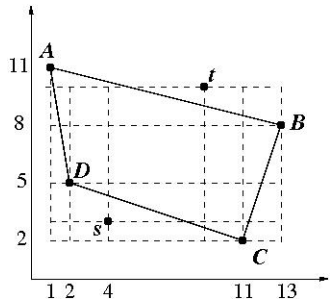
where $(x_{\text{in}}, y_{\text{in}})$ and $(x_{\text{out}}, y_{\text{out}})$ are the coordinates of entrance and exit, respectively, of the congestion region, and $\mathcal{G}_{\text{in}}, \mathcal{G}_{\text{out}}$ are the possible entrances and exits, respectively, restricted to moving on the grid determined by s, t, and the vertices of the congestion region. (The

bounds are redundant if $\mathcal{G}_{\text{in}}, \mathcal{G}_{\text{out}}$ are explicit sets of coordinates; but they could be just grid restrictions.)

The Butt-Cavalier model assumes that there is exactly one entrance and one exit. It is optimal to go around the congestion region for sufficiently large α. Further, the shape of the polygon and locations of s, t could have an optimal path with multiple entrances and exits. Thus, this model is an approximation, where the assumption of exactly one entrance and exit need not hold. Assume it does, as in the following.

Counterexample. Sarkar, Batta, and Nagi[31] define a polytope with four vertices: A, B, C, D. The coordinates of the points are listed in the following table.

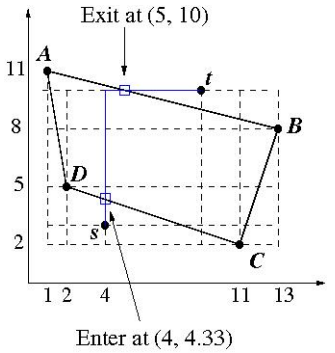
A	(1, 11)
B	(13, 8)
C	(11, 2)
D	(2, 5)
s	(4, 3)
t	(9, 10)



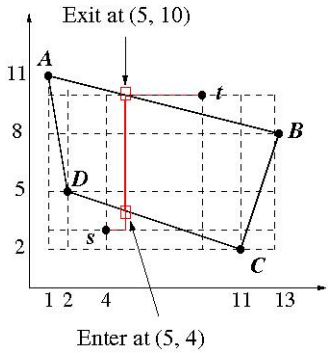
The congestion factor is $\alpha = 0.3$, so the Butt-Cavalier path is determined by choosing entrance and exit points by the mathematical program:

$$\min\ |4 - x_{\text{in}}| + |3 - y_{\text{in}}| + 1.3\,(|x_{\text{out}} - x_{\text{in}}| + |y_{\text{out}} - y_{\text{in}}|) + |x_{\text{out}} - 9| + |y_{\text{out}} - 10| :$$
$$2 \le x_{\text{in}} \le 11,\ 1 \le x_{\text{out}} \le 13,\ 2 \le y_{\text{in}} \le\ 5,\ 8 \le y_{\text{out}} \le 11$$
$$(x_{\text{in}}, y_{\text{in}}) \in \{(2,5),\ (4, 4.333)\},\ (x_{\text{out}}, y_{\text{out}}) \in \{(2, 10.75),\ (5, 10),\ (9, 9.25)\}$$

Every rectilinear path from s to t that always moves either right or up incurs a distance of 12 units (the rectilinear distance from s to t). The additional cost is the congestion factor times the rectilinear distance from the entrance into the congested region to its exit. The Butt-Cavalier path travels 6.67 units in the congestion region, whereas the shortest path (not restricted to the constructed grid) travels only 6 units.



Butt-Cavalier path cost = 14



Shortest path cost = 13.8

SF Myth 18. *The greatest eigenvalue of a real, symmetric matrix, whose diagonal elements are convex functions of a parameter, and whose absolute off-diagonal elements are convex functions of the same parameter, is a convex function of that parameter.*

Let $A(u)$ be an $n \times n$ real, symmetric matrix, where $A_{ii}(u)$ is a convex function of $u \in \mathbb{R}$. Further, assume $A_{ij}(u)$ are non-negative convex functions of u for $i \neq j$. Let $\lambda(u)$ denote the greatest eigenvalue of $A(u)$. Then, Ye[39] proved λ is a convex function of u. He conjectured the myth, which extends this property. Also, it is true for $n = 2$ since

$$\begin{bmatrix} A_{11}(u) & |A_{12}(u)| \\ |A_{21}(u)| & A_{22}(u) \end{bmatrix} \text{ and } \begin{bmatrix} A_{11}(u) & A_{12}(u) \\ A_{21}(u) & A_{22}(u) \end{bmatrix}$$

have the same eigenpolynomials.

Counterexample. Liu and Liu[26] provide the following:

$$A(u) = \begin{bmatrix} 0 & -u & -u \\ -u & 0 & -u^2 \\ -u & -u^2 & 0 \end{bmatrix}.$$

Then,

$$\det(\lambda I - A(u)) = \lambda^3 + 2u^4 - 2\lambda u^2 - \lambda u^4 = (\lambda - u^2)(\lambda^2 + u^2\lambda - 2u^2).$$

The three eigenvalues are thus:

$$u^2,\ \tfrac{1}{2}\left(-u^2 + \sqrt{u^4 + 8u^2}\right),\ \tfrac{1}{2}\left(-u^2 - \sqrt{u^4 + 8u^2}\right),$$

and the greatest of these is given by:

$$\lambda(u) = \begin{cases} u^2 & \text{if } |u| > 1 \\ \frac{1}{2}\left(-u^2 + \sqrt{u^4 + 8u^2}\right) & \text{if } |u| \leq 1 \end{cases}$$

This is not convex over $\mathbb{R}$.

SF Myth 19. *A democratically defined social preference is transitive if individual preferences are transitive.*

This is based on *Arrow's Impossibility Theorem*, a cornerstone of social preference as the sum of individual preferences.

Counterexample. Suppose there are three choices: A, B, C. The population has three groups, X, Y, Z, which differ by their preferences.

Group X has preferences $A \succ B \succ C$, group Y has preferences $C \succ A \succ B$, and group Z has preferences $B \succ C \succ A$.

	Choice		
Group	A	B	C
X	1	2	3
Y	2	3	1
Z	3	1	2

Now suppose the groups are of equal numbers, so a vote of all three choices gives no winner. Each choice receives one group vote.

Consider a vote for each pair, where there is a winner in each case. We see that $A \succ B \succ C$, but $C \succ A$, so the democratically chosen social preference is not transitive.

	Choice		
Group	A vs. B	B vs. C	A vs. C
X	A	B	A
Y	A	C	C
Z	B	B	C
winner	A	B	C

Arrow's Impossibility Theorem has implications in mathematical programming, notably for multiple-objective programming. Also, see Greenberg and Murphy[17] for an application to comparative model assessment.

Special Forms References

[1] E. J. Anderson, P. Nash, and R. R. Weber. A counterexample to a conjecture on optimal list ordering. *Journal of Applied Probability*, 19(3):730–732, 1982.

[2] T. Assavapokee, M. J. Realff, and J. C. Ammons. Min-max regret robust optimization approach on interval data uncertainty. *Journal of Optimization Theory and Applications*, 137(2):297–316, 2008.

[3] R. A. Blau. Stochastic programming and decision analysis: An apparent dilemma. *Management Science*, 21(3):271–276, 1974.

[4] S. E. Butt and T. M. Cavalier. Facility location in the presence of congested regions with the rectilinear distance metric. *Socio-Economic Planning Sciences*, 31(2):103–113, 1997.

[5] A. Charnes and W. W. Cooper. A comment on Blau's dilemma in "stochastic programming" and Bayesian decision analysis. *Management Science*, 22(4):498–500, 1975.

[6] T. Cheevaprawatdomrong and R. L. Smith. A paradox in equipment replacement under technological improvement. *Operations Research Letters*, 31(1):77–82, 2003.

[7] Y. Chen. Measuring super-efficiency in DEA in the presence of infeasibility. *European Journal of Operational Research*, 161(2):545–551, 2005.

[8] S. Climer and W. Zhang. Rearrangement clustering: Pitfalls, remedies, and applications. *Journal of Machine Learning Research*, 7(6):919–943, 2006.

[9] J. J. Cochran. Bowie Kuhn's worst nightmare. *INFORMS Transactions on Education*, 5(1):18–36, 2004. Available at http://archive.ite.journal.informs.org/Vol5No1/Cochran/.

[10] W. L. Cooper. Asymptotic behavior of an allocation policy for revenue management. *Operations Research*, 50(4):720–727, 2002.

[11] W. W. Cooper, B. Gu, and S. Li. Comparisons and evaluations of alternative approaches to the treatment of congestion in DEA. *European Journal of Operational Research*, 132(1):62–74, 2001.

[12] R. G. Dyson, R. Allen, A. S. Camanho, V. V. Podinovski, C. S. Sarrico, and E. A. Shale. Pitfalls and protocols in DEA. *European Journal of Operational Research*, 132(2):245–259, 2001.

[13] M. J. Eisner, R. S. Kaplan, and J. V. Soden. Admissible decision rules for the E-model of chance-constrained programming. *Management Science*, 17(5):337–353, 1971.

[14] Y. Gerchak and D. Mossman. On the effect of demand randomness on inventories and costs. *Operations Research*, 40(4):804–807, 1992.

[15] H. J. Greenberg. On mixed-strategy solutions to chance-constraint mathematical programs. Technical report, Southern Methodist University, Dallas, TX, 1968.

[16] H. J. Greenberg. The use of the optimal partition in a linear programming solution for postoptimal analysis. *Operations Research Letters*, 15:179–185, 1994.

[17] H. J. Greenberg and F. Murphy. Validation and assessment issues of energy models. In S. I. Gass, editor, *Proceedings of a Workshop Held at the National Bureau of Standards*, pages 255–266, Washington, D.C., 1979. National Bureau of Standards.

[18] A. J. Hogan, J. G. Morris, and H. E. Thompson. Decision problems under risk and chance constrained programming: Dilemmas in the transition. *Management Science*, 27(6):698–716, 1981.

[19] R. Jagannathan. Use of sample information in stochastic recourse and chance-constrained programming models. *Management Science*, 31(1):96–108, 1985.

[20] R. Jagannathan. Response to 'on the "Bayesability" of chance-constrained programming problems' by LaValle. *Management Science*, 33(19):1229–1231, 1987.

[21] G. R. Jahanshahloo, F. H. Lofti, and M. Zohrehbandian. Notes on sensitivity and stability of the classifications of returns to scale in data envelopment analysis. *Journal of Productivity Analysis*, 23(3):309–313, 2005.

[22] I. H. Lavalle. On information-augmented chance-constrained programs. *Operations Research Letters*, 4(5):225–230, 1986.

[23] I. H. Lavalle. On the 'Bayesability' of chance-constrained programming problems. *Operations Research Letters*, 4(6):281–283, 1986.

[24] I. H. Lavalle. Response to "Use of Sample Information in Stochastic Recourse and Chance-Constrained Programming Models": On the 'Bayesabillity' of CCP's. *Management Science*, 33(10):1224–1228, 1987.

[25] S. A. Lippman, J. J. McCall, and W. L. Winston. Constant absolute risk aversion, bankruptcy, and wealth-dependent decisions. *Journal of Business*, 53(3):285–296, 1980.

[26] H. W. Liu and S. Y. Liu. Negative answer to the conjecture on the convexity of the greatest eigenvalue. *Journal of Optimization Theory and Applications*, 110(3):677–679, 2001.

[27] R. F. Nau. Blau's dilemma revisited. *Management Science*, 33(10):1232–1237, 1987.

[28] S. Netessine and N. Rudi. Centralized and competitive inventory models with demand substitution. *Operations Research*, 51(2):329–335, 2003.

[29] A. Paul. On the unimodality of the manufacturer's objective function in the newsvendor model. *Operations Research Letters*, 34(1):46–48, 2006.

[30] R. Rivest. On self-organizing sequential search heuristics. *Communications of the ACM*, 19(2):63–67, 1976.

[31] A. Sarkar, R. Butta, and R. Nagi. Commentary on facility location in the presence of congested regions with the rectilinear distance metric. *Socio-Economic Planning Sciences*, 38(4):291–306, 2004.

[32] L. M. Seiford and J. Zhu. Sensitivity and stability of the classifications of returns to scale in data envelopment analysis. *Journal of Productivity Analysis*, 12(1):55–75, 1999.

[33] L. M. Seiford and J. Zhu. Notes on sensitivity and stability of the classifications of returns to scale in data envelopment analysis: A comment. *Journal of Productivity Analysis*, 23:315–316, 2005.

[34] D. D. Sleator and R. E. Tarjan. Amortized efficiency of list update and paging rules. *Communications of the ACM*, 28(2):202–208, 1985.

[35] M. Soleimani-damaneh, G. R. Jahanshahloo, and A. A. Foroughi. A comment on "Measuring super-efficiency in DEA in the presence of infeasibility". *European Journal of Operational Research*, 170(1):323–325, 2006.

[36] A. Tang, J. Wang, and S. H. Low. Counter-intuitive throughput behaviors in networks under end-to-end control. *IEEE/ACM Transactions on Networking*, 14(7):355–368, 2006.

[37] G. L. Wise and E. B. Hall. *Counterexamples in Probability and Real Analysis*. Oxford University Press, New York, NY, 1993.

[38] H. Yang and L. Schrage. Conditions that cause risk pooling to increase inventory. *European Journal of Operational Research*, 192(3):837–851, 2009.

[39] D. Ye. When is the greatest eigenvalue of a parametrized symmetric matrix a convex function of the parameter? *Linear Algebra and Its Applications*, 169:103–109, 1992.

Name Index

Subject Index

If you know of some erroneous result, paradox, fallacy, anomaly, pitfall, or some counter-intuitive result in mathematical programming, please let me know. More generally, I welcome feedback, especially my errors and other opportunities.

Harvey J. Greenberg <hjgreenberg@gmail.com>

This is an ongoing project.