An Annotated Bibliography for Post-solution Analysis in Mixed Integer Programming and Combinatorial Optimization

Harvey J. Greenberg
University of Colorado at Denver
Mathematics Department
PO Box 173364
Denver, CO 80217-3364
e-mail: hgreenbe@carbon.cudenver.edu
http://www-math.cudenver.edu/~hgreenbe

September 5, 1997

DRAFT
Please do not distribute without the author's permission.

Abstract

This annotated bibliography focuses on what has been published since the 1977 Geoffrion-Nauss survey, and it is in BIBTEX format, so it can be searched on the World Wide Web. In addition to postoptimal sensitivity analysis, this survey includes debugging a run, such as when the integer program is unbounded, anomalous or infeasible.

Keywords: integer programming, combinatorial optimization, sensitivity analysis, postoptimal analysis, parametric programming, infeasibility diagnosis, computational economics, computer-assisted analysis.

Contents

1 Introduction

A primary concern of sensitivity analysis is how optimal solution values change when the data changes. There are, however, at least four types of postoptimal sensitivity analyses. The induced change, called an *impulse*, can be either a data object or a decision object. The resulting change that preserves optimality is called a *response*, and it too can be either of these objects. The following table categorizes these four possibilities into primitive forms of postoptimal sensitivity analysis. (More generally, the impulse and/or the response could be some function of the data and/or decision variables.)

	Response	
Impulse	Data	Decision
Data	Drive query	Common query
Decision	Inverse query	Rate of substitution

The *drive query* is concerned with how much one data object needs to change in order to compensate for another data change. For example, one might increase the cost of capital, which eventually makes operating nuclear plants too expensive, in comparison with alternative sources of electricity generation. To keep the optimality of the current levels of nuclear plants, some compensating data object must change. Suppose environmental standards are strengthened to make nuclear generation more economical, such as by requiring more sulfur removal from coal. Then, a drive query is, "If the cost of capital is increased by 2%, how much must the sulfur removal cost increase in order to maintain the same economic disparity between nuclear and coal-fired generation?"

The *inverse query* starts with some decision object, and asks for what data would its value be optimal. For example, one might ask, "For what increase in the cost of capital will some specified level of coal-fired electricity generation become optimal?" (This happens when the cost of coal is equal to that of nuclear generation, so coal can displace the nuclear source.) One of the early methods of infeasibility resolution in linear programming (LP) was with an inverse query approach, using parametric programming to determine a minimal increase in resource levels to meet demands. Along these lines, sensitivity analysis can be an approach to algorithm design. One example is the Generalized Lagrange Multiplier technique (GLM). Enroute to solving the Lagrangian dual, GLM minimizes the Lagrangian for a multiplier, say λ, then posteriori determines a right-hand side vector for which that decision is optimal.

That is, suppose $x^* \in \text{argmin}\{f(x) - \lambda g(x)\colon x \in X\}$, where $\lambda \geq 0$. Then, $x^* \in \text{argmin}\{f(x)\colon g(x) \leq b,\ x \in X\}$ for all b such that $g(x^*) \leq b$ and $\lambda g(x^*) = \lambda b$ (in particular, for $b = g(x^*)$). As an algorithmic strategy, one then changes the Lagrange multipliers, such as by a subgradient of the minimum Lagrangian function.

The remaining non-standard approach is to fix the data but allow some variables to change in response to forced changes in the levels of other variables. We call the relative effect a *rate of substitution*. In LP this is the negative of the tableau entries, using a simplex method, of a compatible basis. That is, we have the linear equations that relate the basic (or dependent) variables to the nonbasic (or independent) variables: $x_B = x_B^* - T x_N$, where $T = \mathrm{B}^{-1}\mathrm{N}$, using the usual partition, $A = [\mathrm{B}\ \mathrm{N}]$, into basic (B) and nonbasic (N) columns. Then, if we force some nonbasic variable, say x_p to change its level from its optimal value of zero, the basic variables change according to the p-th column of the tableau: $\partial^+ x_r / \partial x_p = -T_{rp}$, for each (basic) response variable, assuming the basis is compatible with this perturbation — i.e., $x_r = 0 \Rightarrow T_{rp} \leq 0$ (see Chapter 3 in [44]). This extends to other pairs of variables, giving a rich economic foundation in LP, which has so far not been developed for integer programming.

With attempts to parallel the developments in linear programming, a few results appeared in the 1970's, all pertaining to postoptimal analysis. Since the work prior to 1977 is well surveyed by Geoffrion and Nauss [45], this bibliography begins where that ends. (One exception is [51].) Other surveys on postoptimal sensitivity analysis have been done since 1977: [68] is the most recent one published; [57, 120, 125] have not been published, and an otherwise extensive bibliography on integer programming [124] did not include post-solution analysis as a retrieval category (or even the more narrow topic of postoptimal sensitivity analysis).

Another form of postoptimal analysis is concerned with the *stability* of a solution, where values do not change for a range of parameter values, called the *stability region*. This arises naturally in combinatorial optimization, where a solution is often a subgraph, and the question is for what range of arc weights is this subgraph optimal. Examples include retaining optimality of a Hamiltonian cycle, spanning tree, schedule's job sequence, and shortest path. Although this began before 1977, early works were not published in English, so they are not well known to many in North America. Further, there has been a great deal of attention by a few people for the past several years, and the results have been extended to consider the stability region for a "near optimal" solution. Recent surveys [107, 109] provide details and references to the early literature.

In addition to postoptimal queries, the subject of post-solution analysis includes debugging a scenario, such as when it is anomalous, unbounded or infeasible. (This extends to debugging a model, rather than just one instance.) There is a fairly extensive literature in LP, but the literature is sparse for such matters in integer programming. In principle, one could first see if the LP relaxation is infeasible; if so, LP methods can be used to form a diagnosis. Otherwise, some logical testing [46] could provide a causal substructure. The general approach of finding an *irreducible infeasible subsystem* (IIS) has been extended [54, 55], but it is not as well developed for MIP as it is for LP.

In the next section, basic terms and concepts are presented, along with notation commonly used. More generally, the technical terms used throughout this paper, and in the references cited, are defined in the *Mathematical Programming Glossary* [53]. For perspective, I give a chronology of the citations, and a list by subject in section 3. I also offer some observations about the literature and suggest avenues for research. Finally, the annotated bibliography is given.

2 Terms and Concepts

The mixed integer program has the general form:

$$\text{MIP:}\quad \min\{f(x,y;p)\colon\, g(x,y;p) \geq 0,\, h(x,y;p) = 0,\, x \in X,\, y \in Y\},$$

where p is a parameter vector (could be just 1-dimensional); X is a subset of $\mathbf{Z}^n$, the set of n-dimensional integer vectors (i.e., with integer-valued coordinates); and Y is a subset of $\mathbb{R}^N$, the set of N-dimensional real vectors.

The defining functions, f, g and h, are real-valued with domain in $X \times Y \times P$, where $P \subseteq \mathbb{R}^M$ is the parameter domain. We assume there is at least one parameter ($M > 0$), and there is at least one integer variable ($n > 0$). When $N = 0$, it is a *pure* integer program. When $X \subseteq \{0,1\}^n$, it is a 0-1 program, also called a *binary program*.

Define the *optimal value function*:

$$f^*(p) \stackrel{\text{def}}{=} \inf\{f(x,y;p)\colon\, g(x,y;p) \geq 0,\, h(x,y;p) = 0,\, x \in X,\, y \in Y\}$$

for $p \in P$. (We use inf, rather than min, because it is possible for there to be no optimal solution for some parameter values.) Let $X^*(p)$ denote the optimality region, which is a point-to-set map from P to subsets of $X \times Y$.

Common postoptimal sensitivity analysis is concerned with how a solution changes as a function of p, with much of the focus on properties of f^*. Two types of assumptions are considered:

1. All parameter values are specified in advance. In this case, we can specify the parametric program as follows:

   Find $f^*(p)$ [and some member of $X^*(p)$] for all $p \in P$, where P is the given set of parameter values (typically bounded, and may be finite).

2. Parameter values are specified after the original MIP is solved. Often this is done as a one-parameter variation:

$$z^*(\theta; p, \delta p) \stackrel{\text{def}}{=} f^*(p + \theta \delta p),$$

   where δp is a specified *change direction* and p is the original parameter value.

The "right-hand side case" has the form:

$$\min\{f(x, y)\colon g(x, y) \geq p^1,\ h(x, y) = p^2,\ x \in X,\ y \in Y\},$$

where $p = (p^1, p^2)$. The "objective-coefficient case" has the form:

$$\min\{f(x, y; p)\colon g(x, y) \geq 0,\ h(x, y) = 0,\ x \in X,\ y \in Y\},$$

where p might be decomposed into costs that affect only x and those that affect only y, such as $f_1(x; p^1) + f_2(y; p^2)$.

Marginal analysis pertains to the existence and value of the directional derivative:

$$Df^*(p; \delta p) \stackrel{\text{def}}{=} \lim_{\theta \to 0^+} \frac{z^*(\theta; p, \delta p) - f^*(p)}{\theta}$$

when the limit exists (in the extended reals). Other rates are sometimes considered, which depend upon the behavior of f^* away from the current parameter value, but which have implications for marginal analysis in economics, analogous to LP dual prices.

Range analysis is concerned with the range of θ in the one-parameter model for which some property holds. Classically, the property is the constancy of the functional form of z^*. When z^* is piece-wise linear, as in varying just the right-hand side of an LP, this concern is the constancy of the slope, and the computational problem is the determination of the *breakpoints* — i.e., where the slope changes. This is part of *parametric programming*, which considers variation in z^* (and possibly in X^*), as θ varies over some interval. It is usual to bound the change vectors, in which case they can be scaled such that there is no loss in generality to suppose $\theta \in [0, 1]$.

Adding a variable or a constraint is addressed within this framework. One thinks of the activity or constraint as having been there all along, but with optimal value that makes it equivalent to being absent. In the case of adding a variable, the domain of the given functions are extended in some natural way, such as simply adding a term with the new variable (v): $g'(x, y, v; p) = g(x, y; p) + G(v; p)$. In the case of adding a constraint, one obvious fact is:

> *If an original optimal solution satisfies the new constraint,*
> *it remains optimal in the new problem.*

In the case of adding a variable, a similarly obvious result is obtained through duality.

Some results derive bounds on changes, or subsets of intervals of constancy, using *restriction* or *relaxation*. These are approximating problems from within and without, respectively. Let $F(p)$ denote the feasibility region for a particular parameter, $p \in P$. Then, a restriction is another mathematical program whose feasibility region is always contained in the original one and whose objective function is never better:

$$F'(p) \subseteq F(p) \text{ and } f'(x, y; p) \geq f(x, y; p) \, \forall (x, y) \in F'(p),$$

for all $p \in P$. A relaxation is another mathematical program whose feasibility region always contains the original one and whose objective function is never worse:

$$F'(p) \supseteq F(p) \text{ and } f'(x, y; p) \leq f(x, y; p) \, \forall (x, y) \in F(p),$$

for all $p \in P$.

Some attention has been given to the right-hand side case of the separable form:

$$f^*(b) = \min \left\{ \sum_j f_j(x_j) \colon \ \sum_j g_j(x_j) \leq b, \, x_j \in \{0, \ldots, U_j\} \right\}.$$

One approach that inherently yields parametric analysis of b is dynamic programming (DP). The most straightforward use of DP is with the state equal to the right-hand side and the stage equal to the variable [27]. There are other uses of DP, besides the right-hand side case, such as in [51, 69].

Another special case is the mixed integer linear program,

$$\text{MILP: } \min\{cx + dy \colon \ Ax + By = b, \, (x, y) \geq 0, \, x \in X\},$$

where $X \subseteq \mathbf{Z}^n$. (Inequality constraints can be transformed in the usual manner.) Then,

$$z^*(\theta; r, \delta r) = \inf\{(c + \theta\delta c)x + (d + \theta\delta d)y\colon\ Ax + By = b + \theta\delta b, (x, y) \geq 0,\ x \in X\},$$

where $r \stackrel{\text{def}}{=} (c, d, b)$ and $\delta r \stackrel{\text{def}}{=} (\delta c, \delta d, \delta b)$ (r is sometimes called the *rim data*).

Often we see the right-hand side case with inequalities instead of equations and with $\delta b \geq 0$ or $\delta b \leq 0$. This is called the "monotone" case because the family of feasibility regions is nested. For example, if $\delta b \geq 0$,

$$\theta' > \theta \Rightarrow \{x \in X\colon\ Ax \geq b + \theta'\delta b\} \subseteq \{x \in X\colon\ Ax \geq b + \theta\delta b\}.$$

In general, the univariate optimal value function, z^*, has nice convexity properties in LP, and there are new results for analysis from an interior solution (see relevant chapters in [44] and their references). As summarized in [45], some of the LP properties have some value in sensitivity analysis in MILP.

The pure integer linear program (ILP) has a rich history, stemming from early works before 1977. A definitive body of work is by Blair and Jeroslow [11]-[16], and still continuing (see the surveys in [8, 10]). Unless stated otherwise, the ILP data is assumed to be rational, which is equivalent to assuming they are integer valued. Otherwise, there are pathologies, such as given in [20].

While there are many weak duals for MIP (notably the Lagrangian dual), and strong duals can be defined in (nonlinear) functional spaces, the ILP has a particular strong dual of interest [136]:

<u>Primal</u>	<u>Dual</u>
$\min\{cx\colon\ Ax \geq b,\ x \geq 0,\ x \in \mathbf{Z}^n\}$	$\max\{F(b)\colon\ F(A) \leq c, F \in \Gamma^m\}$,

where Γ^m is the set of subadditive, non-decreasing functions on $\mathbb{R}^m$ such that $F(1) = 0$. We call this the *subadditive dual*. (If the primal is maximization in canonical form, the dual would be minimization, and Γ^m would be the class of superadditive, non-increasing functions. Then, this would be called the *superadditive dual* of the ILP.) A dual solution, F^*, is called a *price function*, and it is simply linear, $F^*(b) = \pi b$, where $\pi \geq 0$, when the LP is structured such that it always has integer extreme points, as in the LP model of the ordinary (i.e., perfect 2-matching) assignment problem or the shortest path problem.

Binary programs arise naturally in connection with logical decision-making, such as whether to do something ($x = 1$) or not ($x = 0$). Here are some examples:

To do, or not to do. Many problems involve a binary decision: invest, or not; open a new plant, or not; shutdown a plant, or not; select a project, or not. In such cases, we assign a binary variable for each decision: $x = 1$ means do it, $x = 0$ means do not do it. Competition among such choices could exist; the range constraint $\alpha \le \sum_j x_j \le \beta$ means choose at least α and at most β affirmative decisions. Another type of competing constraint is a budget, $Fx \le \gamma$, where γ is the total budget, and F_j is the cost of the j-th (affirmative) decision. Alternatively, the cost can enter the objective as a *fixed charge*: Fx, where $F > 0$.

Capacity expansion. Let $\sum_j \kappa_j y_j \le b$ be a capacity limit, where κ_j is the rate of using capacity for the j-th activity, whose level is y_j, and b is the total capacity available. To extend this to allow capacity expansion, let x be a 0-1 variable such that $x = 0$ means no capacity is added, and $x = 1$ means K units of capacity are added. The constraint becomes $\sum_j \kappa_j y_j - Kx \le b$. Then, if some solution has $x = 0$, the original capacity limit applies. If another solution has $x = 1$, y is restricted by the total capacity: $\sum_j \kappa_j y_j \le b + K$. The binary variable can also have a fixed charge, so $x = 1$ only if there are economic incentives, or if it is necessary to be feasible.

Minimum operating level. Suppose y_j is restricted to be either zero or at least $L_j > 0$. An example is if a pipeline is built, it must operate with a specified minimum level of flow. Assume an upper bound, $y_j \le U_j$, which could either be explicitly given, or derived from other constraints. Then, introduce a binary variable, x_j with the bound constraints: $L_j x_j \le y_j \le U_j x_j$. If $x_j = 1$, the bounds are those on y_j when it is desired to have $y_j > 0$ (e.g., the pipeline is built); if $x_j = 0$, the bounds force $y_j = 0$. As usual, there could also be a fixed charge associated with any or all of the binary variables.

Lot-sizing. This is one of the oldest MILPs in operations research, first presented by Wagner and Whitin in 1958. The problem is to minimize cost while satisfying product demands over (discrete) time. Let y_t be the number of units produced in period t, for $t = 1, \ldots, T$ (T is called the *planning horizon*), and let

$$x_t = \begin{cases} 1 & \text{if a setup occurs in period } t \\ 0 & \text{otherwise.} \end{cases}$$

Let the demand from period i to period j, inclusive, be $d_{ij} = \displaystyle\sum_{t=i}^{j} D_t$. Then, a

MILP formulation is:

$$\begin{aligned}
&\min\ cx + dy\colon\ x \in \{0,1\}^n,\ y \geq 0,\\
&\sum_{t=1}^{i} y_t \geq d_{1i} \qquad \text{for } i = 1, \ldots, n-1,\\
&\sum_{t=1}^{n} y_t = d_{1n},\\
&d_{in}x_i - y_i \geq 0 \quad \text{for } i = 1, \ldots, n.
\end{aligned}$$

Scheduling. There are n jobs that must be processed on each of m machines. Let p_{ij} be the processing time for job j on machine i. There are *precedence constraints*, denoted $j \overset{i}{\prec} k$, which means job j must precede job k on machine i. If a pair of jobs has no such relation, even by implication, a *disjunctive constraint* is used to model the fact that exactly one of the two precedence relations must be true in a feasible schedule. Then, one set of fundamental decision variables are the start times, y^0_{ij}, and completion times, y^c_{ij}, related by an assumption of no interruption: $y^c_{ij} = y^0_{ij} + p_{ij}$. The precedence relation, $j \overset{i}{\prec} k$ is represented by the inequality, $y^0_{ik} \geq y^c_{ij}$, but this is only for those precedence relations that are known. Otherwise, the disjunctive constraint, $j \overset{i}{\prec} k$ *or* $k \overset{i}{\prec} j$, is represented by a pair of such inequalities using a binary variable:

$$x_{ijk} = \begin{cases} 1 & \text{if } j \overset{i}{\prec} k \\ 0 & \text{if } k \overset{i}{\prec} j \end{cases}$$

There are various criteria for defining an optimal schedule. One is the *makespan*, $\max_{i,j}\{y^c_{ij}\}$; another is the *mean flow time*, $\sum_{i,j} y^c_{ij}$.

In addition, there are combinatorial optimization problems that can be represented as 0-1 programs. One of the most celebrated is the traveling salesman problem (TSP), and it has more than one representation. The two most common forms are the integer linear program, with subtour elimination constraints included implicitly (there is an exponential number of them), and the quadratic assignment model. Other combinatorial optimization problems included in this bibliography are listed in Table 2.

Another form of sensitivity analysis arises in problems like the TSP, notably in vehicle routing, where locations of points could change. Jones [70, 71, 72] offers an exciting example of finding (and viewing) a region of invariance for each city in a Euclidean TSP. This is an example of *stability analysis*, which is usually concerned with the set of parameters for which an optimal solution remains optimal. More generally, and in Jones' analysis, the *stability region* of a solution, not necessarily optimal, produced by a particular algorithm (or heuristic), is the set of parameter values for which that same algorithm produces the same solution. Let $X^*(p;\mathcal{A})$ be the set of solutions produced by algorithm $\mathcal{A}$ for $p \in P$. Then, the stability region of $x^0 \in X^*(p^0;\mathcal{A})$ is $\{p \in P\colon\ x^0 \in X^*(p;\mathcal{A})\}$. One stability question is: For what range of parameters, like edge weights, does a subgraph, like a tour, remain optimal? The more general question is: For what range of parameters is the same subgraph generated by a particular algorithm?

In many of the combinatorial optimization problems, feasibility does not seem to be of concern (in the context of post-solution analysis). More generally, the recent logic-based approach of inference duality [29, 63] is motivated by the question, "What role does a constraint play in attaining the optimal value?" With a phase I, an answer to this question can be used to determine which constraints can be dropped and still have an infeasible subsystem.

Many of the results, especially the recent works for combinatorial optimization problems, pertain to the *complexity* of the problem, or of an algorithm. (See [53] or any book on combinatorial optimization.) The standard notation is used: $f(n)$ is $O(g(n))$ on $\mathbf{Z}^+$ if there exists a constant, c, and $N \in \mathbf{Z}^+$ such that $f(n) \le cg(n)$ for $n \ge N$. For example, if an algorithm requires $5n^3 + 2n + 10$ fundamental operations on a problem of size n, its time complexity is $O(n^3)$. Another definition also describes the Ω and θ forms:

$$\text{For } \{u_n, v_n\} > 0, \text{ define } s \stackrel{\text{def}}{=} \liminf_{n\to\infty} \frac{u_n}{v_n} \text{ and } S \stackrel{\text{def}}{=} \limsup_{n\to\infty} \frac{u_n}{v_n}.$$

- $S < \infty \Rightarrow u_n = O(v_n)$.
- $s > 0 \Rightarrow u_n = \Omega(v_n)$.
- $u_n = O(v_n)$ and $u_n = \Omega(v_n) \Rightarrow u_n = \theta(v_n)$.

3 Some Observations and Avenues for Research

The literature began in the early 1960's, as linear programming and duality were emerging from their rapid growth the previous decade. That influenced early efforts, and this bibliography begins with a second wave of effort that began in the late 1970's. Table 1 is a chronological list of the citations that follow, with a keyword used to describe something about it.

Table 1: Chronological List of Citations

1977	Branch and bound [89]
	Survey [45]
	Value function [11, 100]
1978	Cutting planes [61]
	Spanning tree [24]
1979	Assignment (generalized)/Facility location [92]
	Cutting planes [75]
	Nonlinear/Complexity [91]
	Scheduling/Knapsack [92]
	Spanning tree [56]
	Survey [92, 131]
	Value function [12, 43, 56]
1980	Cutting planes [3]
	Nonlinear/Branch and bound [90]
	Shortest path [57, 101]
	Stability [49]
	Survey/Matching/Multi-terminal flow [57]
	Uncertainty [81]
	Value function [5, 57]
1981	Matching [129]
	Multicriteria/DP [123]
	Nonlinear/DP [27]
	Shortest path [73]
	Value function/Price [136]
1982	Branch and bound/Cutting planes [95]
	Heuristic/Environment [65]
	Spanning tree [112]
	Value function [13, 65, 99]
1983	Complexity [21]
	Shortest path/Complexity [58]
	Stability [50]
	Value function [59]
1984	Branch and bound [62, 88]
	Communication trees [1]
	Cutting planes [62]
	Nonlinear [6]
	Value function [6, 14]

Table 1: Chronological List of Citations (continued)

1985	Bibliography [124]
	Branch and bound [93, 98]
	Environment model [113]
	Matching [4, 30]
	Spanning tree [77]
	Value function [15]
1986	Complexity [16]
	Value function [16, 26]
1987	Fleet mix [67]
	Heuristic/Knapsack [66]
	Hydropower model [114]
	Nonlinear/Branch and bound [102]
	Unboundedness [20]
1988	DP [69]
	Logical [46, 69]
	Multicriteria [135]
	Price [74]
	Value function [7]
1989	Bibliography [120]
	Lot-sizing [94]
	Scheduling/Stability [104]
	Trees/Sets/Complexity [36]
	Value function [130]
1990	Feasibility [96]
	Scheduling/Lot-sizing/Complexity [125]
	Survey [68, 125]
	TSP [82]
1991	Facility location [79]
	Matroid [84]
	Multicriteria [64]
	Multi-terminal flow [87]
	Nonlinear [23]
	Stability [42, 83, 105]
	TSP [83]
1992	Spanning tree [32]
1993	Lot-sizing/Complexity/DP [121]
	Multicriteria/Environment model [25]
	Stability [106]
	Trees/Sets [35]

Table 1: Chronological List of Citations (continued)

1994	Matching [2]
	Scheduling/Heuristic [76]
	Shortest path [17]
	Trees [17, 37]
	Value function/Sets/TSP [38]
1995	Feasibility [54]
	Knapsack [19]
	Matroid [33, 107]
	Price [28]
	Scheduling/Stability [78, 107]
	Spanning tree [39, 107]
	Survey/TSP/Shortest path [107]
	Value function [8]
1996	Bibliography [52]
	Complexity [122]
	Feasibility [55, 60]
	Knapsack [9, 60]
	Logical [29, 63]
	Scheduling [18]
	Spanning tree [41, 103]
	Stability [18, 70, 86, 85]
	Survey [133]
	Target analysis [47]
	TSP [70, 117]
	Uncertainty [126]
	Value function [127, 128, 132]
1997	Complexity [34, 115, 118]
	Heuristic [97]
	Matroid [31]
	Price [134]
	Scheduling [80, 108, 109, 110]
	Stability [22, 71, 72, 80, 108, 109, 110, 115, 116, 119]
	Survey [10, 44, 109, 115, 134]
	Target analysis [48]
	Trees [31, 40]
	TSP [71, 72, 115, 116, 118]
	Uncertainty [80]
	Value function [111]

Table 2 lists the citations by subject, using the same keywords as in Table 1. The citations for each subject are chronology ordered.

Table 2: Subject List of Citations

	1977-89	1990-97
Assignment (generalized)	[92]	[107]
Bibliography		[52, 120, 124]
Branch and bound	[89, 90, 95, 62, 88, 93]	
	[98, 102]	
Complexity	[91, 21, 58, 16, 36]	[125, 121, 122, 115, 34, 118]
Cutting planes	[61, 75, 3, 95, 62]	
Dynamic programming	[51, 27, 123, 69]	[121]
Environment model	[65, 113]	[25]
Facility location	[92, 130]	[79]
Feasibility		[96, 54, 55, 60]
Fleet mix	[67]	
Heuristic	[65, 66]	[76, 97]
Hydropower model	[114]	
Knapsack	[92, 66]	[19, 9, 60]
Logical	[46, 69]	[29, 63]
Lot-sizing	[94]	[125, 121]
Matching	[57, 129, 4, 30]	[2]
Matroid		[84, 33, 31]
Multicriteria	[123, 135]	[64, 25]
Multi-terminal flow	[57]	[87]
Nonlinear	[91, 90, 27, 6, 102]	[23]
Price	[136, 74]	[28, 134]
Scheduling	[92, 104]	[125, 105, 76, 18, 78]
		[80, 108, 109, 110]
Sets	[36]	[35, 38]
Shortest path/route	[57, 101, 73, 58]	[17]
Stability	[49, 50, 104]	[83, 105, 42, 78, 106, 107]
		[18, 70, 86, 85, 22, 71, 72]
		[115, 116, 119, 80, 108]
		[109, 110]
Survey	[45, 92, 131, 57]	[68, 125, 107, 10, 44]
		[133, 109, 115, 134]
Target analysis		[47, 48]
TSP	[130]	[82, 83, 38, 117, 107, 42, 70]
		[71, 72, 115, 116, 118, 119]
Trees	[24, 56, 57, 112, 1]	[32, 35, 17, 37, 41, 107]
	[77, 36]	[31, 40, 39, 103]
Unboundedness	[20]	
Uncertainty	[81]	[80]
Value function	[100, 11, 12, 43, 57, 5]	[38, 8, 127, 128, 132, 111]
	[136, 13, 65, 132, 99]	
	[59, 6, 14, 15, 16, 26]	
	[7, 130]	

Here is a summary of the scope of MIP post-solution analysis in the literature, with concomitant observations about further research.

- Although there are at least four types of postoptimal sensitivity analyses, the literature is mostly about the most common query: "How does the solution change in response to data changes?" The inverse query is a current area of research in the context of using it to solve a MIP.

- Other scope expansions are the type of MIP and the type of analysis. Most of the literature is about postoptimal solution analysis of MILP. Fewer articles deal with nonlinear MIP, and the only articles I could find that deal with debugging problems are about infeasibility diagnosis. While this is not as advanced as it is in LP, there are important works, old and new, that should be consulted when exploring this avenue of research.

- There are no articles that suggest post-solution analysis could depend upon the particular formulation. It is well known that the effectiveness of a MIP algorithm is very sensitive to the formulation, so it is plausible that computation of response values to impulse changes, as well as the response values themselves, can depend upon the formulation.

- In MILP, focus has been on rim data, with only a few special cases for matrix coefficient perturbation. Further, the right-hand side and cost coefficient cases are generally treated separately, not permitting both to change simultaneously. By contrast, in LP we have results for more general parametric variation, including matrix coefficients, classically from a basic solution and more recently from an interior solution.

- Unlike recent developments in LP, no work has been done based on an interior solution (and an associated algorithm). There are questions that have not been raised before in using an interior point solution method (especially central path following) to solve successive LP subproblems, like relaxations. One family of questions pertains to the sensitivity of the optimal partition of an imputed LP equivalent, such as including generated cuts or (implicitly) all facets of the convex hull of a MILP.

- Recent works (since 1990, with most in the past few years) have focused on stability regions of combinatorial optimization problems. Particular results are for TSP, spanning trees, scheduling, and related problems. These can be extended either by going deeper into the problem classes (e.g., less restrictive assumptions for scheduling problems), or by broadening the problem classes.

- Approaches that determine how best to extend an algorithm to obtain information to be used for postoptimal analysis have been limited to two classical methods: branch and bound and cutting planes. The logic-based approach offers new vistas. Further, what has been done for classical solution methods has not been done for modern meta-heuristics, like genetic algorithms, simulated annealing and tabu search. One exception is the advent of *Integrative Population Analysis*, which uses a form of generalized sensitivity analysis within the context of target analysis to improve algorithm efficiency.

Because of their newness, avenues like the last one listed above are among the most promising for near-term results. The analysis of stability regions is another promising avenue that differs from those conventional approaches that are motivated by LP post-solution analysis. Also, there has been no attention to the stability radius of *partial optimality.* One might ask, "For what range of processing times will some portion of the optimal digraph of a schedule [105] remain optimal? In particular, what is the range for which job j_1 will continue to precede job j_2?"

Wallace [126] points out that care must be taken when using sensitivity analysis, especially stability, to deal with uncertainty. There is no formal theoretical basis for what is often stated in the literature: conducting sensitivity analysis captures the uncertainty of the data, and stable solutions suggest safety in implementing the allegedly optimal policy. Developing such a theory is an area of research and has the potential to build an important bridge between uncertainty and sensitivity analysis.

Commercial codes generally offer no special post-solution analysis methods; an exception is LINDO [97], which uses a method based on [65]. This could be extended further, particularly with an interactive approach to guide heuristics.

In short, this subject has many avenues to explore and a practical need for results. Some avenues are more difficult, such as those that were explored decades ago, but there are new avenues that have not been explored at all.

Acknowledgments

The author gratefully acknowledges help and advice from Hossein Arsham, Charles E. Blair, David Eppstein, David Fernández-Baca, Marek Libura, James Orlin, Linus Schrage, Giora Slutzki, Yuri N. Sotskov, Edo van der Poort, Albert Wagelmans and Frank Werner. I especially want to thank David Woodruff for his patience and help in putting this into Kluwer's style. Last, but not least, I thank our Inter-Library Loan staff, Patricia Connell, Geraldine (Dina) Gold, Connie S. Whitson, and Eveline Yang, without whose help this would not be nearly as complete as it is. Of course, the author remains responsible for whatever errors or omissions might exist. (This bibliography will be maintained on the web, so please send updates to the author.)

Caveat

This bibliography does not include documents I was unable to obtain, particularly some theses, working papers, and monographs that are no longer available.

References

[1] S. Agarwal, A.K. Mittal, and P. Sharma. Constrained optimum communication trees and sensitivity analysis. *SIAM Journal on Computing*, 13(2):315–328, 1984.

> The problem is to find a minimum cost spanning tree in the presence of certain constraints: (i) some cities must be outer nodes, and (ii) some pairs of cities must have a direct connection. (The constraints in (ii) could require the communication network to have a cycle, so there is a tacit assumption that this is not the case.) The cost between a pair of cities (i, j) is the product of the communication requirement, r_{ij}, and the length of the path connecting them. Sensitivity analysis is performed on one communication requirement. There is a finite set of "critical values," $0 < r_{ij}^1 < \ldots < r_{ij}^p$, which are the breakpoints of the optimal value function. Starting at $r_{ij} = 0$, a labeling algorithm based on the earlier work of Gomory and Hu is used to generate the sequence $\{r_{ij}^k\}$, with corresponding optimal trees.

[2] H. Arsham. A solution algorithm with sensitivity analysis for optimal matchings and related problems. *Congressus Numerantium*, 102(2):193–230, 1994.

> This extends [4, 129] to allow simultaneous changes in arc weights. The method is based on LP parametric programming, using tableau entries to obtain ranges. The author emphasizes that the approach deals effectively with both change and chance. (See [126] for some additional thoughts.)

[3] M.G. Bailey and B.E. Gillett. Parametric integer programming analysis: A contraction approach. *Journal of the Operational Research Society*, 31(3):257–262, 1980.

> This solves the pre-defined right-hand side case of a pure ILP with Gomory cutting planes. The "contraction" is the reduction of the feasible region obtained from the cuts.

[4] M.O. Ball and R. Taverna. Sensitivity analysis for the matching problem and its use in solving matching problems with a single side constraint. *Annals of Operations Research*, 4:25–56, 1985/6.

> This considers a special case of linear parameterization of edge weights, where costs of a subset of edges are increased by a constant. The "side constraint" is a limit on the number of edges selected from a given set. GLM is then applied to the side constraint, giving an inherent parametric study of those limits for which the associated LP relaxation has an integer solution (i.e., a matching).

[5] B. Bank. Stability analysis in pure and mixed-integer linear programming. Number 23 in Lecture Notes in Control and Information Science, pages 148–153. Springer-Verlag, Berlin, Germany, 1980.

> Here stability refers to the semi-continuity properties of the feasibility map for the right-hand side and objective coefficient cases, with resulting implications for semi-continuity properties of the optimal value function and the optimality map. This is a very brief introduction without proofs; for a more complete presentation of this approach, see [7]

[6] B. Bank and R. Hansel. Stability of mixed-integer quadratic programming problems. *Mathematical Programming Study*, 21:1–17, 1984.

> This establishes the continuity of the optimal value function and the upper semi-continuity of the optimality region as a point-to-set map under perturbation of the linear part of the quadratic objective. (The "stability" is not to be confused with a region of invariance.)

[7] B. Bank and R. Mandel. *Parametric Integer Optimization.* Mathematische Forschung, Band 39. Akademie-Verlag, Berlin, Germany, 1988. Note: this is

sometimes cited as an article, but it is a monograph with ISBN=3-05-500398-5 and ISSN=0138-3019.

This approaches postoptimal sensitivity questions as semi-continuity properties of the feasibility and optimality regions as point-to-set maps. (These yield semi-continuity properties of the optimal value function, for a continuous objective.) The first few chapters review such properties of mathematical programs in general, and the extensions to allow integer variables begin with Chapter 5, which considers proximity of integer solutions to non-integer ones. Following the authors' notation, let M denote a convex subset of $\mathbf{R}^n$, and $M_s \stackrel{\text{def}}{=} \{x \in M\colon\ x_1, \ldots, x_s \text{ integer}\}$. Then, M_s is defined to be *uniformly distributed* if $\sup_{x \in M}\{d(x, M_s)\} < \infty$, and they prove this is equivalent to: M_s contains an affine basis of M. They also prove M_s is uniformly distributed if

$$M_s = \{x \in \mathbf{R}^n\colon\ x = v + y \text{ for some } v \in V,\ ||y|| < \varepsilon\},$$

where V is a convex cone and $\varepsilon > 0$.

Chapter 6 first considers a quadratic objective function, where only the linear part is perturbed. In particular, they prove f^* is continuous for

$$f^*(p) = \inf\{x^t Q x + px\colon\ Ax \geq b,\ x_1, \ldots, x_s \text{integer}\},$$

where Q need not be positive semi-definite. The second half of the chapter is about convex polynomials, and again only the linear term is perturbed. The authors prove (among other things) that the optimal value is a lower semi-continuous function of the objective coefficients (p).

Chapter 7 considers existence of optimal solutions with concave minimands, and Chapter 8 is about Lipschitz stability. Chapter 10, entitled "On Relations between Parametric Optimization, Solution Concepts and Subadditive Duality for Integer Optimization," uses a penalty function approach, bridging some of the perturbation theory that was developed before the mid-1970's. The more recent advent of the subadditive dual enters the development in the form of response function theory.

[8] C.E. Blair. A closed-form representation of mixed-integer program value functions. *Mathematical Programming*, 71:127–136, 1995.

This introduces *Jeroslow formulas*, which play the same role for MILP as Gomory functions do for ILP. This is the form of the value function of the right-hand side (for rational data). Although the formulas are computable in polynomial time, there is no polynomial-time algorithm to decide whether a given expression is a Jeroslow formula. Drawing primarily from [13], the author summarizes his body of work on characterizing the value function, first for the pure ILP, then for the more difficult case of the MILP. A very brief summary of mathematical details is at the author's web site, http://www.staff.uiuc.edu/~c-blair/mipvsum.html.

[9] C.E. Blair. Sensitivity analysis for knapsack problems: A negative result. Working paper number 96-0131, University of Illinois at Urbana-Champaign, College of Commerce and Business Administration, Office of Research, Champaign, IL, 1996. Note: to appear in *Discrete Applied Mathematics*.

This considers two knapsack problems to be *adjacent* if they have the same data, except that the right-hand sides differ by 1. One might intuitively think that there is some relation between their optimal solutions, such as substantial overlap in the set of positive variables. This report shows this is not the case, and it establishes some theorems in that context.

[10] C.E. Blair. Integer and mixed-integer programming. In Gal and Greenberg [44], chapter 3.

This begins with the author's work on the knapsack problem [9]. He then surveys work on sensitivity for problems solved using the Gomory cutting plane algorithm [136] and implicit enumeration [102]. Next, the author shows how the finite basis theorems for cones and polyhedra can be used to provide unified proofs of early results (viz., [11, 12]). He then surveys results in [13]. These show that the value function of an integer program (for all right-hand sides) is a Gomory function – the maximum of finitely many Chvátal functions. This is analogous to the value function of a linear program being the maximum of finitely many linear functions. The final section, surveying results in [8], introduces *Jeroslow formulas*, which characterize the value function of a MILP, analogous to Gomory functions for an ILP.

[11] C.E. Blair and R.G. Jeroslow. The value function of a mixed integer program: I. *Discrete Mathematics*, 19:121–138, 1977.

This studies the optimal value function for the right-hand side case of a MILP. They show there exist constants, C and D, to bound the difference of optimal solutions by a polyhedral function, $C|\Delta b| + D$. Their main result is that if a MILP is not feasible for all b, we can add additional variables, with specified columns augmented to A, such that the new MILP is feasible for all b and has the same optimal values for the original variables (for those b that are feasible).

[12] C.E. Blair and R.G. Jeroslow. The value function of a mixed integer program: II. *Discrete Mathematics*, 25:7–19, 1979.

This continues the authors' work in [11]. Here a main result is that the difference between the optimal value function of a MILP and its LP relaxation is bounded.

[13] C.E. Blair and R.G. Jeroslow. The value function of an integer program. *Mathematical Programming*, 23:237–273, 1982.

Continuing with the right-hand side case of an ILP, the authors provide a closed-form optimal value function for each feasible right-hand side using Gomory and Chvátal functions. Some of their earlier results then follow as corollaries. They also give some results for the objective-coefficient case, leading to subsequent works [26, 111].

[14] C.E. Blair and R.G. Jeroslow. Constructive characterization of the value function of a mixed-integer program: I. *Discrete Applied Mathematics*, 9:217–233, 1984.

This extends the authors' previous work [13] on pure integer linear programs to the right-hand side case of the more general MILP. A central concept, which is extended, is that of a "consistency tester". These are functions of the right-hand side, that are positive if, and only if, the associated MILP has no feasible solution. A main result is the following:

Every MILP *has a consistency tester that is a Gomory function.*

An example is given to show the converse need not hold, and several characterizations of consistency testers are given. While some of the earlier value function results do not generalize, every value function may be written as the maximum of finitely many Gomory functions.

[15] C.E. Blair and R.G. Jeroslow. Constructive characterization of the value function of a mixed-integer program: II. *Discrete Applied Mathematics*, 10:227–240, 1985.

This is a sequel to [14] in which the right-hand side, b, is replaced by Cv. The aim of the paper is to characterize sets of the form $\{v\colon\ Ax + By = Cv \text{ for some } x, y \geq 0,\ x \in \mathbf{Z}^n\}$.

[16] C.E. Blair and R.G. Jeroslow. Computational complexity of some problems in parametric discrete programming I. *Mathematics of Operations Research*, 11(2):241–260, 1986.

This addresses determining the validity of a statement of the form: "For every right-hand side, the optimal value of two integer linear programs are equal." This is shown to be NP-complete for integer programs, though it has polynomial time complexity for linear programs (given the usual representation). Other complexity results pertain not only to time, but also to space.

[17] H. Booth and J. Westbrook. A linear algorithm for analysis of minimum spanning and shortest-path trees of planar graphs. *Algorithmica*, 11:341–352, 1994.

The algorithm, which is $O(n)$ for n vertices in a given graph, applies to find ranges of edge costs for which a tree remains optimal, with respect to the entitled problems.

[18] H. Bräsel, Y.N. Sotskov, and F. Werner. Stability of a schedule minimizing mean flow time. *Mathematical Computer Modelling*, 24(10):39–53, 1996.

This advances earlier results by considering the objective to be the mean flow time, which is more complicated than the makespan. (See [105] for background and terminology.) They give a general formula for the stability radius, with special attention to conditions under which the radius is 0 or ∞.

[19] R.E. Burkard and U. Pferschy. The inverse-parametric knapsack problem. *European Journal of Operational Research*, 83(2):376–393, 1995.

This considers the one-parameter cost coefficient case of the 0-1 knapsack problem: $\max\{(c + \theta\delta c)x\colon ax \le b,\ x \in \{0,1\}^n\}$, where $c, \delta c, a \ge 0$. The inverse problem is to find a minimum value of $\theta \ge 0$ for which a given solution, x^*, is optimal (or ascertain there is no such θ). One of the authors' results is a complexity bound, $O(n^2 b \log b / log(n \log b))$, which is an improvement over the method in [91]. An empirical study is given, and suggests search methods offer a practical way to get the optimal response values. (Also see [57].)

[20] R.H. Byrd, A.J. Goldman, and M. Heller. Recognizing unbounded integer programs. *Operations Research*, 35(1):140–142, 1987.

The authors give an example to show that a pure ILP can be bounded even though its LP relaxation is unbounded. They proceed to prove this cannot be if the ILP is feasible and has rational data. (You can see the example at http://www-math.cudenver.edu/~hgreenbe/myths.html.)

[21] P.J. Carstensen. Complexity of some parametric integer and network programming problems. *Mathematical Programming*, 26(1):64–75, 1983.

This gives two examples where the number of breakpoints in computing $z^*(\theta; b; \delta b)$ for a pure, 0-1 ILP is $O(2^{\sqrt{n}})$. One significance of this is that a complete parametric analysis of the one-parameter right-hand side case is computationally difficult, at least theoretically.

[22] N. Chakravarti and A.P.M. Wagelmans. Calculation of stability radii for combinatorial optimization problems. Report 9740/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1997.

The entitled problem class is of the form $\min\{f(x;p) : x \in X\}$, where $X \subseteq \{0,1\}^n$, and $f = \sum_j p_j x_j$ or $f = \max_j\{p_j x_j\}$. The authors assume $p \ge 0$, and an approximate solution, x^c, is computed such that $f(x^c; p) \le (1+\varepsilon) f^*(p)$ for given $\varepsilon \ge 0$ (so optimality is the special case of $\varepsilon = 0$). First, a simple upper bound, ρ_u, for the stability radius (ρ) of x^c at p is determined (with the added condition that $p + \delta p \geq 0$) for an arbitrary subset of elements – i.e., where $\delta p_j = 0$ for some j's. (See [105] for terms and concepts.) Their first result is: *The stability radius is the largest $\rho \geq 0$ for which*

$$\min_{x \in X} \left\{ \sum_j \left(p_j(1+\varepsilon) - \rho d_j \right) x_j \right\} \geq \sum_j (p_j + \delta_j) x_j^c,$$

where d is easy to compute from x^c, and $\delta_j = \left\{ \begin{array}{ll} 1 & \text{if } d_j \neq 0 \\ 0 & \text{if } d_j = 0 \end{array} \right\}$. The right-hand side is simply linear in ρ, and the left-hand side is a value function that is piece-wise linear and concave. The authors bound the number of linear pieces on $[0, \rho_u]$ as w^2, where $w = \sum_j \delta_j$ (= number of nonzero d_j). The authors extend their results to the tolerance approach [79], which considers relative, rather than absolute, deviations in a given direction, δp.

[23] M.S. Chern, R.H. Jan, and R.J. Chern. Parametric nonlinear integer programming: The right-hand side case. *European Journal of Operational Research*, 54(2):237–255, 1991.

This considers the multi-dimensional perturbation of right-hand sides for a separable MINLP, with applications to a parametric chance-constrained reliability problem. An empirical study compares a branch-and-bound approach [89] with dynamic programming [27].

[24] F. Chin and D. Houck. Algorithms for updating minimal spanning trees. *Journal of Computer and Systems Sciences*, 16:333–344, 1978.

This presents an $O(n)$ algorithm to update a minimum spanning tree of n vertices when a vertex or edge is added to a graph. It also gives an $O(n^2)$ algorithm to delete any vertex or edge.

[25] A.R. Ciric and S.G. Huchette. Multiobjective optimization approach to sensitivity analysis: Waste treatment costs in discrete process synthesis and optimization problems. *Industrial & Engineering Chemistry Research*, 32(11):2636–2646, 1993.

This considers a nonlinear MIP for the entitled problem, and sensitivity analysis is used here in its connection with trading off multiple criteria. In this case the tradeoff is between maximizing profits and minimizing chemical production wastes. The algorithm is illustrated with an example, and computational results reported suggest the method is practicable, at least for some chemical processes.

[26] W. Cook, A.M.H. Gerards, A. Schrijver, and É. Tardos. Sensitivity theorems in integer linear programming. *Mathematical Programming*, 34(3):251–264, 1986.

This considers variation of c and b in a pure ILP. The main result is a bound on the nearness of an optimal solution to the LP relaxation to an optimal solution to the ILP. A relation this has to sensitivity analysis is that it can be used to prove Blair and Jeroslow's theorem [13] that the optimal value function is a Gomory function.

[27] M.W. Cooper. Postoptimality analysis in non-linear integer programming. *Naval Research Logistics Quarterly*, 28:301–307, 1981.

This addresses the entitled subject with an algorithm that solves the pre-determined right-hand side case. The objective and constraint functions are assumed to be separable, and dynamic programming is used for the parameterization.

[28] A. Crema. Average shadow price in a mixed integer linear programming problem. *European Journal of Operational Research*, 85(3):625–635, 1995.

This extends the results in [74] to MILP. The author provides an algorithm to find the net profit, which he illustrates with a capacitated plant location problem.

[29] M. Dawande and J.N. Hooker. Inference-based sensitivity analysis for mixed integer/linear programming. Technical report, Graduate School of Industrial Administration, Carnegie Mellon University, Pittsburgh, PA, 1996. Note: also available at http://www.gsia.cmu.edu/afs/andrew/gsia/jh38/mipsens.ps.

Building on earlier work [63], this shows how to obtain a resolution proof from the tree of information induced by a branch and cut solution method. A main result is the derivation of linear inequalities on MILP changes, $(\Delta A, \Delta b, \Delta c, \Delta f^*)$. From this, a bound on the objective change, Δf^*, can be inferred from limits on the data changes.

[30] U. Derigs. Postoptimal analysis for matching problems. *Methods of Operations Research*, 49:215–221, 1985.

This modifies the author's shortest augmenting path algorithm to solve a family of problems with varying edge weights. He compares his algorithm to the one in [129].

[31] T.K. Dey. Improved bounds for planar k-sets and related problems. Technical report, Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur, India 721302, 1997.

Given a set P of n points in $\mathbb{R}^d$, a k-set is a subset $P' \subseteq P$ such that $P' = P \cap H$ for a halfspace H and $|P'| = k$ $(0 \leq k \leq n)$. The planar k-set problem ($d = 2$) includes parametric matroids, like the parametric minimum spanning tree problem. This paper proves an $O(n(k+1)^{1/3})$ upper bound for computing planar k-sets, which is a major improvement over the previous bound, $O(n(k+1)^{1/2})$.

[32] B. Dixon, M. Rauch, and R.E. Tarjan. Verification and sensitivity analysis of minimum spanning trees in linear time. *SIAM Journal on Computing*, 21(6):1184–1192, 1992.

This improves upon [77], giving a linear time algorithm to verify the minimality of a spanning tree. This can be used to compute the *radius of stability* [105] with respect to its edge weights.

[33] D. Eppstein. Geometric lower bounds for parametric matroid optimization. Technical report 95-11, Department of Information & Computer Science, University of California at Irvine, Irvine, CA 92697-3425, 1995. Note: to appear in *Discrete & Computational Geometry*.

This gives lower bounds on the number of bases changes for a matroid with linearly varying arc weights, and relates this to problems in computational geometry. The following are proven:

1. There can be $\Omega(nr^{1/3})$ different minimum weight bases in a matroid with n elements and rank r.
2. There can be $\Omega(m\alpha(n))$ different minimum spanning trees with m edges and n nodes, where $\alpha(n)$ denotes the inverse Ackermann function.

[34] D. Eppstein and D.S. Hirschberg. Choosing subsets with maximum weighted average. *Journal of Algorithms*, 24(1):177–193, 1997.

With the principle in [91], the entitled problem is equivalent to parametric selection: given n objects with values decreasing linearly with a parameter, θ (like time), find the θ^* at which the $n-k$ maximum values add to zero. The authors give several algorithms with $O(n)$ time complexity (independent of k). Further, they prove that if the weights are allowed to be negative, the problem is NP-complete.

[35] D. Fernández-Baca and A. Medepalli. Parametric module allocation on partial k-trees. *IEEE Transactions on Computers*, 42:738–742, 1993.

The problem is to allocate modules to processors in a distributed system so as to minimize total cost when the underlying communication graph is a partial k-tree. This is the one-parameter objective case, where arc costs are of the form $c + \theta\delta c$. They give a polynomial-time algorithm to construct the optimal value function with an associated sequence of optimal assignments. They proceed to discuss how to apply their results to parametric versions of the vertex cover, independent set, and 0-1 quadratic programs.

[36] D. Fernández-Baca and G. Slutzki. Solving parametric problems on trees. *Journal of Algorithms*, 10:381–402, 1989.

This is concerned with the complexity of two classes of problems that are NP-hard in general, but have polynomial solution time on trees. The first is the minimum vertex cover (which the authors point out is equivalent to the maximum weighted independent set problem), and the second is a dominating set problem. In each case the graph is presumed to be a tree, and the weights are perturbed as the one-parameter form: $w + \theta\delta w$. The optimal value function is piece-wise linear with a finite number of breakpoints. Since the number of solutions (e.g., the number of vertex covers) is finite, each interval has an optimal vertex cover, and there is a value of θ, denoted t_∞, where the same solution is optimal for all $\theta \geq t_\infty$. This is called the "steady state solution," denoted x^∞. These are the complexity numbers of interest:

1. the number of breakpoints;
2. the time to compute the optimal response (i.e., all breakpoints and associated optima);
3. the time to compute the steady state, t_∞ and x^∞.

The vertex covering problem is a special case of non-serial dynamic programming, as defined by the authors, where each x can be selected from $\{0, \ldots, d-1\}$, where $d \geq 2$. (The vertex covering problem has $d = 2$.) The following table shows the values that are proven in the paper.

	non-serial DP	min dominating set
No. Breakpoints	$O(n^{1+\log_2 d})$	$O(n^{3.322})$
Compute z^*	$O(n^{1+\log_2 d} \log n)$	$O(n^{4.322})$
Compute t_∞	$O(n^{1+\log_2 d})$	$O(n^2)$
Compute x^∞	$O(n^{1+\log_2 d})$	$O(n)$

The improvement to obtain the steady state results without computing the entire response function is due to the procedure in [57].

[37] D. Fernández-Baca and G. Slutzki. Optimal parametric search in graphs of bounded tree-width. In E.M. Schmidt and S. Skyum, editors, *Algorithm Theory – 4th Scandinavian Workshop Proceedings*, volume 824 of *Lecture Notes in Computer Science*, pages 155–166, Berlin, Germany, 1994. Springer-Verlag.

The arc weights are affine functions of one parameter, and the problem is the parametric program: find the breakpoints that define the optimal value function, and an associated optimal solution for each interval. Using the general approach in [91] and tree decomposition, this presents $O(n)$-time approximation algorithms for several NP-hard parametric programs. (Also see [57].)

[38] D. Fernández-Baca and G. Slutzki. Parametric problems on graphs of bounded tree-width. *Journal of Algorithms*, 16:408–430, 1994.

Here the arc weights are polynomial functions of one parameter, which need not be linear, as in related works [37]. The main result is that under mild assumptions, the number of breakpoints of the optimal value function is bounded by a polynomial. They also provide a polynomial-time algorithm to construct the sequence of optimal solutions, and they prove certain parametric problems can be solved in $O(n \log n)$ time, where n is the number of vertices in the graph. The problems to which the results apply include dominating sets, vertex covers, and TSP.

[39] D. Fernández-Baca and G. Slutzki. Linear-time algorithms for parametric minimum spanning tree problems on planar graphs. In R. Baeza-Yates, E. Goles, and P.V. Poblete, editors, *Latin 95: Theoretical Informatics – Second Latin American Proceedings*, volume 911 of *Lecture Notes in Computer Science*, pages 257–271, Berlin, Germany, 1995. Springer-Verlag.

This considers the *minimum ratio spanning tree problem*: Given two edge weights, a and b, find a spanning tree T that minimizes the ratio of their weights, $\frac{\sum_{e \in T} a_e}{\sum_{e \in T} b_e}$. Using the method in [91], this is solved by the parametric spanning tree problem with arc weight $w = a - \theta b$. Their search algorithm is polynomial-time.

[40] D. Fernández-Baca and G. Slutzki. Optimal parametric problems on graphs of bounded tree-width. *Journal of Algorithms*, 22:212–240, 1997.

This includes the work in [37] and considers the additional problem of finding a parameter value that gives an extreme value of the optimal value function [41].

[41] D. Fernández-Baca, G. Slutzki, and D. Eppstein. Using sparsification for parametric minimum spanning tree problems. *Nordic Journal of Computing*, 3:352–366, 1996.

Arc weights have the one-parameter form, $w + \theta\delta w$, and the authors note how this includes the minimum ratio spanning tree problem [39]. They present an algorithm to find the breakpoints that define the optimal value function, z^*, with lower time-complexity than previous results (speedup depends on δw, relative to the graph dimensions). Further, they present an algorithm to find θ^* such that $z^*(\theta^*; w, \delta w) = 0$ (for the monotonic case), and another algorithm to find $\max\{z^*(\theta; w, \delta w)\colon \theta \geq 0\}$. "Sparsification" is a technique that allows computations to be organized efficiently by exploiting the fact that a great portion of the graph is dismissed early. This is progressive in that smaller portions of the graph participate in successive iterations.

[42] D. Fernández-Baca and S. Srinivasan. Constructing the minimization diagram of a two-parameter problem. *Operations Research Letters*, 10:87–93, 1991.

> This considers the general two-parameter objective case: $f^*(\lambda, \mu) \stackrel{\text{def}}{=} \min\{f_1(x)+\lambda f_2(x)+\mu f_3(x):\ x \in X\}$, where X is any finite set. Then, f^* is concave, and X can be subdivided into polytopes such that the same solution remains optimal throughout the relative interior of a polytope. This paper shows how to construct these stability regions.

[43] N.V. Filonenko. Approximate solution of parametric integer programming problems. *Cybernetics*, 15:703–707, 1979. Note: translated from *Kibernetika* 5:91-94, 1979.

> This considers the one-parameter objective coefficient case for the pure ILP. The approximation requires the determination of a change vector, Δx, such that $x+\Delta x$ is feasible, and satisfies a local minimum property of the perturbed objective. The authors assume it is easy to determine such a change or ascertain none exists. From this they provide simple parameter bounds that retain optimality, and they generate breakpoints of z^* by iterative application of their bounding method.

[44] T. Gal and H.J. Greenberg, editors. *Advances in Sensitivity Analysis and Parametric Programming.* Kluwer Academic Press, Boston, MA, 1997.

> This contains 15 chapters on the entitled subject. The chapter by Blair [10] is about integer programming (a table of contents is given at http://www-math.cudenver.edu/~hgreenbe/sapptoc.html).

[45] A.M. Geoffrion and R. Nauss. Parametric and postoptimality analysis in integer linear programming. *Management Science*, 23(5):453–466, 1977.

> This is an excellent survey up to the time of its publication. It is important reading as an introduction to this subject. Its main sections are: • Data Changes that Preserve Optimality; • Families with a Single Parameter in the Objective Function of Right-Hand Side: Drawing Conclusions after Solving but a Few Members; • Using Conventional Branch-and-Bound Wisely to Solve a Family of Related Problems; • Redesigning Branch-and-Bound to Solve a Family of Related Problems.

[46] F. Glover and H.J. Greenberg. Logical testing for rule-base management. *Annals of Operations Research*, 12:199–215, 1988.

This examines precedence constraints for 0-1 variables of the form, IF $x_p = a$ THEN $x_q = c$, where the antecedent value (a) and the consequent value (c) are each specified as 0 or 1. This paper is concerned with feasibility, redundancy, and reduction by identifying forced values and equivalence classes. A main result is showing how the topological sort reduces subsequent time complexities, such as determining forced values. The tests are extended to more general Horn clauses, and shown how they apply to a variety of situations.

[47] F. Glover, J.M. Mulvey, and D.Bai. Improved approaches to optimization via integrative population analysis. Technical report, Business School, University of Colorado, Boulder, CO, 1996.

This contains a "generalized sensitivity analysis" in the context of a new solution method based on the tenets of target analysis. A *grand objective* function is defined on a subset of a population of related problems. This grand objective can depend on the time it takes to solve problems in the set, as well on problem ingredients directly, such as objective functions. A fundamental question is how the grand objective function responds to changes in the population set, such as changing parameter values. Simulation is used to estimate this sensitivity, forming "representatives" that are used within adaptive memory programming. The approach is illustrated with an application to financial planning.

[48] F. Glover, J.M. Mulvey, D.Bai, and M.T. Tapia. Integrative population analysis for better solutions to large-scale mathematical programs. In Gang Yu, editor, *Industrial Applications of Combinatorial Optimization*. Kluwer Academic Press, Boston, MA, 1997.

This is a companion to [47], which includes three additional applications to show how the population of related problems is defined and their similarities exploited with generalized sensitivity analysis. The first application is in flexible manufacturing, where repeated machine scheduling problems arise from new product orders. The second is in marketing, where a population of customers presents varied but similar preferences. The third application is product customization, where a firm seeks to identify new products to a population of potential customers with similar characteristics.

[49] E.N. Gordeev and V.K. Leontev. Stability in bottleneck problems. *Computational Mathematics and Mathematical Physics*, 20(4):275–280, 1981. Note: translated from Russian in *Zhurnal Vychislitelnoi Matematiki Matematicheskoi Fiziki* 20(4):1071-1075, 1980.

This is one of the early papers (first one available in English) that presents a derivation of the stability radius (see [107] for background and terminology). The "bottleneck problems" are defined as follows. A non-negative $n \times n$ matrix, A, is given, along with some set of permissible sequences, which they call "trajectories", consisting of at least two matrix elements. An optimal trajectory is one that minimizes the l_1 norm; the assignment problem is an example of their problem class, where the trajectories are the row-column sequences that define an assignment. They derive a maximal radius for which a trajectory remains optimal under changes to the elements of A.

[50] E.N. Gordeev, V.K. Leontev, and I.Kh. Sigal. Computational algorithms for finding the radius of stability in problems of choice. *Computational Mathematics and Mathematical Physics*, 23(4):973–979, 1983. Note: translated from Russian in *Zhurnal Vychislitelnoi Matematiki Matematicheskoi Fiziki* 23(4):973-979, 1983 ("Problems of Choice" has been translated elsewhere as "Selection Problems").

This extends earlier work on "trajectory problems," which applies to the TSP and bottleneck assignment problems. (See [49].) A modified branch and bound algorithm is presented, with the total computations simply proportional to those needed to find one minimum tour. Experiments with randomly generated cost matrices are reported.

[51] H.J. Greenberg. The use of branching in dynamic programming for parametric analysis. *Operations Research*, 15:976–977, 1968.

This shows how non-serial DP, notably branching, can be used for parametric analysis associated with any pre-determined stage, which corresponds to a variable in a separable MIP.

[52] H.J. Greenberg. A bibliography for the development of an intelligent mathematical programming system. *Annals of Operations Research*, 65:55–90, 1996. Note: also available at http://orcs.bus.okstate.edu/itorms.

> This contains a large list of citations, with cross-referencing categorized by: Analysis, Discourse, Formulation, Model Management, and Software. Some of these (especially in Analysis) pertain to post-solution analysis in MIP and combinatorial optimization.

[53] H.J. Greenberg. *Mathematical Programming Glossary.* World Wide Web, http://www-math.cudenver.edu/~hgreenbe/glossary/glossary.html, 1996-7.

> This is a general resource that contains terms and concepts from many parts of mixed integer programming and combinatorial optimization. It also has links to relevant bibliographies on the web.

[54] O. Guieu. Analyzing infeasible mixed-integer and integer linear programs. M.Sc. thesis, Department of Systems and Computer Engineering, Carleton University, Ottawa, Ontario, Canada, 1995.

> This is the same approach as in [55], but it gives much more detail in the implementation and experimental results. The test problems are partitioned by size, and the statistics include IIS characteristics and measures of efficiency for different isolation and branching schemes. The author gives the following observations:
>
> 1. Depth-first search combined with a best-bound variable selection strategy gave satisfactory results in most cases.
> 2. Efficiency of post-infeasibility schemes are very sensitive to the objective function (e.g., sum of infeasibilities versus maximum infeasibility).
> 3. Due to the presence of unbounded variables, the procedure can terminate with a reducible infeasible subsystem.

[55] O. Guieu and J.W. Chinneck. Analyzing infeasible mixed-integer and integer linear programs. Technical report SCE-96-05, Department of Systems and Computer Engineering, Carleton University, Ottawa, Ontario, Canada, 1996.

> This shows how to compute an irreducible infeasible subsystem (IIS) of a MILP. Unlike LP, there are pathologies where commonly used methods, like branch and bound, do not terminate when a system

> is infeasible. This approach separates the issue of whether the relaxed system is feasible. If not, the LP methodology produces an IIS quickly. If the LP relaxation is feasible, the branch and bound information is used.

[56] D. Gusfield. Bounds for the parametric spanning tree problem. In P.Z. Chinn and D. McCarthy, editors, *Combinatorics and Computing – Proceedings of the West Coast Conference on Graph Theory and Computing*, pages 173–183, Winnipeg, Canada, 1979. Utilitas Mathematica. Note: This was difficult to obtain; its ISBN is 0 919628 26 5.

> This bounds the number of breakpoints in the optimal value function for the one-parameter objective coefficient case, where edge weights are of the form $w + \theta\delta w$. Assuming a connected graph with n nodes and m edges, with $m > n$, the main result is that the number of breakpoints cannot exceed $2m \min\left\{\sqrt{n}, \sqrt{m-n}\right\}$. He also shows $2m$ is an asymptotic lower bound. (These are included in [57].)

[57] D. Gusfield. Sensitivity analysis for combinatorial optimization. Ph.d. thesis; Memo no. UCB/ERL M80/22, Electronics Research Laboratory, University of California, Berkeley, CA, 1980.

> As with any good thesis, this begins with a thorough survey of the literature. The author first develops postoptimal sensitivity results for an extension of matroid optimization, which interleaves two ordered sets. Let M be a matroid with element set E partitioned into two sets, R and W (think of "red" and "white" elements). Given an ordering that preserves the orderings in R and W, let $B_q(R, W)$ denote a minimum weight base containing q elements of R. The *Matroid Selection Problem* is to find $B_q(R, W)$ for all feasible q. Using GLM, a multiplier (λ) is added to the weights of R. As λ is varied, the imputed value of the number of elements of R solves the matroid selection problem for the associated q. (Note that adding the same λ to all weights of R preserves the order.)
> The author then focuses on algorithms for the one-parameter objective case:
> $$z^*(\theta) = \inf\{(c + \theta\delta c)x \colon x \in X\},$$
> where X is generally considered a finite set (for combinatorial optimization). The parametric program is to find the breakpoints of

z^* for some range of θ. The subsequent chapter deals with bounding the number of breakpoints. Stemming from the approach in [91], this presents another method. For the min spanning tree problem with n nodes and m edges, the author proves that the number of breakpoints is not greater than $2m \min\{\sqrt{n}, \sqrt{m-n}\}$. He then returns to the matroid selection problem, showing that this main theorem yields the upper bound: $m \min\left\{\sqrt{\mathrm{r(M)}}, \sqrt{\mathrm{r(M_d)}}\right\}$, where r(M) is the rank of the matroid and $\mathrm{r(M_d)}$ is the rank of its dual. This chapter contains some additional results plus examples, including some analysis of the shortest path problem, where the author brings GLM to the forefront. The last chapter is about multi-terminal network flow synthesis, for a weighted undirected graph, R, with n nodes and m edges. The author states his main results:

"We give simple algorithms which run in time $\max\{m, n\log n\}$, and which construct from R the network G^* with the following desirable properties:

1. G^* is uniformly optimal for R.
2. G^* is planar.
3. No node in G^* has degree greater than four.
4. G^* has as few edges as any uniformly optimal network produced by the Gomory-Hu method.
5. The structure of G^* is easily expressed in terms of R.
6. Routing decisions in G^* can be made locally at each node. This is a very desirable property for communications and computer network applications and hierarchical database applications.

Further, the algorithm shows that the use of the maximum spanning tree, and the revising of the original requirements by the Gomory-Hu method is unnecessary and undesirable."

[58] D. Gusfield. A note on arc tolerances in sparse shortest-path and network flow problems. *Networks*, 13(2):191–196, 1983.

This gives an algorithm with lower computational complexity than [101] for the entitled problem. With n nodes and m arcs, this method requires $O(m)$ space and $O(m \log n)$ time. (Note: "tolerance" is the range of each datum, not to be confused with [79].)

[59] D. Gusfield. Parametric combinatorial computing and a problem in a program module allocation. *Journal of the Association for Computing Machinery*, 30(3):551–563, 1983.

This is concerned with varying the cost in a MILP. The author first considers the one-parameter response and shows that if z^* can be evaluated in polynomial time, then the next break point, θ^{k+1}, given the current break point, θ^k, can be computed in polynomial time. (In light of [21], this does not imply the parametric program is solvable in polynomial time.) He then uses the approach in [91] to find the next breakpoint. The identification of breakpoints in one dimension is a problem of identifying regions of constancy in higher dimensions, and the author considers this for the entitled problem.

[60] P. Hansen and J. Ryan. Testing integer knapsacks for feasibility. *European Journal of Operational Research*, 88(3):578–582, 1996.

This gives an $O(a_1^2+n)$-time algorithm to test the feasibility of $ax = b, x \in Z^{n+}$, where $a_1 < \ldots < a_n$, each a_i is a positive integer, and g.c.d.$\{a_i\} = 1$. The space complexity is $O(a_1)$. For comparison, they note the DP time and space complexities are $O(nb)$ and $O(b)$, respectively.

[61] S. Holm and D. Klein. Discrete right hand side parameterization for linear integer programs. *European Journal of Operational Research*, 2:50–53, 1978.

The ILP is $\max\{cx\colon Ax = b + \Delta,\, x \geq 0,\, x \in \mathbf{Z}^n\}$, and we suppose we have a Gomory cut for the original problem ($\Delta = 0$):

$$\sum_{j \in J} \left(\left\lfloor h\overline{A}_{kj} \right\rfloor - \lfloor h \rfloor \overline{A}_{kj} \right) x_j \geq \lfloor h\overline{x}_k \rfloor - \lfloor h \rfloor \overline{x}_k,$$

where h is a scalar, $\overline{A} = B^{-1}A$, $\overline{x} = B^{-1}b$, and $\lfloor v \rfloor$ denotes the floor (integer round-down) of v. The authors note a simple shift is a Gomory cut for the perturbed problem:

$$\begin{aligned} \sum_{j \in J} \left(\left\lfloor h\overline{A}_{kj} \right\rfloor - \lfloor h \rfloor \overline{A}_{kj} \right) x_j \quad \geq \quad & \lfloor h\overline{x}_k \rfloor - \lfloor h \rfloor \overline{x}_k \; + \\ & \left(\left\lfloor hB^{-1}_{k\bullet} \right\rfloor - \lfloor h \rfloor B^{-1}_{k\bullet} \right) \Delta \; + \; \delta, \end{aligned}$$

where $\delta \stackrel{\text{def}}{=} \left\lfloor \{h\overline{x}_k\} + \{hB_{k\bullet}^{-1}\}\Delta \right\rfloor$; $\{v\}$ is the fractional part of v (so $0 \leq \{v\} < 1$ and $v = \lfloor v \rfloor + \{v\}$). Then, by saving the information that defines δ, they obtain a Gomory cut for the shifted right-hand side.

[62] S. Holm and D. Klein. Three methods for postoptimal analysis in integer linear programming. *Mathematical Programming Study*, 21:97–109, 1984.

The authors address postoptimal sensitivity analysis for right-hand sides of a pure ILP. The "three methods" are as follows.

Method A: Roodman's approach, which applies information obtained when solving the original 0-1 ILP by implicit enumeration to new right-hand sides.

Method B: Similar to Method A (generalized), but for pre-assigned right-hand side values.

Method C: Cutting plane approach, using the authors' earlier works [61, 75]

The experiments consist simply of one ILP with 4 variables and 2 constraints. Thirty problems are solved that differ by the two right-hand side values.

[63] J.N. Hooker. Inference duality as a basis for sensitivity analysis. In *Principles and Practice of Constraint Programming – CP96*, volume 1118 of *Lecture Notes in Computer Science*, pages 224–236. Springer-Verlag, Berlin, Germany, 1996. Note: also available at http://www.gsia.cmu.edu/afs/andrew/gsia/jh38/jnh.html.

The author defines the *inference dual* as the problem of inferring from the constraints a best possible bound on the optimal value, and a solution to this dual is a *proof*. He proceeds to view sensitivity analysis as the role each constraint has in this proof. If it plays no role, one would say (at least marginally) that the optimal value is unaffected by that constraint, analogous to an inactive constraint in LP. This innovative approach is compared with more traditional approaches, notably that of Schrage and Wolsey [98].

[64] D.R. Insua and S. French. A framework for sensitivity analysis in discrete multi-objective decision-making. *European Journal of Operational Research*, 54(2):176–190, 1991.

This is like an artificial intelligence approach, employing "rules of thumb" to reflect how we intuitively establish stability regions for decision-making among a finite set of alternatives. Uncertainty is represented by Baysian probabilities, which provide weights for multiple objectives. These are related, in the usual manner, to Pareto optimality, which they call "potentially optimal" (p.o.). Then, a Chebyschev norm is used to measure nearness of optimality, forming the "framework" with algorithms, paying special attention to the specification of weights.

[65] L. Jenkins. Parametric mixed integer programming: An application to solid waste management. *Management Science*, 28(11):1270–1284, 1982.

This considers the cases of changing the right-hand side or the objective coefficients (separately) in a MILP. The postoptimal analysis is motivated by a fixed charge location model of reclamation and disposal of municipal solid waste. The procedure is heuristic, and computational experiments are given to suggest it is practicable. A central part of the method is to suppose the rule: *If the optimal solution for* θ_1 *is the same as for* θ_2*, assume that solution is optimal throughout the interval,* $[\theta_1, \theta_2]$. (See [128] for an example that this rule can fail.) One of the observations for the entitled application is that plants can be ordered by their economic properties so that their opening can be determined with a parametric 0-1 MILP, letting the number of plant openings increase.

[66] L. Jenkins. Parametric-objective integer programming using knapsack facets and Gomory cutting planes. *European Journal of Operational Research*, 31(1):102–109, 1987.

This addresses the one-parameter variation of the pure integer program for a specified direction of change. The author exploits the concavity of z^* for the objective coefficient case in an ILP with a heuristic that avoids solving independent ILPs. This is done by Gomory's group knapsack formulation.

[67] L. Jenkins. Using parametric integer programming to plan the mix of an air transport fleet. *INFOR*, 25(2):117–135, 1987.

This applies the heuristic in [65] to the entitled problem for the one-parameter objective case. In addition, a budget constraint is varied (separately), which is a special case of the monotone right-hand side. The price coefficients in the budget constraint are also subjected to one-parameter variation with a method similar to the right-hand side case. Finally, another set of matrix coefficients is varied by one parameter: the number of aircraft that appears in the logical constraint for the associated binary fixed charge variable ($\overline{x}$). With a positive fixed charge, $(m_i + \lambda\delta m_i)\overline{x}_i - x_i \geq 0$ implies $\overline{x}_i = 1 \Leftrightarrow x_i > 0$, and $x_i \in \{0, \ldots, \lfloor m_i + \lambda\delta m_i \rfloor\}$. Monotonicity of f^* as a function of λ is exploited in finding breakpoints for $\lambda \in [0, 1]$.

[68] L. Jenkins. Parametric methods in integer linear programming. *Annals of Operations Research*, 27:77–96, 1990.

This is a succinct review, clarifying some of the earlier works by putting them into perspective.

[69] R.G. Jeroslow and J. Wang. Dynamic programming, integral polyhedra, and horn clause knowledge bases. *ORSA Journal on Computing*, 1(1):7–19, 1988.

This applies integer programming to inference in expert systems that assume horn clauses in their knowledgebase (which was the usual case at that time, and which is often the case now). The variables are the truth values of propositions, and the procedure provides a proof structure. Further, the procedure can detect a "near proof" when the original theorem is not true (maximizing some truth variable that cannot reach the value of 1).

[70] C.V. Jones. Sensitivity Analysis for the Euclidean Traveling Salesman Problem. Technical report, University of Washington, World Wide Web, http://weber.u.washington.edu/~cvj/tsp/tspnew.html, 1996.

This is a highly innovative use of web technology to demonstrate the authors' analysis of the TSP. Associated with each city is a region such that if the city is relocated anywhere in that region, the tour remains optimal. Using color coding, the author illustrates that the regions need not be connected. Complete documentation is available from the web site.

[71] C.V. Jones. The Stability of Solutions to the Euclidean Traveling Salesman Problem: Part I: Experimental Results. Technical report, School of Business Administration, University of Washington, Seattle, WA 98195, 1997. Note: also available at http://is.badm.washington.edu/tsp.

This shows, with an extensive empirical study, the results established in its companion paper [72]. Using the image processing technique introduced in [70], this designs experiments with carefully chosen algorithms and problems. The algorithms chosen to obtain optimal solutions are a branch-and-bound code and a cutting plane method, plus more than 30 heuristics were included in the study. In some cases, the code used was by the author; in other cases they were published by others. For the statistical inference, the author generated hundreds of problems, for 10, 20 and 50 cities, using a uniform distribution on coordinate locations. In addition, the author executed the algorithms and heuristics on the classical 49 city problem (48 contiguous U.S. state capitals plus Washington, D.C.), and on a 6-city problem to demonstrate that the regions can have holes. Statistical tests were used to draw the following conclusions:

1. In most cases, the optimal solution was more stable than the heuristics. One exception is a variant of the nearest neighbor; and, the space-filling heuristic was more stable for only two of the four measures of stability.
2. Changing a heuristic, such as adding 2-Opt or 3-Opt postprocessing, can sometimes yield more stable solutions.
3. The image processing technique used can study other problem domains that can be mapped into the plane. For example, a knapsack problem has two data elements for each item: size (a) and benefit (b). Then, one could map (a, b) into the plane and ask about the stability with respect to these parameters.

[72] C.V. Jones. The Stability of Solutions to the Euclidean Traveling Salesman Problem: Part II: Theoretical Results. Technical report, School of Business Administration, University of Washington, Seattle, WA 98195, 1997. Note: also available at http://is.badm.washington.edu/tsp.

This companion to [71] presents terms, concepts and theorems for the entitled topic. A *region* is the usual definition of stability region, and a *piece* is a maximal connected subregion. Regions can be disconnected, and they are unions of their pieces. The author defines four measures of stability:

Number of Regions : The more regions for a given area, the greater the chance the tour will change as a city is moved.

Number of Pieces : A connected stability region is more stable than one composed of disconnected pieces.

Number of "Sharp Corners" : Measured by polygonal approximation to each piece. A square or circle is more stable than an odd-shaped polygon.

Degree of Gerrymandering : This is measured by $G = \frac{P_R^2}{4\pi A_r}$, where P_R is the perimeter of the region, and A_R is the area of the piece. The minimum value of G is 1, which is for a circle.

Some of the theoretical results are given in the authors' summary table:

Algorithm/ Heuristic	Regions		Pieces		
	Max Number	Connected	Convex	Hole-Free	Star-Shaped
OPTIMAL	$O(n^2)$	No	No	?	?
NEAREST NEIGHBOR (variant)	$O(n)$	Yes, except for possibly one	Yes, except for possibly one	Yes	Yes
SPACE-FILLING	$O(n)$	Yes	No	Yes	No
CONVEX HULL	$O(n^2)$	Yes	Yes	Yes	Yes
ASSIGNMENT	$O(n^2)$	No	No	?	?
MATCHING	$O(n)$	Yes, for n even	No	Yes, for n even	Yes, for n even

[73] R.M. Karp and J.B. Orlin. Parametric shortest path algorithms with an application to cyclic staffing. *Discrete Applied Mathematics*, 3:37–45, 1981.

Given a digraph, this adds the same parameter to the cost of each arc in a given subset of arcs. Then, the shortest path is computed as a function of this parameter. The "cyclic staffing" problem has the mathematical form: $\min \sum_j x_j$: $Ax \geq b$, $x \geq 0$, $x \in \mathbf{Z}^n$, where A is a column circular matrix. One example is to minimize the number of persons needed to staff a weekly schedule that vary from day to day but repeat weekly, such that each person has two consecutive days off per week.

[74] S. Kim and S. Cho. A shadow price in integer programming for management decision. *European Journal of Operational Research*, 37:328–335, 1988.

This defines the *average shadow price* for an ILP, perturbing the i-th "resource constraint:" $A_{i\bullet}x \le b_i + \theta$. The perturbed maximum value is $z_i(\theta) \stackrel{\text{def}}{=} z^*(\theta; b, \mathrm{e}_i)$ (where e_i denotes the i-th unit vector). The associated *net profit function* is $e_i(p) \stackrel{\text{def}}{=} \sup_{\theta \ge 0}\{z_i(\theta) - z_i(0) - p\theta\}$ (where e_i is their notation), and they define the associated shadow price as: $y_i \stackrel{\text{def}}{=} \inf\{p\colon\, p \ge 0,\, e_i(p) \le 0\}$. This is the slope of the envelope function of z_i (which is a step function that is non-decreasing and left-continuous), following the usual conjugate function theory. The reason they call this an "average" price is that it is equivalent to the greatest average rate on $\mathbf{R}^+$: $y_i = \sup_{\theta > 0}\left\{\frac{\Delta z_i(\theta)}{\theta}\right\}$. The authors proceed to prove mathematical/economic properties that parallel LP prices in connection with market equilibria.

[75] D. Klein and S. Holm. Integer programming postoptimal analysis with cutting planes. *Management Science*, 25(1):64–72, 1979.

This describes when a generated Gomory cut remains valid for a new right-hand side. Their approach extends to other cutting planes.

[76] A.W.J. Kolen, A.H.G. Rinnooy Kan, C.P.M. van Hoesel, and A.P.M. Wagelmans. Sensitivity analysis of list scheduling heuristics. *Discrete Applied Mathematics*, 55(2):145–162, 1994.

This is contained in [125], with some added commentary for insights.

[77] J. Komlös. Linear verification for spanning trees. *Combinatorica*, 5(1):57–65, 1985.

This gives an $O(n)$ algorithm to verify the optimality of a spanning tree on n vertices. This applies to obtain the range of each edge weight for which a tree remains optimal.

[78] S.A. Kravchenko, Y.N. Sotskov, and F. Werner. Optimal schedules with infinitely large stability radius. *Optimization*, 33:271–280, 1995.

This continues with earlier works [105], focusing on when processing times can change by any amount, yet the optimal digraph remains the same. One simple case is when the precedence constraints force just one feasible solution. More generally, define the following:

$Q_k \stackrel{\text{def}}{=}$ operations that must be processed on the k-th machine;

$A_k \stackrel{\text{def}}{=}$ jobs that must precede at least one job in Q_k;

$B_k \stackrel{\text{def}}{=}$ jobs that must succeed at least one job in Q_k.

Then, one of the theorems is the following for minimizing makespan:

There exists an optimal digraph with infinite stability radius if, and only if, the following conditions hold:

1. $|Q_k| > 1 \Rightarrow \max\{|A_k|, |B_k|\} \leq 1$;
2. *If there exist* $g \in A_k$ *and* $f \in B_k$ *of some job, then there exists a path from* f *to* g *in the optimal digraph (possibly* $f = g$*).*

The authors' algorithm to find an optimal schedule with infinite radius (if it exists) has $O(n^2)$ time complexity.

[79] M. Labbé, J-F. Thisse, and R.E. Wendell. Sensitivity analysis in minisum facility location problems. *Operations Research*, 39(6):961–969, 1991.

This considers the objective coefficient case for the problem: $f^*(w) \stackrel{\text{def}}{=} \min\left\{\sum_{j=1}^n w_j d(x, y^j) \colon x \in X\right\}$, where X is a finite set, $\{y^j\}$ is a set of given points, d is a distance function, and $\{w_j\}$ are non-negative weights that comprise the objective coefficients of interest. Following the tolerance approach in LP, the weights are varied by $\gamma{\bullet}\delta w \stackrel{\text{def}}{=} (\gamma_1 \delta w_1, \ldots, \gamma_n \delta w_n)$, where γ is in a polytope, Γ, that contains the origin. Two special cases are noted: $\gamma_j = \theta$ for all j is the one-parameter case, and $\delta w = w$ varies the weights by percentages. Let x^0 be an optimal solution for $\gamma = 0$, and define $F(x^0, \gamma) = \sum_j (w_j + \gamma_j \delta w_j) d(x^0, y^j)$. The response function is a *degree of optimality* for a tolerance, $\tau > 0$, given by the objective difference:

$$\alpha^*(\tau) \stackrel{\text{def}}{=} \max_{\substack{\gamma \in \Gamma \\ ||\gamma||_\infty \leq \tau}} \left\{F(x^0, \gamma) - f^*(w + \gamma{\bullet}\delta w)\right\}.$$

The parametric problem is to compute, or characterize, α^* more efficiently than solving the optimization problem to evaluate f^*. The authors provide an algorithm, based on LP, for location on a tree, and under a block norm in $\mathbf{R}^2$.

[80] T-C. Lai, Yu.N. Sotskov, N.Yu. Sotskova, and F. Werner. Optimal makespan scheduling with given bounds of processing times. Preprint 20/97, Otto-von-Guericke-Universität, Fakultaät Mathematik, Magdeburg, Germany, 1997.

> This uses a stability approach [18, 108] to deal with uncertain processing times. An expression for the stability radius is established that is a generalization of [105], and an inverse query approach is taken to move towards pre-set bounds. The algorithm stops when a solution is obtained for processing times whose stability region includes one member within the bounds.

[81] M. Libura. Integer programming problems with inexact objective function. *Control and Cybernetics*, 9(4):189–202, 1980.

> This considers the objective coefficient case of the ILP. For any feasible x, the range of the objective value over the set of possible costs, C, is $v_{\min}(x) = \min_{c \in C}\{cx\}$ to $v_{\max}(x) = \max_{c \in C}\{cx\}$. Letting X be the feasible region, the objective range is $\min\{v_{\min}(x)\colon x \in X\}$ to $\min\{v_{\max}(x)\colon x \in X\}$, and associated solutions are called *optimistic* and *pessimistic*, respectively. These extremes can be obtained by minimizing (maximizing) a variable, t, and augmenting constraints, $t \le cx$ (resp., $t \ge cx$) for all $c \in C$. The author suggests that these values offer some help when nothing else is known about the costs. (One could solve each extreme, to see what the range is, in order to determine whether it is worth obtaining more information.)

[82] M. Libura. On traveling salesman problem with side constraints. In R. Kulikowski and J.Sosnowski, editors, *Badania Systemowe, t.2. Metody Optymalizacji i Sterowania Komputerowego*, pages 134–142, Warszawa, Poland, 1990. Omnitech Press.

> This approaches the problem of finding the stability region for the arc weights of the entitled problem, drawing from [83] (which had been written in 1988). The "constraints" are those that force edges into or out of the tour. The idea is to solve a relaxation, using a penalty value added to (or subtracted from) the arc weights.

[83] M. Libura. Sensitivity analysis for minimum Hamiltonian path and traveling salesman problems. *Discrete Applied Mathematics*, 30(2):197–211, 1991.

This uses a relaxation to give lower bounds on each arc weight for a TSP solution to remain optimal. One of the theorems is the following: *Suppose e is an edge in an optimal tour, with weight* w_e*, and* v^* *is the value of that tour. Let* $v(e)$ *be the value of an optimal tour with the added restriction that edge e cannot be included. Then, the range of the change in edge weight,* Δw_e*, for which the original tour remains optimal is* $(-\infty, v(e) - v*]$.

[84] M. Libura. Sensitivity analysis for the minimum weight base of a matroid. *Control and Cybernetics*, 20(3):7–24, 1991.

The arc weights are allowed to vary individually, and the goal is to find the range for which a base remains minimal. Let B denote the minimum base, with optimal value $v(B)$. Suppose $x \in B$. Let $v(B^-)$ be the value of the minimum weight base of the same matroid, but without x. Then, the weight of x can increase by $v(B^-) - v(B)$, and B remains a minimum weight base of the original matroid, so its range is $(-\infty, w_x + v(B^-) - v(B))$. Now suppose $x \notin B$, and let $v(B^+)$ be the minimum weight base of the same matroid, with x required to be in it. Then, its weight can decrease by $v(B^+) - v(B)$, and B remains a minimum weight base of the original matroid, so its range is $(w_x - v(B^+) + v(B), \infty)$.

[85] M. Libura. On accuracy of solutions for discrete optimization problems with perturbed coefficients of the objective function. Report PMMi0-1/96, Systems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447, Warszawa, Poland, 1996. Note: to appear in *Annals of Operations Research*.

This is for the 0-1 ILP with $c > 0$. Two types of perturbations are considered: (1) absolute: $|\delta c_j| \leq r$; (2) relative: $|\delta c_j| \leq rc_j$, where the radius (r) is the same for each j. The *relative error function* of any feasible solution, x, is the relative deviation from the optimum: $e(x;c) \stackrel{\text{def}}{=} \dfrac{cx - f^*(c)}{f^*(c)}$ (where $x^* \neq 0$, so $f^*(c) > 0$). The *sensitivity function* measures the worst absolute perturbation: $s(x,r) \stackrel{\text{def}}{=} \max_{|\delta c_j| \leq r} e(x, c + \delta c)$, for $0 < r \leq \max\{c_j\}$. The *accuracy function* measures the worst relative perturbation: $a(x,r) \stackrel{\text{def}}{=} \max_{|\delta c| \leq rc} e(xc + \delta c)$, for $0 < r \leq 1$. The author derives these functions, and some approximations that require less computation.

[86] M. Libura. Optimality conditions and sensitivity analysis for combinatorial optimization problems. *Control and Cybernetics*, 25(6):1165–1180, 1996.

> This is concerned with the stability region of an optimum. After indicating how to use simple relaxations and restrictions to characterize subsets, the author considers the use of k-best solutions. (This is for the generic combinatorial optimization problem, where a solution is, for example, a particular subgraph (tour, cover, schedule, or tree), so "optimal values" do not change level – i.e., $x^*(p)$ is constant throughout a stability region.) The author lastly considers the use of the Lagrangian (weak) dual and the subadditive (strong) dual.

[87] S.C. Lin and E. Ma. Sensitivity analysis of 0-1 multiterminal network flows. *Networks*, 21(7):713–745, 1991.

> This considers the addition or deletion of an edge from a 0-1 flow in an undirected network with n nodes. As in the Gomory-Hu algorithm, a "cut-tree" is constructed, but the authors' algorithm could require fewer than $n-1$ maximum flow problems to be solved.

[88] E. Loukakis and A.P. Muhlemann. Parameterisation algorithms for the integer linear programs in binary variables. *European Journal of Operational Research*, 17:104–115, 1984.

> This considers the pure 0-1 ILP, extending earlier works of Roodman for the one-parameter postoptimal sensitivity analysis by sharpening Roodman's LP bounds in the parametric program. The impulse can be any one data element in $\{c_j, b_i, A_{ij}\}$. The response is the optimal value function. The essential idea is to consider the parameter throughout a (classical) branch and bound method. At each test, the parameter range is divided into intervals: where it passes and where it fails.

[89] R.E. Marsten and T.L. Morin. Parametric integer programming: The right-hand side case. *Annals of Discrete Mathematics*, 1:375–390, 1977.

> This modifies the generic branch and bound algorithm for the one-parameter right-hand side case by having objective lower and upper bound functions of θ. The "bounding test" becomes a comparison of

two functions on the unit interval, rather than a point comparison for $\theta = 0$. The parametric problem is to solve

$$\max\ cx\colon\ Ax \le b + \theta\delta b,\ x \in \{0,1\}^n$$

for all θ in [0,1]. The lower bound function is simply the best feasible solution generated. At a node in the search tree, J^0 is the set of j for which $x_j = 0$, and J^1 is the set of j for which $x_j = 1$. Then, the upper bound function is the parametric LP relaxation:

$$\max\ cx\colon\ Ax \le b + \theta\delta b,\ x \in [0,1]^n,$$
$$x_j = 0 \text{ for } j \in J^0,\ x_j = 1 \text{ for } j \in J^1.$$

[90] R.D. McBride and J.S. Yormark. Finding all solutions for a class of parametric quadratic integer programming problems. *Management Science*, 26(8):784–795, 1980.

This considers the monotone right-hand side case of a pure 0-1 quadratic program. The authors show how to apply information from a branch and bound algorithm to multiple parameter values.

[91] N. Megiddo. Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4(4):414–424, 1979.

This applies the one-parameter ILP case to solve a fractional integer program: $F^*(c,d) = \min\left\{\frac{cx}{dx}\colon\ x \in X\right\}$, where X is a non-empty finite set and $dx > 0$ for all $x \in X$. He uses the parametric pure ILP: $z^*(\theta; c,d) = \min\left\{(c+\theta d)x\colon\ x \in X\right\}$, showing that $z^*(\theta; c,d) = 0 \Leftrightarrow F^*(c,d) = -\theta$; $z^*(\theta; c,d) < 0 \Leftrightarrow F^*(c,d) < -\theta$; and, $z^*(\theta; c,d) > 0 \Leftrightarrow F^*(c,d) > -\theta$. He uses these relations to find a solution to the fractional IP for particular cost data (c,d).

[92] R.M. Nauss. *Parametric Integer Programming.* University of Missouri Press, Columbia, Missouri, 1979.

This is the author's Ph.D. thesis, and it gives a detailed account of what had been done up to that time (also see [45]). After some general theorems, the thesis concentrates on particular problem classes to improve algorithm efficiency for parametric analysis. Particular problems analyzed are: scheduling, 0-1 knapsack, generalized assignment, and capacitated facility location.

[93] Y. Ohtake and N. Nishida. A branch-and-bound algorithm for 0-1 parametric mixed integer programming. *Operations Research Letters*, 4(1):41–45, 1985.

> This considers the right-hand side case of a binary MILP. An example is used to compare this exact algorithm with the heuristic in [65].

[94] K. Richter and J. Vörös. A parametric analysis of the dynamic lot-sizing problem. *Journal of Information Processing Cybernetics*, 25(3):67–73, 1989.

> The one-parameter case is considered for varying setup and holding costs or demand. Then, these are considered jointly in a 2-parameter variation.

[95] S.L.K. Rountree and B.E. Gillett. Parametric integer linear programming: A synthesis of branch and bound with cutting planes. *European Journal of Operational Research*, 10:183–189, 1982.

> This assumes prior knowledge of the set of right-hand side changes. (They can also be derived if only a change vector is specified.) The approach is to save appropriate information from tests used in a branch and bound framework that apply to at least one right-hand side.

[96] J. Ryan. The structure of an integral monoid and integer programming feasibility. *Discrete Applied Mathematics*, 28(3):251–264, 1990.

> This considers the right-hand side case of a MILP. The author casts the feasibility problem as whether b belongs to the monoid induced by A: $\mathrm{mon}(A) \stackrel{\mathrm{def}}{=} \{Ax\colon x \in Z^{n+}\}$. One of the theorems is an analogue of Weyl's theorem for cones: *there exists m-dimensional Chvátal functions,* $f_1, \ldots, f_p$, *such that* $\mathrm{mon}(A) = \{y \in Z^m\colon f_i(y) \geq 0,$ *for* $i = 1, \ldots, p\}$.

[97] L. Schrage. *Optimization Modeling with LINDO.* Duxbury Press, Pacific Grove, CA, 5th edition, 1997.

> Chapter 18 contains a section entitled "Parametric Analysis of Integer Programs," which is for the pure 0-1 ILP. The software uses the technique in [65] for the one-parameter objective coefficient case.

[98] L. Schrage and L.A. Wolsey. Sensitivity analysis for branch and bound integer programming. *Operations Research*, 33(5):1008–1023, 1985.

This considers the right-hand side case of a pure 0-1 ILP. It also considers the second question of how great must the cost coefficient be of a new 0-1 activity in order for it to be 0 in an optimal solution (augmented to the current optimum). First, it supposes the original ILP is solved by branch and bound, then it considers a branch and cut scheme. In both cases, the issue is how to use information from the algorithm to minimize the amount of computation that needs to be performed in order to answer the postoptimal sensitivity questions. (The authors assume all node and cut information have been saved.)

[99] I.V. Sergienko and L.N. Kozeratskaya. Solution of a parametric integer programming problem. *Cybernetics*, 18:360–367, 1982. Note: translated from *Kibernetika* 3:80-84, 1982.

This considers perturbing one constraint:

$$\max\ cx\colon\ x \in \mathbf{Z}^n \cap D \cap \{x\colon\ (a + \theta\delta a)x \leq b + \theta\delta b\},$$

where D is a non-empty, convex polyhedron in $\mathbf{R}^n$, and cx is bounded on D. The problem is the parametric program: compute this maximum (or ascertain there is no feasible solution) for all $\theta \in [\underline{\theta}, \overline{\theta}]$, where $\underline{\theta} < 0 < \overline{\theta}$. The author finds an effective algorithm, based on some simple properties.

[100] J.F. Shapiro. Sensitivity analysis in integer programming. *Annals of Discrete Mathematics*, 1:467–477, 1977.

This uses the Lagrangian dual, as in GLM, to produce breakpoints where the LP relaxation solves the ILP. For the analysis, the author assumes no duality gap and considers the question of retaining this under perturbation. The range for a cost, c_j, is simple when $x_j^* = 0$. For the more difficult case of $x_j^* = 1$, the bound on the change Δc_j is when x^* is no longer optimal for the Lagrangian (it is weak in that x^* could lose optimality before this threshold is reached). Right-hand side ranges are approximated, and matrix coefficients are considered in a special way. The case where the associated Lagrange multiplier is zero is easy, for then A_{ij} can vary without changing the optimality of x^*.

[101] D.R. Shier and C. Witzgall. Arc tolerances in shortest path and network flow problems. *Networks*, 10(4):277–291, 1980.

> This uses network methodology to determine how much an arc's cost can increase or decrease while preserving the optimality of a tree for the shortest path. The tree corresponds to a basis in LP, so this is an alternative computational approach to the standard parametric programming for the objective coefficient case. Its advantage is that all arc tolerances can be determined at once. (Note: "tolerance" is the range of each datum, not to be confused with [79].)

[102] J. Skorin-Kapov and F. Granot. Non-linear integer programming: Sensitivity analysis for branch and bound. *Operations Research Letters*, 6(6):269–274, 1987.

> This extends the Schrage-Wolsey [98] approach to 0-1 integer linear programs to those whose relaxation is a convex program. Particular attention is given to a quadratic program with linear constraints. The Wolfe (and Dorn) duality is used to establish the validity of bounds obtained during the original solution process to perturbed right-hand side and objective coefficient cases.

[103] P.T. Sokkalingam, R.K. Ahuja, and J.B. Orlin. Solving inverse spanning tree problems through network flow techniques. Working paper WP 3914-96, MIT Sloan School, Cambridge, MA, 1996.

> This is a form of the inverse query: given x^*, and a cost vector, c, the problem is to find another cost vector, d, such that x^* corresponds to a minimum spanning tree for d, and $||d - c||$ is a minimum for which this is true. They prove this is the dual of an assignment problem with special structure, which they exploit in their algorithm design. They extend this to a weighted spanning tree problem, showing that inverse query is the dual of a transportation problem.

[104] Yu.N. Sotskov. Stability of high-speed optimal schedules. *Computational Mathematics and Mathematical Physics*, 29(3):57–63, 1989. Note: translated from Russian in *Zhurnal Vychislitelnoi Matematiki Matematicheskoi Fiziki* 29(5):723-731, 1989.

This extends earlier works [49, 50] that were given as the stability of "trajectory problems," meaning graph tours, like the TSP. Earlier works introduced the *stability radius* for one optimal solution [49]. (See [105] for definitions and further insights.) Analogous to the range for one basis in an LP, this would have zero radius if there are alternative optima. This paper deals more generally with the radius for scheduling problems, and it shows the existence of schedules for which the stability radius is infinite – i.e., the optimal digraph does not depend upon the processing times. This lays the foundation for a series of papers by this author and others.

[105] Yu.N. Sotskov. Stability of an optimal schedule. *European Journal of Operational Research*, 55:91–102, 1991.

The *radius* of the stability region of an optimal solution, $x^* \in X^*(p)$, for processing times, p, is the largest l_1 ball of processing times that preserve optimality:

$$\max\{r\colon |\delta p_i| \le r\, \forall i,\ p + \delta p \ge 0, \text{ and } x^* \in X^*(p + \delta p)\}.$$

The *optimal digraph* is the digraph whose nodes are jobs and whose arcs are precedence relations, determined by an optimal schedule. (The relations include the ones given as constraints, but more generally represents a qualitative property of which jobs precede which in an optimal schedule, x^*.) The optimal digraph could remain the same even if the schedule changes, and the author considers the stability region for the optimality of the digraph. This finds the radius of an optimal schedule and suggests its application to stochastic scheduling. While some results are given for any of the usual criteria, simple bounds are derived for the makespan.

[106] Yu.N. Sotskov. The stability of the approximate Boolean minimization of a linear form. *Computational Mathematics and Mathematical Physics*, 33(5):699–707, 1993. Note: translated from Russian in *Zhurnal Vychislitelnoi Matematiki Matematicheskoi Fiziki* 33(5):785-795, 1993.

This presents an algorithm to compute the *radius of stability* [105] of an approximate solution for a variety of combinatorial optimization problems that include scheduling and the TSP.

[107] Yu.N. Sotskov, V.K. Leontev, and E.N. Gordeev. Some concepts of stability analysis in combinatorial optimization. *Discrete Applied Mathematics*, 58(2):169–190, 1995.

> The entitled subject started more than two decades ago, but it has accelerated its pace since 1989. This survey includes the early results that were available only in Russian. (See [105] for central concepts and terminology.) This reviews the computation of the stability radius, and more easily computed bounds, of both an optimal digraph and that of an approximate solution. Applications discussed in this survey include TSP, assignment, shortest path, Steiner/spanning tree, scheduling, and matroids. (See [109] for a more recent survey on this topic.)

[108] Yu.N. Sotskov, N.Yu. Sotskova, and F. Werner. Stability of an optimal schedule in a job shop. *OMEGA*, 25(5), 1997. To appear.

> The criteria for the entitled problem is to minimize mean or maximum flow times. The former is more complicated, and the latter is also called the makespan. This is a computational study to compute maximal, average and minimal values of the stability radius (see [109] for background). They generated about 10,000 job shop scheduling problems, addressing the following questions: • How often is the stability radius equal to zero? • May the stability radius be infinity? • How many 'best' schedules do we need to consider? • How can one combine this approach with the branch and bound method? • How can we use this approach for problems of practical size? In the last question, "this approach" refers to enumeration of feasible digraphs that can be optimal for some set of processing times.

[109] Yu.N. Sotskov, V.S. Tanaev, and F. Werner. Stability radius of an optimal schedule: A survey and recent developments. In G. Yu, editor, *Industrial Applications of Combinatorial Optimization*. Kluwer Academic Press, Boston, MA, 1997. To appear.

> This paper updates the previous survey [107], bringing the calculation and use of stability regions of optimal digraphs into focus. Their empirical study reflects the indication in [18] that the computation of the stability radius for an optimal mean flow digraph is more complicated than that of an optimal makespan digraph. In addition, the

authors suggest avenues for further research: • improve bounds for which a digraph remains optimal; • reduce path computations; • refine algorithms, like branch-and-bound, and heuristics, like SPT and LPT, to combine computation of k-best solutions with that of stability regions. (See [105] for central concepts and terminology.)

[110] Yu.N. Sotskov, A.P.M. Wagelmans, and F. Werner. On the calculation of the stability radius of an optimal or an approximate schedule. Report 9718/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1997.

This advances the basic ideas and results in [18], with special attention given to where the stability radius is 0 or ∞. They give a polynomial algorithm to compute the stability radius of an approximate solution when the number of unstable parameters fixed at 1 grows as $O(\log n)$ for n jobs. In addition, they establish new bounds on the radius and some improvement in the exact computation for the makespan criterion.

[111] B. Sturmfels and R.R. Thomas. Variation of cost functions in integer programming. *Mathematical Programming*, 77:357–387, 1997.

This considers the following for an ILP: two cost vectors are defined to be "equivalent" if they give the same optimal solutions for every b. The equivalence classes of c vectors are shown to be the normal cones of a union of polytopes defined by reduced Gröbner bases of A. Much of the paper depends on knowing about Gröbner bases and their (recently shown) connections with ILP. Of interest is a table that they give in their introduction to summarize the interrelations among the main concepts in their paper.

[112] R.E. Tarjan. Sensitivity analysis of minimum spanning trees and shortest path problems. *Information Processing Letters*, 14(1):30–33, 1982.

This provides an algorithm for the entitled applications that has lower computational complexity than those in [58, 101].

[113] G.K. Tayi. Sensitivity analysis of mixed integer programs: An application to environmental policy making. *European Journal of Operational Research*, 22:224–233, 1985.

This exploits the special structure of a MIP for the entitled application. The constraints are linear, but the objective is a nonlinear, concave maximand. The purpose is to deal with three objectives, and postoptimal sensitivity analysis for the relevant right-hand sides is used to conduct trade-offs.

[114] A. Turgeon. An application of parametric mixed-integer linear programming to hydropower development. *Water Resources Research*, 23(3):399–407, 1987.

This considers the right-hand side case of a 0-1 MILP, where the binary variables represent whether or not to build a dam at a site. The change vector is for capacity and energy demand, keeping the other right-hand sides fixed. The method uses the heuristic in [65], and some computational experiments suggest this works well.

[115] E.S. van der Poort. Aspects of sensitivity analysis for the traveling salesman problem. Ph.d. thesis, Graduate School/Research Institute Systems, Organizations and Management, University of Groningen, P.O. Box 800, Groningen, The Netherlands, 1997.

This begins with basic concepts and a literature review. Most of the results are contained in technical reports [116, 117, 118, 119]. Not cited here are Chapter 4: *Solving the k-best TSP*, and Chapter 7: *The maximum number of tours Hamiltonian in graphs.* The former is a review of prior results, leading to the subsequent chapter [116]. The latter gives lower and upper bounds on the number of Hamiltonian tours in a graph.

[116] E.S. van der Poort, V. Dimitrijević, G. Sierksma, and J.A.A. van der Veen. Using stability information for finding the k-best traveling salesman problem. Research report 97A29, Graduate School/Research Institute Systems, Organizations and Management, University of Groningen, P.O. Box 800, Groningen, The Netherlands, 1997. Note: this is Chapter 5 in [115].

This addresses the question, "How can we use stability information to solve the k-best TSP?" The authors consider the 2-best TSP, given an optimal tour and its stability ranges. They argue that finding the second best tour without these ranges takes exponential time, whereas they present a polynomial-time algorithm once the ranges are found (under mild conditions). They proceed to prove that the

k-best TSP remains NP-hard for $k > 2$, even when stability ranges are known.

[117] E.S. van der Poort, M. Libura, G. Sierksma, and J.A.A. van der Veen. Sensitivity analysis based on k-best solutions of the traveling salesman problem. Research report 96A14, Graduate School/Research Institute Systems, Organizations and Management, University of Groningen, P.O. Box 800, Groningen, The Netherlands, 1996. Note: this is Chapter 6 in [115], and it is to appear in *Annals of Operations Research*.

Following [135] and the earlier works of Piper and Zoltners, the k-best tours are used to determine the range of cost values for which the optimal tour remains optimal. (Whereas the earlier works are for general 0-1 ILP, this is specifically for the TSP.) Subsets of the (polyhedral) stability region are determined from knowing value differences among the best k tours and whether a particular edge remains in or out of the tour (versus it leaving or entering, resp.).

[118] E.S. van der Poort, G. Sierksma, and J.A.A. van der Veen. Determining tolerances for the traveling salesman problem. Research report 97A27, Graduate School/Research Institute Systems, Organizations and Management, University of Groningen, P.O. Box 800, Groningen, The Netherlands, 1997. Note: this is Chapter 2 in [115].

This begins with basic terms and a review of known results. Given an optimal solution, the *tolerance problem* is to find the range of each cost for which this remains optimal. (Note: "tolerance" is the range of each datum, not to be confused with [79].) They then address the problem complexity, which is NP-hard, followed by how one can solve a TSP by solving a polynomial number of tolerance problems. Each problem in the sequence is related to its predecessor, so a solution might be computed quickly. (Theoretically, it is not polynomial-time.) One could start with all costs equal to zero, so any Hamiltonian circuit is optimal. The idea is to move towards the original costs by an inverse query approach. The results are extended to consider the complexity of determining approximate solution values, following the results in [122].

[119] E.S. van der Poort, G. Sierksma, and J.A.A. van der Veen. Stability regions for the symmetric traveling salesman problem. Research report 97A28, Graduate

School/Research Institute Systems, Organizations and Management, University of Groningen, P.O. Box 800, Groningen, The Netherlands, 1997. Note: this is Chapter 3 in [115].

This characterizes the stability region in terms of the Hamiltonian cycle polytope, showing too that the set of all constant length cost vectors is the lineality space of the stability region. In the process, the following preliminary results are proven (where m is number of costs (called "lengths" in the paper) – viz., $m = n(n-1)/2$ for the symmetric case with n nodes):

1. The stability region is an m-dimensional, non-pointed, convex cone.
2. The intersection of all stability regions equals the set of constant length vectors, and the union equals $\mathbb{R}^m$.

[120] C.P.M. van Hoesel, A.W.J. Kolen, A.H.G. Rinnooy Kan, and A.P.M. Wagelmans. Sensitivity analysis in combinatorial optimization: A bibliography. Technical report 8944/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1989.

This is simply a bibliography without annotation or survey. It includes problems that can be solved by linear programming, such as single-commodity network flows.

[121] S. van Hoesel and A. Wagelmans. Sensitivity analysis of the economic lot-sizing problem. *Discrete Applied Mathematics*, 45(3):291–312, 1993.

This addresses ranges of setup, production and holding costs, and ranges of demands, that preserve optimality for the Wagner-Whiten lot-sizing problem. The method uses DP, as in [125]. Here are some of the results:

1. *The maximal allowable decrease of each setup cost can be calculated in constant time.*
2. *The maximal allowable decrease of each production cost can be calculated in constant time for a production period, and in* $O(\log n)$ *for a non-production period.*
3. *The maximal allowable increases of all setup and production costs can be calculated simultaneously in* $O(n \log n)$ *time.*

4. *The maximal allowable increases of all holding costs can be calculated simultaneously in* $O(n)$ *time.*
5. *The maximal allowable decreases of all holding costs can be calculated simultaneously in* $O(n^2)$ *time.*
6. *The maximal allowable increases of all demands can be calculated simultaneously in* $O(n \log n)$ *time.*
7. *The maximal allowable decreases of all demands can be calculated simultaneously in* $O(n^2)$ *time.*

[122] S. van Hoesel and A. Wagelmans. On the complexity of postoptimality analysis of 0/1 programs. Report 9660/A, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1996. Note: supersedes 1991 report, 9167/A, by same authors; to appear in *Discrete Applied Mathematics.*

> The authors address the cost-coefficient case of a combinatorial optimization problem, $\min\{cx\colon x \in X \cap \{0,1\}^n\}$, where $c \in \mathcal{Q}_+^n \stackrel{\text{def}}{=}$ non-negative rational values in $\mathbb{R}^n$. (X may, or may not, be representable by linear inequalities.) The first of their six propositions yields the result: under mild conditions, the existence of a polynomial method to solve the maximal ranges of each cost coefficient in a 0-1 program implies a polynomial method to solve the 0-1 program, itself. The first three propositions suggest the determination of one cost range that preserves optimality is computationally difficult (unless, maybe, $\mathcal{P} = \mathcal{NP}$). Proposition 4 is the proverbial nail (in the coffin) that says it is just as difficult to preserve the near optimality produced by a heuristic. Propositions 5 and 6 say it is also just as difficult to find an approximate range.

[123] B. Villarreal and M.H. Karwan. Parametric multicriteria integer programming. *Annals of Discrete Mathematics*, 11:371–379, 1981.

> This uses dynamic programming for the pre-determined right-hand side case of a multicriteria MILP.

[124] R. von Randow, editor. *Integer Programming and Related Areas: A Classified Bibliography 1981-1984*, volume 243 of *Lecture Notes in Economics and Mathematical Systems.* Springer-Verlag, Berlin, Germany, 1985.

This is a comprehensive bibliography with respect to covering the following topics: • theory and methods of general integer programming; • combinatorial and graph theoretical optimization problems related to integer programming; • applications of integer programming. The topic of post-solution analysis is absent as a category, but there are some citations related to postoptimal sensitivity analysis interspersed (and hard to find).

[125] A.P.M. Wagelmans. Sensitivity analysis in combinatorial optimization. Ph.d. thesis, Econometric Institute, Erasmus University, Rotterdam, The Netherlands, 1990.

The first chapter is a thorough literature review, taking more than half the thesis in its 107 pages. The scope of this review includes "easy" problems that can be solved by LP methods, like network flows. The bibliography is large and covers some feasibility problems, as well as postoptimal sensitivity analysis. The rest of the thesis, which contains new results, is divided into three chapters:

Speeding up machines in a two machine flow shop. This studies the minimum makespan on two machines as a function of speedup factors. The value function is piece-wise linear, but not convex. The author gives bounds on the number of breakpoints and on their location.

Sensitivity analysis of the economic lot-sizing problem. This pertains to the classical MILP model, with binary variables used to determine whether a setup occurs in each period. The solution is obtained with DP, and sensitivity analysis is concerned with changing production, holding and setup costs. The author uses the optimality of the partial path up to where the costs change, similar to what was done in [51].

Sensitivity analysis of list scheduling heuristics. This is concerned with how the quality of a heuristic solution is affected by changes in data. The particular (NP-hard) problem is the minimization of makespan on parallel identical machines. The heuristics are the *Shortest Processing Time* (SPT) and *Longest Processing Time* (LPT) rules. The solution depends upon an ordering of the processing times, and the author considers changing this for one job. This defines a response function that equals the heuristic makespan value as a function of one job's processing time. The heuristic response functions are shown to be piecewise linear. It is further shown that the slopes and breakpoint values of SPT can be computed in polynomial time, but there can be an exponential number of breakpoints for the LPT rule.

[126] S.W. Wallace. Is sensitivity analysis of any use? Technical note, Faculty of Social Sciences and Technology Management, Norwegian University of Science and Technology, N-7034 Trondheim, Norway, 1996.

This gives examples to illustrate the fallacy of using sensitivity analysis to deal with uncertainty. It is possible for a solution to be optimal for all values of a right-hand side in a MILP, yet that is not a solution to the 2-stage recourse model. The example is chosen such that the recourse model represents the definition of optimality, thereby making the usual stability conclusions fallacious (for the example).

[127] H-F. Wang and J-S. Horng. Directed perturbation analysis of an integer program. *Journal of Mathematical Analysis and Applications*, 201(2):447–460, 1996.

This is concerned with obtaining the one-parameter optimal value function for the right-hand side case of an ILP:

$$\min\{cx\colon\ Ax \geq b + \theta\delta b,\ x \geq 0,\ x \in \mathbf{Z}^n\}$$

for $\theta \in [0,1]$. They first show that the step function, $z^*(\theta; b, \delta b)$, is constant when $(w-1)d < \theta < wd$ for $w = 1, \ldots, 1/d$, where d is the reciprocal of the least common multiple of $\{|\delta b_1|, \ldots, |\delta b_m|\}$. For example, if $\delta b = \binom{100}{26}$, $d = \frac{1}{1300}$, and the only values of θ that are candidates to be breakpoints are those in $\{0, \frac{1}{1300}, \frac{2}{1300}, \ldots, 1\}$. Some other value cannot be a breakpoint because then $\theta\delta b$ has no integer coordinate. This set can be reduced, however, since the l.c.m. condition is necessary, but not sufficient. For example, $\frac{1}{1300}\binom{100}{26}$ has no integer coordinate, and $\frac{1}{100}$ is the least positive value of θ for which $\theta\delta b$ has an integer coordinate. (The l.c.m. condition is sharp; for example, if $\delta b = \binom{100}{25}$, $d = \frac{1}{100}$, and the candidate values of θ are all of $\{0, \frac{1}{100}, \frac{2}{100}, \ldots, \frac{99}{100}, 1\}$.) Focus is then on the monotone cases, $\delta b \geq 0$ (restriction) and $\delta b \leq 0$ (relaxation), improving upon the general one-parameter right-hand side case in [128].

[128] H-F. Wang and J-S. Horng. Structural approach to parametric analysis of an IP on the case of right-hand side. *European Journal of Operational Research*, 92(1):148–156, 1996.

This addresses the ILP, as in [127], and begins with the same fundamental theorems about changing θ. They also point out that the rule in [65] is also not exact for the right-hand side case with a simple example such that x^* is optimal at $b + \theta_1 \delta b$ and at $b + \theta_2 \delta b$, but fails to be optimal for some $\theta \in (\theta_1, \theta_2)$. (The example is on the web at http://www-math.cudenver.edu/~hgreenbe/myths.html.) They introduce *adjacent principal candidates* to prove a correct version of when optimality throughout the interval $[\theta_1, \theta_2]$ can be inferred. The *candidate* values of $\theta \in [0, 1]$, for which z^* changes, are $\{0, d, \ldots, 1\}$, where $d = \frac{1}{\text{l.c.m.}\{|\delta b_i|\}}$. The "principle candidates" are those breakpoint values of θ that ensure z^* remains constant between adjacent principle candidates, thereby retaining the optimality of $x^*(\theta_1)$ throughout the interval, $[\theta_1, \theta_2)$.

[129] G.M. Weber. Sensitivity analysis of optimal matchings. *Networks*, 11(1):41–56, 1981.

This is concerned with changing one edge weight in a matching problem. It shows how to use information from Edmonds' algorithm to solve the family of matchings more efficiently than solving each one independently, proving a speedup of $O(n)$, where n is the number of vertices.

[130] A.C. Williams. Marginal values in mixed integer linear programming. *Mathematical Programming*, 44:67–75, 1989.

This uses the LP framework to determine the continuity and directional differentiability of the optimal value function of a MILP, considering all data objects (rim and matrix values). After proving a necessary and sufficient condition for continuity of a MILP with bounded feasibility region, he notes special applications, such as TSP, Chinese postman, crew scheduling, and plant location.

[131] H.P. Williams. The economic interpretation of duality for practical mixed integer programming problems. In A. Prékopa, editor, *Survey: Mathematical Programming, Proceedings of the 9th International Mathematical Programming Symposium*, volume 2, pages 567–586, Amsterdam, The Netherlands, 1979. North Holland.

> This begins by attempting to use dual prices in an ILP the way one would in an LP. Using a small example of production shows the reader the difference, leading the author to give "motives for seeking the dual of an integer programming model." At this point, the dual is the Lagrangian dual, which is not strong, and the results are superseded by the author's more recent presentation [134].

[132] H.P. Williams. Constructing the value function for an integer linear programme over a cone. *Computational Optimization and Applications*, 6:15–26, 1996.

> This gives a recursive procedure for constructing the optimal value function, based on [13]. The assumed form of the constraints is $Ax \geq b$, $x \in \mathbf{Z}^n$, in which A is assumed to be $n \times n$, rational and nonsingular. In this case, either the ILP is unbounded for all b, or it is bounded for all b. (The former holds if there is no solution to its linear dual: $yA = c$, $y \geq 0$; otherwise, $cx = yAx \geq yb$ for any feasible x.) The "cone" is the translated one defined by the feasible region.

[133] H.P. Williams. Duality in mathematics and linear and integer programming. *Journal of Optimization Theory and Applications*, 90(2):257–278, 1996.

> This considers duality from a vantage of fundamental properties, such as reflexivity (i.e., involutionary property), and that each relation and operation in the primal has a correspondence in its dual. With several duals in different branches of mathematics (e.g., in projective geometry), the importance of LP duality in sensitivity analysis becomes a dominant consideration in evaluating ILP duals. Besides the Lagrangian, surrogates are considered as weak duals. The strong superadditive dual is presented with a numerical example, which shows the reader the structure of the value function for the right-hand side case [13, 132].

[134] H.P. Williams. Integer programming and pricing revisited. *IMA Journal of Mathematics Applied in Business & Industry*, 8:1–11, 1997.

This is an update of his earlier tutorial [131], beginning with an early method of imputing prices, due to Gomory and Baumol, which can be used for some kinds of sensitivity analysis, notably pricing a new activity. Those properties of LP that are preserved are: • *Activities in use make a zero profit*; • *No activity makes positive profit*; • *Any resource (good or service) with a zero price is free and can be increased arbitrarily without affecting the optimal solution*; • *No resource has a negative price*. Due to the presence of a duality gap, properties not preserved are given as: • *The optimal value of the outputs need not equal the optimal value of the inputs* (the difference in I/O values is the price of the integrality constraint); • *Marginal changes in the resource levels need not result in continuously changing solutions* (i.e., $f^*(b)$ need not be continuous). The author then closes with a comparative analysis using the strong superadditive dual, based on Chvátal functions [13, 136] (which were developed after his earlier tutorial).

[135] G.R. Wilson and H.K. Jain. An approach to postoptimality and sensitivity analysis of zero-one goal programs. *Naval Research Logistics*, 35:73–84, 1988.

This approaches the entitled problem by finding a set of k-best solutions (introduced earlier by Piper and Zoltners). The goal program here is viewed in the context of multiple objectives, where goals are used for trade-off analysis in obtaining Pareto optima. Individual objective coefficients and right-hand sides (goal values) are the parameters, and the response is the invariance of the set of k best solutions. Ranges of each datum are derived using rules that ensure the invariance, but the range need not be exact – viz., a right-hand side can exceed the derived upper limit, yet the k best solutions could still be the same.

[136] L.A. Wolsey. Integer programming duality: Price functions and sensitivity analysis. *Mathematical Programming*, 20(2):173–195, 1981.

This is a comprehensive treatment of the entitled subjects, using the superadditive dual to provide sensitivity information much like the LP dual does. Let x^* be a primal solution (there are no y variables), and let F^* be a dual solution. Four simple cases are treated first:

1. If only the right-hand side changes, F^* is still feasible in the dual, so $f^*(b+\Delta b) \geq F^*(b+\Delta b)$. Equality holds if F^* is optimal for the dual of the perturbed problem, in which case a primal optimum is any feasible x that satisfies $F^*(Ax) = cx$.
2. If only the objective coefficient changes, x^* remains primal feasible, so $f^*(c + \Delta c) \leq f^*(c) + \Delta c\, x^*$. If $\Delta c \geq F^*(A) - c$, F^* remains dual feasible, so $f^*(c + \Delta c) \geq F^*(b) = f^*(c)$; if additionally $\Delta c\, x^* = 0$, x^* remains optimal.
3. If a new activity is introduced with objective coefficient c_{n+1} and matrix column A_{n+1}, $(x^*, 0)$ is primal feasible, and it is optimal if $F^*(A_{n+1}) \leq c_{n+1}$.
4. If a new constraint is introduced, x^* remains optimal if it is feasible. The extended dual function, $\overline{F}(a, \overline{a}) \stackrel{\mathrm{def}}{=} F^*(a)$, is dual feasible for the new problem, so $F^*(b)$ is a lower bound on the optimal value.

The study continues with some theorems of the alternative, useful for feasibility analysis, then focuses on some special structures.