

Twenty Years of Strategic Distribution System Design: An Evolutionary Perspective

ARTHUR M. GEOFFRION

*Anderson Graduate School of Management
University of California
Los Angeles, California 90095*

RICHARD F. POWERS

*INSIGHT, Inc.
PO Box 1609
Bend, Oregon 97702*

Using optimization to design distribution systems became technically feasible a little more than two decades ago, and developments have occurred at a rapid rate ever since. These developments can be understood in terms of six evolutionary processes. Four are core: evolution of algorithms, data development tools, model features and software capabilities, and how companies actually use software for designing distribution systems. The other two are environmental: evolution of logistics as a corporate function and of computer and communications technology.

The strategic problem of designing distribution systems has been important since not long after the dawn of the industrial revolution, but only in recent times has it become possible to design complete distribution systems that are truly optimal for all practical purposes. This is now true for even the largest companies.

We have been designing distribution systems since 1970 when we began developing and applying optimization technology for multi-product problems. We were

deeply involved in the main periods of this application domain since that time: the optimizer-centric period of the 1970s, the data-centric period of the late 1970s through the late 1980s, and the present user-centric period. Our experience has been both academic and commercial and includes projects with the federal government and more than 50 companies. In most cases, we have been able to reduce distribution costs by five to 15 percent while maintaining or improving customer

Copyright © 1995, Institute for Operations Research
and the Management Sciences
0092-2102/95/2505/0105\$01.25
This paper was refereed.

PROGRAMMING—INTEGER—APPLICATIONS
INFORMATION SYSTEMS—DECISION SUPPORT SYSTEMS

service. An important side benefit has been that the very act of building a comprehensive model has helped most of these organizations to understand their logistical dimension more profoundly.

Involvement over such a long period teaches lessons of a different sort than does involvement over a few years. In summarizing this long experience, our objective is to share our historical perspectives and the lessons most likely to interest others.

The standard problem we address is to find a minimal-annual-cost configuration of a company's production and distribution network that satisfies product demands at specified customer service levels. Input data consist of aggregated product and customer lists, facility data for plants and stocking points (processing rates, costs, capacities, and locations), transportation options and rates for all applicable lanes, demand forecasts, and various policy considerations, such as shipment planning rules, customer service requirements, and stocking point inventory restrictions. Desired outputs include answers to such questions as these:

—How many stocking points should there be, and where should they be located? Should they be owned?

—Should all stocking points carry all products or specialize by product line?

—Which customers should be served by each stocking point for each product?

—Where should the plants be located?

—What should each plant produce and how much?

—Which suppliers should be used and at what levels?

—What should the annual transportation

flows be throughout the system? Should pool points be used, and if so where should they be?

These questions can be visualized in terms of Figure 1, which portrays a company with two kinds of stocking points: plant warehouses and field warehouses.

Logisticians and logistics executives generally agree that these questions cover the most important strategic logistical issues [Byrne and Markham 1991]. For further details, see Geoffrion, Graves, and Lee [1982] or Geoffrion and Powers [1980].

Evolutionary Perspectives

In this paper, we view the strategic design of distribution systems over the last 20-plus years in terms of six evolutionary processes:

- (1) The evolution of logistics as a corporate function;
- (2) The evolution of computer and communications technology;
- (3) The evolution of algorithms;
- (4) The evolution of data development and management tools;
- (5) The evolution of model features and software capabilities; and
- (6) The evolution of how companies actually use software for designing distribution systems.

The first two are powerful historical currents that have swept along everyone who designs distribution systems. In contrast, we had considerable opportunity to influence the other four processes.

All of these processes have produced surprises compared to our expectations of more than 20 years ago. We did not predict nor anticipate

—The remarkable elevation of logistics' role in corporate organization and strategy;

DISTRIBUTION SYSTEM DESIGN

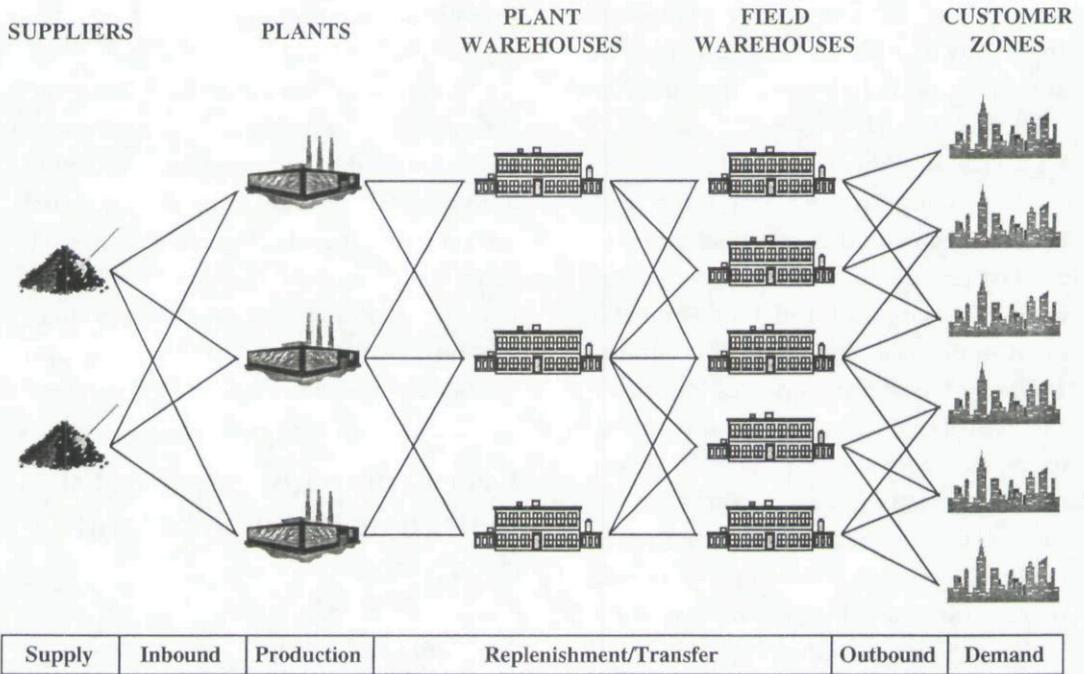


Figure 1: This typical logistics network includes several suppliers, plants, and two kinds of stocking points: plant warehouses and field warehouses.

- The devastating impact of desktop computers on user acceptance of mainframe implementations;
- The amazing longevity of technically inferior solution techniques;
- The lack of new algorithmic breakthroughs since the mid-1970s;
- The huge proportion of effort required for the development of data and related tools compared to that for the implementation of optimization technology; and
- The astonishing variety of real management issues that would parade before us under a banner that originally read “warehouse location.”

The six evolutionary processes are not all independent; some mutual impacts are

- 1 → 4, 5, 6
- 2 → 1, 3, 4, 5, 6

- 3 → 5, 6
- 4 → 5, 6
- 5 → 4, 6
- 6 → 1, 3, 4, 5.

We comment occasionally on these influences in what follows, but our main concern is with interpreting the importance and significance of each evolutionary process and with culling the lessons they teach.

Evolution of Logistics as a Corporate Function

The corporate status of logistics has changed dramatically during the last two decades. Within many companies, it has gone from a neglected and disdained function to a highly visible one respected for its profit impact and key strategic role. This

transformation has been commented upon frequently and has been documented by surveys, especially those done annually at the Ohio State University (for example, La Londe and Masters [1994]).

Masters and Pohlen [1994] write specifically on the evolution of logistics during the last few decades. They discern three evolutionary phases: functional management of physical distribution (1960s and 1970s), internal integration of logistics functions (1980s), and external integration of logistics between firms (1990s). Their observations are generally consistent with those presented in other publications, including a series of major studies conducted by A. T. Kearney, Inc., the most recent of which is Byrne and Markham [1991].

The first phase was marked by the emergence of

- (1) Materials management as an integrated approach to managing materials that flow into an organization (for example, managing purchasing, raw materials, inbound transportation, and work-in-process inventory control), and
- (2) Physical distribution as an integrated approach to managing the movement of finished products to customers (for example, managing demand forecasting, inventory control, order processing, outbound transportation, warehousing, and customer service).

Corporations reorganized to bring fragmented functions under more centralized managerial control. The managers involved rose in the corporate hierarchy and in educational attainment. This created conditions favoring the development of such professional associations as the Council of Logistics Management, associations that

provide an important vehicle for propagating OR/MS technology.

According to Masters and Pohlen, this phase was powered by growing acceptance of the concept of total cost analysis, by the emergence of simulation models of distribution systems and other computer applications such as data analysis, distribution resource planning, and material resource planning (MRP) that addressed important managerial decisions, and by the economic

Every company pays for the inefficiencies up-chain and down-chain.

necessity to exploit the opportunities for reducing costs that these developments made possible. Most OR/MS professionals would find the first factor surprising, as this community has taken it for granted for half a century, but its relatively recent introduction in logistics was a crucial event that enabled proper trade-offs among the cost consequences of traditionally separate functions [Stock and Lambert 1987].

Customer service became prominent during the latter part of this phase, with the result that more management attention was focused on the trade-off between cost and service. In addition, the deregulation of the transportation industry, the OPEC oil shock, double-digit inflation, and the emerging environmental movement caused companies to rethink many logistical issues.

Masters and Pohlen's second evolutionary phase (1980s) was marked by the effective integration in many organizations of materials management, physical distri-

DISTRIBUTION SYSTEM DESIGN

bution, and a few functions previously performed by marketing and manufacturing into a single all-encompassing managerial responsibility for integrated logistics. Naturally, executives carrying this responsibility rose still higher in the corporate hierarchy and needed still more formal education [La Londe and Masters 1994]. Many top logistics jobs were created at the vice-presidential level.

Among the factors that helped make the logistics executive's job more complex during the 1980s were

- The challenges associated with rapidly expanding international operations;
- The emergence of third-party logistics providers offering to take over many functions traditionally performed in-house;
- The need to exploit rapidly developing computer and communications technologies for logistical applications (for example, business process automation, electronic data interchange, and bar coding);
- The trend toward stockless retailers and just-in-time (JIT) production and distribution;
- The craze for mergers, acquisitions, and buy-outs that often resulted in the integration of previously separate logistics operations;
- The restructuring of many companies to trim their labor forces (including middle management) and reduce their asset bases;
- The new emphasis on organizational flexibility and reduced cycle times; and
- The mandates of the quality movement.

Whereas the first two phases looked inward within the firm for functional management and integration, Masters and Pohlen's third evolutionary phase of the 1990s (external integration of logistics be-

tween firms) looks outward from a company to opportunities for logistical coordination with other firms in the supply chain(s) of which the company is a part. The fundamental observation justifying this approach is that, one way or another, every company pays for the inefficiencies of its suppliers up-chain and its customers down-chain.

The objective of intercompany coordination is to reduce costs, reduce risks, and leverage the resources of each supply chain member. This coordination can take many forms and usually is cooperative rather than manipulative. Often the result is a mutually beneficial logistics partnership or strategic alliance. A few examples: well-informed product and packaging designs can enable more economical handling, storage, and transportation; sharing a company's sales forecasts and new product intentions with a supplier can help the supplier plan for future requirements; and collective planning of asset investments (for example, information systems) on the interface between two firms can avoid redundancy. Bowersox et al. [1989] and Byrne and Markham [1991] give many other examples.

The dramatic transformation of logistics as a corporate function has provided fertile ground for computer-based logistical applications. Without the demand pull it created, the supply push from academia and consulting firms would have sputtered along with little effect.

Evolution of Computer and Communications Technology

The rapid progress in computer and communications technology since the late 1970s is evident to all, and its impact on

the design of logistics systems has been profound.

La Londe and Masters [1994, p. 7.3] describe it as follows in a paper on the evolution of logistics since 1970:

Perhaps the most pervasive change in the business environment during this period was the flowering of information technology which occurred. In 1970, logistics organizations used information systems on mainframe computers . . . Processing was batch-oriented, data storage was accomplished on magnetic tape, data input was via keystroke, and outputs were often voluminous paper reports which inundated the manager with a great deal of data but very little useful information. Changes to the system software were expensive and slow and required the efforts of a legion of programmers and systems analysts. It was very difficult to integrate the operation of data systems within a firm, and integration between firms was very rare. By 1990 the world had really changed. Information processing was based on desktop machines, networked to one another and perhaps to a mainframe. Machines were small, sturdy, reliable, faster, more powerful, and vastly cheaper. Systems processing took place in real time, mass data storage was via disk with near instantaneous retrieval, raw data input was accomplished by automated scanning devices, and output reports were presented on video screen. User-friendly software and relational data bases were widely available which allowed a manager to selectively retrieve relevant information and design tailored reports. Electronic data interchange telecommunications protocols allowed direct interaction and machine-controlled business transactions between corporations around the world. In the traditionally information intensive activities of the logistics process, these new capabilities paved the way towards totally new ways of doing business.

To round out this portrait, certain other changes require mention:

- Character-based user interfaces have been replaced by graphical ones (GUIs), now a requirement for most business application software, and there is a strong trend toward visual information displays.
- The price/performance ratio of desktop

computing equipment has improved rapidly, with mass semiconductor memory and high processing speeds becoming so affordable that even the largest logistics optimization problems now can be solved on the desktop.

—The price/performance ratio of personal communications equipment has also improved rapidly; for example, today's modems are more than 10 times faster than those of a decade ago, have better error correction and compression features, include fax capabilities, and cost less in real terms.

—The monopoly of corporate data processing and MIS departments has been broken on many of the data and computing resources needed for designing logistics systems. Some of the logistics executives we have worked with were so disenchanted with their central computing departments that they would have turned elsewhere if we had not provided alternatives. Highly centralized administration of data and applications during the 1970s gave way to a comparatively anarchic situation during the 1980s when end users rushed to PCs that were often stand-alone or on a LAN that did not communicate well with the mainframe world—either way, they often bypassed the central administrative apparatus. The price of this freedom frequently included partially redundant database and spreadsheet applications that tended to fall into disagreement, with much ensuing confusion.

—There has been a strong trend toward client-server architecture with improved mainframe access. This renders central administration of data and applications possible once again in organizations that place

servers under central control, and it has begun to change the design of logistics software.

—Electronic data interchange (EDI) and data capture technology (bar coding systems, hand-held data entry terminals) for logistics applications have been proliferating [Bowersox et al. 1989; Byrne and Markham 1991; Gopal and Cypress 1993]. —Desktop software development environments and off-the-shelf components have emerged that are far superior to their mainframe predecessors.

The importance of these dramatic developments in computers and communications has not been lost on senior logistics managers. An annual survey of logistics executives shows that, between 1972 and 1992, information technology replaced logistics as the most favored topic for executive education programs [La Londe and Masters 1994]. Another major survey [Bowersox et al. 1989] showed that companies on the "leading edge" of logistics are distinguished by their comparatively frequent application of information technology, and still another [Byrne and Markham 1991] showed that participating companies expected greater future productivity improvement in information systems than in any other logistics activity.

Our own software for designing logistics systems was almost completely mainframe-oriented until the mid-1980s. INSIGHT was among the first to get large-scale optimization software running on a PC [Bausch and Brown 1988]. But because of the massive data requirements inherent in such applications and because of the time it took to make our many data development tools accessible via a graphic user

interface, our commercial software was not available for desktop use until 1992. Desktop computing has now almost completely supplanted mainframe computing for development work and production runs.

By the late 1980s, computer-to-computer communications had become important to INSIGHT: between our offices, between our offices and contract developers and consulting associates, and between our offices and many clients. It reduced travel, improved coordination in dispersed workgroups, and improved service through remote software installation, troubleshooting, debugging, and results analysis.

Telecommunications also have helped the companies we work with. People in the field upload data files to model builders, and many companies find it convenient to run their models remotely.

Evolution of Algorithms

In 1970, the following computational approaches were in commercial use for designing distribution systems [Geoffrion 1975; Bender 1985a]:

Nonoptimizing cost calculators simply compute total annual cost and its various components for a succession of manually determined distribution system designs. (Compare this with the modern practice of using spreadsheet software for purely evaluative purposes.) The aim was to improve on the current or leading proposed design or to compare specific design alternatives. For convenience, we include under this approach dynamic simulation, which also deals with one fixed design alternative at a time, enabling the user to make a very detailed evaluation.

Heuristic methods were popular because they allowed designers to "deal with" al-

most any desired model feature and to control execution speed and memory requirements. But ironically, some of the heuristic methods of this era actually were slower than optimizing methods for the same model class, because heuristic methods tended to be less sophisticated in conception and implementation. One could show that improved designs obtained via heuristics were indeed better simply by costing them out against the current or leading proposed design. Of course, this did not insure against the existence of still better designs.

Casewise linear programming was often used in the absence of a computationally adequate mixed-integer programming optimizer; many practical studies resorted to repeated runs of a general purpose LP code or of a minimal cost network flow optimizer. Each run locked in a manually determined logical configuration consisting of specific choices for all of the discrete decisions that would be modeled via 0-1 variables in an integer programming formulation. The aim was to improve on the current or leading proposed logical configuration or to compare specific logical configuration alternatives.

Specialized optimization packages mitigated the shortcomings of commercial integer programming software to some extent by exploiting special problem structure, such as a single stage of distribution or a single product. Branch-and-bound typically provided the organizing algorithmic framework.

Commercial mixed-integer linear programming (MIP) software was appealing because it promised realistic models, but such software was awkward to use unless

special application-specific front- and back-end modules were written, and computational performance seldom was adequate for problems of realistic size.

In addition, 1970s designers used heuristics in connection with the last three approaches in an attempt to reduce computing times or memory requirements at the expense of strict optimality and to permit additional model features not within the scope of optimization.

We began our own work during 1970 with a project at Hunt-Wesson Foods, and

One would expect the gradual demise of nonoptimizing approaches.

in 1971 we found an essential improvement that made a variant of Benders' primal decomposition approach the optimization method of choice for models within its scope [Geoffrion and Graves 1974; Geoffrion 1976a].

Many well-known algorithmic advances in optimization have been made since those days, but it turns out that most have not been important for designing distribution systems. For example, the new interior point methods are of little use because they are not well suited to the task of solving many related LPs, a principal requirement of branch-and-bound [Schultz and Pulleyblank 1991]. Only two developments have been of fundamental importance to this application domain:

Primal network simplex algorithms emerged in the 1970s as a revolutionary advance in the state of the art for minimum cost network flow optimization

[Ahuja, Magnanti, and Orlin 1993]. Improvements have been made continually since then in other approaches as well as in the primal network simplex method.

Following the seminal work of G. Dantzig and R. Van Slyke [1967] on exploiting generalized upper bounding (GUB) structure, the 1970s saw substantial progress on other structure-exploiting factorizations of the essential information needed to execute LP iterations. Most important among these were factorizations exploiting network-related structures; Brown and Olson [1994] give a historical review. This meant that larger LPs could be solved within available computational resources, especially if they had certain special structures.

The speed improvements and memory reductions achieved by the primal network simplex method soon benefited software for distribution system design that made use of network optimization. Our software assimilated this technology in 1976 when we embedded GNET [Bradley, Brown, and Graves 1977] in our code following earlier use of a factorization method that was an order of magnitude faster than our previous out-of-kilter implementation for transportation models.

The new factorization technology also found its way into practical use promptly, since the LP engine is crucial in most optimizing approaches to logistic system design. Our software first incorporated a version of GUB factorization in 1974 [Graves and McBride 1976], and in the late 1980s, we incorporated factorization for embedded network structures into our main optimizer [Brown and Olson 1994]. This made it practical to solve models with

more than three distribution echelons and arbitrarily complex linear constraints linking commodities, thereby overcoming serious implementation restrictions of our Benders decomposition software that we are only now removing. Very recently, new advances in factorizing embedded network structures have yielded remarkable computational improvements that already have proven valuable for an immense logistics model (about 550,000 constraints and 7,000,000 variables) [Mamer and McBride 1994].

In addition to exploiting these advances of general interest, we also introduced specialized innovations in response to the increasingly complex problems we encountered in practice. These include a "double" version of primal (Benders) decomposition in the mid-1970s to cope with a very large Department of Defense study [Hall 1980] and implicit elastic treatment of lower and upper distribution-center-throughput capacities at about the same time.

Meanwhile, in keeping with the move to desktop computing, a constantly growing proportion of the organizations we have worked with have been choosing to install our software in-house. A major implication is that, with end users in full control of their own models, we have had to make our optimizers far more robust than they were when we ourselves performed most optimization runs. This has had a major impact on our algorithmic development program, including introducing a bias toward factorization (on machines with a lot of main memory) in preference to decomposition technology.

Given this history of algorithmic progress and the compelling arguments for

optimal rather than heuristic methods for this type of application [Bender 1985a; Geoffrion and Van Roy 1979; Powers 1989; Shapiro 1985], one would expect the gradual demise of nonoptimizing approaches in actual practice. But if this is happening, the process is agonizingly slow. All the approaches mentioned earlier remain in commercial use, except two specialized optimizers, which have been replaced by commercial general-purpose MIP codes, and the most primitive forms of casewise linear programming, which nowadays incorporate heuristic routines to generate promising logical system configurations. In fact, some of the most popular commercial software packages use heuristics in a way that sacrifices even optimality bounds. According to Ballou and Masters [1991, p. 172], "over 75 percent of the firms that have purchased location models have selected heuristic-based approaches." The market has even seen the reintroduction of commercial software based on center-of-gravity models, known for many years to be hopelessly unrealistic for all but the most trivial applications.

This situation provides a lesson in humility to the OR/MS profession. The factors contributing to this situation include the following:

First, the computer revolution is a rising tide that lifts all boats; heuristic software benefits from it just as much as optimization-based software does. Well-designed heuristic packages therefore can maintain their advantage over optimizing packages in terms of the computing resources required, a consideration unlikely to diminish in importance so long as the size and complexity of models arising in practice

continue to increase.

Second, optimization-based software for designing distribution systems has been slower than heuristic software to migrate from mainframe to desktop and slower to develop graphic user interfaces. Nowadays, logistics executives nearly always choose desktop analytical software over mainframe software and a GUI over an

This situation provides a lesson in humility to the OR/MS profession.

older style interface, even if it means accepting a heuristic rather than optimizing package.

Other factors include discomfort with "high" technology when easily understandable "low" technology appears adequate, inadequate managerial appreciation of the added benefits of optimization, difficulties in reliably obtaining optimal solutions using commercial optimization software, and the lingering memory of past failed attempts to do optimization.

Not only has optimization failed to become the dominant approach but, contrary to our early expectations, Benders decomposition has not become popular among those taking an optimization approach. In fact, ours may be the only commercially available software for designing logistics systems that uses Benders decomposition [Andersen Consulting 1994; Ballou and Masters 1991; Magnanti and Wong 1990].

Why? Perhaps the main reasons are economics and access. Making Benders decomposition work well requires a lot of technical expertise and implementation ef-

fort. It is easier to turn to off-the-shelf commercial MIP software even if its computational performance permits only sub-optimal solutions in practice.

In any case, we find that most real logistic system design applications require model features—most notably, single-sourcing of customer demands by product category—that are impractical to handle optimally at large scale without Benders decomposition or a structure-exploiting factorization.

Evolution of Data Development and Management Tools

By the mid-70s it had become clear that the lion's share of the work in optimizing logistics networks lay in developing the necessary data. Powers, Karrenbauer, and Doolittle [1983] estimated that 75 to 80 percent of all time and effort typically goes into preparing data.

The amount of data depends on the business being modeled, but even for small companies the amount needed is daunting. A small company doing business nationwide may have 100 products, 10 to 15 distribution centers, one to five plants, and 10,000 customers placing 200,000 orders per year. But even a network this small would require 500,000 to 1,000,000 freight rates. At the other extreme is the Department of Defense project we did in the mid-70's with over 3,000,000 stock items, 30 million orders per year, and thousands of sources. In this case the freight rates number well into the millions.

The Glidden project described by Powers, Karrenbauer, and Doolittle [1983] is typical of the network structures and data volumes encountered today in logistics network design studies:

- Thousands of customer accounts aggregated into 212 demand zones;
- Three separate types of ship-to customers, giving the model an effective number of 636 customers;
- 6.3 million demand transactions in a year;
- 3,000 individual product codes;
- 61 existing and potential warehouse locations;
- Seven owned plants and hundreds of vendor sources; and
- 1.2 million freight rates.

By 1980, when the Glidden project was carried out, we had already expended a great deal of effort developing data management tools for logistics network design. We turn now to the evolution of these tools. Our discussion is limited largely to our own experience. Others who regularly perform similar studies also have developed special tools, but little public-domain information is available. (However, see Shapiro, Singhal, and Wagner's [1993] discussion of SLIM and Palmer's [1984] discussion of PLATOFORM.) One reason for the paucity of information may be that, as a topic, data development is so burdened with application-specific details that it does not lend itself easily to research, articles, or software of general applicability. Another impediment is concern for proprietary interest.

Here is a developmental chronology of our tools for data development and management.

Our first system for distribution network design, created in 1971, gave the user no assistance in preparing the problem file required by the optimizer. The problem file, a collection of FORTRAN-formatted, card-

image data files, was readable but catered mainly to the optimizer's needs. Typically, the user prepared it partly by hand with a mainframe text editor and partly in batch mode on a mainframe using ad hoc programs. The optimizer took care of report writing.

By 1973, we had developed a bulk data preprocessor to facilitate preparing the bulkiest components of the problem file, namely the demand data and the list of permissible transportation links and their unit costs. We added data checking logic to the data reader, thereby beginning a long

The computer revolution is a rising tide that lifts all boats.

series of developments aimed at relieving the optimizer proper from the task of finding bad or missing data.

In 1975, we expanded the bulk data preprocessor into a comprehensive preprocessor able to prepare the entire problem file needed by the optimizer. This preprocessor accepted as input either a complete input data file (in creation mode) or an already existing problem file (in editing mode). We designed the input data file for readability and compactness (through abbreviations). The preprocessor ran independently of the optimizer and had these main functions:

- Input data file error detection, reporting, and recovery;
- Data extension (user-controllable defaults, data replication rules, and generation formulas) and modeling options (for example, criteria for inbound link inclusion);
- Generation of new problem files suitable

- for direct optimizer access;
- Editing of old problem files (delete, insert, revise, rescale); and
- Generation of summary reports (with options) on the problem file.

In 1980, we completed DATA-1, an elaborate front-end software module to prepare the preprocessor's input data file [Powers, Karrenbauer, and Doolittle 1983]. Its key innovation was permitting a user to provide a file of historical demand transactions for whatever period was being modeled (typically a year); this was used to calculate portions of the input data file (for example, demand data) via automatic customer and product aggregation applied to these actual detailed transactions.

In 1981, we launched SHIPCONS, a discrete event shipment planning simulator designed to mimic the behavior of an experienced traffic manager. It uses a historical demand transaction file, standard freight rate databases, and custom freight rates from the user. Multiple transportation modes, rate types, and shipment-planning policies (hold times, stop-offs, pooling, and so forth) can be accommodated to generate realistic shipment sizes and costs. Each simulation run produces extensive reports summarizing the simulated performance of the target distribution system and also the bulkiest section of the optimizer's problem file, namely the weighted average outbound freight rates. In addition, SHIPCONS can be used to establish the base-line cost for historical outbound flows needed for model validation and credibility. Also noteworthy this year was the incorporation of a standard georeference database into DATA-1 to permit automatic generation of distances, service areas, and

map displays.

In 1984, the optimizer and all data management functions previously implemented were subsumed and enhanced within a comprehensive integrated package called SAILS (Strategic Analysis of Integrated Logistics Systems). SAILS also offered new scenario-generation capabilities, elaborate menu-based report selection, a system log (computational progress, resource utilization, diagnostic messages), and the ability to build automatically complete aggregate models from publicly available demographic and commercial databases.

From 1985 to the present, we have continually enhanced the data management features of SAILS. Now included are additional standard databases (for example, the Yellow Freight Systems 500 tariff), reports on the inventory implications of various network structures, and a module that provides a very detailed comparison of the results of any optimized scenario with the model baseline or with any previous scenario. All of these features are now available with a full graphic user interface on a PC running Windows.

The road to easier data development has been long and arduous for us, with no foreseeable end. What lessons of general interest have emerged from this journey? One is that to make proper use of modern optimization technology for strategic logistical planning, one needs a far larger investment in data development tools than in software for optimization. One comparative measure is lines of custom code (mostly FORTRAN): in our case, the ratio is approximately eight lines of data development code for every line of optimizer code.

Another lesson is that data preparation, validation, and analysis offer very important benefits to a company quite separate and apart from the results that can be computed using these data (cf. [Shapiro, Singhal, and Wagner 1993]). For example, we find that companies often greet the purely descriptive analyses needed to prepare demand inputs to the optimizer as revelations.

A third observation is that there are major "database economies of scale" for this kind of modeling. Acquiring or developing reusable databases and integrating them into data development tools can be very expensive, yet such databases are essential for building accurate models with a reasonable amount of effort. Vendors have a great advantage over individual companies in this respect because they can spread costs over many clients.

Our final observation is that many important theoretical research topics in the area of data development are languishing for lack of attention. Vital topics include customer and product aggregation, the effect of errors in either real or regression-based transportation rates, and the consequences of using linear approximations to nonlinear warehousing costs (inventory costs are nonlinear). Ballou [1991, 1994] and Geoffrion [1976b, 1977] and others have written papers on these topics. Good theoretical advice could have a substantial effect on future practice, which must wrestle with such questions in nearly every application.

Evolution of Model Features and Software Capabilities

If someone had told us at the outset that our apparently complete software for de-

signing distribution systems would undergo vigorous evolution for more than two decades, we would not have believed them. Yet this is exactly what happened. Clearly, we did not appreciate the powerful forces that would drive this evolution.

One such force was the rapid progress in computer and communications technology. We commented on this previously, but we elaborate here on one particularly impor-

The road to easier data development has been long and arduous.

tant aspect, namely the powerful influence of desktop computing on user expectations for logistics software.

Generally speaking, desktop application software is not restricted to lease-only and is much less expensive than mainframe software, has a colorful and relatively easy graphic user interface, and is nicely documented and packaged. Few users would choose a mainframe application package over a desktop package with even barely adequate functionality.

It would be difficult to exaggerate the significance of the radical shift in user expectations during the last decade. This shift made it imperative to move our mainframe systems to the desktop, which was not a small undertaking.

A second force that drove the evolution of our software, much weaker than the first, was external algorithmic progress (discussed previously).

A third driving force was our desire to respond to client requests. Typically, a successful engagement—either performing a

study for a client or installing our software on their machine and working with their people—leads to requests for enriching the standard model in various useful ways. These are a few examples of requests for model features to be added:

- Customer-specific product subsets that are single-sourced, a relaxation of the requirement that all products demanded by a customer must come from one distribution center;
- Lower as well as upper limits on the shipments of a given product at a given plant;
- Product-specific weighting factors for distribution-center throughput measures;
- User-controllable elastic treatment of distribution-center throughput limits;
- Piecewise linear approximations to non-linear distribution-center costs;
- The ability to locate plants as well as distribution centers;
- Joint capacity constraints across products at plants;
- Raw material conversion activities at one or two echelons; and
- Additional distribution and production echelons.

We have solved models with all of these features. Geoffrion, Graves, and Lee [1982] give details on the first five items, which we dealt with efficiently within the Benders decomposition framework set forth by Geoffrion and Graves [1974]. In contrast, generally we have found it more practical to use structure-exploiting factorization to deal with the remaining items on the list, as decomposition proved until recently to be somewhat fragile with respect to such model extensions.

Software installations also lead to re-

DISTRIBUTION SYSTEM DESIGN

quests for enhancements other than new model features, typically to improve ease of use. These are a few examples of implemented requests:

- Arbitrary monetary and physical quantity units (essential for international models);
- Easy selection of transportation links for inclusion via customer proximity thresholds;
- Easy scaling and substitution options for most data;
- Easy masking (in or out) of any model entity;
- Spreadsheet data import facilities; and
- “Live” map displays with zoom capability.

Most of these demands for new model features and software capabilities derive ultimately from the maturation of logistics as a corporate function. The increasing modeling demands simply reflect the increasing complexity and functional scope of the issues facing logistics management and the rising sophistication of logistics executives who better appreciate the need for logistics data and modeling.

These factors have been the fountainhead not only of requests to extend our software, but also of requests to integrate strategic design models with tactical planning models (for example, for inventory and production planning) and with operational distribution models (for example, vehicle scheduling and routing). In response, we are in the process of complementing the tactical SHIPCONS module (with its dynamic simulator for shipment consolidation) with an operating-level system called SHIPCONS II for optimizing shipment consolidation and transportation plans at a

daily level of detail. In addition, we have completed several strategic and tactical applications combining distribution and production planning [Brown, Graves, and Honczarenko 1987; Arntzen et al. 1995].

A fourth force that drove the evolution of our software was the desire to expand software coverage to more of the modeling life-cycle, which has many phases before and after optimization proper (Geoffrion, Graves, and Lee [1982] discuss this life-cycle in the distribution planning context; also Geoffrion [1989]). In the following list of life-cycle phases, the details depend on whether an application is a one-shot study or is embedded in an ongoing business process. An ideal modeling environment should support most of these phases in an integrated way that facilitates passing output from one phase to the next and that facilitates the necessary looping and iterating.

- (1) Recognize a modeling opportunity.
- (2) Analyze the requirements and feasibility, and negotiate the project charter.
- (3) Finalize the model structure, organize a project team, and make a detailed project plan.
- (4) Develop and verify data (also acquire or develop pertinent software tools).
- (5) Prepare the solver and customize it as necessary.
- (6) Verify and validate the instantiated model(s) and solver.
- (7) Do optimization runs (after installation and training if an embedded application).
- (8) Analyze the results.
- (9) Report and explain the findings and conclusions (for a one-shot study) or explain the operational performance

(for an embedded application).

- (10) Document the model, solver, data development, and analysis.
- (11) Maintain, update, and improve.
- (12) Re-evaluate everything periodically and eventually terminate or make a transition to a new model or solver (for an embedded application).

Commercial software for designing logistics systems must support those phases primarily concerned with optimization: 5, 6, and 7. But such systems differ widely in their support for the other phases. We have always believed that the more of these other phases our software can support, the more useful it will be. We have concentrated largely on the immediately adjacent phases: 4, 8, and 9.

We have already described the evolution of our software's data-development capabilities. Our results analysis and reporting features have also evolved:

- We have improved the existing reports and developed new ones that bring the current total to nearly 300.
- We have interfaced our system to commercial mapping software so that we can produce reports that facilitate visualizing the configuration of distribution systems (for example, Atlas Pro and Atlas GIS from Strategic Mapping, Inc.).
- We have developed a post-optimal comparator so that we can easily contrast an incumbent optimal solution with a baseline solution or a previously computed solution.
- We have made optimal solutions exportable to files so that end users can further analyze them using their own software, and graphics and reports are now copyable to the Windows clipboard for subsequent

pasting into other Windows documents.

We still face many challenges in phases 4 to 9, but we hope eventually to add some support for the other phases as well.

The four forces that drove the development of our software can best be understood in the context of the competitive market for consulting services and software for designing logistics systems. This market has become increasingly competitive, with a large increase in the number of companies offering consulting services in this area and an increase in the number of commercial software packages specifically for this purpose from a handful in 1970 to about 20 today [Andersen Consulting 1994; Ballou and Masters 1991].

Competition leads to price reduction, product differentiation, and creative efforts to meet client needs. Generally the client benefits, but not always. Some vendors have devoted most of their effort to making visually attractive user interfaces, neglecting functional content.

Evolution of How Companies Actually Use Distribution System Design Software

Just as hardware and software have evolved over the past 20 years, so have the ways users apply distribution system design packages to their business decisions. They have found new ways to use these packages beyond the originally envisioned uses.

To illustrate, we consider four companies that have used the SAILS package for seven to 12 years. Each company originally acquired SAILS to address issues associated with the classic warehouse-location problem, but over time they developed a widening variety of new applications

DISTRIBUTION SYSTEM DESIGN

[Geoffrion and Powers 1980].

Baxter Healthcare Corporation began using the package in 1980 to resolve classical distribution network strategy issues and successfully identified a number of cost-effective changes. In 1985, Baxter acquired the American Hospital Supply Corporation. The combination of two complete national networks greatly increased the number of facilities in use and the number of SKUs in inventory. Baxter used SAILS over a two-year period to evaluate many different structures for consolidating its assets into a "One Baxter" network that would serve the hospital market.

In 1988, Baxter established the Network 2000 Program to implement the results of the consolidation studies. As part of this initiative, Baxter distribution teams formulated prototypes of four different distribution center designs differing by size and the use of automation for order-picking. They then used SAILS to define the optimal network from alternatives of new, technologically advanced facilities versus expanded and enhanced existing facilities. They formulated the cost factors to optimize the net present value of the investment and operating costs over an extended time period.

This process yielded a refined network design, and the analysis results were directly compatible with the capital analysis procedures used in requesting funds for facility projects. Since 1988, Baxter has invested over \$300,000,000 in new distribution centers to meet its customer requirements in a logistically and financially cost-effective manner.

Pet, Incorporated started using SAILS in 1982. Initially, the company saw it simply

as a strategic tool to ensure minimal-cost alignment of the distribution network, and it was used only periodically. Then, in 1984, Pet was considering the acquisition of another food company, Progresso. It used the package to evaluate how Progresso's product line could best be integrated into a single distribution network and to evaluate the synergies of jointly handling the sale and distribution of the two product lines. Pet repeated this same process in 1989 when it acquired Van de Camp Foods and had to create a still larger and more diverse integrated distribution network.

Pet's vice-president for logistics, David Tarr, says that Pet now uses the package more in a tactical mode than for strategic analysis, the purpose for which it was acquired. According to Tarr, "Today we use SAILS to evaluate every proposal about product shifts among plants, capital investments, new product introductions, or fundamental changes in operations." He adds that the institutionalized use of this package has had a big impact on the kind of organization Pet is and the kinds of skills Pet looks for in the people it hires.

Clorox Company began using the package in 1984. After spending some time rationalizing its distribution network, Clorox analyzed new product introductions and how those products should be manufactured and distributed based on raw material supplies. In 1990, the company evaluated the distribution synergy and implications of acquiring Pine Sol and integrating the distribution of Pine Sol with the bleach business. Partly because of the SAILS analyses, John Leggett of Clorox said the acquisition and integration of Pine Sol

went more smoothly than anyone expected.

Clorox is currently using the package to evaluate the economics and business implications of setting up alternate distribution channels for nonbleach products. Activity-based costing is emphasized increasingly because, when all products go through a common channel, costs figured as averages across all weight moved can be misleading owing to great disparities in volume between bleach and other products, such as foods. SAILS is helping Clorox to clarify the activity-based costs of distributing diverse products through common and separate channels.

Eastman Kodak has been one of the most creative SAILS users since 1986, with a variety of applications in several business units. Recent uses include "reverse logistics," "focused factories," and strategic alliances.

With the increased emphasis on the environment and recycling, Kodak has used the package to evaluate how best to collect spent film products from film processors and get them back to recycling facilities. To accomplish this recycling project, Kodak inverted the usual model, with customers becoming suppliers and the recycling facilities becoming the customers.

In looking at options for global manufacturing and distribution, Kodak found a way to use SAILS to evaluate the creation of "focused factories," which specialize in producing selected products in various parts of the world. An important cost factor in analyzing the global flows of such products is the duty charged on entering a given country. But if value is added in the country and the more complex product is

then exported, a duty "drawback" or credit can be realized. These duties and drawbacks can have profound influence on the economics of how and where in the world a company manufactures its products.

In another recent application, Kodak evaluated a strategic alliance between Sterling Winthrop, a Kodak unit, and Sanofi, a large French pharmaceutical company, for the distribution of complementary phar-

Companies often greet the purely descriptive analyses needed as revelations.

maceutical products throughout Europe and the Far East. The two companies together had 21 distribution centers in Europe. They cut that number in half following the use of SAILS, with both companies' products flowing through the same distribution network.

These examples are typical of the diverse ways in which users have adapted SAILS to cope with rapidly changing market environments and new technologies for manufacturing, distribution, and information processing. We expect that this evolution of new applications will continue. Strategic alliances, product proliferation, market segmentation, environmental concerns, and the global marketplace will continue to test the ingenuity and skills of logistics planners and decision makers in the future.

The Future

Our review of more than two decades of experience with the strategic design of distribution systems offers clues to the future as well as historical insights. We present

now our expectations based on these clues, together with a few brief summary comments.

Logistics as a Corporate Function

One could say that the dominant theme in logistics for the last three decades in corporate America has been one of increasing integration: during the 1960s and 1970s a grouping of many individual management functions under materials management and many others under physical distribution; during the 1980s a grouping of those two management clusters under integrated logistics; and during the 1990s an increase in intercompany coordination.

From the OR/MS perspective, this increased integration sets the stage for modeling applications that deal with the many new trade-offs and choices that the logistics executive must face. Indeed, what success we have had in designing distribution systems might not have been possible without the receptivity created by this need.

Logistics managers will continue to take an ever more comprehensive perspective, because that is where the greatest opportunities for managerial improvement lie. OR/MS practitioners will therefore need to create ever more comprehensive logistics models. Additional challenges will arise from the factors that made the job of logistics executives more complex during the 1980s, for nearly all of those factors still operate. Moreover, there are new pressures such as reengineering projects, and a shrinking entry-level labor pool combined with a deteriorating educational system at a time when increasingly complex logistical systems demand more technologically sophisticated workers.

Computer and Communications Technology

Future developments in computer and communications technology should continue to exert a powerful influence on the model-based design of logistics systems:

—Further improvements in the price/performance ratio will encourage large companies to formulate still larger and more complex models and will enable more small companies to build logistics models.

—As networks become more popular, client-server compatibility will become mandatory for most data-intensive and computation-intensive business application software (including that for logistic system design), much as graphic user interfaces became mandatory. The benefits for data acquisition via SQL (Structured Query Language) databases will be particularly striking.

—The explosive growth of the Internet and related network-based resources will profoundly alter the ways in which models are built and used. Core modeling teams will become smaller and more dispersed, communications with people not on the core modeling team will increase greatly (Internet addresses will become as obligatory as postal and fax addresses), more modeling data will come from outside the company being modeled, total modeling project costs will decrease due to increased productivity and faster project completion, and model quality will increase.

—Continued progress toward the cross-operability of popular desktop software on the major platforms eventually will make cross-operability with such software mandatory for most nonembedded business application software. For example, Micro-

soft Corporation offers widely used database, electronic-mail, presentation-graphics, project-management, spreadsheet, and word-processing packages that work together and with other software that supports dynamic data exchange (DDE), object linking and embedding (OLE), and other standards set forth in their Windows Open Services Architecture. Modeling software that cross-operates synergistically with such packages will be superior to modeling software that doesn't, because it will support much more of the modeling life-cycle. Once a reasonably capable package for designing logistic systems becomes available that cross-operates with popular desktop software, the market will soon demand this capability of all such packages.

Cross-operability benefits are superadditive with respect to possibly cooperating categories of OR/MS software, because if two modeling packages—say one for demand forecasting and one for logistics system design—are cross-operable with a popular desktop software suite, then they are cross-operable with each other.

—A more speculative prediction is the addition of new data and results imaging features based on scientific visualization technology. All vendors of software for designing logistic systems have discovered the necessity for a graphic user interface, but they must offer something more now that virtually all packages have one. Scientific visualization with or without animation is a logical place to look for other visually appealing displays (after standard business graphics, which are an immediate by-product of cross-operability). Visualization has been thriving in the scientific community since the mid-1980s [Jones 1994;

McCormick, DeFanti, and Brown 1987; Schultz and Pulleyblank 1991]. Its technology could be adapted for viewing data and results in fresh and informative ways, with demand and transportation data being the most likely initial applications.

Algorithms

Algorithmic advances of major importance for logistic system design have not occurred very often. Since 1970, the only truly useful advances have been the adaptation of Benders decomposition, the ascendancy of the primal network simplex method, and the continuing progress of factorization. All other algorithmic progress has been incremental or of marginal utility for the design of logistics systems.

We expect this situation to persist until the currently dominant computing paradigm yields to a new one. Moreover, no matter what the rate of algorithmic progress, we expect that practitioners will always want to solve larger problems than are comfortably within the state of the art.

Improved algorithms for logistic system design have propagated in the world of practice much more slowly than one might expect. In particular, nonoptimizing methods have shown surprising longevity. If our explanations for this state of affairs are valid, technological improvements will continue to propagate at a leisurely rate.

Data Development and Management Tools

The challenges of practical logistics design problems have required far more effort in data development and management tools than in building optimizers. We see no reason why the relative allocation of effort should change, except possibly to tilt still more toward the data side as a result

of the much greater complexity of international studies [Bender 1985b], which are becoming increasingly common.

Many clients find great value in the data developed for their models even before any optimization runs are performed. We expect this experience to become ubiquitous when sufficient progress has been made toward the cross-operability of logistics design software and popular desktop software. Advanced data visualization features would add further value to modeling data. Companies with successful logistical models would be well advised to use them as a guide to improving their main managerial accounting and information systems, but widespread adoption of this approach may still be years away.

Database economies of scale can be expected to abate somewhat as internetworking becomes more common and both public-domain and commercial database services become more plentiful over the Internet and other networks. Downloadable data sources and tools for processing such data are steadily becoming better and more readily available. This should lead to major improvements in the way models are built and used.

As external data sources proliferate and become more accessible, internal data sources are also becoming more accessible thanks to new database management system (DBMS) access tools, especially for DBMSs that support SQL. Several end-user database access packages can read and write directly to most of the leading desktop DBMSs and can act as front-end clients for most of the leading SQL database servers. There are also dynamic link libraries (DLLs) that enable all database functions—

read, insert, update, delete, table creation, and so forth—to be performed for any major SQL DBMS not only from such programming languages as C, but also from the macro or script language of any of the many DLL-supporting applications written for Microsoft Windows and similar systems. These recent developments will, in their maturity, enable greatly improved data development and management tools for logistics design and many other data-intensive modeling applications.

Model Features and Software Capabilities

Four main forces drove the evolution of the model features and other capabilities supported by our software: (1) rapid progress in computer and communications technology, (2) algorithmic progress, (3) client requests, and (4) the desire to expand the coverage of our software to a greater portion of the modeling life-cycle. The likely directions resulting from the first two forces have already been discussed.

Client requests are the most fundamental force in setting direction and will remain powerful because logistics continues to gain in scope and influence as a corporate function. Requests from users mirror this increased scope. We and others [Byrne and Markham 1991; Gopal and Cypress 1993; Shapiro 1985] see model integration as written in the stars of logistics for many years.

Finally, we expect the leading software for distribution system design gradually to provide more software support to modeling life-cycle phases not immediately concerned with optimization. This destiny would be accelerated greatly by making such software cross-operable with popular

desktop software.

How Companies Actually Use Distribution System Design Software

Creative logistics analysts and planners have found ingenious and unanticipated ways to use and embellish classic warehouse location models. These models played a significant role in the mergers and acquisitions of the 1980s, and today they are being used to facilitate global mergers and logistics partnerships and to solve contemporary problems such as recycling spent containers and other packaging materials.

We expect that creativity in applying software tools for designing logistics networks will continue and increase as these tools become more widely used and accessible to a broader range of applications through desktop computers. Speaking as the originators of more than a few such tools, we find this creativity to be a source of considerable personal satisfaction.

Acknowledgments

We appreciate the valuable comments provided by R. Ballou, P. Bender, G. Brown, D. Fylstra, J. Karrenbauer, J. Shapiro, and the guest editors.

References

- Ahuja, R.; Magnanti, T.; and Orlin, J. 1993, *Network Flows*, Prentice-Hall, Englewood Cliffs, New Jersey.
- Andersen Consulting 1994, *Logistics Software*, Council of Logistics Management, Oak Brook, Illinois.
- Arntzen, B. C.; Brown, G. G.; Harrison, T. P.; and Trafton, L. L. 1995, "Global supply chain management at Digital Equipment Corporation," *Interfaces*, Vol. 25, No. 1, pp. 69-93.
- Ballou, R. H. 1991, "The accuracy in estimating truck class rates for logistical planning," *Transportation Research-A*, Vol. 25A, No. 6, pp. 327-337.
- Ballou, R. H. 1994, "Measuring transport cost-
- ing error in customer aggregation for facility location," *Transportation Journal*, Vol. 33, No. 3 (Spring), pp. 49-59.
- Ballou, R. H. and Masters, J. M. 1991, "A survey of commercial-grade warehouse location models," *Council of Logistics Management Annual Meeting Proceedings, Vol 1*, pp. 145-180. An abbreviated version appears as "Commercial software for locating warehouses and other facilities" in *Journal of Business Logistics*, Vol. 14, No. 2 (Fall 1993), pp. 71-106.
- Bausch, D. O. and Brown, G. G. 1988, "NDP FORTRAN and Phar Lap tools," *OR/MS Today*, Vol. 15, No. 3 (June), pp. 20-25.
- Bender, P. S. 1985a, "Logistic system design," *The Distribution Handbook*, eds., J. F. Robeson and R. G. House, Free Press, New York, pp. 143-224.
- Bender, P. S. 1985b, "The international dimension of physical distribution management," *The Distribution Handbook*, eds., J. F. Robeson and R. G. House, Free Press, New York, pp. 777-814.
- Bowersox, D. J.; Daugherty, P. J.; Dröge, C. L.; Rogers, D. S.; and Wardlow, D. L. 1989, *Leading Edge Logistics: Competitive Positioning for the 1990s*, Council of Logistics Management, Oak Brook, Illinois.
- Bradley, G. H.; Brown, G. G.; and Graves, G. W. 1977, "Design and implementation of large scale primal transshipment algorithms," *Management Science*, Vol. 24, No. 1 (September), pp. 1-34.
- Brown, G. G.; Graves, G. W.; and Honczarenko, M. D. 1987, "Design and operation of a multicommodity production/distribution system using primal goal decomposition," *Management Science*, Vol. 33, No. 11 (November), pp. 1469-1480.
- Brown, G. G. and Olson, M. P. 1994, "Dynamic factorization in large-scale optimization," *Mathematical Programming*, Vol. 64, No. 1 (March), pp. 17-51.
- Byrne, P. M. and Markham, W. J. 1991, *Improving Quality and Productivity in the Logistics Process*, Council of Logistics Management, Oak Brook, Illinois.
- Dantzig, G. B. and Van Slyke, R. M. 1967, "Generalized upper bounded techniques for linear programming," *Journal of Computer and Systems Science*, Vol. 1, pp. 213-226.
- Geoffrion, A. M. 1975, "A guide to computer-

- assisted methods for distribution systems planning," *Sloan Management Review*, Vol. 16, No. 2 (Winter), pp. 17-41.
- Geoffrion, A. M. 1976a, "Better distribution planning with computer models," *Harvard Business Review*, Vol. 54, No. 4 (July-August), pp. 92-99.
- Geoffrion, A. M. 1976b, "Customer aggregation in distribution modeling," working paper 259, Western Management Science Institute, UCLA, Los Angeles, October.
- Geoffrion, A. M. 1977, "A priori error bounds for procurement commodity aggregation in logistics planning models," *Naval Research Logistics Quarterly*, Vol. 24, No. 2 (June), pp. 201-212.
- Geoffrion, A. M. 1989, "Computer-based modeling environments," *European Journal of Operational Research*, Vol. 41, No. 1 (July), pp. 33-43.
- Geoffrion, A. M. and Graves, G. W. 1974, "Multicommodity distribution system design by Benders decomposition," *Management Science*, Vol. 20, No. 5 (January), pp. 822-844.
- Geoffrion, A. M.; Graves, G. W.; and Lee, S. J. 1982, "A management support system for distribution planning," *INFOR*, Vol. 20, No. 4 (November), pp. 287-314.
- Geoffrion, A. M. and Powers, R. F. 1980, "Facility location analysis is just the beginning," *Interfaces*, Vol. 10, No. 2 (April), pp. 22-30.
- Geoffrion, A. M. and Van Roy, T. J. 1979, "Caution: Common sense planning methods can be hazardous to your corporate health," *Sloan Management Review*, Vol. 20, No. 4 (Summer), pp. 31-42.
- Gopal, C. and Cypress, H. 1993, *Integrated Distribution Management*, Business One Irwin, Homewood, Illinois.
- Graves, G. W. and McBride, R. D. 1976, "The factorization approach to large-scale linear programming," *Mathematical Programming*, Vol. 10, No. 1 (February), pp. 91-110.
- Hall, J. R., Jr. 1980, "Management science update column: Multicommodity distribution system design by Benders decomposition, Part I," *Management Science*, Vol. 26, No. 8 (August), pp. 855-856.
- Jones, C. V. 1994, "Visualization and optimization," *ORSA Journal on Computing*, Vol. 6, No. 3 (Summer), pp. 221-257.
- La Londe, B. J. and Masters, J. M. 1994, "Organizational trends and career paths in distribution," *The Distribution Management Handbook*, eds., J. A. Tompkins and D. Harmelink, McGraw-Hill, New York, pp. 7.1-7.26.
- Magnanti, T. L. and Wong, R. T. 1990, "Decomposition methods for facility location problems," *Discrete Location Theory*, eds., P. Mirchandani and R. Francis, John Wiley and Sons, New York, pp. 209-262.
- Mamer, J. and McBride, R. D. 1994, "Solving multicommodity flow problems with a primal embedded network simplex algorithm," working paper, Anderson Graduate School of Management, UCLA, Los Angeles, June 13.
- Masters, J. M. and Pohlen, T. L. 1994, "Evolution of the logistics profession," *The Logistics Handbook*, eds., J. F. Robeson and W. C. Copacino, The Free Press, New York, pp. 13-34.
- McCormick, B.; DeFanti, T.; and Brown, M., eds. 1987, "Visualization in scientific computing: A special report of ACM SIGGRAPH," *Computer Graphics*, Vol. 21, No. 6 (November), pp. 1-14.
- Palmer, K. H. 1984, *A Model-Management Framework for Mathematical Programming*, John Wiley and Sons, New York.
- Powers, R. F. 1989, "Optimization models for logistics decisions," *Journal of Business Logistics*, Vol. 10, No. 1, pp. 106-121.
- Powers, R. F.; Karrenbauer, J. J.; and Doolittle, G. 1983, "The myth of the simple model," *Interfaces*, Vol. 13, No. 6 (December), pp. 84-91.
- Schultz, H. and Pulleyblank, W. 1991, "Trends in optimization," *OR/MS Today*, August, pp. 20-25.
- Shapiro, J. F. 1985, "Quantitative Methods in Distribution," *The Distribution Handbook*, eds., J. F. Robeson and R. G. House, Free Press, New York, pp. 373-409.
- Shapiro, J. F.; Singhal, V. M.; and Wagner, S. N. 1993, "Optimizing the value chain," *Interfaces*, Vol. 23, No. 2 (March-April), pp. 102-117.
- Stock, J. R. and Lambert, D. M. 1987, *Strategic Logistics Management*, second edition, Richard D. Irwin, Homewood, Illinois.

Copyright 1995, by INFORMS, all rights reserved. Copyright of Interfaces is the property of INFORMS: Institute for Operations Research and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.