PART VI

Case Studies and Applications

The variety of applications for automated planning is becoming quite significant. For example, the web repository of PLANET, the European network on automated planning, refers to about 20 applications in aeronautics and space, agricultural, industrial, commercial, and military domains,[1] and this list is doubtless incomplete. The Chapters in this part do not give a comprehensive survey of these planning applications. Rather, they provide case studies to illustrate some of the issues and approaches.

The application domains for our case studies include control of spacecraft (Chapter 19), robotics (Chapter 20), computer-aided design and manufacturing (Chapter 21), emergency rescue operations (Chapter 22), and the game of bridge (Chapter 23). In some cases (Chapters 19, 20, and 23), the systems have been successfully deployed, and some of them have made newspaper headlines. Other cases (Chapters 21 and 22) remain at the level of research prototypes. But in all cases, the focus is on how to address the issues that are important in the application domain at hand.

The planning paradigms that appear in our case studies include forward search, HTN planning, temporal and resource planning, MDP planning, plan merging, and case-based planning. HTN planning seems to be especially useful—various forms of it appear in all but one of our case studies—because it provides a flexible and efficient way to encode domain-specific knowledge about how to solve a problem. In several of the case studies, HTN planning is combined with other planning techniques: e.g., MDPs and HTNs in the robotics chapter, and HTNs and case-based planning in the chapter on emergency evacuation.

1. See http://www.planet-noe.org/ and http://scom.hud.ac.uk/planet/repository/.

Although the planning systems described in these chapters are based on general planning paradigms, most of the systems are *domain-specific*: they rely on representations and algorithms designed specifically for their respective application domains, and most of the systems were built by teams that included both planning researchers and domain experts. From a software engineering perspective, domain-independent planning is far behind areas such as database management, where it is comparatively easy to use off-the-shelf database server products.

In order to work effectively on practical applications, planning researchers must adjust how they think about planning problems. In research on automated planning, the typical approach has been to create an abstract class of problems that omit various details of a problem and then to look for general ways to solve problems in that class. In contrast, practical applications generally require a planning researcher to develop an understanding of the specific characteristics and requirements of the application domain at hand.

Some of the issues include the following.

- *How to acquire, validate, and verify information about the planning domain.* Some of the necessary information may be informal, anecdotal, proprietary, or otherwise hard to obtain and formalize.
- *What constitutes a planning problem, what constitutes a solution, and which simplifying assumptions might be appropriate.* What a planning researcher may consider to be an "unimportant detail" may, from the viewpoint of a domain expert, be a critical part of the problem. Assumptions that look appropriate to a planning researcher may not look credible to a domain expert, and vice versa.
- *How to represent the information and what planning paradigm(s) to use.* Often the approach will need to incorporate some domain-specific heuristics and algorithms, as well as *ad hoc* tweaks to handle special cases.
- *How best to interact with human users and/or how to integrate the planning system into the architecture of a larger system.* In some cases a fully automated planning system or subsystem may be the most appropriate approach, and in others it may be better to build a system that provides decision support while keeping the human user in charge of the planning process.

Developing an understanding of an application domain can be time-consuming, but the effort can be very worthwhile—not just to enable the development of useful planning tools but also to develop planning techniques that may be useful in other settings. In many of our case studies, research on specific applications has led to significant advances in planning theory, and advances in planning theory have led to better planning applications.