Shedding some light on Mixed Integer Linear Programming

by Michael G. Lamoureux, PhD

3:30 pm Friday, December 2, 2005 ITC 317

Faculty of Computer Science, University of New Brunswick

Extended Abstract

In Applied Mathematical Programming by Bradley, Hax, and Magnanti (Addison-Wesley, 1977), the authors state a fact that is still true 28 years later:

"Mathematical programming, and especially linear programming, is one of the best developed and most used branches of management science."

Given a linear program, we have a large number of techniques to find solutions and multiple variations of a simple technique (simplex algorithm) that is expected to (,and usually does,) find an optimal solution in polynomial time. In comparison, the expected solution time of almost any other mathematical representation of an optimization problem is exponential. Moreover, once we have a solution to a relaxed linear program, we can use heuristic techniques to quickly find near-optimal (local minima) solutions to a mixed integer linear program in polynomial time as well.

Furthermore, a (mixed-integer) linear program is a well defined construction. It is simply a set of decision variables, a set of (linear) constraint (equation)s on those decision variables, and a (linear) objective function. The decision variables represent your questions, the constraint equations represent the (real-world) restrictions on valid answers, and the objective function is your end goal. [*1]

And if that is not enough to satisfy you, we have entire textbooks of models we can use to solve manufacturing, production, sourcing, profit, risk, diet, blending, transportation, routing, and even scheduling models.

However, this is not the whole story. What Bradley, Hax, and Magnanti do not come right out and tell you, like most of the authors who follow them, is that this statement is really only true from a mathematical perspective. Given a mixed integer linear program, we can solve it very efficiently, but given a new management science problem, we are still more-or-less in the dark with respect to how to formulate it as a (mixed integer) linear program. [Brad Honnef's implication that model building is often done by a wizard is the best illustration the author has ever seen on the state-of-the-art of modeling for mathematical optimization. [*3]] If our problem does not match a model that has already been constructed, we usually have to start from scratch - and all we have to fall back on are generic problem solving methodologies, such as:

(1) understand the problem (2) identify the decisions (3) identify the constraints
(4) postulate a model (5) analyze the model, if valid, implement else repeat and refine

Considering that even Bradley, Hax, and Magnanti noted that (almost) all of "[mathematical programming] concerns the optimum allocation of limited resources among competing activities, under a set of constraints imposed by the nature of the problem being studied", one would expect that there should be an extensive inventory of common techniques to model and solve mathematical programs, especially if we restrict ourselves to (mixed integer) linear programming.

However, if you peruse the state-of-the-art, there is not. Recently, there has been a considerable amount of research in common model representations to allow problems to be shared between solvers (for example, the COIN-OR effort that was managed by IBM until recently), and a lot of research in mathematical modeling languages (and algebraic modeling languages in particular, such as GAMS) to allow models to be expressed quickly and easily, but very little research in common modeling paradigms or common model frameworks.

This leads one to ask - especially given the state of advancement in modeling in certain domains of computer science (database models, object models, patterns, etc) –

Does it have to be this way?
NO!

We can do better. We might never be able to develop a methodology that will solve all of our problems (that is just the nature of mathematics and modeling), but we can certainly make progress. What kind? You'll just have to give up part of your Friday afternoon to find out. [*2]

[*1] In a manufacturing problem, where you are trying to determine if you should build cars, trucks, or both you might have a decision variable that represents the number of cars you are going to build, a decision variable that represents the number of trucks you are going to build, a set of constraints limiting the number of cars and trucks you can build based on available raw materials (e.g. metal, rubber), and an objective to maximize the overall expected profit based on the profit associated with each vehicle.

*2 The author is not promising you all the answers, but it is highly probable that you will walk away with a few insights and a few interesting problems.

*3