

# An Event-Based Task Framework for Disaster Planning and Decision Support

Adriaan ter Mors, Jeroen Valk, Cees Witteveen

Faculty of Electrical Engineering, Mathematics and Computer Science,  
Delft University of Technology,  
{a.w.termors, j.m.valk, c.witteveen}@ewi.tudelft.nl

## ABSTRACT

Because of the apparent ineffectiveness of current disaster plans, we focus our research on modeling emergency response activities. If we can capture the crucial concepts of emergency response in a mathematical framework and apply this framework to construct disaster plans, then we pave the way for the development of automated decisions support systems for emergency response.

## Keywords

disaster planning, decision support, events, task framework

## INTRODUCTION

Many studies concerning disaster planning have found that actual operations in response to a crisis do not follow the instructions specified in the disaster plan. To gain better control over the emergency response operations, we should replace disaster plans in their current form – long, paper-based, and difficult to read – with disaster plans in a format that would allow more emergency responders to easily extract the information that is relevant for them. In our research, we have concerned ourselves with determining what the crucial concepts are in emergency response, and we are developing a mathematical framework in which these concepts can be modeled both clearly and intuitively. For one thing, we should be able to express which tasks should be performed, but also how one task depends on or is related to another, as dependencies between tasks will determine whether coordination and communication will be required to perform the tasks; furthermore, we should also be able to specify when and under which circumstances a particular task should be performed.

In our view, a disaster plan specified in a formal framework can form an integral part of an *automated decision support* system for emergency response – such a system could assist in deciding when a certain task should be performed, who should perform it, and which actions should be taken in case the actions planned in the disaster plan become inappropriate or impossible. Note that some researchers have suggested that disaster plans might be disbanded altogether in view of their ineffectiveness, and that research should focus instead on e.g. encouraging improvisation [5] and on improving communication between emergency responders [1]. However, we need an explicit disaster plan representation for the type of decision support system we envisage. Therefore, we view the research into e.g. improvisation and communications as complementary to our research.

Our research is still very much in the preliminary stages, so in the next section we will elaborate on our research goals that we have introduced above: to specify disaster plans in a formal framework, and to provide decision support using an information system based on this modeling framework. In the remainder of the paper, we will illustrate the use of our framework using a number of examples, but we do not present the framework in its entirety.

## RESEARCH GOALS

### Better disaster plans

We cannot give an objective answer to the question “what is a good disaster plan?” – and this issue hasn’t been thoroughly analyzed as far as we know<sup>1</sup> – but we can highlight the shortcomings that we have come across in current disaster plans. We have studied disaster plans from the Dutch municipalities of Amsterdam (2003), Sevenum (2000), Volendam (1994), and Eindhoven (1993), and we have noted the following points:

- disaster plans are long, paper-based, and difficult to read;

---

<sup>1</sup> In [3], recommendations are given with regard to how disaster plans should be structured, and what information they should contain.

- information on one particular subject is dispersed throughout the document, some pieces of information are often repeated;
- it is unclear which parts of the disaster plan are meant for which emergency responders;
- how one activity depends on another is often specified vaguely, and in very general terms.

When we consider the points above, it is not surprising that studies have shown that only a handful of emergency responders are usually aware of the existence of the disaster plan – let alone of its contents [2,7]. To remedy this problem, we recommend that a disaster plan format should be concise and easily understandable, so that the majority of the emergency responders for which the disaster plan is relevant can become familiar with its contents.

What we propose is to define a formal framework in which to specify those parts of the disaster plan concerning the actual emergency operations. Such a framework would specify clearly the tasks to be performed, and how the execution of one task depends on the execution of another. Furthermore, an emergency response information system based on the formalized disaster plan would be able to provide each individual emergency responder with the information he (or she) needs, while shielding him from any information he doesn't need.

### Execution monitoring and plan repair

In the aftermath of a disaster it often – if not always – turns out that the emergency response operations did not follow the instructions specified in the disaster plan. Naturally, we would like to make this observation earlier, namely when there is still time to do something about it. We envisage a system that, based on the formalized disaster plan it has stored internally, *monitors* the emergency response operations and can notify an emergency responder when a certain task must be performed.

Of course, deviations from the disaster plan are not just caused by forgetfulness on the part of emergency responders. Frequently, *contingencies* arise that disrupt normal execution of the disaster plan; for example, equipment can break down, fires can spread, buildings can collapse, etc. In general, emergency response is one area where you should expect the unexpected. In the face of these uncertain conditions, one inclination might be to abandon all notions of planning, and focus instead on providing ideal circumstances for improvisation [5]. Although recognizing the need for improvisation is a good idea, we feel it can be complemented by contingency planning.

The disaster plan can explicitly plan for a contingency by specifying that if a certain *event* occurs, then we should perform a certain set of tasks. Contingency planning can also be done more implicitly; for example, we can specify that a certain task can be performed in more than one way. If one way of performing the task fails, the other *method* can be tried instead. In this way, the disaster plan is made more robust.

### Dynamic task allocation

Having specified the tasks that need to be performed in the disaster plan, we then need to decide who should perform which task. Sometimes, the disaster plan directly specifies who should perform a certain task, sometimes the responsibility for a task is even stipulated by law – though experience shows that in both cases this is no guarantee that this person will actually perform the specified task [2,7]. We should note, however, that assigning responsibilities beforehand can only be done for very high-level tasks, e.g., the fire department should be responsible for the task fight fire. Moreover, several studies have shown [6] that in emergency situations many people and organizations lend a hand. In short, which *agents*<sup>2</sup> are available – and which are not – is not predictable in advance. Thus, especially in situations when large numbers of relatively unknown agents are involved, we have to solve a problem of *dynamic task allocation*.

There are many factors to take into account when we must decide which tasks to allocate to which agents. First, and obviously, we must know which agents are available to perform a certain task. Each agent may have a number of characteristics that affect the suitability of this agent for the task at hand, such as its *capabilities*, its *current location* (in relation to where the agent should perform the task), and its *reputation* (cf. [4] for research into the use of reputation mechanisms for agents). In addition, we must take into account that it is not always ideal to assign a task to the most suitable agent. Once an agent has been allocated a task, it will be temporarily unavailable to perform other, perhaps more pressing tasks.

A second issue we have to consider when deciding how to allocate a task is in what way we will perform a task. In many cases, there is more than one way to perform a task (for example, we can extinguish a fire using water but also using foam), and each method of performing a task will have its own characteristics, such as:

---

<sup>2</sup> An agent is an autonomous actor, which in the setting of emergency response may correspond to an individual emergency responder, but also to any group of emergency responders, such as a specific police department division.

- the duration of the method;
- the capabilities it requires of the performing agent;
- the relation this method has to other tasks; for example, putting a fire out using dynamite conflicts with the task of rescuing people;
- the side effects of the method; for example, using foam to put out a fire will make a lot of mess (environmental effects), while using dynamite generates a lot of noise.

Clearly, there are a lot of factors that must be taken into account in the task allocation process. An information system can assist in collecting and presenting all this information, as well as process the information to suggest an allocation of tasks to agents.

### A HIERARCHICAL TASK FRAMEWORK

One view on incident management – a static, structural view – is that we must perform a set of *tasks*: victims must be rescued, their wounds must be tended to, fires must be put out; the list goes on. In addition, the execution of one task may depend on the execution of another. For example, before ambulances can reach the disaster scene, we must first clear the roads of the worst debris. Another, more dynamic view on incident management is that emergency response situations are characterized by incoming events: buildings collapse, equipment breaks down, alarm calls come in, key emergency response personnel arrives at the scene, etc.

To model both static and dynamic aspects, we are developing a modeling framework that combines both concepts. We will omit a formal and comprehensive description of our framework in favor of a brief listing of its elements, followed by a number of examples that show how the framework can be used to model portions of disaster plans. The framework contains the following elements:

- The processes in the disaster plan are modeled as a set of tasks  $T$ .
- Between the tasks in  $T$ , there can exist *precedence constraints*, expressing that one task should be completed before another can start; in addition, we may specify that a task can be completed by performing a number of its *subtasks* – all of its subtasks if we want to define an AND relation between, and at least one of the subtasks when specifying an OR relation.
- *Events* can occur due to some uncontrolled change in the environment (*external events*), but also on the completion of a task. In fact, a task can have more than one *outcome*, and we can model each outcome as the event that occurs on that particular completion of the task.
- An event can *trigger* a task, by which we mean that after the event has occurred, then the task can be executed (it has been *enabled*); an event can also *disable* a task, after which the task can no longer be performed. If a task was already being executed at the time the disabling event occurs, then the task will be *suspended*, and it will remain suspended until it is triggered again by a different event.

The following two examples aim to give an idea how a formal modeling framework can be used in disaster plan modeling, and they illustrate the framework elements listed above.

#### Example: evacuate or not

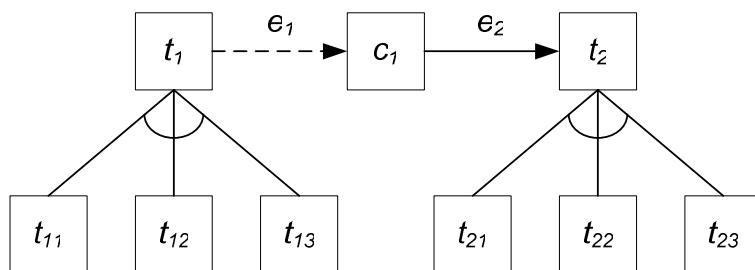


Figure 1

Imagine there has been an explosion in a chemical factory that is sited near a residential area. In that case, we might consider performing task  $t_2$ : evacuate citizens (see Figure 1). To perform this task, we must perform each of its subtasks (the AND relation is indicated by the arc connecting the lines underneath  $t_2$ ):

- $t_{21}$  : arrange transportation
- $t_{22}$  : prepare accommodation
- $t_{23}$  : order food supplies

Preceding the evacuation operations, however, is the decision to evacuate, here modeled as task  $c_1$ . Only if the outcome of  $c_1$  is to go ahead with the evacuations – event  $e_2$  – do we start with evacuating. One event that might trigger the decision to evacuate is the event  $e_1$  : hazardous gases released, which is one possible outcome of the task  $t_1$  : measure release of hazardous materials, which consists of three subtasks, all of which must be performed:

- $t_{11}$  : determine nature of released materials
- $t_{12}$  : determine area of infection
- $t_{13}$  : estimate the development of the spread of the materials

#### Example: plan repair

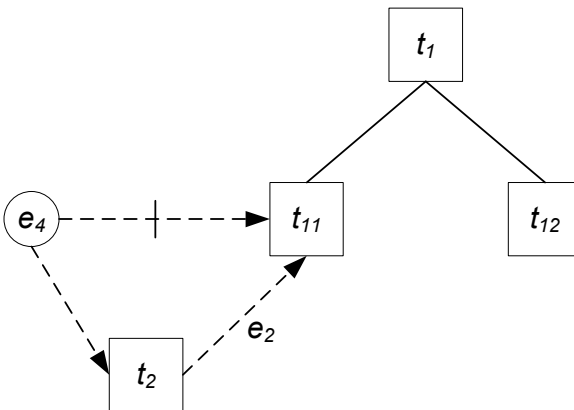


Figure 2

In most disasters, there is need to transport wounded people from the disaster area to a hospital. Thus, we have the task  $t_1$ : transport wounded (see Figure 2). This task can be accomplished by performing either or both of the two subtasks (there is no arc connecting the lines underneath  $t_1$ ):

- $t_{11}$  : ambulance transport
- $t_{12}$  : medical helicopter transport

Unfortunately, the event  $e_4$  : roads blocked will make task  $t_{11}$  impossible. To achieve the goal of transporting victims in spite of event  $e_4$ , the disaster plan in Figure 2 suggests two possibilities. First, we might use medical helicopters instead; second, we could perform the task  $t_2$  : clear roads, after which the ambulances would be able to reach the disaster area again.

#### CONCLUDING REMARKS

In view of the ineffectiveness of current disaster plans, we have set out to develop a modeling framework that we believe captures the most important aspects of emergency response: the tasks that must be performed, the dependencies between them, and the events that either trigger their execution, or disrupt their execution. Note that such a framework not only enables the construction of a decision support system, it also facilitates its construction, since for example the range of responses to an arising contingency may already be encoded in the disaster plan.

Future work will consist of further developing the framework both in terms of stabilizing the current concepts and extending the framework to handle e.g. resource usage.

#### ACKNOWLEDGMENTS

This research is supported by the Dutch Ministry of Economic affairs under the SENTER TSIT program Cybernetic Incident Management (TSIT2021).

## REFERENCES

1. Dynes, Russell, and E.L. Quarantelli. Organizational communications and decision making in crises. *Disaster Research Center Report 17*, 1977.
2. Directie Brandweer en Rampenbestrijding Den Haag. Eindrapport vliegtuigongeval Eindhoven (15 juli 1996), 1996.
3. Kay C. Goss. Guide for All-Hazard Emergency Operations Planning, September 1996.
4. Tomas Klos and Han La Poutré. Using reputation-based trust for assessing agent reliability. In *Seventh International Workshop on Trust in Agent Societies*, July 2004.
5. David Mendonça and Frank Fiedrich. Design for improvisation in computer-based emergency response systems. In *Proceedings of the First International Workshop on Information Systems for Crisis Response and Management*, May 2004.
6. Kathleen J. Tierney. Research Overview: Emergency Response, 1997.
7. M.J. van Duin, E.R. Muller, and U. Rosenthal. Cindu 8 juli 1992: een procesmatige analyse van een crisis, 1992.