Oper Manag Res (2010) 3:68–79
DOI 10.1007/s12063-010-0028-0

Decision support for disaster management

Erik Rolland · Raymond A. Patterson · Keith Ward · Bajis Dodin

Received: 1 September 2009 / Revised: 19 January 2010 / Accepted: 19 January 2010 / Published online: 10 March 2010

Abstract Disaster response and recovery are crucial phases of disaster management. Decision-support systems used in disaster management must cope with the complexity and uncertainty involved with the scheduling and assignment of differentially-skilled personnel and assets to specific tasks. Operational constraints—such as workload and labor requirements, precedence constraints, resource availability, and critical deadlines among others—make timely and appropriate task assignment and sequencing difficult. Failure to assign personnel in an efficient and effective manner may result in unnecessary fatalities and significant additional loss of property as well as damaging the reputation of the disaster management organizations. Therefore, this paper proposes a decision-support system for disaster response and recovery using hybrid meta-heuristics.

Keywords Disaster management · Meta-heuristics · Project management · Scheduling · Decision support systems

E. Rolland (✉) · B. Dodin
The Anderson School of Business Administration,
University of California,
Riverside, CA 92521, USA
e-mail: Erik.Rolland@ucr.edu

B. Dodin
e-mail: Bajis.Dodin@ucr.edu

R. A. Patterson
School of Business, The University of Alberta,
3-21E Business Building Edmonton,
Edmonton, AB, Canada T6G-2R6
e-mail: Ray.Patterson@ualberta.ca

K. Ward
School of Management and Business, St. Edward's University,
3001 S. Congress Ave.,
Austin, TX 78704, USA
e-mail: keithfw@stedwards.edu

1 Introduction

The ability of governmental agencies and relief organizations to respond quickly and appropriately to a natural or man-made disaster is crucial in saving lives and/or preventing additional loss of property. In the case of Hurricane Katrina's strike on the southern US coastline in 2005, the US Congressional investigation into the aftermath of the disaster found that the governmental hierarchy (federal, state, and local) failed to act decisively. Many of the elements of the nation's disaster management plan were either poorly implemented or never attempted resulting in an increased fatality rate and the preventable suffering of many of the survivors (U.S. House of Representatives 2006).

The failure to allocate needed resources in a timely manner is not uncommon and may exacerbate the disaster situation. As noted by Quarantelli (1988), inter-organizational communication and collaboration problems in disaster situations often heighten stress levels for the response team managers. To assist in the possible prevention or amelioration of poor plan implementation in the face of a disaster, we propose the use of a meta-heuristic decision-support system tool to solve scheduling problems in near real-time. These scheduling problems are specifically addressed in the response and recovery phases of disaster management. By utilizing the appropriate decision support system, disaster management officials can mitigate resource allocation problems in a timely manner (Thompson et al. 2006).

2 Purpose

Altay and Green (2006) outline four phases of disaster operations management: mitigation, preparedness, response and recovery. Their disaster response phase involves activities such as implementing plans, establishing command posts and shelters, and provisioning of all necessary emergency services. In this phase time-critical activities must be scheduled and sequenced, and appropriate personnel must be assigned in a timely manner, otherwise severe late-penalties might accrue. Although the recovery phase (including cleanup, infrastructure repair and replacement, and continued provision to the displaced—Altay and Green 2006) may be slower paced, the need for efficiency and effectiveness is equally important.

Detailed preparation for potential disaster situations is a necessity, but once the actual disaster occurs, managers on the scene must often develop complex responses in a compressed time-frame without support from higher-level managers. The more overwhelming the disaster, the more likely the need for managers to adapt as the situation evolves. As Quarantelli (2006) pointed out, planning from the ground up by local response personnel is often more useful than a 'top–down' plan developed by senior managers that are removed from the scene. Thus, this paper will address the resource-constrained scheduling aspect of disaster response and recovery using a ground-up approach.

From an academic point of view, the resource allocation issues noted above may be conceptualized as a resource-constrained scheduling problem, and will be referred to in this paper as the disaster response (and recovery) scheduling problem, or *DRSP*. The *DRSP* can be classified as a complex variation of the multiple resource-constrained project scheduling problem (or *MRCPSP)*, simultaneously addressing both scheduling and assignment issues (see Section 4 for a detailed discussion). This paper proposes an efficient algorithm to solve the *DRSP* within the imposing deadlines that disaster management requires—indeed, a major contribution of this paper is the introduction of modeling and solution methods that enable near-real time solutions. This ability to deal simultaneously with scheduling and assignment in near-real time allows managers to re-solve a particular response or recovery problem as conditions at the scene change.

In the next section we examine the related literature in disaster management. In Section 4 we discuss how the DRSP maps as a *MRCPSP* and the unique characteristics of that class of problems. We present a solution technique for the *DRSP* in Section 5, and computational results are discussed in Section 6. We conclude the paper with a discussion of practical application, conceptual limitations, and ideas for future research in the final section.

3 Disaster management and resource constrained project scheduling

The early disaster management research utilizing management science tools focused on the location of emergency service facilities and equipment (Toregas et al. 1971; Kolesar and Blum 1973; Carter and Ignall 1970; Kolesar and Walker 1974). For example, Kolesar and Walker (1974) proposed a mathematical programming model and an associated heuristic algorithm to address the location of fire trucks in New York City. Going a step further, Brown and Vassiliou (1993) developed a decision support system that addressed methods for assigning tasks in a post-disaster relief effort. However, a critical element in disaster response and recovery is the need for appropriate sequencing (e.g. the need for a hazardous materials team to neutralize a chemical spill before allowing emergency medical teams to assist the injured), as well as recognizing the constraint of scarce resources. In this paper, we add the consideration of time-periodic scheduling of assignments, rather than just assigning people to tasks, which recognizes precedence as well as scarcity.

Realizing the need for emergency response planning as opposed to a reactionary response, Mendonca et al. (2001) proposed the use of group decision support systems. Bryson et al. (2002) extended this thinking by formulating a subplan selection problem (a limited part of the disaster recovery planning process) utilizing mathematical programming and discussed phases (or objectives) of this system that could be applied under various conditions or timings of a disaster. In this paper we are not concerned with a priori planning. Instead, this paper primarily addresses operational disaster response while recognizing that much of our work also applies to the disaster recovery phase.

In more recent works, disaster management has again been examined from the facility location perspective. Liu and Zhao (2007) discussed a logistic relief network consisting of relief suppliers, relief distribution centers, and relief demand areas. They proposed a multi-objective model for quick response to relief demands. In a more specific case, Özdamar and Yi (2008) discussed vehicle dispatch for disaster relief planning. They proposed a mathematical model, a solution procedure, and then demonstrated these on randomly generated problems on grid networks. Balcik and Beamon (2008) proposed a mixed integer programming formulation for facility location and stock pre-processing for disaster relief. Their model allows organizations to understand their response capacity, and make adjustments to their operations. While this approach examines an important aspect of disaster management, in this paper we are not explicitly concerned with facility location as we view locations as being tied to tasks.

4 Classifying and understanding the *DRSP*

As seen by previous work done in this area, disaster management is complex in that managers must assign specific personnel and resources to multiple locations and tasks. For instance, although ambulance crews, fire crews, police, SWAT teams, and the bomb squad all have basic first responder skills, each has specific specialty skills and specialized equipment that are most useful (or required) at a certain point in the disaster management effort. This type of complexity suggests the *DRSP* is a form of the multiple resource-constrained project scheduling set of problems, and occurs when personnel can perform the various tasks of assorted projects with differing time and cost requirements (Brucker et al. 1999). *MRCPSP* problems are subject to a multitude of operational constraints, such as workload and labor requirements, safety issues, logistics and equipment availability, which are subject to precedence constraints and ultimately will be associated with specific time-periods (Drexl 1991). As was seen in the response to Hurricane Katrina, in real-world situations task/personnel assignments are often made in an unstructured manner, usually without the aid of sophisticated software. The end result is a disjointed response that often results in increased delays and higher casualties. Thus, we see a need for using *MRCPSP*-related models in disaster management.

The *MRCPSP* is conceptually and computationally complex since it involves both an assignment of resources to tasks, as well as a schedule for executing the tasks. It is known to be NP-complete in the strong sense (see Garey and Johnson 1979). In fact, even generating a test problem instance for the *MRCPSP* problem is complex, and several papers have proposed problem generators to address this need (Kolisch et al. 1995; Drexl et al. 2000). Despite the large amount of research on the *MRCPSP* (Boctor 1996; Böttcher et al. 1999; Drexl and Gruenewald 1993; Hartmann 2001; Hartmann and Drexl 1998; Jozefowska et al. 2001; Kolisch and Drexl 1997; Mori and Tseng 1997; Özdamar and Alanya 2001; Özdamar 1999; Serafini and Speranza 1994), researchers may become confused over the details of this problem given that its literature is alive with variations of the problem and that the problem is structurally related to a class of machine scheduling problems. In response to this confusion, and as a tool for keeping track of new and relevant developments, several taxonomies of the problem domain have been proposed (Brucker et al. 1999; Herroelen et al. 2001).

Recent events such as Hurricane Katrina, the tsunami in Southeast Asia in 2004, the Chinese earthquake in 2008, and the Myanmar cyclone in 2008 demonstrate that time is becoming an increasingly critical factor for many of the real-world applications of this problem. Reducing the amount of time required to appropriately allocate needed resources to disaster sites can result in large amounts of precious resources being saved. The generalized *MRCPSP* is a case where the make-span objective (minimizing the project duration) is replaced by any other objective (Sprecher and Drexl 1998), i.e. a time/cost tradeoff. The time/cost tradeoffs allow "processing times to *vary* according to how much the planner is willing to pay for it" (Brucker et al. 1999).

In disaster management, this "willingness to pay" may be a direct function of available resources, and the objective may be to save lives rather than minimize cost. Since assigning a monetary value to a life is controversial, costs can be interpreted as an efficiency measure such as the increased probability of loss of life or other resources due to time delay. While managing the disaster response, the finding of scheduling efficiencies and the understanding of time/cost tradeoffs, especially during multiple disaster periods, are clearly critical issues for both the relief organizations and their response teams. The mapping between the elements of the *MRCPSP* and the *DRSP* are presented in Table 1.

The previous research literature in this area includes a wide range of papers (Summers 1972; Bolenz and Frank 1977; Balachandran and Zoltner 1981; Chan and Dodin 1986; Drexl 1990, 1991; Dodin and Chan 1991; Knechl and Benson 1991; Salewski and Bartsch 1994; Salewski 1995; Dodin and Elimam 1997; Salewski et al. 1997; Dodin et al. 1998). Using the mapping presented in Table 1, we interpret the formulation as presented in Dodin et al. (1998) and utilize it to represent the DRSP (see online Appendix A). The mixed linear-integer programming problem formulation seeks to assign disaster response (or recovery) teams to tasks to minimize total costs subject to a variety of constraints. The objective function included the cost of where response teams are performing each assigned task—typically including a high mismatch cost when the response teams are assigned improperly. The mismatch cost may include training costs or the risk associated with errors due to the lack of familiarity with the task in question. The objective function also includes a cost for transferring response teams and equipment to another disaster location (engagement); this is known as the setup cost. The setup cost is designed to capture the trade-off resulting from a response team switching from one location to another and the match-up between the response team and the tasks. The third cost component is a significant penalty for completing the project (or the last task of a project) after its due date, in this case a loss of life.

Constraints include precedence relations and response team availability and preferences. Constraints also include the fact that a response/recovery team cannot process more than one task at a time, and account for delays and team transitions. These cost components and constraints make *DRSP* structurally similar to *MRCPSP*s used in managing certain services, such as consulting, audit, and law firms discussed in the literature review, as it simultaneously incorporates both the budget problem (a fixed upper bound on the non-renewable resource) and the deadline problem (a bound on the project duration) as defined by Brucker et al. (1999).

Table 1 Mapping between *MRCPSP* and the *DRSP*

MRCPSP	Disaster Response Scheduling Problem (*DRSP*)
Project/engagements	Disaster locations
Assignment of personnel to task	Sufficient man-power
Mismatching issues	Skill sets for specialized jobs are assigned correctly
Sequencing of tasks (precedence)	Some tasks can only be started after other tasks have completed
Task overlap	Some tasks can overlap in time
Late costs	Loss of life
Setup costs	When moving personnel between tasks, a setup cost is incurred. This includes things like removal of expert personnel and their equipment, and re-insertion into other (geographic) areas.
Teams	Response/recovery teams (or rescue crews)
Planning horizon	The time horizon for the response/recovery project
Deadlines	Deadlines for certain aspect of the response/recovery efforts (such as finding survivors, supplying food, …)
Multiple jobs	A response/recovery effort could have multiple unique goals (survivor extraction, contamination minimization, housing provisioning)

Dodin et al. (1998) found that even modest instances of the *MRCPSP* become computationally intractable, and presented a solution technique based on the Tabu Search (*TS*) meta-heuristic. Their *TS* implementation utilized memory to direct or constrain the search process to find local neighborhood improvements given a starting point solution, but still experienced frequent and large optimality gaps. Thus, part of the focus of this paper is to develop a new solution procedure that addresses this weakness, and is appropriate to the time-sensitive requirements of the disaster management domain.

5 Using adaptive reasoning technique to provide periodic redirection to solve the *DRSP*

To overcome some of the weaknesses of neighborhood search heuristics, Patterson et al. (1999) and Patterson and Rolland (2002) proposed the *Adaptive Reasoning Technique* (*ART*). *ART* is a constructive, iterative, and memory-based meta-heuristic, and is illustrated in Fig. 1. A domain-specific solver generates a complete solution for the problem at each iteration of the *ART* algorithm. The term "solver" refers to a problem-class specific solution method, such as a greedy heuristic. If a solution found by the solver is sufficiently promising, a local search is executed to find a local optimum. *ART* is based on memory concepts such as learning, remembering, and forgetting which are used to adjust the behavior of the solver during subsequent iterations. In essence, the memory captures facets of the solver's performance from prior iterations, and this drives the modification and future behavior of the solver.

ART's memory is comprised of three components: 1) a short-term memory consisting of a list of prohibited solution choices; 2) a long-term memory containing the best solutions found; and 3) an operational memory of how to learn, including counters, memory length, and the parameters of memory manipulation and how to manipulate these learning parameters throughout the algorithm. *ART* learns about the behavior of the solver, and then imposes changes to prevent the solver from making seemingly myopic, or otherwise poor, choices. As shown in Fig. 1, *ART* is not static in the way it learns: the learning parameters are modified throughout the execution of the *ART* meta-heuristic algorithm. While the local search found in *ART* does manipulate the solutions found by the solver, *ART* manipulates the solver. We next discuss the elements of *ART* in more detail.

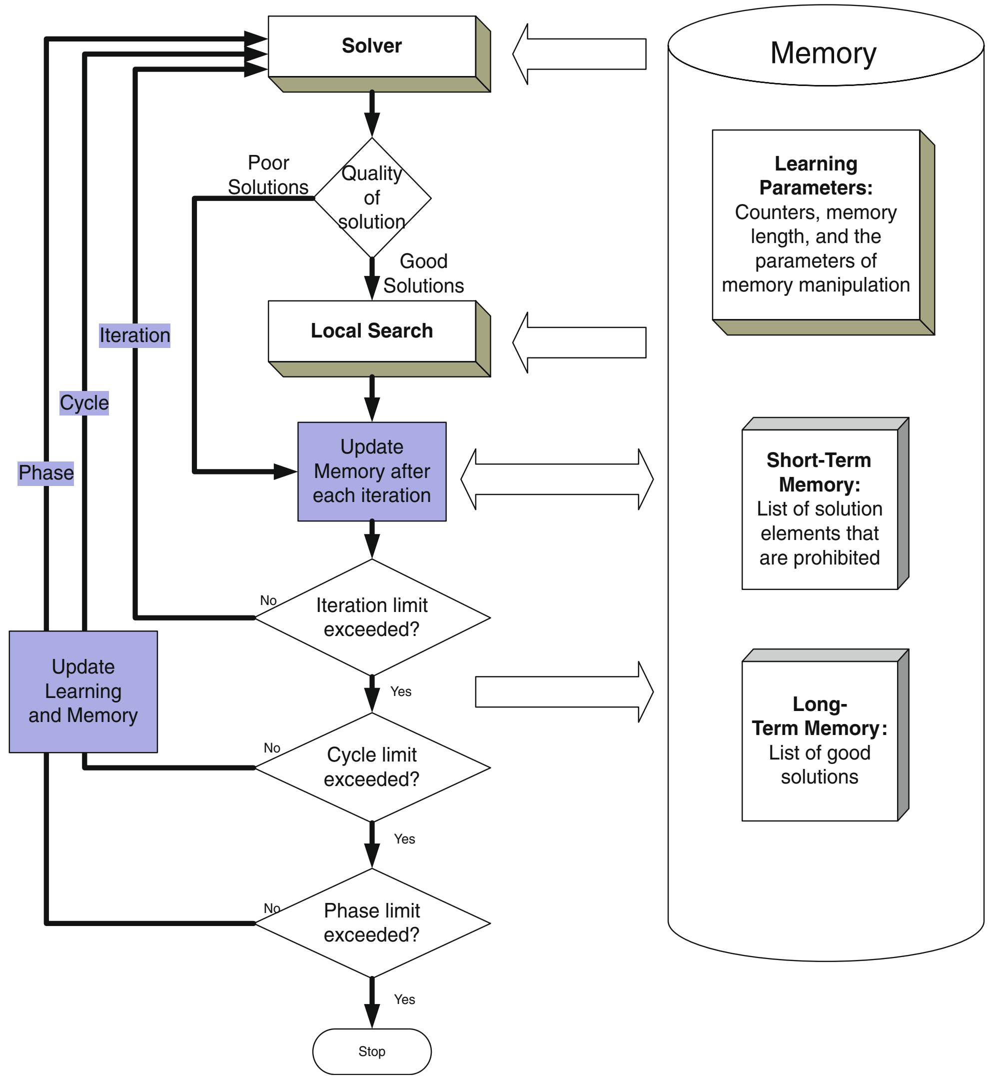
5.1 The solver

The solver used in *ART* can be any algorithm that constructs a solution for the problem. We utilize the Forward Loading (*FL*) heuristic, a simplistic and well-known procedure for solving *MRCPSP*s (Dodin et al. 1998) as our solver. *FL* uses a dispatching rule that assigns tasks to available responder/response teams from the beginning to the end of the available time horizon, giving preference to the responder/response team with the least mismatching cost subject to capacity, precedence, and any potential additional constraints.

5.2 Local search

Local search is performed using two moves: within-responder/response team (horizontal) moves; and between-responder/response team (vertical) moves. Both moves are performed on the basis of maximum cost savings, subject to memory restrictions and feasibility. The within-responder/response team move is an attempt to find a better time slot

Fig. 1 The principles of ART



for a task already assigned to the responder/response team. The between-responder/response team move is an attempt to find a new responder/response team for a task currently assigned to somebody else. For additional details, see Dodin et al. (1998). Because local optimization is very time consuming, *ART* only performs this local optimization when the solution produced by the solver is better than any of the previously found 10 best post-local optimization solutions.

5.3 Updating memory and learning parameters

ART consists of three (3) loops to modify the memory that imposes restrictions on the operation of the solver (see Fig. 1): *iteration loops* (repeat until 35 consecutive iterations of the solver have been performed without an improvement in the best solution); 7 *cycle loops*; and 4 *phase* loops. Memory is adjusted after each iteration by prohibiting selected decision variables. With a random probability larger than a threshold T, a response team i selected to process task j in period h will be prohibited from being selected for a certain number of iterations as follows:

$$\textit{Length of ART Prohibition} = rand[0,1]*|I_j|*d*(l_j-\theta_j)*\left(\frac{c_{ij}}{\max\limits_k\{c_{kj}\}}\right)$$

where:

- *rand* [0,1] = a random number in the range [0, 1]
- $|I_j|$ = the number of response teams who can perform task j
- d = the depth-of-learning multiplier. The depth-of-learning is a multiplier used for determining how long a prohibition will be maintained.
- $(l_j\text{-}\theta_j)$ = latest minus earliest completion time for task j
- $\left(\frac{c_{ij}}{\max\limits_k\{c_{kj}\}}\right)$ = the mismatching cost of response team i performing task j divided by the maximum mismatching cost for any response team k performing task j

The learning parameters are adjusted after each cycle. The threshold T is initially set to 3% at the beginning of each phase, and is increased by 5% at the end of each cycle within a phase. Threshold modification is done to vary the difficulty of prohibiting a choice as the algorithm proceeds. Thus, T is reset to 3% at the end of each phase. Varying T up and down is analogous to the "heating and cooling" typically performed in simulated annealing.

Memory and learning parameters are adjusted after each phase. In addition to resetting T to 3%, the *ART* memory is reset to the *ART* memory that produced the best solution, with a 20% probability, the length in iterations of the best *ART* memory will be "forgotten" by a random proportion in the range [0, 1]. Resetting the *ART* memory allows the algorithm to go back and begin the search anew from the seemingly most fruitful location. Reducing the length of prohibition is akin to memory loss, or at least to reducing the intensity of the memory, and allows the algorithm to more rapidly search new areas.

The depth-of-learning multiplier is set initially to 1 and it is increased by 15% at the end of each phase. The depth-of-learning multiplier determines how many iterations a prohibition will be maintained; increasing the multiplier serves to increase the length of prohibition. In essence, memory becomes more intense, or longer, as the algorithm proceeds.

6 Experimental results

For consistency and comparability, the experiments are conducted using the same experimental data used by Dodin et al. (1998). Five (5) different types of data sets are used, with seven (7) problem instances in each data set for a total of 35 problems. For each of the 35 problems, the algorithm was run 50 times using *ART* with *FL* and 10 times using *ART* with *TS*. All experiments were coded using Delphi Studio Architect 7.0, and run on a dual core processor (Intel 6400 @ 2.13 GHz) Dell XPS computer with 4 Gb of RAM, under Windows XP Professional. Tables 2, 3, and 4 contain data describing the 5 problem sets and the 7 problem instances within each set as reported in Dodin et al. (1998).

Table 5 displays the percentage gap comparison using the best *ART* solution values and the best reported solution values in Dodin et al. (1998). The choice of solver does have a significant impact on the performance of *ART*. The *ART* procedure in Table 5 uses *FL* as the solver for problem sets 1, 2, 4, and 5, and *TS* as the solver for problem set 3. We found that for data sets 1, 2, 4, and 5, *ART* performed best using the Forward Loading (*FL*) solver. For data set 3, *ART* performed best using *TS* as the solver. The reason for this interesting dichotomy is due to the extreme nature of problem set 3. Problem set 3 consists of small, randomly generated problems with a large amount of flexibility due to the minimal precedence relations. As a result of the lack of precedence relations, *FL* performs very poorly on problem set 3. *TS*, however, serves as a good solver on problem set 3 for *ART*.

As shown in Table 5, *ART* found the best-known upper bound (best feasible solution) in 83% of the cases. The average percentage gap from the best upper bounds for each problem set is also extremely small for *ART*, ranging from 0% to 2.2%. This is in comparison to an average percentage gap ranging from 3.9% to 52.0% for *TS*. These computational results clearly demonstrate that the *ART* procedure is able to make dramatic improvements in solution quality. The computational times reported in Table 5 indicate that these dramatic results can be achieved in a reasonable amount of time. On average, problem set 1 took 1.6 seconds per iteration, set 2 took 12.31 seconds, set 3 took 7.96 seconds, set 4 took 7.44 seconds, and the largest problem set (set 5) took 59.39 seconds of CPU time per iteration on average.

7 Conclusions

This paper proposed the use of resource-constrained project scheduling models for disaster response and recovery. We discussed the disaster response scheduling problem, and proposed an efficient algorithm to solve the *DRSP* within the reasonable time limits consistent with the requirements of disaster management efforts. Our research advances operations management theory in several ways. First, we present a theoretical link, showing that the *DRSP* is a multi-mode case of the resource-constrained project-scheduling problem. Second, we propose a method by which this complex version of the *MRCPSP* problem, the *DRSP*, can be solved in near-real time, enabling the generally complex

Table 2 Experimental problem profile

Problem Number	Instances	(G) Engagements	(I) Respondents/Teams	(J) Tasks	(H) Horizon
1	7	2	4	19	12
2	7	2	4	35	35
3	7	2	4	20	15
4	7	4	6	32	20
5	7	5	6	75	45

Table 3 Description of problem set characteristics

Problem Characteristics					
Problem Number	Problem Size	Precedence Relations	Planning Horizon	[ej,lj] Intervals	Source of Problem
1	Small	Rigid	Short	Narrow ~ 6 periods	Real life
	Notes:	*Real life case studies presented in Dodin and Elimam* (1997)			
2	Larger than #1	Similar to #1	Longer than #1	Wider than #1 ~ 20 periods	Pseudo real life
	Notes:	*Wider intervals provide more feasible schedules to choose from*			
3*	Small, similar to #1	Minimal	Short, similar to #1	In between #1 and #2	Randomly generated
	Notes:	*Tasks can be processed in parallel and more flexibility in problem due to minimal precedence relations*			
4*	Larger than #1, similar to #2	More than #3	Longer than #3	In between #1 and #2	Randomly generated
	Notes:	*More respondents makes #4 less restrictive than #2*			
5*	Very large, ~ 22 times	Similar to #4	Longer than #4	In between #1 and #2	Randomly generated
	Notes:	*Similar to #4, only much larger*			

*While problem sets #3–#5 are randomly generated, the problem parameters are similar to, and based on, real-life problem instances

multiple resource-constrained models to be utilized in disaster response and recovery. The ability to deal simultaneously with the combination of scheduling and assignment in near-real time also translates to an ability to re-solve the problem as conditions on the ground change.

For practitioners, this proposed model and solution method constitute a decision-support system that can be readily used by managers dealing with the various aspects of disaster response and recovery. Given sufficient experience in the field, these managers will be able to provide the necessary data (types of teams, resources, and rule of thumb guidelines for the necessary time to complete certain tasks) to develop a complex schedule for assignment of resources and personnel. This schedule can be re-solved as data and parameters evolve due to changing conditions. The model allows these solutions to be generated in real time and will likely serve to prevent some of the untimely responses that have characterized many recent disaster management efforts.

The results presented in this paper also suggest significant time and cost savings may accrue and improved operational effectiveness between multiple disaster management organizations may be obtained. Assuming that the teams and resources are scattered throughout multiple organizations, our method facilitates inter-organizational collaboration and reduces uncertainty and the noise resulting from this uncertainty in the communication channels. Our proposal involves a modeling and solution tool that would allow for coordinated planning, by a single agency or command center, to achieve a desirable inter-organizational result. Based on recent efforts by Congress to improve disaster management in the wake of the Katrina disaster, the creation of a unified Joint Field Office (as a part of the Incident Command System) providing this coordination is a near-term possibility (National Response Framework 2008).

For future researchers in this area, the paper demonstrates the use of simple heuristics (*Forward Loading*) and meta-heuristics (*Tabu Search*) as the solvers for the *Adaptive Reasoning Technique* in order to solve the *DRSP*. The *ART* algorithm is unique in that it modifies the decision tree of the solver. This insight will be useful to researchers in the meta-heuristics area, as it provides a unique way of thinking regarding meta-heuristic design. A promising area of future research is to design meta-heuristics that manipulate the decision tree process of a problem-class-specific solution method. Also, additional explorations of *ART*'s learning and memory functions are expected to be fruitful

Table 4 Description of characteristics for all problem sets

Problem Instance	Instance Characteristics
0	The base problem
1	As in #0 except the parameters $H^-_i = 0$, and $H^+_i = H$ (The respondent/team utilization constraints are ignored)
2	Lateness penalty is zero, otherwise same as #0
3	Lateness penalty is set high (~10 times that of #0)
4	Higher mismatching costs (c_{ij}), such that $c_{ij} > 0$ for all (i,j)
5	High setup costs as respondent/teams move between engagements
6	Low setup costs as respondent/teams move between engagements

Table 5 Computational results

Problem Name	Best Upper Bound	Percentage gap from best upper bound		Average ART CPU Time in seconds
		TS 1998 Dodin et al.	Best ART	
1.0	6,300	25.4%	0.0%	1.56
1.1	6,000	21.7%	0.0%	1.59
1.2	2,900	24.1%	0.0%	1.62
1.3	20,300	6.4%	0.0%	1.65
1.4	14,800	0.0%	0.7%	1.62
1.5	13,200	58.3%	0.0%	1.67
1.6	4,280	6.5%	0.0%	1.55
Average		20.4%	0.1%	1.61
2.0	11,800	33.9%	0.0%	12.18
2.1	10,600	49.1%	0.0%	12.71
2.2	11,600	41.4%	0.0%	12.62
2.3	11,600	41.4%	0.0%	12.15
2.4	14,600	52.1%	0.0%	12.86
2.5	22,600	40.7%	0.0%	12.17
2.6	9,340	43.5%	0.0%	11.45
Average		43.1%	0.0%	12.31
3.0	83	0.0%	2.4%	8.05[a]
3.1	78	6.4%	0.0%	7.76[a]
3.2	83	0.0%	2.4%	8.03[a]
3.3	85	232.9%	0.0%	8.04[a]
3.4	144	0.0%	0.0%	7.85[a]
3.5	85	57.6%	0.0%	8.06[a]
3.6	85	67.1%	0.0%	7.95[a]
Average		52.0%	0.7%	7.96
4.0	139	0.0%	2.9%	8.36
4.1	139	0.0%	5.0%	6.37
4.2	138	0.7%	0.0%	8.58
4.3	137	1.5%	0.0%	7.77
4.4	180	12.2%	0.0%	7.59
4.5	369	13.0%	0.0%	4.75
4.6	111	0.0%	7.2%	8.69
Average		3.9%	2.2%	7.44
5.0	49,700	29.8%	0.0%	59.55
5.1	46,400	7.1%	0.0%	57.55
5.2	48,000	29.4%	0.0%	59.74
5.3	55,000	2.9%	0.0%	60.07
5.4	78,200	7.4%	0.0%	60.07
5.5	136,000	10.3%	0.0%	58.65
5.6	36,000	34.7%	0.0%	60.14
Average		17.4%	0.0%	59.39

[a] ART with TS

avenues for future research. We note that the *ART* heuristic improved substantially on most of the solution values for known problems presented in previous research.

This paper also provides a strong argument for researchers desiring to use advanced meta-heuristic algorithms in disaster management research. The computational results suggest that there is room for future research, and further improvements, in the solution procedures for this very important real-life problem. Specifically, since *ART* is a relatively new technique, it could benefit from additional examination and utilization across new problem domains as well as other *MRCPSP* variants. An example of a very practical variant of the *DRSP* is an examination of the tradeoff between the cost of hiring permanent versus temporary personnel and other needed resources and the potential penalty for lack of preparedness. The value of our proposed model and solution procedure is one of quickly assessing a disaster scenario and adapting to the dynamic nature of the situation in a timely manner.

A unique challenge to the implementation of the decision support model in practice exists in that perfect information is typically not available for disaster management planning. All that is known, at best, are estimates. We would argue that for many issues, the latest completion times can be estimated. For example, we do have good measures for life-expectancy for earth-quake survival—meaning a time beyond which search for survivors does not make sense. Similarly, there are time measures for supplying drinking water, corpse removal, and other necessary assistance provision activities. The minimum time between tasks could be spelled out by the need for moving equipment between locations, which is an issue due to resource constraints. As such, there are estimates and experiences to aid in this regard. Earliest completion times are estimates as to when a task will end given assumptions regarding the situation on the ground. The strength of our model and solution approach is indeed that the model can be quickly re-solved by the disaster coordinator at any time should more accurate information become available, or should the situation on the ground change significantly—which is customary for these types of situations.

Limitations in this area of research are twofold. First, finding good real-life datasets is difficult because of a lack of a standard formal representation that is easily understood by managers. This hampers both the documentation of real-life projects, as well as collection of data from such projects. Second, the setup of the problem structure in order to enable the *ART* heuristic is complex. Indeed, this shows that there is ample room for integration of decision support systems that aid the disaster management team in setting up the problem with the *ART* method.

Appendix A Problem formulation

Parameters

A_{ij} The set of response tasks, which can be processed by response team i after they completes task j.

a_{jk} Portion of task j to be completed before the start of task k, where j precedes k, and $0 < a_{jk} \leq 1$.

A_j The set of tasks that directly succeed task j.

B_j The set of tasks that are connected to task j by a direct precedence relationship.

c_{ij} The mismatching cost between response team i and task j.

d_g Due date of engagement g.

e_j The earliest completion time of task j: $e_j = \max_{k \in B_j} \left\{ e_k - FS_{kj} + \min_{i \in I_j} [t_{ij}] \right\}$.

FS_{jk} The minimum number of time units that must transpire from the completion of j prior to the start of k; if $FS_{jk} = 0$, then we have the usual CPM/PERT. $FS_{kj} = t_{ik} - SS_{kj}$

g 1, 2,..., $|G|$, response engagement index, where G is the set and $|G|$ is the number of engagements.

h 1, 2,..., $|H|$, time period index where H is the planning horizon and $|H|$ is the number of periods in the planning horizon.

H_i^- Minimum number of periods response team i is available within H.

H_i^+ Maximum number of periods response team i is available within H.

i 1, 2,..., $|I|$, responder/response team index, where I is the set and $|I|$ is the number of response teams.

I_j The set of response teams nominated for task j.

j 1, 2,..., $|J|$, response task index, where J is the set and $|J|$ is the number of all response tasks in all engagements.

J The set of response tasks which can be processed by response team i.

l_j The latest completion time of task j: $l_j = \min_{k \in A_j} \left\{ e_k - \min_{i \in I_k} [t_{ik}] + FS_{jk} \right\}$.

p_g Cost of having engagement g late (tardy) for one period.

s_{ijk} The cost of having response team i process task k after completing task j.

SS_{jk} The time lag required for the start of task k after the start of task j. The overlapping relationship between two related activities, j and k, is given by the ratio a_{jk}. Task k can start any time after the completion of $a_{jk}\, t_{ij}$ periods of work on task j. Hence, $SS_{jk} = \lceil a_{jk} t_{ij} \rceil$, where $[q]$ is the smallest integer greater than or equal to q. The precedence relationship becomes strict when $a_{jk} = 1$.

t_{ij} The time it takes response team i to process response task j.

Decision variables

$$x_{ijh} = \begin{cases} 1 & \text{if response team } i \text{ completes task } j \text{ at the end of period } h; \\ 0 & \text{otherwise.} \end{cases}$$

$$z_{ijk} = \begin{cases} 1 & \text{if response team } i \text{ completes task } k \text{ after task } j; \\ 0 & \text{otherwise.} \end{cases}$$

Integer program formulation

$$\min f(\underline{x},\underline{z}) = \sum_{i=1}^{I}\sum_{j\in J}\sum_{h=e_j}^{l_j} c_{ij}x_{ijh} + \sum_{i=1}^{I}\sum_{j\in J}\sum_{k\in A_{ij}} s_{ijk}z_{ijk} + \sum_{i=1}^{I}\sum_{g=l}^{G}\sum_{h=d_{g+1}}^{H} p_g(h-d_g)x_{iM_gh} \quad (1)$$

where M_g is the last task in engagement g.

The model contains five sets of constraints:

Availability of response teams and the requirement that each response team must work no less than H_i^- periods and no more than H_i^+ periods within the planning horizon H. Therefore, for each response team i we have:

$$\sum_{j\in J}\sum_{h=e_j}^{l_j} t_{ij}x_{ijh} \geq H_i^- \quad (2)$$

$$\sum_{j\in J}\sum_{h=e_j}^{l_j} t_{ij}x_{ijh} \leq H_i^+ \quad (3)$$

Each response task must be processed by one response team. Therefore for each task j we have the constraint

$$\sum_{i\in I_j}\sum_{h=e_j}^{l_j} x_{ijh} = 1 \quad (4)$$

A response task cannot start before its preceding requirements are satisfied. Consequently for every task j

$$\sum_{h=e_k}^{l_k}\sum_{i\in I_k}(h - FS_{kj})x_{ikh} - \sum_{h=e_j}^{l_j}\sum_{i\in I_j}(h-t_{ij})x_{ijh} \leq 0 \quad (5)$$

for all $k \in Bj$

A response team cannot process more than one task at a time. A task j is processed by response team i in period h if it is completed in period r where

$h \leq r \leq h + t_{ij} - 1$, and $e_j \leq r \leq l_j$

Therefore, let $u_j = \max\{e_j, h\}$ and $v_j = \min\{l_j, h + t_{ij} - 1\}$, then for each response team i:

$$\sum_{j\in J_i(h)}\sum_{r=u_j}^{v_j} x_{ijr} \leq 1 \qquad \text{for all } h = 1,2,\ldots,H \quad (6)$$

where $J_i(h)$ is the set of tasks response team i can process in period h.

Setup costs: Whenever a response team changes between response tasks, a setup cost is incurred. The setup cost depends on the response team and on the "from" and the "to" tasks. Therefore, as response team i completes task j, they might be assigned to a task $k \in A_{ij}$:

$$z_{ijk} - \sum_{r=\theta_k}^{l_k} x_{ikr} \leq 0 \qquad \text{for all } k \in A_{ij} \quad (7.a)$$

where $\theta_k = \max\{e_k, e_j + t_{ik}\}$

$$\sum_{k\in A_{ij}} z_{ijk} - \sum_{r=e_j}^{l_j} x_{ijr} = 0 \qquad \text{for all } i \text{ and } j \quad (7.b)$$

The analysis of constraint set (7.b) indicates that the response team, after completing their assignments, must exit the project. Consequently, for each response team i we add a dummy task, ∂_i, at the end of the project. This task is also added to the appropriate sets A_{ij}, but not to the set J_i. To guarantee that ∂_i is the last task response team i is assigned to, and that this task is scheduled after the completion of all the engagements, the following constraints must be observed for each $i \in I$:

$$\sum_{e_{M_g}}^{l_{M_g}}\sum_{i\in I_{M_g}} h x_{iM_gh} - Hx_{i\partial_iH} \leq 0 \qquad \text{for all } g \in G \quad (7.c)$$

Analyzing constraint sets (7.a) and (7.b) further indicates the possibility of cycling in the changeover decision variable z. To eliminate cycling, the following constraint sets must be enforced:

$$\sum_{j\in J_i} z_{ijk} \leq 1 \text{ for all } i \in I,\ k \in J_i' \quad (7.d)$$

$$\sum_{k\in J_i} z_{ijk} \leq 1 \text{ for all } i \in I,\ k \in J_i \quad (7.e)$$

$$\sum_{r=e_k}^{l_k} r\,x_{ikr} - \sum_{r=e_j}^{l_j} r\,x_{ijr} \geq t\,z_{ijk} - M(1 - z_{ijk}) \text{ for all } k \in A_{ij} \quad (7.f)$$

where, $J_i' = J_i \cup \partial_i$ and M is a large number.

Constraint sets (7.a) and (7.b) are dependent on the set A_{ij}. Theoretically, the set A_{ij} consists of all the activities in the set J_i . However, early and late completion times, along with precedence, can be used to reduce the size of such set. First, a task k ε A_{ij} only if $e_j + t_{ik} \leq l_k$. Then, the interval for the completion time of each element k ε A_{ij} is determined. The earliest completion time of k, denoted by θ_k, is given by

$$\theta_k = \max\{e_k, e_j + t_{ik}\}$$

The latest completion time of task $k \varepsilon A_{ij}$ is l_k, which is determined as stated above.

References

Altay N, Green WG (2006) OR/MS Research in disaster operations management. Eur J Oper Res 175:475–493

Balachandran BV, Zoltner AA (1981) An interactive audit-staff scheduling decision support system. Account Rev 56:801–812

Balcik B, Beamon BM (2008) Facility location in humanitarian relief. Int J Logist: Research and Applications 11:101–121

Boctor FF (1996) A new and efficient heuristic for scheduling projects with resource restrictions and multiple execution modes. Eur J Oper Res 90:349–361

Bolenz G, Frank R (1977) Das Zuordnungsproblem von Prüfern zu Prüffeldern unter Berücksichtigung von Reihenfolgebedingungen—Ein Lösungsansatz der binären Optimierung. Z Betriebswirtschaftliche Forschung 29:427–447

Böttcher J, Drexl A, Kolisch R, Salewski F (1999) Project scheduling under partially renewable resource constraints. Manage Sci 45:543–559

Brown G, Vassiliou A (1993) Optimizing disaster relief: real-time operational and tactical decision support. Nav Res Logist 40 (1):1–23

Brucker P, Drexl A, Mohring R, Neumann K, Pesch E (1999) Resource-constrained project scheduling: notation, classification, models, and methods. Eur J Oper Res 112:3–41

Bryson K, Millar H, Joseph A, Mobolurin A (2002) Using formal MS/OR modeling to support disaster recovery planning. Eur J Oper Res 141:679–688

Carter G, Ignall E (1970) A simulation model of fire department operations. IEEE System Science and Cybernetics 6:282–293

Chan KH, Dodin B (1986) A decision support system for audit staff scheduling with precedence constraints and due dates. Account Rev 61:125–140

Dodin B, Chan H (1991) Application of production scheduling methods to external and internal audit scheduling. Eur J Oper Res 52:267–279

Dodin B, Elimam AA (1997) Audit scheduling with overlapping activities and sequence dependent setup costs. Eur J Oper Res 97:22–33

Dodin B, Elimam AA, Rolland E (1998) Tabu search in audit scheduling. Eur J Oper Res 106:373–392

Drexl A, Nissen R, Patterson JH, Salewski F (2000) ProGen/πx—An instance generator for resource-constrained project scheduling problems with partially renewable resources and further extensions. Eur J Oper Res 125:59–72

Drexl A, Gruenewald J (1993) Multi-mode resource constrained project scheduling. IIE Trans 25:74–81

Drexl A (1991) Scheduling of project networks by job assignment. Manage Sci 37:1590–1602

Drexl A (1990) Planung Des Ablaufs von Unternehmensprüfungen. Poeschel, Stuttgart

Garey MR, Johnson DS (1979) Computers and intractability: a guide to the theory of Np-completeness. Freeman, San Francisco

Hartmann S (2001) Project scheduling with multiple modes: a genetic algorithm. Ann Oper Res 101:111–135

Hartmann S, Drexl A (1998) Project scheduling with multiple modes: a comparison of exact algorithms. Networks 32:283–297

Herroelen W, Demeulemeester E, De Reyck B (2001) A note on the paper "Resource-constrained project scheduling: notation, classification, models and methods" by Brucker et al. Eur J Oper Res 128:679–688

Jozefowska J, Mika M, Rozycki R, Waligora G, Weglarz J (2001) Simulated annealing for multi-mode resource-constrained project scheduling. Ann Oper Res 101:137–155

Knechl WR, Benson HP (1991) An optimization approach for scheduling internal audits of divisions. Decis Sci 38:1245–1263

Kolesar P, Walker WE (1974) An algorithm for the dynamic relocation of fire companies. Oper Res 22:249–275

Kolesar P, Blum EH (1973) Square root laws for fire engine response distances. Manage Sci 19:1368–1378

Kolisch R, Drexl A (1997) Local search for non-preemptive multi-mode resource-constrained project scheduling. IIE Trans 29:987–999

Kolisch R, Sprecher A, Drexl A (1995) Characterization and generation of a general class of resource-constrained project scheduling problems. Manage Sci 41:1693–1703

Liu M, Zhao L (2007) A composite weighted multi-objective optimal approach for emergency logistics distribution. Proceedings of the 2007 IEEE International Conference on Industrial Engineering and Engineering Management 968–972

Mendonca D, Beroggi GEG, Wallace WA (2001) Decision support for improvisation during emergency response operations. Int J Emerg Manage 1:30–39

Mori M, Tseng CC (1997) A genetic algorithm for multi-mode resource constrained project-scheduling problem. Eur J Oper Res 100:134–141

National Response Framework (2008) Department of homeland security. Accessed on November 20, 2009 from http://www.fema.gov/emergency/nrf/index.htm

Özdamar L, Yi W (2008) Greedy neighborhood search for disaster relief and evacuation logistics. IEEE Intell Syst 23:14–23

Özdamar L, Alanya E (2001) Uncertainty modeling in software development projects (with case study). Ann Oper Res 102:157–178

Özdamar L (1999) A genetic algorithm approach to a general category project scheduling problem. IEEE Trans Sys Man & Cybern—Part C: Applications And Reviews 29:44–59

Patterson RA, Rolland E (2002) Hybrid fiber co-axial network design. Oper Res 50:538–551

Patterson RA, Pirkul H, Rolland E (1999) A memory adaptive reasoning technique for solving the capacitated minimum spanning tree problem. Journal of Heuristics 5:159–180

Quarantelli EL (2006) Catastrophes are different from disasters: some implications for crisis planning and managing drawn from Katrina. Disaster Research Center, University of Delaware. Retrieved December 1, 2009 from http://understandingkatrina.ssrc.org/Quarantelli/

Quarantelli EL (1988) Disaster crisis management: a summary of findings. J Manag Stud 25:373–385

Salewski F, Schirmer A, Drexl A (1997) Project scheduling under resource and mode identity constraints: model, complexity, methods, and application. Eur J Oper Res 102:88–110

Salewski F, Bartsch T (1994) A comparison of genetic and greedy randomized algorithms for medium-to-short-term audit-staff scheduling, Working Paper No. 356, Department of Production and Logistics, Institute for Management, Christian-Albrechts University at Kiel, Germany

Salewski F (1995) Hierarchische Personaleinsatzplanung in Wirtschaftsprüfungsgesellschaften. Physica, Heidelberg

Serafini P, Speranza MG (1994) A decomposition approach in a DSS for a resource constrained scheduling problem. Eur J Oper Res 79:208–219

Sprecher A, Drexl A (1998) Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm. Eur J Oper Res 107:431–450

Summers EL (1972) The audit staff assignment problem: a linear programming analysis. Account Rev 47:443–453

Thompson SJ, Altay N, Green WG, Lapetina J (2006) Improving disaster response efforts with decision support systems. Int J Emerg Manage 3:250–263

Toregas C, Swain R, ReVelle C, Bergeman L (1971) The location of emergency service facilities. Oper Res 19:1363–1373

U.S. House of Representatives (2006) A failure of initiative: final report of the select bipartisan committee to investigate the preparation for and response to Hurricane Katrina, Retrieved May 5, 2008 from http://www.gpoaccess.gov/Katrinareport/mainreport.pdf