DTIC FILE COPY

AD-A223 637

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 1990	Final Report, 15 Sep 89 to 14 Mar 90

4. TITLE AND SUBTITLE
AN INTEGRATED CONCEPT FOR SCHEDULING TRANSPORTATION N NETWORKS

5. FUNDING NUMBERS
~~AFOSR~~
F49620-89-C-0098
65502F 3005/A1

6. AUTHOR(S)
William G. Nulty

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Computer Aided Planning and Scheduling Inc.
2900 Paces Ferry Road, Building D
Atlanta, GA 30339-3719

8. PERFORMING ORGANIZATION REPORT NUMBER
AFOSR-TR- 90 0733

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AFOSR/NM
Building 410
Bolling AFB, DC 20332-6448

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
F49620-89-C-0098

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

Methods for the integration of optimization human interactions, simulation, and knowledge basis were related to problems of scheduling transportation networks.

14. SUBJECT TERMS	15. NUMBER OF PAGES	16. PRICE CODE
	31	

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

90 06 27 042

AFOSR-TR- 90 0733

An Integrated Concept for
Scheduling Transportation Networks

SBIR PHASE I (F49620-89-C-0098)

FINAL REPORT

March 1990

Prepared By

COMPUTER AIDED PLANNING & SCHEDULING INC.
2900 Paces Ferry Road, Building D
Atlanta, Georgia 30339-3719

TABLE OF CONTENTS

1. Introduction

The focus of this research is to investigate and develop methodologies for the integration of optimization, human interactions, simulation, and knowledge base to address the problems of scheduling transportation networks. These four techniques have been traditionally applied in isolation when addressing scheduling problems, resulting in serious modeling limitations. However, the complementary strengths of these techniques suggest a synthesis that would provide dramatic improvement in the ability to solve these problems. We have developed a prototype model that demonstrates the novel power and benefit of this integration. The concept and methodologies have been tested and proven successful.

Transportation networks are the fundamental structures associated with the movement and storage of material. The basic elements of the transportation networks include material movement requirements, transportation vehicles, points (facilities for supplying and receiving movement requirements), links (relationships between points), and crews. Through the generalization of the transportation network structure, a large variety of logistics problems can be modeled in the same fashion, bearing similar mathematical properties, and can studied and solved in a disciplined (as opposed to ad hoc) manner. The following figure illustrates the underlying structure and applications of transportation networks.

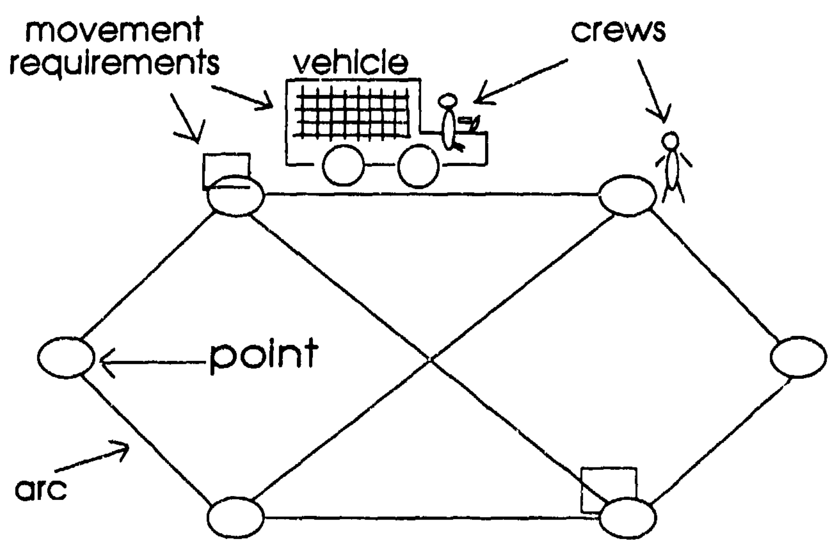

Figure 1: Transportation Network

Applications: military airlift, military sealift, helicopter routing, deployment, pipeline scheduling, pickup and delivery, postal delivery, courier route, location/allocation, dynamic dispatching, etc.

Points: airports, seaports, service stations, cities, depots, distribution centers, warehouses, refueling stations, manufacturing plants, customers, dates (in time), logical stages, etc.

Arcs: air lanes, sea routes, railroads, highways, streets, pipelines, aisles, conveyors, logical relationship (e.g. precedence), etc.

Vehicles: airplanes, helicopters, trains, trucks, ships, pallets, robots, AGVs, etc.

Crews: pilots, drivers, service crews, maintenance crews, etc.

Movement Requirements: goods, pallets, passengers, special task force, mail, etc.

Constraints: time windows at a point, vehicle capacity, maximum number of vehicles allowed at a point at the same time, type matching between vehicles and materials, maximum vehicle operating hours/miles, refueling, safety factors, management policy, etc.

Objectives: on-time delivery, quick response, minimal distance/time, minimal cost, minimal number of vehicles used, balanced work loads, minimal potential danger, etc.

The performance of a transportation system depends especially on the routing and scheduling of the transportation elements. Scheduling transportation networks is an extremely challenging task for both practitioners and researchers. The complexity of these problems is three-fold. First, these problems are usually combinatorial in nature and require enumerative search techniques. Second, human judgement and non-quantifiable considerations are usually involved in the decision making. Third, the scheduling process is dynamic because of changing problem parameters, objectives, and status. For example, the objectives of a transportation network may include any combinations of the following: prompt delivery of materials, minimizing the number of transportation vehicles and crews involved, balancing the assignment to each crew or vehicle, geographical considerations (e.g. keep drivers within a specified region), route patterns, etc. Quite often, the final solution accepted by the user is one that is a compromise of many unspoken objectives.

Traditionally, mathematical optimization-based methods and simulation-based methods have been viewed as the competing approaches for solving routing and scheduling problems. More recently, interactive approaches and artificial intelligence approaches have also been introduced. In general, optimization models provide a global quantitative approach at a crude level of detail. Simulation models provide a local myopic approach but include a great deal of detail. Knowledge-based systems and interactive techniques allow the incorporation of qualitative issues. Each of these approaches possesses attractive features, but all have severe limitations when addressing the network transportation problems in practice. The major strengths and limitations of these methods are summarized as follows:

Optimization Approaches

The mathematical optimization approach offers a global perspective of the problem. The optimization models are able to consider many alternatives simultaneously, and easily handle the various quantifiable objectives and constraints. Occasionally these models can be solved to optimality; more often they provide a basis for heuristics which provide near-optimal solutions. Optimization models also have the attractive side benefit of providing insight into the structure of a model through post-optimality analysis such as sensitivity and parametric analysis.

Optimization models generally sacrifice detail. These models characteristically grow to prohibitive size and complexity, forcing the use of aggregate, average, and notional data. For example, time is almost always considered in discrete increments (such as days or weeks), and the models have no resolution within a time increment. Conflicts which occur within a time period are not considered by the models. This lack of detail generates solutions which are generally fairly crude approximations to the actual systems, and in fact might include small but hidden inconsistencies.

Optimization models assume that the data is known in advance. This is almost never the case in practice.

Simulation Approaches

Simulation is a tool which has primarily been used in a planning as opposed to an operational environment. Even in the planning environment, simulation is primarily used to evaluate a plan rather than to generate a plan. Most of the research in simulation has been directed at handling the stochastic elements of the system, those over which we have no control. However, the major issues of scheduling transportation networks are about what events we plan to happen as opposed to stochastic events. While stochastic simulations are useful to the scheduler in answering "what if" questions, what is really desired is a schedule which works under the predicted environment.

When modeling non-stochastic processes, a simulation can be thought of as a detailed event processor. If the simulation is driven by a predicted event list or schedule, then the result will indicate in detail how the system behaves over time. It is possible, though, to allow stochastic elements in the more detailed level of a schedule. By doing so, we can test the robustness of the schedule. Again, the simulation is useful only in evaluating the schedule generated by optimization models.

Interactive Approaches

Interactive approaches are currently undergoing intensive study in conjunction with both optimization and simulation. This is causing rethinking in both of these areas (see Cullen, Jarvis, and Ratliff [1981], and Fisher [1986]). In the traditional environment the user would generate the model, run it in batch mode, and then try to interpret the result in terms of the problem being addressed. With the rapid development in computer processing power and graphics capabilities, there is now opportunity for the user to interact in a meaningful way with the models as they execute. A fundamental problem which arises when trying to have meaningful human interaction is the limitation on the human ability to handle a great deal of information. This problem is greater when interacting with simulation models since they embody more detail.

Successful application of transportation scheduling models has been hindered due in part to the lack of interactive approaches. In the context of routing and scheduling problems, Bodin et. al. [1983] state that "many computer-generated solutions are rejected based on relatively minor issues that could be corrected if certain controls over the computer system were given to the user." Interactive decision models overcome the difficulties associated with hard-to-quantify objectives and constraints, excessive data requirements, and computational complexity by having human planners guide and control the solution process, and make the critical strategic decisions. In this context the human planners are integrated elements of the solution process, rather than evaluators and managers of the final results.

This framework is desirable because the human is better equipped to make complex judgments involving nebulous information. When data is presented to him in a graphical form, he is able to perform complex spatial analysis and pattern processing far beyond the capability of today's computers to produce useful and powerful information and insight.

Knowledge-based System Approaches

Knowledge-based approaches have found many applications in the manufacturing environment (e.g. flexible manufacturing systems, just-in-time systems), and diagnostic systems. Typically, these systems are associated with problems and issues that are ill-defined, qualitative, and knowledge-rich, and the solution procedures are composed of a set of rules rather than mathematical optimization models. Knowledge-based approaches have some potential in scheduling transportation networks. In particular, they are useful in reaching a compromise among various objectives that are hard to quantify.

Significance of An Integrated Approach

In short, the relationship between optimization and simulation approaches (and methodology defining the coordination of these two approaches) has not received much attention in the research community. In addition, to date quantitative and qualitative solution approaches have remained isolated. Recently, Nobel Prize recipient Herbert Simon wrote "Two Heads Are Better than One: The Collaboration between AI and OR" (Simon [1987]), in which he exhorts the need for researchers to work on integrating qualitative and quantitative models. Simon [1987] suggests applying optimization techniques to well-defined subproblems within larger environments that are too complex for mathematical modeling. This knowledge-based control of the optimization algorithms would provide guidance to the algorithms, eliminating *mathematically feasible but undesirable* real-world solutions. By applying AI techniques to cull out feasible but illogical solutions, the mathematical models could be precisely focused, allowing more detail to be studied in shorter processing time. Forayce et. al. [1987] state that artificial intelligence methodology is well-suited for heuristic problem-solving and thus would nicely blend with more traditional optimization models. The overall result of this integration would be an improvement in the quality of solutions.

In Phase I we successfully built an integrated model to address a common instance of the vehicle routing and scheduling problem. Though in its prototype stage, the model already demonstrates novel power resulting from the integration of optimization, simulation, human interaction, and knowledge base system. Several important issues for the integration were identified and successfully dealt with. Such issues include mode' modularity and flexibility, object-oriented presentation, information filtering communication. The results of Phase I indicate that an integrated model to addr variety of transportation network problems can certainly be developed and will be a most valuable tool to a wide range of users.

2. Phase I Research Objectives

The primary objective of Phase I research effort was to prove the feasibility of the integrated concept by developing a prototype model to address the network transportation problems. The specific technical objectives were:

(1) Identify the transportation network components to be modeled, and characterize their structural relationships.

(2) Develop a knowledge-based simulation model and associated framework for emulating the transportation network schedule at different levels of detail.

(3) Develop the control flow framework which refines the gross transportation schedules resulting from optimization models by executing simulation models incorporating additional detail.

(4) Develop the control flow framework which integrates optimization/simulation systems with knowledge-based systems. Study incorporating the optimization/simulation system under the control of a knowledge-based system, and integrating the knowledge-based system with the knowledge-based simulation for schedule refinement.

(5) Develop methodology to track the schedule estimated by the scheduling models as the real-world execution of the schedule takes place, using actual time/event information received from the execution to help the user make scheduling revisions and adjustments in conjunction with the knowledge base.

(6) Develop a user interface concept which will allow modifications to the transportation network schedule during execution of the simulation.

(7) Delineate additional research issues resulting from the study.

3. Phase I Accomplishments

In Phase I, we extensively explored and analyzed various approaches and techniques that could be used for the integration of optimization, simulation, human interactions, and knowledge base approaches. Then we built an integrated model to address the vehicle routing and scheduling problem. Though in its prototype stage, the model has demonstrated the concept feasibility and powerful results of the integration. Various critical issues for the integration were identified and successfully resolved. The most significant contribution is its ability to schedule a system under a dynamic environment.

3.1 Problem Scenario for the Prototype Model

In Phase I, a representative problem was selected from the class of transportation network problems: the vehicle routing and scheduling problem. This problem is concerned with the assignment of transportation requirements to vehicles and the routing, usually associated with time, of the vehicles in accomplishing their tasks. This problem is common in both military and non-military applications. The following is a partial list:

1) Military Airlift
2) Military Sealift
3) Helicopter Routing
4) Pickup and Delivery
5) Postal Delivery
6) Product Distribution

Vehicle routing and scheduling is an extremely difficult problem, especially with time constraints (e.g. service open only from 8:00 am to 10:00 am). In practice, this problem is further complicated by the fact that many factors in consideration cannot easily be quantified or expressed, such as the importance of delivering goods to a station on time, the conflicting claims of competing objectives, etc. Therefore, the vehicle routing and scheduling problem presents a perfect scenario for the study of the integration of optimization, human interaction, simulation and knowledge base.

3.2 The Integrated Approach

The prototype model encompasses four basic modules: Optimization, Simulation, Knowledge Base, and User Interface. These four modules are integrated so that they can communicate with each other and the user can best utilize them to address the vehicle routing and scheduling problem. The integration of these modules is described by the logic flow of decision-making and modification as follows:

First, the Optimization Module allows the user, at any point in time, to pose vehicle routing and scheduling decisions as optimization models and then to solve these models. The user can either directly select an algorithm and parameter values for a particular solution, or "talk" interactively with the Knowledge Base and let it select the most suitable algorithm with the most appropriate set of parameter values for that problem. If the solution is not satisfactory, the user can feed back or adjust his preferences using the Knowledge Base and improve the solution.

After a tentative solution is generated, the Simulation Module "plays out" the solution by **stepping through** the schedule over time. This allows the user to evaluate his plan in more detail and study its effect through time. For example, the user can see the movement of vehicles on the screen and can track the flow of goods over time. Like running a tape deck, the user can move easily forward and backward in time through the plan. This is an innovative feature in itself, as most simulation systems only simulate forward. This freedom to move through time will enable the user to identify and track bottlenecks in the logistics system.

The simulation can also be run in real-time as an "emulation." This can be a tool with which the user can monitor the actual system. At any time the user can intervene in the emulation to redirect the unfolding solution. In the event that the actual events deviate significantly from the schedule, the simulation module allows the user to update the model with the new or changed data and re-schedule the system. For example, if a vehicle arrives late at a destination, the user can simply point at the corresponding vehicle icon on the emulation and drag it back to the appropriate position, re-schedule the system (using the optimization module) from that point on, and resume the emulation.

Such a capability to quickly respond to unpredictable changes in the actual execution of a schedule and to re-schedule the system is most useful and important, especially in crisis management situations. We believe this capability is one of the most novel and useful accomplishments of this research effort, and one which deserves continued development.

The User Interface plays an important role in making all the information easily accessible to the user, in facilitating communication between the modules, and in aiding the user in expressing vague goals and preferences.

The relationship between the four modules may be illustrated graphically by the following figure:

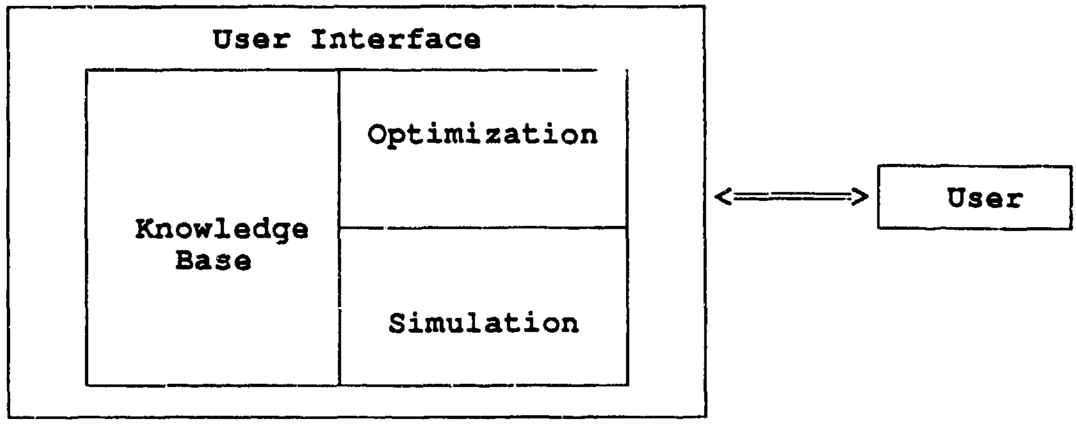

Figure 2: An Integrated System

While these modules are integrated together, they are each designed to maintain its own modularity and flexibility. Functionally, the Optimization Module can work without support of other modules. After the solutions are generated (from whatever means), the Simulation Module can work independently. Both the Optimization Module and the Simulation Module can be directly accessed by the user through the User Interface, or they can be accessed by the Knowledge Base. The user can "talk", through the User Interface, with the Knowledge Base which then calls the Optimization and/or Simulation Modules, or the user can directly call, through the User Interface, the Optimization and/or Simulation Modules.

The critical issue in integrating the Optimization Module and the Simulation Module is being able to interrupt the simulation and call the Optimization Module to re-schedule the interrupted system. Note that the algorithms in the Optimization Module usually handle well-defined problems with **static** data. They cannot directly handle real-time routing and scheduling problems in a dynamic environment. However, when the system needs to be re-scheduled, the components of the system are no longer static. Their status will be complicated by the past, present, and future commitments. For instance, a vehicle may be carrying several orders each to a different destination. To re-schedule this vehicle, its current condition and obligation must also be considered. It may be possible to develop specialized algorithms to handle this kind of situation. But since these situations vary and are unpredictable, it is impossible to prepare an enormous number of specialized algorithms for all uncertain situations. Therefore, we handle it in a more generalized and disciplined approach. That is, we keep the algorithms to their standard form, and **transform** the dynamic problem into an **equivalent** static problem. So the problem presented to the Optimization Module always looks as if the system has not started. This approach has proven to be fast, stable, and correct.

3.3 The Four Basic Modules

The Optimization, Simulation, Knowledge Base, and User Interface all are designed so that their structure will work not only for the prototype model but also for a full-fledged system. In other words, though it is a prototype, the general structure is designed for a full-fledged system.

Optimization Module

This module currently includes two algorithms to solve the vehicle routing and scheduling problem. However, the module was designed so that more algorithms can be easily added in future.

Boundary Router Algorithm: This algorithm processes the links (i.e. transportation requirements) in order of decreasing distance between link pickup point and link delivery point. Vehicles are assigned to links according to the parameters such as minimum angle, maximum angle and cut-off radius set by the user. The angles specify how far offline a vehicle may travel from its starting point in order to process a link. This algorithm works well for the problems where out-and-back geographical routing patterns are important and time windows are not critical.

Weighted Constraint Router Algorithm: This algorithm uses six special route parameters: volume, distance, pickup time window span, pickup close time, drop-off time window span, and drop-off close time. These parameters are used to score the relative importance of various routing constraints. The links are processed in the order of their relative importance. This algorithm work well when the problem is highly constrained by timing requirements.

Notice that both these algorithms generate solutions (schedules) for a system before the system starts. The input data to these algorithms are "static". As we shall see later, when a system requires re-scheduling, most of the system data are not **static** and are complicated by past, present, and future commitments. Therefore, these algorithms cannot be directly applied for re-scheduling the system. This is where the integration of the optimization module and simulation shows its power.

The Optimization Module can be used directly by the user, or it can be called by the Knowledge Base.

Simulation Module

The Simulation Module is equipped with many powerful features. Each object (element) of the system is graphically represented by an icon on the computer screen. As tne system evolves over time, the movement and current locations of the vehicles are indicated by the icon objects. Information of each object (vehicle, point, schedule, link, etc.) can be displayed at a mouse double click on the icon. The information box may stay on the screen as tne simulation continues, or it can be canceled by a simple mouse click on the cancel button.

The user runs the simulation through a panel of control buttons (like a tape deck) at the bottom of the screen. By mouse clicking on the buttons or scrolling the time bar, the use can select a target time (either future or past) and let the system gradually evolve to the target time. The use can choose the speed at which the simulation runs. Most importantly, the user may suspend the simulaticn at any point in time in order to examine the system in more detail, adjust simulation options, or update the system status with actual events and re-schedule the system.

Updating the system status requires little effort of the user. For example, the user may update the current location of a vehicle by mouse "dragging" the vehicle icon to its corresponding location on the screen. The user may even indicate that the vehicle i longer functioning (break-down) by a mouse click on the INACTIVE button in vehicle's information box. The primary theme is to help the user concentrate on those unpredicted events. Therefore, for all predicted events their status are automatically updated unless the user wishes to override the information.

Currently, the simulation is driven by a predefined event list and emulates the results in detail how the system behaves over time. However, the very framework and structure we have designed will allow the simulation to also model stochastic processes. We expect to fully implement all these in Phase II.

Knowledge Base

The Knowledge Base is composed of sets of rules derived from the experience and expertise of logistics specialists at CAPS. The Knowledge Base works as a technical expert who interprets the users objectives and vague goals ana then selects the most suitable algorithm and parameter values for solving the problem. Its main purpose is to provide technical guidance and enhanced human interaction. Therefore, the user does not have to be concerned with the selection of algorithms and parameters. Instead, the user can concentrate on the more strategic issues. However, if the user chooses to deal at a detailed level with the optimization algorithms, he can directly access those algorithms and parameters. The Knowledge Base opens a door to non-technical personnel who might be more experienced in strategic decisions. By eliminating the gap between technical and non-technical users, the system offers great potential for high quality solutions.

For the vehicle routing and scheduling problem, the Knowledge Base let the user interactively indicate, the relative importance of his objectives/goals:

a) Minimize vehicles' total travel time;

b) Minimize vehicles' total travel distance;

c) Minimize system's cost;

d) Balance the assignment to each vehicle/crew;

e) Minimize the number vehicles used.

The Knowledge Base is designed to "learn" and remember the user's preferences from session to session and incorporate them in subsequent decisions. The user can also interactively express his degree of willingness to relax certain constraints such as vehicle capacity constraints and time windows constraint.

For users with technical background, the Knowledge Base provides an Internal Logic Display box which displays the internal logic (path) of the Knowledge Base including the sequence of rules that are effected, the algorithms used, and the selected parameter values. This box can be turned on or off as the user chooses. For non-technical users this display box may serve as a tutor who explains what algorithm and parameter values are good for what problems.

Although the Knowledge Base was developed in a short time, its power and benefit have been immediately obvious. We realize, however, there are still a lot we can do to improve the Knowledge Base. For instance, currently the Knowledge Base influences the solution through selecting an algorithm and parameters values. It has no direct control over the inner logic of the algorithms. To be more effective, the Knowledge Base may construct its own algorithms through a rich set of rules. In this way the Knowledge Base could provide the logic of the solution procedures, among other things. All of these improvements will be easily realized under the framework we have developed.

User Interface

The User Interface was developed with many new concepts and techniques. We highlight them as follows.

Object-Oriented Information Communication: All conceptual objects in the model are presented to the user as objects that can be directly manipulated (for example, by pointing and clicking on the object icon on the screen). The goal here is to present the user with information that he can comprehend immediately and directly since it is consonant with his experience. Therefore as much as possible, information is packaged in ways that correspond to actual objects (vehicles, points/stations, and so on) and not as abstractions (tables, lists, and so on - although such are available).

Hierarchical Information Presentation: Information hierarchy is necessary for effective communication. The user must be informed in enough detail to make decisions. On the other hand, the user must not be burdened with unnecessary details of information which would impair his ability to make correct judgement. We have organized the information into three levels. The first level of information is the aggregate level. At this level, the system is described by a geographical map of the network system, where points/stations, vehicles, etc. are represented by icons. The second level of information is object oriented. For each physical object of the network, its main information is presented in a summary box. The third level of information is the relational information associated with other objects. The lower level (more detail) of information about an item in the current information box can be retrieved by mouse clicking on that item. Once the mouse cursor clicks on that item, a more detailed information box about that particular item will pop up on the screen. So the user can easily step through different levels of information without losing track.

Multi-Centric View: The user can examine decisions, constraints, etc. from the point of view of any participant (e.g. vehicle, point/station, orders), not just the single point of view of the aggregate planner.

Multiple Channels of Communication: Most information can be presented and accessed by mouse pointing and clicking on the icons. This greatly enhances the man-machine interaction. However, we do not restrict the channel of communication to graphical techniques only. For example, the vehicle's currently location is always indicated by its icon position on the screen. But if the user double-clicks the mouse cursor on the vehicle icon, then its information box will pop up on the screen, showing the corresponding vehicle location's longitude and latitude readings. To input the vehicle's new location, the user can click on the vehicle icon and, with button pressed, drags the vehicle icon to a new location. Alternately, the user can use the keyboard to type the new longitude and latitude readings into the information box. Another example is the time clock. The time can be set by mouse dragging or clicking, and it can also be set by typing in the readings through the keyboard.

Stability of Solution Presentation: Each time the network system is suspended for re-scheduling, it sends a different instance of the problem to the Optimization Module, although the user thinks that he is dealing with the same instance. To illustrate, the new schedule will not include the orders (i.e. transportation requirements) that have been delivered by the time the system is suspended for re-scheduling. But on the other hand, the fact that those orders have been delivered should be visible to the user. So the final schedule presented to the user should be composed of two parts: 1) part of the old schedule which has been actually executed, and 2) the new schedule of the system. It should be pointed out that the system may be re-scheduled many times, but each time we re-schedule it we consider what has been actually executed as the old schedule and we append the new schedule to it. This presentation of a schedule not only keeps all the system information as a whole, but also makes the user comfortable by hiding the potentially disturbing fact that he is dealing with a new instance of the problem.

4. Implementation and Functional Description

The prototype model incorporates the CAPS microcomputer Logistics Toolkit with the new modules and functions developed in Phase I. The Logistics Toolkit provides an overall framework within which specific logistics problems can be easily modeled and analyzed. It provides data management interface, modular algorithmic functions, and interactive graphics. It was coded in the C programming language and runs on the IBM PC (and compatible) family of microcomputers. By making the Logistics Toolkit part of the new system, the prototype makes full use of all the features of the Logistics Toolkit. Therefore, the Phase I effort could be focused on developing the new modules and functions that can communicate with all the Logistics Toolkit modules, especially under the same Data Base Management. The following figure describes the logical relationship between the Phase I work with the Logistics Toolkit:

The Integrated System (Prototype)	
Logistics Toolkit	Phase I Work
Data Base Management Interactive Graphics Algorithmic Functions	Simulation Knowledge Base User Interface Data Processing & Flow Control

Figure 3: Logical Relationship between the Prototype and the Logistics Toolkit.

4.1 Optimization Module

This module is based on the CAPS Logistics Toolkit. All the algorithms are modularized, and data are passed back and forth as arguments of a call function. Therefore, there is no structural limitation for future addition or modification of algorithms. To integrate the Optimization Module with the Simulation Module so that the algorithms works under the dynamic environment, additional work is needed modify the **PreProcessingDataStructure** function and the **PostProcessingSolution** function.

PreProcessingDataStructure Function: This function converts the data of a problem instance to a standard data structure used by the algorithms. Depending on the system's current environment (e.g. is it in the middle of a simulation, or is it simply in the planning stage), this function will treat the data differently.

PostProcessingSolution: This function translates the output of an algorithm into a solution format. In a static environment, this transformation is straightforward. However, in a dynamic environment it becomes very complicated because the output of an algorithm may be only part of a larger solution. Consider, for example, the re-scheduling situation. Each time the network system is suspended for re-scheduling, it sends a different instance of the problem to the Optimization Module, although the user thinks that he is dealing with the same instance. To illustrate, the new schedule will not include the orders (i.e. transportation requirements) that have been delivered by the time the system is suspended for re-scheduling. But on the other hand, the fact that those orders have been delivered should be visible to the user. So the final schedule presented to the user should be composed of two parts: 1) part of the old schedule which has been actually executed, and 2) the new schedule of the system. It should be pointed out that the system may be re-scheduled many times, but each time we re-schedule it we consider what has been actually executed as the old schedule and we append the new schedule to it. This presentation of a schedule not only keeps all the system information as a whole, but also makes the user comfortable by hiding the potentially disturbing fact that he is dealing with a new instance of the problem.

4.2 Simulation Module

The Simulation Module is accessed by checking on the Master Menu Bar displayed on the screen. The submenus under the simulation menu allow the user to initialize the simulation, select options, and run the simulation. The submenu items include:

(1) Initialize Simulation: This identifies the schedules to be simulated, creates an event list based on the schedules, creates an icon for each object (e.g. airplane or truck), and creates a panel of display and control devices at the bottom of the screen:

Time Display: This includes a time scroll bar accompanied with time readings. Both the scroll position and the readings will advance according to simulated time of the system, indicating the current time.

Set Time: This allows the user to select a target time by keyboard input or using mouse to scroll the time bar.

Step Forward: When this control button is clicked, the simulation advances to the next event time.

Step Backward: When this control button is clicked, the simulation reverts the direction and advances to the previous event time.

Suspend/Go: This button allows the user to suspend the system at any time he wants to examine the system in detail, adjust simulation options, prepare to re-schedule the system. A second press on the button resumes the simulation.

Viewing Mode: This allows the user to adjust the viewing speed of the simulation. If the maximum viewing speed is selected, the simulation will jump from current system status to system status at the target time, without displaying the intermediate stages.

(2) Select Simulation Options: This allows the user to select the event times at which he would examine the system in more detail. However, the user still can suspend the simulation whenever he presses the **Suspend/Go** button:

Time-Driven: This option automatically suspends the system at specified intervals of time to let the user study and query the system.

Event-Driven: This option automatically suspends the system any time when an arrival of a vehicle at a station occurs. This gives the user a chance to examine that vehicle's schedule as well as other information of the system.

Note that either one or both Time-Driven and Event-Driven options can be selected at the same time.

Stepwise/Automatic Advancing: This allows the user to choose if the simulation automatically suspends the system during run time or the simulation continues without automatic suspending.

Stepsize: This allows the user to determine the time interval (e.g. half a day, one day) at which the system is automatically suspended for examination.

(3) Exit Simulation: This gives the user a chance to update the system data with the actual events so that the system can be re-scheduled accordingly. After updating the system information, the user may exit the simulation and call the Optimization Module to re-schedule the system. The user may simply exit the simulation without updating the system.

Updating System Information: Once this button is checked, the user may update the system status according to the actual events. The powerful User Interface has make the data updating work extremely convenient, usually at a mouse click. For example, the vehicle's currently location is always indicated by its icon position on the screen. When the user double-clicks the mouse cursor on the vehicle icon, then its information box will pop up on the screen, showing the corresponding vehicle location's longitude and latitude readings. To input the vehicle's new location, the user can click on the vehicle icon and, with button pressed, drags the vehicle icon to a new location. Alternately, the user can use the keyboard to type the new longitude and latitude readings into the information box. The types of information to be updated include: a) vehicle's location; b) vehicle's operating condition (e.g. break-down or working); and c) new transportation requirements.

A special data processing function is automatically called to prepare the system for re-scheduling. It transforms the system into the initial zero-time status, which makes it possible to treat the problem as if the system has not started. In essence, it re-formulates the system model so that the Optimization Module can be directly used to obtain a new schedule for the system. For the vehicle routing and scheduling problem, this transformation involves the following steps:

A) Let the system's start time be the time when it is suspended for correction.

B) Delete all orders that have actually been delivered to their destination.

C) If a vehicle is at a point/station either loading or unloading an order, let the vehicle continue the loading/unloading (unless told otherwise by the user), and let the vehicle's start time be the time when the loading/unloading is completed.

D) If a vehicle is half way between two points/stations, let its current location be a new point and include the new point into the network, and let the vehicle's start time be the current time.

E) If a vehicle is not functioning (break down), set its start time to the system's finish time.

F) Treat all vehicles as empty at their current location and ready to transport.

G) For all orders that are aboard a vehicle, change their starting point to the vehicle's current location, set their required start times to the current time with zero loading times, treat these orders as if they had not been transported but they are designated to their current vehicle.

H) For all orders that have not been touched by a vehicle, keep all information the same.

Quit Simulation: This cancels all the display and control devices that are created when the simulation is initialized. It also frees the memory requirements needed for the simulation.

4.3 Knowledge Base

The Knowledge Base is accessed by checking on the Master Menu Bar displayed on the screen. The submenu items include:

(1) Internal Logic Display: This can be toggled on or off. When it is toggled on, the Internal Logic Display box will pop up on the screen displaying every action that the Knowledge Base is doing internally (e.g. which rule is affected, which algorithm is selected, how the parameter values are set, how a solution is evaluated at each iteration, etc.). In addition, the display box provides the three option buttons:

> **Next Step**: When this button is pressed, the Knowledge Base will continue to work until the next displayed message, and then automatically suspend to give the user a chance to read the display message. If the user wants to study how the Knowledge Base functions throughout the process, this button should be used at every step.
>
> **Non-Stop**: When this button is pressed, the Knowledge Base will continue to work without further suspension until a solution is represented to the user. The Internal Logic Display box will still be displaying messages as the Knowle Base progresses. However, since each displayed message is so quickly repla by the next message, the user may not be able to read or see all the messages.
>
> **Cancel Display**: When this button is pressed, the Knowledge Base will continue its work without further suspension, and the Internal Logic Display box will disappear. Therefore, the user will not see any message while the Knowledge Base progresses normally.

(2) Write the Logic to A Disk File: This can be toggled on or off. When it is toggled on, every action that the Knowledge Base does internally (e.g. which rule is affected, which algorithm is selected, how the parameter values are set, how a solution is evaluated at each iteration, etc.) will be written to a DOS text file. This file may later be used for post analysis.

(3) Run the Knowledge Base: The Knowledge Base starts by letting the user indicate his objectives through a dialogue box. Then the inference engine is automatically called which directs the control to the Algorithm Selection procedure, the Parameter Value Selection procedure, the Solution Procedure, and the Evaluation procedure. If the solution passes the evaluation based on the user's objectives, the it is presented to the user. Otherwise, the inference engine will again direct the control to the Algorithm Selection procedure, the Parameter Value Selection procedure, ..., usually with different set of selections and an improved solution.

Objective Dialogue Box: This box lets the user express his objectives by indicating the relative importance on a scale of 0 to 100. A set of common objectives for the vehicle routing and scheduling problem are listed as follows:

a) minimizing vehicles' total travel time;

b) minimizing vehicles' total travel distance;

c) minimizing system's cost;

d) balancing the assignment to each vehicle/crew; and,

e) minimizing the number vehicles used.

Along each objective item, there is a scroll bar accompanied with readings through which the user may interactively adjust the relative importance.

Besides the objectives, the dialogue box also includes two types of constraints that may sometimes "soft" on the vehicle routing and scheduling problem: the vehicle capacity constraints and the time window constraints. In the same manner through the scroll bar and the readings, the user may interactively express his degree of willingness to relax these constraints.

Algorithm Selection: This procedure consists of a set of rules which diagnose the characteristics of the problem and then select an algorithm that is considered most suitable for the given instance.

Parameter Values Selection: This procedure consists of a set of rules which, according to the selected algorithm, select values to be used by the algorithm.

Solution Evaluation: This procedure evaluates the solution obtained according to the objectives expressed by the user.

Update Best: This function records the "best" solutions obtained so far. A solution is considered the "best" when it cannot be replaced by any other solution in achieving one of the objectives. Note that the function does not need to record the solution itself which would require a lot of memory storage; it only records the algorithm index and the relevant parameter values which can be used later to re-generate the solution. Besides saving the memory storage, this technique also makes it convenient for the Knowledge Base to "learn" and remember the user's preferences from session to session and incorporate them in subsequent decisions.

Softening Constraints: This function "softens" the constraints by adding a certain amount of resource. For example, if the user has indicated that he is willing (to certain degree) to relax the time window constraints, then this function will "soften" the time window constraints by widening (by certain degree) the time window spans (e.g. change the span of 9:00-11:00 into a span of 8:00-12:00). The amount of resources added is dependent on the degree of flexibility that the user wants the constraints to be.

4.4 User Interface

This module makes full use of the CAPS Logistics Toolkit functions. The new functions and tools developed in Phase I are basically in the form of dialogue boxes, icons, and pull-down menus. The dialogue between the user and the computer is made as natural and simple as possible. Therefore, the user can "talk" to the computer by simple mouse clicks.

5. Relationship to Future Research

The integrated approach applied in the prototype model has proven very successful. The modeling structure, the concept and techniques will be equally applicable to a variety of network transportation problems. Therefore, the prototype model provides a guideline as to how a full-fledged integrated system should be developed. The Phase II will extend the concept and techniques developed in Phase I to other types of transportation network problems, and further explore the issues identified in Phase I and strengthen the modules and their integration. The final product of Phase II effort should be a fully integrated scheduling system that can be used to address a variety of transportation network problems, and it can be used by a wide range of users (from non-technical person to specialist).

5.1 More Types of Transportation Network Problems To Be Addressed in Phase II

Among the family of transportation network problems, the following problems distinguish themselves from the vehicle routing and scheduling problem in structure and modeling approaches.

Multi-Mode Scheduling

In some applications, the transportation network requires several scheduling decisions to be made at the same time. Quite often, these scheduling decisions are interrelated and dependent upon one another. For example, there are master routes (and schedules) for vehicles which pick up material at some points and deliver to other points. But at each point, there is a schedule for the crew which prepares(e.g. package) the material to be picked up or receives the delivered material. As another example, in a helicopter dispatching system CAPS is developing, some points (offshore platforms) need to be serviced by maintenance crews from their bases; and the crews are transported by helicopters. So there is not only a schedule for the crews, but also a schedule for the helicopters. Note that a helicopter is not designated for only one crew; while the crew is servicing a platform, the helicopter can fly other crews. The prompt service response, the efficient use of the crews and the helicopters depends on how well both kinds of schedules are done.

The essence of the interrelationship between the different levels of scheduling is the temporal issue: time coordination between different schedules. It has been demonstrated that the best technique of analyzing the temporal issues is to "play it out" over time. Therefore, the integration of optimization and simulation provides the best platform to generate and analyze the schedules.

Pipeline Scheduling Problem

The pipeline scheduling problem is unique in its way of moving materials between two terminal points. Materials are transported through a pipeline from one end to another by pushing more materials into the pipeline at the sending end. Since a pipeline may simultaneously contain segments(blocks) of materials (each segment corresponds to a different kind of material, e.g. unleaded gas), transporting a particular kind of material to a terminal point involves complicated scheduling. For example, a particular kind of material may be preceded by several other kinds of materials in the same pipeline. In order to transport the material to the demanding terminal point, it is not only necessary to push certain amount of material into the pipeline from the sending terminal, but also necessary to make sure how to handle the unwanted materials that precede it. So the order's type, size/amount, position, predecessors, successors, the piping speed, etc. all need to be considered in the scheduling. The graphical representation of the pipeline scheduling problem is as follows:

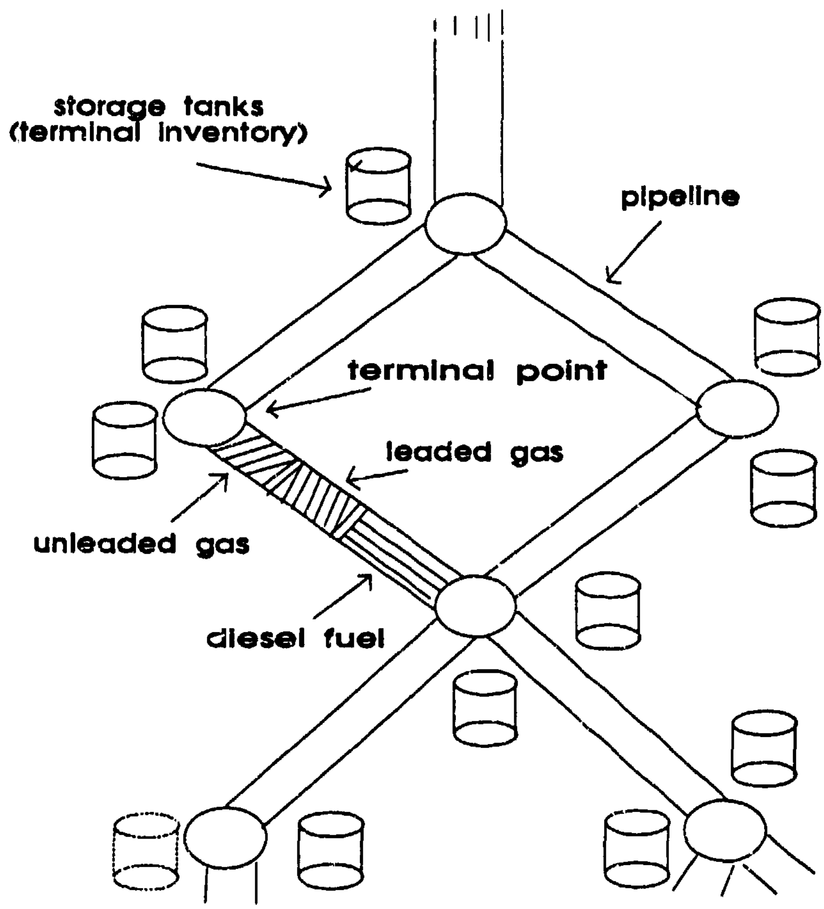

Figure 4: Pipeline Transportation Network

There is no known efficient algorithm for the pipeline scheduling problem. Its complexity is partly due to its additional spatial issue and the physical precedence relationship. Mathematical optimization models usually have limited power in dealing with the spatial issue. On the other hand, the human is better equipped to perform complex spatial analysis and pattern processing when data is presented in a graphical form. Therefore, the integration of optimization and simulation will provide a perfect platform for the user to perform complex spatial analysis and improve the solution quality.

Location/Allocation/Sourcing Problem

This problem is concerned with the physical location and size of a variety of different kinds of facilities including production, storage, and service facilities. The new facilities are either for the facilities(both new and existing) themselves or for customers. In general, there are two levels of decisions involved here. The first is deciding where and of what size to build each facility. The second is assigning the transportation flows to and from these facilities. The first decision restricts the second decision, and the second decision will influence the cost of the first decision. The primary cost tradeoff is the cost of building and running the facilities versus the transportation cost associated with getting material and services to and from the facilities. So in order to minimize the total cost, both levels of decisions should be made simultaneously.

The typical mathematical optimization model is to formulate the first decision as the master problem and the second decision as the subproblem. The assumptions (e.g. cost, transportation requirements) in the model are usually very rough and restrictive. Temporal issues, such as the inventory level at each facility over time, are usually ignored.

The limitation of the optimization model may be eliminated by integrating it with the simulation model: The optimization produces an aggregate plan in which some decisions are not fully specified; then the simulation disaggregates the plan and determine the details of the decisions postponed by the optimization. Whenever the simulation has trouble disaggregating the plan, it will be a potential bottleneck in the logistics plan and will need to be examined carefully by the planner.

More specifically, the simulation will focus on the inventory level of each facility over finer periods of time. Beside the data that are used in the optimization model, the simulation will require the data concerning the demand/supply during each finer time period, and the detailed shipping schedules. These additional data are not explicit in the optimization model. Rather, they are usually assumptions for the optimization model. In this context, the simulation is to check the feasibility of those assumptions.

5.2 Tasks Identified for Phase II Research

The concept feasibility has been proven in Phase I, and the mission of Phase II is to develop a full-blown integrated dynamic scheduling system based on the approaches and modeling structure of Phase I. The specific research issues identified for Phase II are:

(1) Build a modeling structure for a general (arbitrary) transportation problem

The integrated scheduling system will include general **data object abstraction, general graphics tools** to operate on these objects. It will provide the capability to model and analyze an arbitrary transportation network problems, including the Multi-Mode Scheduling, Pipeline Scheduling, Location/Allocation, as well as Vehicle Routing and Scheduling problem.

In the prototype model, the data objects are in the form of Vehicles, Schedules, Points, Links, Zones, Parameters, Maps, etc. Each data object captures the relevant attributes of the physical or logical component of the transportation system. These abstractions have worked well. For a full-fledged system, however, the graphics representations should be flexible enough to handle various types of objects in different problem scenarios.

(2) Develop general aggregation/disaggregation tools

There are several kinds of disaggregation:

1) Continuous flows (e.g. steady-state airplane cycles) disaggregated into discrete airplane takeoff/landing events.

2) Continuous time periods(e.g. the distribution center at week 1) disaggregated into finer periods(e.g. at day 1).

3) Disaggregating combined with more detail:

distance/time estimation: "crow-fly" estimates disaggregated to network estimates or road map estimates.

transportation requirements: an "order" disaggregated to many different products and quantities.

resources: ports have tons/day throughput at one level, berths, cranes, and marshalling areas at a lower level.

4) Disaggregating with randomness, stochastic properties(e.g. break down time)

In the prototype model, the disaggregation is done at the simulation stage. This will still be the case for the more general model. The disaggregation technique used in Phase I (using one abstract function to handle the disaggregation, and the rest remain at the aggregate level) allows an easy generalization for the full-fledged system.

(3) Develop general tools for reconciling a planned network schedule with actual events

The prototype model provides the mechanism to do dynamic scheduling and controlling. But the additional tools will provide more convenience to the user:

1) Tools to predict the impact of actual events' variation from the plan (e.g. does the lateness of a truck require the system to be re-scheduled entirely, or does the system allow the lateness without modification of the whole plan). This will determine when to replan.

2) Tools to do partial re-scheduling (e.g. re-schedule only one particular route).

These tools can be readily developed since the prototype model already has such potential. The additional work is to make these capabilities more handy.

(4) Develop more convenient operational tools for user interface

These tools include adding new requirements or changing existing requirements on the fly, updating changes in behavior of assets, and modifying routes on the fly. In the prototype, most of the required tools were developed, but more tools are definitely needed as the full-fledged system is used to address various types of problems.

(5) Enhance the Knowledge Base

The framework and control logic for the Knowledge Base developed in the prototype will be the same for the full-fledged system. However, the effectiveness of the Knowledge Base may clearly be improved by letting it construct problem-specific heuristics/algorithms. This will require a lot of work of "knowledge gathering", but will be well worthwhile.

(6) Develop system ports for user-specific functions

The ports let the user hook up his own functions for specified data operations. Since the functions should also be able to supersede the built-in functions, the functional abstraction should provide the flag indicate when the user wants to use his own function. This may be achieved at the macro level.

(7) Provide macro functions

The macro functions are to be used by the user to organize new control flow of the modules, select objects for simulation, construct Knowledge Based heuristics/algorithms, and activate/inactivate rules in the Knowledge Base.

6. Conclusion

In Phase I, we extensively explored and analyzed various approaches and techniques that could be used for the integration of optimization, simulation, human interactions, and knowledge base approaches. An integrated model was built to address the vehicle routing and scheduling problem. Though in its prototype stage, the model already demonstrated novel power resulting from the integration of optimization, simulation, human interaction, and knowledge base system. Most noteworthy is the surprising power of the integrated model to handle the **real-time dynamic** controlling and scheduling of the network system. Several important issues for the integration were identified and successfully dealt with. Such issues include model integration, modularity and flexibility, object-oriented presentation, effective techniques of information filtering and communication. In addition, research directions and tasks for Phase II were identified. The results of Phase I indicate that an integrated model to address a variety of transportation network problems can certainly be developed and will be a most valuable tool to a wide range of users.

References

1. Bodin, L., and B. Golden, A. Assad, M. Ball, *Computers & Operations Research Special Issue: Routing and Scheduling Of Vehicles and Crews*, Volume 10, No.2, 1983.

2. Cullen, F.H., J.J. Jarvis, and H.D. Ratliff, "Set Partitioning Based Heuristics for Interactive Routing," *Networks*, Volume II, No.2, Summer 1981.

3. Fisher, M.L., "Interactive Optimization," *Annals of Operations Research*, 5, 1985-86, pp.541-556.

4. Fordyce, K., P. Nordon, and G. Sullivan, "Artificial Intelligence and the Management Science Practitioner: Links between Operations Research and Expert Systems," *Interfaces*, 17:4, 34-40, July-August 1987.

5. Kearney, A., "Improving Productivity in Physical Distribution," Report undertaken for CPDM, London, 1980.

6. Simon, H.A., "Two Heads Are Better than One: The Collaboration between AI and OR," *Interfaces*, 17:4, 7-15, July-August 1987.