REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE

3. REPORT TYPE AND DATES COVERED
technical

4. TITLE AND SUBTITLE
Models and Techniques for Dynamic Demand-Responsive Transportation Planning: A State-of-the-Art Assessment Inspired by the Aeromedical Regulation and Evacuation Problem

5. FUNDING NUMBERS

6. AUTHOR(S)
Norman M. Sadeh and Alexander Kott

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

8. PERFORMING ORGANIZATION REPORT NUMBER
CMU-RI-TR-96-09

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

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

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This article provides an overview of state-of-the-art technologies relevant to dynamic transportation planning problems that involve the reactive routing and scheduling of a fleet of vehicles in response to dynamically changing transportation demands. Specifically, we focus on a new class of complex transportation planning problems, which we refer to as the "Dynamic Dial-A-Ride Problem with Multiple Acceptable Destinations and/or Origins" (D-DARP-MADO). While this class of dynamic problems is representative of a number of practical transportation problems, it does not appear to have been the object of prior studies. This is not to say that techniques proposed for simpler routing and scheduling problems cannot be brought to bear on this problem. To the contrary, our survey shows that a number of techniques developed in the fields of vehicle routing and scheduling, including reactive techniques proposed in the constraint-directed scheduling and manufacturing scheduling literature, appear quite relevant and can be adapted and/or combined to design effective solution procedures for the D-DARP-MADO.

14. SUBJECT TERMS

15. NUMBER OF PAGES
pp

16. PRICE CODE

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

MODELS AND TECHNIQUES FOR DYNAMIC DEMAND-RESPONSIVE TRANSPORTATION PLANNING

A State-Of-The-Art Assessment Inspired by the Aeromedical Regulation and Evacuation Problem

Norman M. Sadeh* and Alexander Kott+

CMU-RI-TR-96-09

*The Robotics Institute
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213-3891 - U.S.A.
Tel: (412)268-8827
Fax: (412)268-5569
sadeh@cs.cmu.edu

+Carnegie Group, Inc.
Five PPG Place
Pittsburgh, PA 15222 - U.S.A.
Tel: (412)642-6900
Fax: (412)642-6906
kott@cgi.com

This research has been a part of the TRAC2ES (TRANSCOM Regulating and Command and Control Evacuation System) project -- an effort to develop a global Command and Control decision support system that will help the U.S. Transportation Command (USTRANSCOM) conduct aeromedical evacuation of patients worldwide. Carnegie Group, Inc. provides research and software development support to the TRAC2ES project.

Abstract

This article provides an overview of state-of-the-art technologies relevant to dynamic transportation planning problems that involve the *reactive routing and scheduling* of a fleet of vehicles in response to dynamically changing transportation demands. Specifically, we focus on a new class of complex transportation planning problems, which we refer to as the "Dynamic Dial-A-Ride Problem with Multiple Acceptable Destinations and/or Origins" (D-DARP-MADO). While this class of dynamic problems is representative of a number of practical transportation problems, it does not appear to have been the object of prior studies. This is not to say that techniques proposed for simpler routing and scheduling problems cannot be brought to bear on this problem. To the contrary, our survey shows that a number of techniques developed in the fields of vehicle routing and scheduling, including reactive techniques proposed in the constraint-directed scheduling and manufacturing scheduling literature, appear quite relevant and can be adapted and/or combined to design effective solution procedures for the D-DARP-MADO.

Table of Contents

1. Objective and Scope

This article provides an overview of state-of-the-art technologies relevant to dynamic transportation planning problems that involve the *reactive routing and scheduling* of a fleet of vehicles in response to dynamically changing transportation demands. Specifically, we focus on a new class of complex transportation planning problems, which we refer to as the "Dynamic Dial-A-Ride Problem with Multiple Acceptable Destinations and/or Origins" (D-DARP-MADO). While this class of dynamic problems is representative of a number of practical transportation problems, it does not appear to have been the object of prior studies. This is not to say that techniques proposed for simpler routing and scheduling problems cannot be brought to bear on this problem. To the contrary, our survey shows that a number of techniques developed in the fields of vehicle routing and scheduling, including reactive techniques proposed in the constraint-directed scheduling and manufacturing scheduling literature, appear quite relevant and can be adapted and/or combined to design effective solution procedures for the D-DARP-MADO.

This paper is organized as follows. In the next section, we introduce the Aeromedical Regulation and Evacuation problem faced by the US Transportation Command when planning the evacuation of patients to adequate Medical Treatment Facilities during both wartime and peace. We show that this problem is in a number of ways more complex than dynamic Vehicle Routing Problems (VRPs) and dynamic Dial-A-Ride Problems (DARPs) considered in the transportation literature. Section 3 provides a more formal taxonomy of modeling assumptions considered in the VRP/DARP literature and provides a formal definition of the D-DARP-MADO problem by abstracting some of the added sources of complexity found in the Aeromedical Regulation and Evacuation problem. The second part of this section provides a broad taxonomy of techniques proposed in the literature for solving complex routing and scheduling problems. In Section 4, we review a number of relevant replanning/rescheduling techniques for the D-DARP-MADO, distinguishing between approaches relying on constructive techniques and approaches relying on iterative repair techniques. The review attempts to discuss the merits and advantages of each approach, often drawing on examples from the the Aeromedical Regulation and Evacuation domain to illustrate how these techniques can be generalized and/or combined to solve D-DARP-MADO problems. We argue that constructive and iterative repair techniques can be seen as complementary and that practical solutions to D-DARP-MADO can be devised that combine the strengths of both approaches. A summary and some concluding remarks are provided in Section 5.

2. Motivation and a Problem Example

Research in dynamic demand-responsive transportation planning is generally inspired and supported by application development in such domains as dynamically dispatched delivery of cargo by trucks and chartered planes or passenger delivery by demand-responsive bus and van companies.

Although this survey is not limited to a particular problem instance, it was particularly motivated by an application domain in which transportation planning is highly dynamic, complex and critical: the military Aeromedical Regulation and Evacuation (ARE) of patients to medical treatment facilities. A complex version of the well-known Dial-A-Ride-Problem (DARP), to these authors the ARE problem became an important measuring stick for determining viability and practicality of models and techniques proposed in the literature. Below, we summarize, some of the major characteristics of this problem. On numerous occasions in this survey, we will draw on examples taken from this domain. It should be emphasized that the ARE domain is but an instance of the much broader class of DARP-MADO transportation problems we define in Section 3.

2.1. The Aeromedical Evacuation Problem in Armed Services

U.S. Transportation Command (USTRANSCOM) is the DoD agency responsible for evacuating patients during both wartime and peace. Doctrinally, patients requiring extended treatment must be evacuated by air to a suitable Medical Treatment Facility (MTF). The process of identifying an MTF that constitutes a suitable destination for a given patient (based on consideration of a match between patient's condition and MTF's capability, and also based on economics and transportation availability) is called *regulating*. The process of routing and scheduling the required aeromedical evacuation flights (*missions*) and assigning patients to suitable missions is a part of the *evacuation* planning and execution.

There has been very limited experience with this approach to handling patients other than in peace time. The Persian Gulf war was the first significant armed conflict in which this concept has been put to a serious test. The results were far from satisfactory -- about 60% of the patients ended up at the wrong destinations and half in the wrong country [29].

In early 1993, the Department of Defense tasked USTRANSCOM to consolidate the command and control of medical regulation (i.e. assignment of patients to suitable medical treatment facilities) and aeromedical evacuation operations during peace, war and projected contingencies. The ensuing analysis led to TRAC2ES (TRANSCOM Regulating and Command and Control Evacuation System), a decision support system under development for USTRANSCOM [39].

The integrated medical regulation/evacuation problem requires the dynamic identification of appropriate Medical Treatment Facilities (MTFs) for new patients and the planning/scheduling of aeromedical evacuation operations to transport these patients from their current locations to the selected MTFs. This is a large-scale, highly dynamic planning and scheduling problem that can involve hundreds or even thousands of simultaneous patient movement requests. Each patient has one or several medical requirements that constrain the type of MTF to which he or she can be evacuated and a ready-time prior to which evacuation cannot start. Additional constraints can include a maximum altitude above which the evacuation aircraft cannot take the patient, a maximum number of hours that a patient can spend in a flight before requiring an overnight rest, a maximum number of stops the patient can tolerate during evacuation, etc. Planning/scheduling operations in this domain require the dynamic coordination and (re)allocation of a large number of resources subject to a wide variety of constraints. Key assets/resources and associated constraints include aircrafts and their different characteristics (e.g. capacity, refueling requirements, etc.), air and medical crews and restrictions on the number of hours they can work in any given day, airports and their different characteristics (e.g. capacity, types of aircraft they can accommodate, etc.), hospital beds at MTFs located all around the globe and the types of patients each MTF can accommodate, etc. Probably the most challenging aspect in planning and scheduling medical evacuation operations has to do with the dynamics of a domain in which requirements and constraints continuously change over time. New patient requests come in, others get canceled. Patient conditions change over time, both prior and during evacuation, possibly requiring the delay, acceleration or cancellation of a patient's evacuation or a change in the patient's destination. Availability of key assets is also subject to unpredictable events (e.g. aircraft maintenance problems, hospital beds not getting freed on time, airfield attrition, etc.). Further complicating the problem, weather conditions can affect evacuation plans at anytime, requiring that a mission be delayed, rerouted or simply scrapped.

In building and revising evacuation plans a number of objectives and preferences need to be taken into account. The number of patients evacuated to adequate MTFs has to be maximized, with urgent patients given priority over routine ones. The time to pickup and deliver patients to their MTFs, especially urgent ones, also has to be as short as possible. Simultaneously, the time each patient flies and the number of stops during his/her evacuation have to be minimized. Other important considerations include minimizing the number of missions, maximizing aircraft capacity utilization, etc.

2.2. General Features of Dynamic Transportation Problems

The ARE problem exhibits a number of features commonly found in dynamic transportation problems:

- There are multiple demands to transport commodities or entities (in our example -- patients) from/to origin or destination points (e.g., airports and hospitals);
- Multiple vehicle resources (e.g., planes) are to be routed and scheduled to meet the demands;
- Each demand (e.g., patient) is to be assigned to one or more vehicles (e.g., a patient may have to be transported via one or several missions);
- The destination and/or origin points may have to be determined dynamically, e.g., patient's destination may be determined based on the available missions.

- There are time window constraints, e.g., in our example a patient cannot be picked up at the origin point until he/she is prepared for departure and delivered to the airport;
- There are constraints on the vehicle capacity, e.g., an aeromedical evacuation mission has a limited number of seats available;
- There are multiple other constraints of varied nature, such as constraints on duration of tours associated with vehicles and/or with particular requests (e.g., aeromedical mission durations have limits and a patient's allowable flying time is also limited by his specific medical condition);
- The demands can change dynamically while the schedule is executing, e.g., new patients may need to be evacuated, or medical condition of a patient may change;
- The resources may also change dynamically, e.g., a mission can be delayed or cancelled, or an airport may be closed due to the weather.

A number of these problem features are found in Dial-A-Ride Problems studied in the vehicle routing literature. Others, particularly the dynamic changes in demands and resources, and the dynamic determination of origin and destination points, are not reflected in any canonic models. In Section 3, we briefly review major modeling assumptions considered in the Vehicle Routing/Dial-A-Ride literature and introduce a new canonic model, which we refer to as the Dynamic Dial-A-Ride Problem with Multiple Acceptable Destinations and/or Origins(D-DARP-MADO). This model, which attempts to capture some of the major sources of difficulty found in a number of practical dynamic transportation domains, such as the ARE problem, is shown to be in a number of ways more complex than models traditionally considered in the literature.

3. A Taxonomy of Models and Techniques

This section briefly reviews major Vehicle Routing/Dial-a-Ride models studied in the literature, introducing some basic terminology and identifying sources of complexity associated with dynamic transportation problems. The section proceeds with the introduction of a simple taxonomy which is used to categorize techniques discussed in Section 3.

3.1. A Taxonomy of Models

In its canonical form, the Vehicle Routing Problem (VRP) requires the design of a set of minimum cost routes originating and terminating at a central depot for a fleet of vehicles that has to service a set of customers with known demands. The problem can easily be seen to be NP-hard. For instance, the single uncapacitated vehicle version of the problem with the objective of minimizing total travel time reduces to a Traveling Salesman Problem (TSP), a well-known NP-hard problem (e.g. [50]).

Comprehensive surveys of VRP models and techniques include [7, 1, 101]. In many practical situations, as illustrated by the ARE problem, assumptions made in the canonical VRP formulation are too simplistic and a variety of additional considerations need to be taken into account. The following is a brief summary of more complex modeling assumptions considered in the literature:

- **Time Windows**: In the VRP with Time Windows (VRPTW), customers impose earliest/latest possible service time windows. Examples of similar constraints in the ARE problem include a patient's earliest ready-to-move time or a latest acceptable arrival time at a MTF . A good review of techniques for VRPTWs is provided in [101] (see also [100, 105, 14, 86]).

- **Capacitated Vehicles**: When servicing a customer/location involves picking up or delivering goods or people, vehicle capacities restrict the number of customers/locations or the number of demands that can be serviced by a given vehicle (e.g. [64]). For example, in ARE problem, planes can carry particular number of patients, depending on plane type and configuration.

- **Multiple Depots**: In a number of environments, not all vehicle routes originate from the same depot. Work on VRPs with Multiple Depots includes [32, 5, 99, 96]. In ARE domain, evacuation missions may originate from numerous airfields around the world.

- **Constraints on Lengths and/or Durations of Vehicle Tours**: Some VRP formulations also allow for constraints on the lengths and/or durations of vehicle tours. For example, in the ARE domain, such considerations arise from constraints on the range of a given aircraft, refueling requirements, restrictions on the crew duty day (the maximum number of hours a crew is allowed to work without an extended rest), and others.

- **Multiple Types of Vehicles**: In many actual VRPs, not all vehicles are the same and some customer requests can only be serviced by some vehicles (e.g. different aircraft types can be used in the ARE domain to meet specific demands).

- **Pickup and Delivery (Dial-A-Ride) Problems** arise when a vehicle is required to pickup an entity (e.g., a passenger or a parcel) at one location and then deliver it to another location. In this case, the pickup location must precede the dropoff location. Thus, Dial-A-Ride Problems are equivalent to VRPs with the addition of precedence constraints between pickup and dropoff locations. Early work on the Dial-A-Ride Problem (DARP) has been reported in [25, 111, 112, 113]. Work on the single vehicle DARP includes [71, 72, 15]. Techniques for the multi-vehicle DARP, which are more relevant to this survey, include those described in [79, 80, 36, 19]. It should be noted that multi-vehicle DARP does not capture the added complexity of many practical dynamic transportation problems which require matching demands with appropriate origin or destination points. In other words, there may be more than one pickup locations (e.g., when the customer will accept delivery of a commodity from any warehouse where it can be found) or there may be more than one acceptable dropoff location to choose from (e.g., a patient can be delivered to any hospital with suitable medical capability).

- **Dynamic Models:** Static formulations assume that customer demand is known ahead of time (i.e. models assuming "advance reservations") and that vehicle routes have to be optimized once and for all (e.g. School Bus problem). In contrast, in dynamic models, new customer requests are eligible for immediate consideration (e.g. [71, 43, 5]) and in general require revisions of already established routes and schedules. For example, the ARE problem is an instance of a dial-a-ride problem that includes a mix of both advance and immediate reservations.

From the above discussion, it should be clear that the dynamic transportation problem domain is in many ways more complex than VRPs/DARPs traditionally discussed in the literature. Using the VRP/DARP terminology, one can characterize a broad class of dynamic transportation planning and scheduling problems as a *Dynamic Dial-A-Ride Problem with Multiple Acceptable Destinations/Origins* (D-DARP-MADO). The D-DARP-MADO model expands DARP along two directions:

1. there may be multiple acceptable destination and/or origin locations for a given demand; the solution to this problem must include assignment of each demand to a destination and/or origin location;
2. both the demands and the resources can change dynamically, while the initial plan/schedule is being executed; the solution must include reactive revision of routes and schedules of resources, and of assignments of demands to origins, destinations and resources.

D-DARP-MADO captures some but by no means all the complexities found in practical dynamic transportation problems. Different domains present a great variety of domain-specific constraints which are not treated in literature and which are difficult to generalize. For example, in the ARE domain we find refueling constraints, overflight restrictions, maximum number of flying hours before a mandatory overnight rest for different patients, altitude restrictions on a patient's itinerary, etc. Similarly, different domains present a great variety of domain-specific preferences and objectives, which must be simultaneously considered. Again using the ARE problem example, we can find preferences and objectives such such as maximizing the number of patients delivered to appropriate treatment facilities, while giving a higher weight to urgent patients, minimizing disruptions and maximizing notification time for new missions/itineraries, minimizing delivery time especially for urgent patients, minimizing number of missions, etc.

By listing these complexities we do not intend to say that techniques proposed for simpler VRPs/DARPs cannot be brought to bear on more complex, real-world problems. To the contrary, this survey attempts to show that a number of insights and techniques proposed for simpler problems can in fact be extended, refined and combined to support realistic dynamic transportation planning and scheduling problems.

3.2. A Taxonomy of Techniques

Because the general VRP and DARP are NP-hard, attempts at developing optimal solutions to variations of these problems have been limited to fairly simple models and problems of relatively small size (e.g. see [7, 74, 1, 14]) This survey deliberately focuses on more pragmatic approaches that have emphasized the development of heuristic procedures capable of supporting the dynamic requirements of the ARE domain.

Heuristic search procedures developed for the VRP/DARP and related routing and scheduling problems can be categorized along a number dimensions. In particular, we distinguish between the following three basic dimensions:

- **Constructive versus Iterative/Repair Procedures**: Constructive techniques build solutions by incrementally instantiating decision variables (e.g. assignment of patients to missions) until a complete feasible and satisfactory solution is obtained. In contrast, iterative procedures evolve in a (search) space of complete solutions (i.e. solutions in which all decision variables have been instantiated), using transformation operators to move from one complete solution to another until a feasible and satisfactory solution has been found or until some predetermined condition becomes satisfied (e.g. a time limit has been reached). Simple examples of transformation operators in the ARE domain could include reassigning a patient from one mission to another or modifying the itinerary of a mission. A number of constructive and iterative procedures have been proposed in the literature to solve various routing and scheduling problems. They include simple one-pass constructive procedures which stop as soon as a first satisfactory solution has been obtained (e.g. [36]) as well as more expensive constructive procedures such as enumerative procedures described in [72, 98]. Iterative/repair procedures developed for routing and scheduling problems include simulated annealing procedures (e.g [109, 70, 87, 86]), Tabu search procedures (e.g. [64, 49, 62]) and genetic algorithm procedures (e.g. [106, 110]). As it turns out, generative and iterative techniques are not necessarily incompatible but can sometimes be viewed as complementary:

simpler constructive approaches can provide a basis for quickly producing new though potentially highly sub-optimal solutions while iterative techniques offer a way to spend additional time post-processing these solutions to improve their quality.

- **Informed versus Brute Force Search**: Another useful dimension along which to differentiate between techniques for routing and scheduling problems involves looking at the amount of analysis performed by a procedure before making a decision. At one extreme of the spectrum, we find brute-force techniques such as branch-and-bound or simulated annealing, which attempt to rapidly cover wide areas of the search space but spend very little time deciding which area to explore next, At the other extreme sit more sophisticated techniques, which emphasize more selective (and hence slower) exploration of smaller areas in the search space (e.g. [96]). In between, some techniques rely on constraint propagation to quickly prune unpromising areas of the search space but spend little time differentiating between remaining alternatives (e.g. [98]).

- **Adaptive versus Fixed Search Procedures**: The adaptability of a search procedure refers to its ability to gather information as it searches for a solution and exploit this information to improve search performance. This is also referred to as the ability of a procedure to "learn" as it goes. Learning can be used in many different ways to enhance search efficiency. Tabu Search, e.g., [26, 64] relies on simple short-term and long-term memories to dynamically adapt its behavior. Learning mechanisms have also been developed to enhance the performance of fast brute force search techniques such as Simulated Annealing (e.g. [60, 86]) or to determine how to best select among a set of repair operators (e.g. [57, 58])

Clearly, these three dimensions are not the only ones along which one can classify routing and scheduling techniques. In fact, at times in the following sections, we will either have to refine this taxonomy or discuss aspects of proposed solutions that are orthogonal to the dimensions identified above. Nevertheless, this simple 3-dimensional taxonomy provides a useful starting point for differentiating among the many techniques discussed in the following section.

4. A Selective Overview of Dynamic Replanning/Rescheduling Techniques

While considerable efforts have been expended developing optimal algorithms and heuristic procedures for the static version of the Vehicle Routing Problem and its many variants (VRPTW, DARP, etc), work on dynamic versions of these problems has been rather scarce. In the opening of his 1988 survey of dynamic vehicle routing problems [74], Psaraftis noted: "For all the explosive growth in the vehicle routing literature over the past several years,.., very little has been published on dynamic variants of the vehicle routing problems. Of the 62 references cited in [28], only three include phrases such as 'dynamic', 'real-time' or 'on-line' in their titles". Even though the body of research in dynamic vehicle routing problems has grown significantly since the time these observations were made, dynamic aspects still occupy a relatively small fraction of the overall research in VRP.

As already stated, our objective in this survey is to focus on dynamic routing and scheduling approaches and discuss their merits in light of the dynamic planning and scheduling requirements of D-DARP-MADO domains such as the ARE domain.

Requirements of dynamic problems differ from those of static ones in a number of ways (e.g. "real-time" requirements, higher criticality of near-term decisions, open-ended problems, uncertain requirements, etc.). As a result, solutions to static problems are often difficult to adapt to dynamic situations. When it comes to building solutions to dynamic problems, it is important to distinguish between procedures that rely on constructive techniques and those that rely on iterative techniques:

1. **Dynamic Replanning and Rescheduling Using Constructive Techniques**: In this class of techniques, the process begins with an incomplete or even empty solution and constructs the missing elements of the solution. Typically, this process proceeds through a path which does not include infeasible solutions. When relying on constructive search procedures, reactive functionalities require techniques that help determine which part of the solution to undo before invoking the constructive procedure to rebuild a new solution. This is a critical barrier to adapting constructive techniques to replanning and rescheduling tasks. An extreme variant of this approach involves freezing all decision variables associated with those parts of the solution that have already been executed, while rebuilding a brand new solution for the remaining decision variables (e.g. [71]). Rebuilding brand new solutions in this fashion raises two concerns. One concern is that in large-scale domains, such as the ARE domain, the computational requirements of such an approach could be prohibitive. By the time a new solution has been constructed, additional contingencies may have occurred, rendering the new solution obsolete. Another and a more profound concern is that in situations where it is possible to build a brand new solution each time a contingency occurs, this approach may still be undesirable because it introduces too many disruptions. Often it is preferable to restrict solution revisions to small parts of the solution (e.g. to avoid creating confusion, account for difficulties in communicating new solutions in real-time or difficulties in adapting to new solutions). Below, we discuss a number of approaches that have been proposed to determine which part

of a routing/scheduling solution to revise, including matchup approaches [3], conflict propagation approaches [83] and truth maintenance approaches [17, 21, 69, 41].

2. **Dynamic Replanning and Rescheduling Using Iterative Repair Techniques:** Iterative repair techniques traverse a path of complete solutions, both feasible and infeasible, eliminating constraint violations and improving the quality of the solution. In theory, these techniques can directly support reactive capabilities, provided one can ensure they eventually extricate themselves from infeasible regions and converge to a feasible solution. Examples of such techniques include genetic algorithm procedures such as the one described in [106, 105], simulated annealing procedures such as the ones described in [114, 86] or constraint-directed repair procedures such as the ones described in [55, 93, 58]. Iterative improvement techniques that do not allow for infeasible solutions can still be used to reoptimize solutions when favorable contingencies occur that make the problem easier and offer opportunities for improving the quality of the existing solution (e.g. cancellation of a request, addition of a new vehicle, duration of a trip is shorter than expected, etc.). In the face of contingencies that invalidate an existing solution (e.g. a transportation asset becoming unavailable for some period of time), iterative techniques require heuristics that can decide which part of the solution to undo, just like constructive techniques.

Instances of each approach are further discussed below.

4.1. Dynamic Rescheduling Using Constructive Techniques

4.1.1. Rebuilding New Solutions From Scratch

When relying on a constructive planning/scheduling procedure, the most obvious approach to dynamic replanning/rescheduling involves rerunning the procedure from scratch while freezing decision variables corresponding to activities that have already been executed or can no longer be replanned (e.g. because they are already executing or because of difficulties in replanning those activities).

An example of a technique based on this approach is described in [71] for the single vehicle DARP with many-to-many immediate requests. The technique is based on a dynamic programming solution to the static version of the problem. Under the dynamic scenario, the technique is generalized by (1) imposing constraints that reflect the state of affairs when the contingency occurs (i.e. customers already serviced and customers currently on board of the vehicle), and (2) rebuilding a new solution that satisfies these added constraints.

In complex highly dynamic environments such as the ARE domain, such an approach, at least in its pure form, is impractical, due to the significant solution modifications it would continuously introduce (e.g. continuous changes in mission assignments and times would make effective command and control impossible). However, a modification of this approach reported in [46] appears to be promising. This *commitment-constrained* technique uses the most recent solution as a constraint on the new solution. This minimizes the differences between the new solution and the preceding solution.

4.1.2. Insertion Techniques

The next most obvious approach to handling dynamic events such as the arrival of a new move request is to incrementally insert new requests into existing vehicle routes. A similar approach can be used when requests are canceled: simply deleting the corresponding stops from existing vehicle routes[1]. This approach can be simpler and faster than one that rebuilds brand new solutions. It was first considered by Wilson et al. [111, 112, 113]. More recent work using insertion techniques is described in [100, 36, 43]. In particular, Kikuchi and Rhee describe an insertion procedure for the many-to-many pickup and dropoff problem with multiple vehicles, time windows and advance reservations [43]. Their approach builds vehicle routes one vehicle at a time, using a two-step procedure in which an initial vehicle route is first constructed and then additional requests are inserted in the vehicle route. The approach uses a tree search technique to identify the maximum number of requests that can be inserted in a given vehicle route. This amounts to a local optimization procedure where vehicles are considered one by one and requests are optimally assigned to the vehicle under consideration. Prior to inserting a request, the procedure checks for feasibility using time window updating mechanisms reminiscent of the earliest/latest start time propagations found in constraint-directed scheduling systems (e.g. [51, 81, 59]). Like other insertion approaches, this technique can be generalized to handle dynamic requests. The new requests are simply inserted into existing vehicle routes, and, new routes are initiated when none of the existing routes can accommodate a new request.

Insertion techniques are generally quite fast, e.g., [36].

However, when used in dynamic contexts where new requests are incrementally inserted into existing solutions, insertion techniques are likely to produce highly sub-optimal solutions. This is because these techniques do not take advantage of reoptimization opportunities, namely opportunities to produce better solutions by resequencing trips/stops already assigned to vehicle routes or reassigning requests to different vehicles. As with all constructive approaches, this problem can be alleviated, using local reoptimization procedures to post-process the resulting solution. This can be done using local interchange and reassignment operators (e.g. [73, 64, 86]),

[1] In the ARE domain, multiple patients may be picked and dropped off at the same locations. In this case, stops are only eliminated if there are no other patients requiring them.

as discussed below in our subsection on iterative improvement/repair techniques.

4.1.3. Partial Revision

Insertion/deletion procedures are relatively narrow in scope. They are mainly intended to handle the arrival of new requests or the cancellation of existing ones[2]. Short of rebuilding brand new solutions each time a contingency occurs, a more general approach involves rebuilding a portion of the existing solution. Clearly, reconstructing a large portion of the existing solution is likely to yield a new high quality solution (assuming a good constructive procedure), but it can also be time consuming. In the extreme case, this approach actually reduces to rebuilding an entirely new solution. Generally, when deciding how much of the current solution and which specific part(s) of the solution to be rebuild, a number of factors need to be considered:

- *solution feasibility*: when new conditions need to be accommodated (e.g. new requests with tight time windows or an important loss of capacity), undoing too few decisions may cause the resulting problem to be infeasible;

- *disruptions*: as already indicated, important changes in the current solution can be detrimental even if apriori resulting in a "better" solution. This is due to the added difficulty in communicating the new solution to all the players involved, difficulties in adapting to the requirements of the new solution, etc. In general, there is a tradeoff between obtaining a highly optimized solution and maintaining stability in the system.

- *solution quality*: a large enough number of decisions should be undone to provide sufficient opportunities for reoptimization (e.g. opportunities to come up with new sequencing decisions on a given vehicle route or different vehicle assignments);

- *real-time requirements*: The computation time necessary to rebuild a solution should not exceed the amount of time available;

Below, we discuss different approaches proposed in the literature to determine which part of a scheduling/routing solution to revise, given a new situation.

The Matchup Scheduling Approach:

Bean et al. describe a "matchup" procedure to determine which part of a production schedule to rebuild, given contingencies such as machine breakdowns [3]. They consider a manufacturing

[2] They can possibly also be used to react to contingencies involving a loss of capacity by deleting trips that relied on the lost capacity and assigning them to different vehicles (or to the same vehicle but at a different time).

facility in which a set of jobs needs to be processed on a set of resources with some degree of processing compatibility. Different jobs have different release dates at which they become available for processing and different due dates by which they need to be completed. Tardiness costs are incurred for finishing a job past its due date and earliness costs (e.g. inventory costs) for finishing it too early. Jobs in this model require two resources, a machine and a tool. The machines are subject to stochastic breakdowns, requiring that the schedule be revised. Additionally, switching production from one job to another on a given resource requires a setup whose duration may be sequence-dependent (i.e. whose duration may depend on the previous job processed on the same resource).

Matchup scheduling is based on the assumption that, given a long enough scheduling horizon and a small enough disruption (e.g. a small enough machine breakdown), it is possible to find a *matchup time* beyond which an optimal solution to the scheduling problem before the disruption occurred remains optimal with the disruption. However, rather than looking for an optimal solution, which is typically impractical, matchup scheduling looks for satisfactory solutions. A solution is considered satisfactory when the sum of tardiness and inventory costs of all jobs on any given machine is below some acceptable value. Given a contingency involving one or several machine breakdowns, matchup scheduling picks a matchup time and rebuilds new schedules for each of the machines involved up to the selected matchup time. If the resulting machine schedules are not considered satisfactory, the procedure increments the value of the matchup time by some prespecified duration and tries again. If in the process of revising machine schedules, the matchup time comes to exceed some predetermined value, the procedure attempts to reallocate jobs across machines so as to reduce the number of setups and achieve a better balance between the loads of the different machines.

The procedure has the merit of being relatively fast, provided that matchup times are properly selected and that the underlying constructive procedure used to rebuild new solutions is not too time consuming. Selecting too tight a matchup time can potentially result in a number of iterations before the procedure finally finds a new satisfactory solution. Selecting too large a matchup time could force the procedure to revise a large portion of the solution and, hence, could also prove computationally expensive. Ideally, selection of a matchup point should account for the severity of the disruption at hand, the amount of time available to come up with a new solution, and the tightness of the existing solution or more specifically the ease/difficulty with which the new disruption could be absorbed by the existing solution. For instance, a breakdown on a bottleneck resource is likely to be more difficult to absorb than one on an underutilized resource. Unfortunately, these issues are not explicitly addressed in [3].

The Conflict Propagation Approach:

The concept of Conflict Propagation was first introduced in the context of the Micro-Boss scheduling system [83]. Conflict Propagation is somewhat similar to Matchup scheduling but is more selective in the way it determines which part of the solution to rebuild, because it adapts to the severity of the disruption. Rather than using a common matchup time for all resources affected by a contingency, Conflict Propagation analyzes the impact of the disruption on individual operations. The result typically amounts to unscheduling operations with different "matchup times" on different resources or even unscheduling multiple non-contiguous groups of operations on the same resource. The approach can also account for precedence constraints such as those found between related operations in a manufacturing job. Specifically, given a constraint violation such as a machine breakdown, Conflict Propagation considers dependencies in the original schedule (e.g. precedence constraints between multiple operations) and propagates the conflict at hand until it is totally absorbed by slack and/or resequencing opportunities present in the schedule. Conflict Propagation unschedules all the operations it traverses, then rebuilds a new solution for these operations. In the Micro-Boss system, a micro-opportunistic search procedure is used to rebuild the new solution [84]. In general, any constructive procedure could possibly be used, though the quality of the resulting solution will depend on how good a procedure is used. Conflict Propagation is guaranteed to always unschedule enough operations to restore integrity in the solution, though it will sometimes unschedule more than the strict minimum. Because it is more selective than matchup scheduling, Conflict Propagation is expected to generally reschedule fewer operations and does not need multiple trials to identify the set of operations to be rescheduled. Conflict propagation can also be complemented with heuristics that selectively unschedule additional reservations to provide more flexibility to the constructive procedure used to rebuild a new solution, and, hence, allow for the construction of a higher quality solution. [83] provides examples of such heuristics in the factory scheduling domain.

Truth/Reason Maintenance Approaches:

Truth/Reason Maintenance Systems (TMS/RMS) [17] and their breadth-first counterpart, Assumption-based Truth Maintenance Systems (ATMS) [13], provide yet an alternative approach to determining which part of the solution to modify. The idea here is that, during the construction of a solution, one should record the impact of each decision on the domain of possible values of other decision variables. If later a contingency invalidates assignments in the current solution, this information can be used to help determine the minimum set of earlier decisions that needs to be undone to restore integrity in the solution. For instance, in the ARE domain, one could record the impact of the assignments of new patients to a given mission on the time window for pickup and delivery of a particular patient already assigned to that mission, say $patient_1$ (i.e. as new patients are assigned to the mission, we would record for each one of them whether they caused the time windows for pickup and delivery of $patient_1$ to shrink and if so, which specific time intervals became unavailable). If during execution of the evacuation plan, the mission gets delayed to the point that $patient_1$ may not longer be delivered on time to his/her destination MTF (Medical Treatment Facility), the recorded information could be used to help identify patients within the same mission, which if reassigned to other missions would allow patient_1 to be delivered on time. More complex dependencies between missions could possibly be recorded as well. Clearly, recording these dependencies and updating them over time as the solution evolves can become rather complex and time consuming[3].

Early work using truth maintenance in reactive scheduling systems includes that of Elleby et al. [21, 22]. Prosser reported using a TMS approach to keep track of dependencies in dynamic single-resource scheduling problems found in the context of a larger distributed scheduling system called DAS [69, 9]. Prosser's approach combines truth maintenance with a simple shallow learning mechanism [12] that can dynamically identify more complex dependencies as conflicts arise and help determine which scheduling decisions to undo. More recently, Kelleher et al. have developed a Lazy/Focused RMS approach to tackle factory scheduling problems at Pirelli and airline scheduling problems at Iberia [40, 41]. In Kelleher's Lazy RMS approach, bookkeeping activities are performed on demand to answer specific queries. The authors report two-order of magnitude speedups over standard ATMS bookkeeping mechanisms.

To the best of our knowledge, the application of truth maintenance concepts to dynamic scheduling problems has only considered the problem of building *feasible* solutions. Ideally, in domains such as ARE where a number of preferences and objectives have to be taken into account, one should also keep track of optimization-related dependencies. Consider the following situation. Suppose that, based on proximity considerations, our transportation scheduling system has determined that aircraft_X would be ideal to evacuate patient_Y. However, because aircraft_X is temporarily out of service, the current best solution involves introducing a detour in the route of a second aircraft, say aircraft_Z, to come and pick up patient_Y. An interesting dependency to record would be that the assignment of patient_Y to aircraft_Z and the detour in aircraft_Z's original route are motivated by the fact that aircraft_X is temporarily out of service. Later, if aircraft_X happens to be fixed faster than originally anticipated, this information could be used to reassign patient_Y to aircraft_X. In practice, keeping track of such dependencies and using this information in an effective manner is far from trivial. For instance, suppose that, following the decision to introduce a detour in aircraft_Z's original route, new patients in the vicinity of patient_Y also get assigned to aircraft_Z. Aircraft_X may not have sufficient capacity or meet other requirements to transport these new patients and patient_Y. Given this new situation, aircraft_Z may now be a better choice for patient_Y. A similar condition could also arise if, by the time aircraft_X is fixed, aircraft_Z is already en-route to patient_Y's location. Keeping track of dependencies required to recognize these conditions could become extremely complex and time consuming in a highly dynamic domains such as the ARE domain.

[3]For instance, it is easy to show that in its general form the process of creating labels required for keeping track of dependencies in an ATMS has an exponential worst-case complexity.

4.2. Dynamic Rescheduling Using Iterative/Repair Techniques

In contrast to dynamic rescheduling approaches relying on constructive techniques, iterative/repair approaches navigate in a search space of complete, though possibly infeasible, solutions. These procedures move from one complete solution to another, while trying to eliminate constraint violations and improve the overall quality of the solution. Several approaches to dynamic rescheduling using iterative/repair techniques are discussed below.

4.2.1. Interchange Approaches

Over the years, interchange procedures have been successfully used to solve a number of routing and scheduling problems (e.g. [72]). These procedures are based on the concept of "k-interchange" introduced by Lin and Kernighan in the context of the Traveling Salesman Problem [53]. In general, interchange procedures can be used to resequence vehicle trips or reassign trips between multiple vehicles (e.g. lambda-interchange [64]). In its simplest form, an interchange procedure iteratively considers possible interchanges in the neighborhood of the current solution. If a given interchange improves the quality of the solution, it is performed and a new solution is obtained. The procedure can be applied until the solution that can no longer be improved (local optimum). Alternatively, it can be run for some predetermined amount of time. Interchange procedures can be used to post-process solutions obtained using other techniques, whether dealing with static or dynamic problems. For instance, a simple insertion procedure can be used to handle incoming requests and the resulting solution can be further optimized, using an interchange procedure. In their simplest form, interchange procedures are only allowed to move from one feasible solution to another. An example of such a procedure is the one developed by Psaraftis for precedence constrained routing problems [72]. Sometimes, by allowing the procedure to wander into infeasible regions of the search space, it is possible to eventually reach better solutions. While attractive, this approach is far from straightforward. It requires ensuring that the procedure eventually extricates itself from infeasible regions and converges towards a feasible solution. One approach here involves introducing cost penalties in the objective function to account for constraint violations in the solution. When properly adjusted, these cost penalties can help find feasible solutions, though some infeasibilities are more difficult to get rid of than others. As a result, interchange procedures may allow for some constraint violations that are easy to get rid of but not others.

Interchange procedures that allow for infeasible solutions can be used to directly support dynamic reactive capabilities. Contingencies causing one or more constraints to be violated introduce cost penalties in the objective function. The iterative procedure then works on trying to get rid of these penalties, moving the solution back into a feasible region.

As already indicated earlier, in their simplest form, interchange procedures generally get stuck in local optima. This can be particularly annoying if the local optimum is infeasible. One way of alleviating this problem is to sometimes allow the procedure to transition to neighboring solutions that are not as good as the current one in the hope of eventually reaching better solutions. A number of techniques have been developed to do just that. They include Simulated Annealing procedures [114, 70] and adaptive enhancements of Simulated Annealing procedures relying on speedup learning mechanisms [86], Genetic Algorithm procedures such as the one used in the Gideon system [106], as well as Tabu Search procedures [64].

4.2.2. Constraint-Directed Repair

Interchange procedures typically rely on brute force search. They attempt to find better solutions and/or eliminate constraint violations by rapidly covering large areas of the search space, while spending little time deciding which interchange to try next. An alternative approach is to spend more time analyzing the problem at hand and deciding how to modify/repair the current solution. This "informed" approach to solution repair, which is more selective in deciding what to do next and tends to cover smaller areas of the search space has been emphasized in constraint-directed repair work, including work on the Opis factory scheduling system [93] and its transportation scheduling counterpart, DITOPS [96]. Work in this area also includes that of Miyashita et al. in the context of Cabins [56, 58], work by Minton et al. on the Min-Conflict repair heuristic [55], and work by Hildum in the context of the DSS system [34].

In the Opis scheduling system, a number of metrics are used to analyze flexibility of the solution and opportunities for improvement. Given one or more constraint violations such as those caused by a temporary loss of capacity on a resource, the system evaluates these metrics in the vicinity of the conflict to select among a set of alternative repair operators (e.g. whether to right shift the current solution to absorb the conflict at hand, or reassign conflicting operations to other equivalent resources, or resequence operations involved in the current conflict, etc. or some combination of the above). Examples of useful metrics include resource utilizations in the vicinity of the conflict, fragmentation of reservations on different resources, amount of downstream and possibly upstream slack within different jobs affected by the conflict at hand, variance in these metrics, etc. Based on these metrics, an initial repair operator is selected. Application of the operator may introduce new conflicts, which in turn need to be repaired. The process goes on until all conflicts have been eliminated and a satisfactory solution has been obtained. An agenda-based mechanism is used to keep track of new conflicts as they arise and to help the system sequence the application of multiple repair operators. As with other repair procedures, special care must be taken to ensure that the procedure eventually converges to a feasible solution. A somewhat similar approach is also described in [34], in the context of the DSS system, though here the author places a higher emphasis on search flexibility, allowing the repair procedure to continuously revise its search strategy in a way reminiscent of the micro-opportunistic search procedures described in [81, 84].

Miyashita and Sycara have reported using Case-Based Reasoning (CBR) to train a scheduling system to recognize situations where one repair operator is preferable to others [58]. This is used in their Cabins system. Cabins computes metrics similar to those found in Opis and relies on earlier experience gathered in the form of cases to decide which repair operator to apply next. Given a new domain, this system would have to be retrained, a process which could possibly be tedious and time consuming.

5. Conclusions

Relatively little work has been done on the class of problems that involve routing and scheduling of multiple vehicles responding to dynamically changing transportation demands. However, a number of techniques developed for related factory scheduling problems (e.g. matchup scheduling) and/or developed in Artificial Intelligence (e.g. conflict propagation, constraint-directed repair) can be also brought to bear on this challenging class of problems.

Broadly speaking, we have distinguished between two main lines of approach to react to contingencies such as the arrival of a new patient request (or its cancellation), a loss in available capacity, a variation in the expected duration of a particular activity (e.g. aircraft maintenance, trip, etc.). One approach -- partial revision -- involves identifying ahead of time a group of decision variables affected by the contingency and relying on a constructive technique to rebuild a solution for these decision variables. Examples of such techniques include matchup scheduling, conflict propagation and truth maintenance. A second approach -- iterative/repair -- relies on techniques where a complete, though possibly infeasible solution, is iteratively manipulated using (local) transformation operators until all constraint violations have been removed and a satisfactory solution has been obtained. Examples of this approach include interchange techniques and other brute force variations of these techniques such as Simulated Annealing, Genetic Algorithms or Tabu Search, as well as more knowledge-intensive approaches such as the ones implemented in constraint-directed repair techniques.

These two lines of approach are not incompatible. In fact, they can be viewed as complementary. For instance, a solution obtained by using constructive techniques for partial revision can be further optimized using an iterative repair procedure. Similarly, a solution invalidated by dynamic disruptions can be first repaired partially with simple repair operators and then further improved by reconstructing a significant part of the solution.

Acknowledgements

This work was supported by the United States Transportation Command. The authors are indebted to MAJ J. Simpson (USTRANSCOM), Mr. R. Simpson (MITRE) and Mr. W. Elm who provided valuable insights and guided this research.

List of Relevant References

[1] Assad, Arjang A.
Modeling and Implementation Issues in Vehicle Routing.
Vehicle Routing: Methods and Studies.
In B.L. Golden and A.A. Assad,
Elsevier Science Publishers B.V. (North Holland), 1988, pages 7-45.

[2] Barnhart,C., and Y. Sheffi.
A Network-Based Primal-Dual Heuristic for the Solution of Multicommodity Network Flow Problems.
Transportation Science 27(2):102-117, May, 1993.

[3] Bean, J.C.,J.R. Birge, J. Mittenthal, C.E. Noon.
Matchup Scheduling with Multiple Resources, Release Dates and Disruptions.
Operations Research 39(3):470-483, May-June, 1991.

[4] Matthew E. Berge and Craig A. Hopperstad.
Demand Driven Dispatch: A Method for Dynamic Aircraft Capacity Assignment, Models and Algorithms.
Operations Research 41(1):153-168, January-February, 1993.
Special Issue on Stochastic and Dynamic Models in Transportation.

[5] Bertsimas, D. and G. Van Ryzin.
Stochastic and Dynamic Vehicle Routing in the Euclidean Plane with Multiple Capacitated Vehicles.
Operations Research 41(1):60-76, January-February, 1993.
Special Issue on Stochastic and Dynamic Models in Transportation.

[6] Blythe, Jim, and Scott W. Reilly.
Integrating Reactive and Deliberative Planning for Agents.
Technical Report CMU-CS-93-155, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA 15213, 1993.

[7] Bodin, L., B. Golden, A. Assad, and M. Ball.
Routing and Scheduling of Vehicles and Crews: The State of the Art.
Computers and Operations Research 10:63-211, 1983.

[8] Brown, G.G. and G.W. Graves.
Real-Time Dispatching of Petroleum Tank Trucks.
Management Science 27:19-32, 1981.

[9] Peter Burke and Patrick Prosser.
A Distributed Asynchronous System for Predictive and Reactive Scheduling.
Technical Report AISL-42, Department of Computer Science, University of Strathclyde, 26 Richmond Street, Glasgow, GI IXH, United Kingdom, October, 1989.

[10] Cerny,.V.
Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm.
J. Opt. Theory Appl. 45:41-51, 1985.

[11] L.D. Davis.
Job Shop Scheduling with Genetic Algorithms.
In J.J. Grenfenstette (editor), *Proceedings of the 1st International Conference on Genetic Algorithms*, pages 136-140. Lawrence Erlbaum Associates, Carnegie Mellon University, Pittsburgh, PA 15213, July, 1985.

[12] Rina Dechter.
Enhancement Schemes for Constraint Processing: Backjumping, Learning, and Cutset Decomposition.
Artificial Intelligence 41:273-312, 1989.

[13] De Kleer, J.
An Assumption-Based Truth Maintenance System.
Artificial Intelligence 28:127-162, 1986.

[14] Desrochers, Martin, Jacques Desrosiers and Marius Solomon.
A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows.
Operations Research 40(2):342-354, March-April, 1992.

[15] J. Desrosiers, Y. Dumas and F. Soumis.
A Dynamic Programming Solution of the Large-Scale Single Vehicle Dial-a-Ride Problem with Time Windows.
Am. J. Math. Mgmt. Sci. 6:301-325, 1986.

[16] J. Desrosiers, Y. Dumas and F. Soumis.
The Multiple Vehicle Many-To-Many Routing Problem With Time Windows.
Technical Report Cahiers du GERAD G-84-13, Ecoles des Hautes Etudes Commerciales, Montreal, Canada, 1986.

[17] John Doyle.
A Truth Maintenance System.
Artificial Intelligence 12(3):231-272, 1979.

[18] Dror, M., and W. Powell.
Stochastic and Dynamic Models in Transportation.
Operations Research 41(1):11-14, January-February, 1993.
Preface to Special Issue on Stochastic and Dynamic Models in Transportation.

[19] Dumas, Y, J. Desrosiers and F. Soumis.
Large-Scale Multi-Vehicle Dial-A-Ride Problems.
Technical Report Cahiers du GERAD G-89-30, Ecoles des H.E.C., Montreal, Canada, 1989.

[20] Edwards, R.G. and T.G. Yow.
Strategic Mobility Modeling at Oak Ridge National Laboratory.
Technical Report CONF-900157-1, Oak Ridge National Laboratory, Oak Ridge, TN, January, 1990.

[21] P. Elleby, H.E. Fargher and T.R. Addis.
Reactive Constraint-Based Job-Shop Scheduling.
Expert Systems and Intelligent Manufacturing.
In Oliff, M.D.,
Elsevier Science Publishing Co., 1988, pages 1-10.

[22] P. Elleby, H.E. Fargher and T.R. Addis.
A Constraint-based Scheduling System for VLSI Wafer Fabrication.
Knowledge Based Production Management Systems.
In J. Browne,
Elsevier Science Publishers B.V. (North Holland), 1989.

[23] Mark S. Fox, Norman Sadeh, and Can Baykan.
Constrained Heuristic Search.
In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence,* pages 309-315. 1989.

[24] M.R. Garey and D.S. Johnson.
Computers and Intractability: A Guide to the Theory of NP-Completeness.
Freeman and Co., 1979.

[25] B. Gavish and K. Srikanth.
Mathematical Formulations for the Dial-A-Ride Problem.
Technical Report Working Paper 7909, University of Rochester, Rochester, NY, 1979.

[26] Fred Glover.
TABU Search.
Technical Report CAAI-88-3, Center for Applied Artificial Intelligence, Graduate School of Business, Box 419, University of Colorado, Boulder, Colorado 80903-0419, November, 1988.

[27] D.E. Goldberg.
Genetic Algorithms in Search, Optimization, and Machine Learning.
Addison-Wesley, New York, 1989.

[28] Golden, B.L. and Assad, A.
Perspectives on Vehicle Routing: Exciting New Developments.
Operations Research 34:803-810, 1987.

[29] Endoso, J.
TRANSCOM wants casualties to get where they belong.
Government Computing News , May, 1994.

[30] Khosrow Hadavi, Wen-Ling Hsu, Tony Chen, and Cheoung-Nam Lee.
An Architecture for Real Time Distributed Scheduling.
Technical Report, Siemens Corporate Research, 755 College Road East, 1991.

[31] Haimovich, M. and H.G. Rinnooy Kan.
Bounds and Heuristics for Capacitated Routing Problems.
Mathematics of Operations Research 10:527-542, 1985.

[32] Haouari,M., P. Dejax and M. Desrochers.
Modelling and Solving Complex Vehicle Routing Problems Using Column Generation.
Technical Report Working Paper, Ecole Centrale de Paris, LEIS, 1990.

[33] Harrison, G., Southworth, F., Sexton, A. and Hilliard M.
Functional Description of the Airlift Deployment Analysis System.
Technical Report ORNL-6560, Oak Ridge National Laboratory, Oak Ridge, TN, May, 1991.

[34] David Hildum.
Flexibility in a Knowledge-Based System for Solving Dynamic Resource-Constrained Scheduling Problems.
PhD thesis, COINS, Univ. of Massachusetts, 1994.

[35] Huang, Yuangeng.
Dynamic Scheduling Problem Solving.
PhD thesis, , 1990.

[36] J. Jaw, A. Odoni, H. Psaraftis and N. Wilson.
A Heuristic Algorithm for the Multi-Vehicle Advance Request Dial-A-Ride Problem With Time Windows.
Transportation Research 20B:243-257, 1986.

[37] Johnson, D. S., Aragon, C. R., McGeoch, L. A. and Schevon, C.
Optimization by Simulated Annealing: Experimental Evaluation; Part I, Graph Partitioning.
Operations Research 37(6):865-892, January-February, 1989.

[38] Johnson, D. S., Aragon, C. R., McGeoch, L. A. and Schevon, C.
Optimization by Simulated Annealing: Experimental Evaluation; Part II, Graph Coloring and Number Partitioning.
Operations Research 39(3):378-406, January-February, 1991.

[39] I. Johnson, V. Saks, MAJ. J. Simpson and R.H. Simpson.
Constraint-Directed Object-Oriented Medical Evacuation Planning.
In *American Defense Preparedness Association Conference.* 1994.

[40] Gerry Kelleher and Phillipe Retif.
Controlling Constraint-Based Scheduling Using Focussed RMS.
In *Proceedings of the IJCAI-93 Workshop on Knowledge-based Production Planning, Scheduling, and Control*, pages 195-204. Chambery, France, August, 1993.

[41] G. Kelleher and J Spragg.
Time-Critical Rescheduling Using Truth Maintenance.
Technical Report TRUTH.D43, Commission of the European Communities ESPRIT III Programme, Truth Consortium - Project 6463, 1994.

[42] Naiping Keng and David Y.Y. Yun.
A Planning/Scheduling Methodology for the Constrained Resource Problem.
In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 998-1003. 1989.

[43] Kikuchi, Shinya and Jong-Ho Rhee.
Scheduling Method for Demand-Responsive Transportation System.
Journal of Transportation Engineering 115(6):630-645, November, 1989.

[44] S. Kirkpatrick, C.D. Gelatt, Jr., and M.P. Vecchi.
Optimization by Simulated Annealing.
Science 220:671-680, 1983.

[45] Kleinrock, L.
Queueing Systems.
Wiley, 1975.

[46]
Commitment-Constrained Regeneration: an Approach to Reactive Replanning, and Rescheduling.
Carnegie Group, Inc., Pittsburgh, PA, 1995 .
working paper.

[47] Kraemer, R.D. and I.G. Harrison.
Decision Support System for the Military Airlift Command, the Airlift Deployment Analysis System.
Technical Report ORNL/TM-11201, Oak Ridge National Laboratory, Oak Ridge, TN, September, 1989.

[48] Yoshitaka Kuwata, David M. Hart and Paul R. Cohen.
Steering the Execution of a Large-Scale Planning and Scheduling System.
In *Proceedings of the AAAI-92 SIGMAN Workshop on Knowledge-Based Production Planning, Scheduling and Control.* San Jose, CA, July, 1992.

[49] Manuel Laguna and Fred Glover.
Integrating Target Analysis and Tabu Search for Improved Scheduling Systems.
Expert Systems With Applications 6(3):287-297, July-September, 1993.
Special Issue on Scheduling Expert Systems and their Performances.

[50] Lenstra, J. and A. Rinnooy Kan.
Complexity of Vehicle Routing and Scheduling Problems.
Networks 11:221-228, 1981.

[51] Claude Le Pape and Stephen F. Smith.
Management of Temporal Constraints for Factory Scheduling.
Technical Report, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, 1987.
Also appeared in Proc. Working Conference on Temporal Aspects in Information Systems, Sponsored by AFCET and IFIP Technical Committee TC8, North Holland Publishers, Paris, France, May 1987.

[52] Claude Le Pape.
Implementation of Resource Constraints in ILOG SCHEDULE: A Library for the Development of Constraint-Based Scheduling Systems.
Intelligent Systems Engineering , 1994.

[53] Lin, S. and Kernighan, B.W.
An Effective Heuristic Algorithm for the Traveling Salesman Problem.
Operations Research 21:498-516, 1973.

[54] Minkoff, Alan S.
A Markov Decision Model and Decomposition Heuristic for Dynamic Vehicle Dispatching.
Operations Research 41(1):77-90, January-February, 1993.
Special Issue on Stochastic and Dynamic Models in Transportation.

[55] S. Minton, M.D. Johnston, A.B. Philips, P. Laird.
Solving Large-Scale Constraint Satisfaction and Scheduling Problems Using a Heuristic Repair Method.
In *Proceedings of the Eighth National Conference on Artificial Intelligence*, pages 17-24. 1990.

[56] Kazuo Miyashita and Katia Sycara.
Predictive and Reactive Scheduling through Iterative Revision.
In *Proceedings of the IJCAI-93 Workshop on Knowledge-based Production Planning, Scheduling, and Control.* Chambery, France, August, 1993.

[57] K. Miyashita and K. Sycara.
Learning Control Knowledge through Cases in Schedule Optimization Problems.
In *Proceedings of the 10th IEEE Conference on Artificial Intelligence Applications (CAIA-94).* March, 1994.

[58] Kazuo Miyashita and Katia Sycara.
CABINS: A Framework of Knowledge Acquisition and Iterative Revision for Schedule Improvement and Reactive Repair.
Technical Report CMU-RI-TR-94-34, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, 1994.

[59] Nicola Muscettola.
HSTS: Integrating Planning and Scheduling.
Intelligent Scheduling.
In Mark Fox and Monte Zweben,
Morgan Kaufmann Publishers, 1994, pages 169-212, Chapter 6.

[60] Nakakuki, Yoichiro, and Norman Sadeh.
Increasing the Efficiency of Simulated Annealing Search by Learning to Recognize (Un)Promising Runs.
In *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pages 1316-1322. 1994.

[61] G.L. Nemhauser and L.A. Wolsey.
Integer and Combinatorial Optimization.
Wiley, 1988.

[62] Eugeniusz Nowicki and Czeslaw Smutnicki.
A Fast Taboo Search Algorithm for the Job Shop Problem.
Technical Report Working Paper, Technical University of Wroclaw, Institute of Engineering and Cybernetics, ul. Janiszewskiego 11/17, 50-372 Wroclaw, Poland, 1994.

[63] Wilhelmus P.M. Nuijten.
Time and Resource Constrained Scheduling.
PhD thesis, Technische Universiteit Eindhoven, December, 1994.

[64] I.H. Osman.
Meta-Strategy Simulated Annealing and Tabu Search Algorithms for the Vehicle Routing Problem.
Technical Report, Institute of Mathematics and Statistics, University of Kent at Canterbury, Canterbury, Kent CT2 7NF, UK, 1992.

[65] P.S. Ow, S.F. Smith, and A. Thiriez.
Reactive Plan Revision.
In *Proceedings of the Seventh National Conference on Artificial Intelligence*, pages 77-82. 1988.

[66] Peng Si Ow and Stephen F. Smith.
Viewing Scheduling as an Opportunistic Problem-Solving Process.
Annals of Operations Research 12:85-108, 1988.

[67] Don T. Phillips and Alberto Garcia-Diaz.
Fundamentals of Network Analysis.
Prentice-Hall, 1981.

[68] Powell, W.
An Operational Planning Model for the Dynamic Vehicle Allocation Problem with Uncertain Demands.
Technical Report, Department of Civil Engineering, Princeton University, Princton, NJ, 1985.

[69] Patrick Prosser.
A Reactive Scheduling Agent.
In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, pages 1004-1009. 1989.

[70] Patrick Prosser, Claude Muller and Craig Brind.
A Preliminary Study of Stochastic Search Techniques Applied to Vehicle Routing Problems.
Technical Report, Department of Computer Science, University of Strathclyde, 26 Richmond Street, Glasgow, GI IXH, United Kingdom, February, 1994.
Working Paper.

[71] Psaraftis, H.N.
A Dynamic Programming Solution to the Single Vehicle, Many-to-Many, Immediate Request Dial-A-Ride Problem.
Transportation Science 14:130-154, August, 1980.

[72] Psaraftis, H.N.
An Exact Algorithm for the Single Vehicle Many-to-Many Dial-A-Ride Problem with Time Windows.
Transportation Science 17(3):351-357, August, 1983.
Technical Note.

[73] Psaraftis, H.N.
k-Interchange Procedures for Local Search in a Precedence-Constrained Routing Problem.
European Journal of Operational Research 13:391-402, 1983.

[74] Psaraftis, Harilaos N.
Dynamic Vehicle Routing Problems.
Vehicle Routing: Methods and Studies.
In B.L. Golden and A.A. Assad,
Elsevier Science Publishers B.V. (North Holland), 1988, pages 223-248.

[75] Psaraftis, H.,J. Orlin, B. Bienstock and P. Thompson.
Analysis and Solution Algorithms of the Sealift Routing and Scheduling Problems: Final Report.
Technical Report, Sloan School of Management, MIT, Cambridge, MA, 1988.

[76] Ravela, Srinivas.
A Survey of Reactivity.
Technical Report UMA 92-61, Dept. of Computer and Information Science, University of Massachusetts at Amherst, September, 1992.

[77] Rhee, J.-H.
Vehicle Routing and Scheduling Strategies for Demand Responsive Transportation Systems.
PhD thesis, University of Delaware, 1987.

[78] Mark J. Ringer.
Time Phased Abstractions for Combining Predictive and Reactive Scheduling Methods.
In *Proceedings of the AAAI-92 SIGMAN Workshop on Knowledge-Based Production Planning, Scheduling and Control.* San Jose, CA, July, 1992.

[79] S. Roy, J.-M. Rousseau, G. Lapalme and J.A. Ferland.
Routing and Scheduling for the Transportation of Disabled Persons - The Algorithm.
Technical Report Working Paper TP 5596E, Transport Development Center, Montreal, Canada, 1984.

[80] S. Roy, J.-M. Rousseau, G. Lapalme and J.A. Ferland.
Routing and Scheduling for the Transportation of Disabled Persons - The Tests.
Technical Report Working Paper TP 5598E, Transport Development Center, Montreal, Canada, 1984.

[81] Norman Sadeh.
Look-ahead Techniques for Micro-opportunistic Job Shop Scheduling.
PhD thesis, School of Computer Science, Carnegie Mellon University, March, 1991.

[82] Sadeh, N.M., S. Otsuka, and R. Schnelbach.
Predictive and Reactive Scheduling with the Micro-Boss Production Scheduling and Control System.
In *Proceedings of the IJCAI-93 Workshop on Knowledge-based Production Planning, Scheduling, and Control.* Chambery, France, August, 1993.

[83] Norman M. Sadeh.
Micro-Boss: Towards a New Generation of Manufacturing Scheduling Shells.
In *Proceedings of the ARPA/Rome Laboratory Knowledge-Based Planning and Scheduling Initiative*, pages 191-203. Tucson, AZ, Februrary, 1994.

[84] Norman Sadeh.
Micro-Opportunistic Scheduling: The MICRO-BOSS Factory Scheduler.
Intelligent Scheduling.
In Mark Fox and Monte Zweben,
Morgan Kaufmann Publishers, 1994, pages 99-135, Chapter 4.

[85] Norman Sadeh, Katia Sycara, and Yalin Xiong.
Backtracking Techniques for the Job Shop Scheduling Constraint Satisfaction Problem.
Artificial Intelligence , 1994.
To appear in Special Issue on 'Planning and Scheduling'.

[86] Norman Sadeh, Yoichiro Nakakuki and Sam Thangiah.
Learning to Recognize (Un)Promising Simulated Annealing Runs: Efficient Search Procedures for Job Shop Schedulng and Vehicle Routing.
Technical Report, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3891, April, 1995.
Submitted to International Conference on Metaheuristics (MIC'95).

[87] Sadeh, Norman, and Yoichiro Nakakuki.
Focused Simulated Annealing Search: An Application to Job Shop Scheduling.
Annals of Operations Research , 1995.
To appear in issue on 'Metaheuristics in Combinatorial Optimization. Also available as technical report CMU-RI-TR-94-30 at Carnegie Mellon's Robotics Institute.

[88] Victor Saks, Al Kepner, and Ivan Johnson.
Knowledge Based Distribution Planning.
In *American Defense Preparedness Association Conference.* 1992.

[89] J. Schank, M. Mattock, G. Sumner, I. Greenberg, J. Rothenberg, and J.P. Stucker.
A Review of Strategic Mobility Models and Analysis.
RAND, 1991.
Prepared for the Joint Staff.

[90] Sexton, A. and H.-L. Hwang.
Automated Support for Processing Special Assignment Airlift Missions (SAAMs): A Concept Paper.
Technical Report ORNL/TM-11024, Oak Ridge National Laboratory, Oak Ridge, TN, 1989.

[91] Stephen F. Smith, Peng Si Ow, Nicola Muscettola, Jean-Yves Potvin, Dirk Matthys.
An Integrated Framework for Generating and Revising Factory Schedules.
Journal of the Operational Research Society 41"(6):539-552, 1990.

[92] Stephen F. Smith.
Knowledge-based Production Management: Approaches, Results, and Prospects.
Production Planning and Control 3(4), 1992.

[93] Stephen F. Smith.
OPIS: A Methodology and Architecture for Reactive Scheduling.
Intelligent Scheduling.
In Mark Fox and Monte Zweben,
Morgan Kaufmann Publishers, 1994.

[94] S.F. Smith and O. Lassila.
Toward the Development of Flexible Mixed-Initiative Scheduling Tools.
In *Proceedings of the 1994 Rome Lab/ARPA Planning Initiative Workshop held in Tucson, AZ.* February, 1994.

[95] Smith, S.F. and O. Lassila.
Configurable Systems for Reactive Prodcution Management.
IFIP Transactions. Volume 15-B.*Knowledge-Based Reactive Scheduling.*
In R. Kerr and K. Szelke,
North Holland, Amsterdam, 1994.

[96] Smith, S.F. and Sycara,K.P.
Flexible Coordination in Resource-Constrained Domains.
Technical Report CMU-RI-TR-95-02, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, 1995.

[97] Smith, Stephen F., Keng, Naiping, and Kempf, Karl.
Exploiting local flexibility during execution of pre-computed schedules.
Technical Report CMU-RI-TR-90-13, Center for Integrated Manufacturing Decision Systems, The Robotics Institute, Carnegie Mellon University, June, 1990.

[98] Douglas R. Smith.
Transformational Approach to Scheduling.
Technical Report KES.U.92.2, Kestrel Institute, 3260 Hillview Avenue, Palo Alto, CA 94304, December, 1992.

[99] Smith, Douglas R. and Eduardo A. Parra.
Transformational Approach to Transportation Scheduling.
In *Proceedings of the 1994 Rome Lab/ARPA Planning Initiative Workshop held in Tucson, AZ*, pages 205-216. February, 1994.

[100] Marius M. Solomon.
Vehicle Routing and Scheduling With Time Window Constraints: Models and Algorithms.
Technical Report 83-42, College of Business Administration, Northeastern University, 360 Huntington Ave - Boston, MA 02115, December, 1983.

[101] Solomon, M.M. and J. Desrosiers.
Survey Paper: Time-Window Constrained Routing and Scheduling Problems.
Transportation Science 22(1), 1988.

[102] R. Stallman and G. Sussman.
Forward Reasoning and Dependency-directed Backtracking in a Sysem for Computer-aided Circuit Analysis.
Artificial Intelligence 9:135-196, 1977.

[103] Storer, Robert H., S. David Wu, and Renzo Vaccari.
New Search Spaces for Sequencing Problems with Applications to Job Shop Scheduling.
Management Science 38(10):1495-1509, October, 1992.

[104] Robert E. Tarjan.
Network Flows.
CBMS-NSF Regional Conference Series in Applied Mathematics. Number 44:*Data Structures and Network Algorithms.*
Society for Industrial and Applied Mathematics, 1983, Chapter 7.

[105] Sam R. Thangiah.
A Genetic Algorithm System for Vehicle Routing With Time Windows.
PhD thesis, North Dakota State University of Agriculture and Applied Sciences, May, 1991.

[106] Thangiah, Sam R., Kendall E. Nygard and Paul L. Juell.
GIDEON: A Genetic Algorithm System for Vehicle Routing With Time Windows.
In *Proceedings of the Seventh IEEE Conference on Artificial Intelligence Applications*, pages 322-328. 1991.

[107] Paul Thompson.
Local Search Algorithms for Vehicle Routing and Other Combinatorial Problems.
PhD thesis, MIT, 1988.

[108] Truett, L.F., R.S. Loffman and S.S. Stevens.
Airlift Deployment Analysis (ADANS) Development Guidelines: User Interface.
Technical Report ORNL/TM-11391, Oak Ridge National Laboratory, Oak Ridge, TN, 1990.

[109] Van Laarhoven, P.J., Aarts, E.H.L., and J.K. Lenstra.
Job Shop Scheduling by Simulated Annealing.
Operations Research 40(1):113-125, 1992.

[110] Whitley, D., T. Starkweather and D. Shaner.
The Traveling Salesman and Sequence Scheduling: Quality Solutions Using Genetic Edge Recombination.
Technical Report CS-91-111, Dept. of Computer Science, Colorado State University, Fort Collins, Colorado 80523, January, 1991.

[111] Wilson, N., J. Sussman, H. Wang and B. Higonnet.
Scheduling Algorithms for Dial-a-Ride Systems.
Technical Report Report USL TR-70-13, Urban Systems Laboratory, MIT, Cambridge, MA, 1971.

[112] Wilson and H. Weissberg.
Advanced Dial-a-Ride Algorithms Research Project: Final Report.
Technical Report Report R76-20, Dept of Civil Engineering, MIT, Cambridge, MA, 1976.

[113] Wilson and N. Colvin.
Computer Control of the Rochester Dial-a-Ride System.
Technical Report Report R77-31, Dept of Civil Engineering, MIT, Cambridge, MA, 1977.

[114] Monte Zweben, Eugene Davis, and Michael Deale.
Iterative Repair for Scheduling and Rescheduling.
Technical Report, NASA Ames Reserch Center, MS 244-17, Moffett Field, CA 94035, 1991.