Chapter IV

Plan Optimization by Plan Rewriting

José Luis Ambite, University of Southern California, USA
Craig A. Knoblock, University of Southern California, USA
Steven Minton, Fetch Technologies, USA

ABSTRACT

Planning by Rewriting (PbR) is a paradigm for efficient high-quality planning that exploits declarative plan rewriting rules and efficient local search techniques to transform an easy-to-generate, but possibly suboptimal, initial plan into a high-quality plan. In addition to addressing planning efficiency and plan quality, PbR offers a new anytime planning algorithm. The plan rewriting rules can be either specified by a domain expert or automatically learned. We describe a learning approach based on comparing initial and optimal plans that produce rules competitive with manually specified ones. PbR is fully implemented and has been applied to several existing domains. The experimental results show that the PbR approach provides significant savings in planning effort while generating high-quality plans.

INTRODUCTION

Planning is the process of generating a network of actions, a plan that achieves a desired goal from an initial state of the world. Many problems of practical importance can be cast as planning problems. Instead of crafting an individual planner to solve each specific problem, a long line of research has focused on constructing domain-independent planning algorithms. Domain-independent planning accepts as input not only
descriptions of the initial state and the goal for each particular problem instance, but also a declarative domain specification, that is, the set of actions that transform a state into a new state. Domain-independent planning makes the development of planning algorithms more efficient, allows for software and domain reuse, and facilitates the principled extension of the capabilities of the planner. Unfortunately, domain-independent planning is computationally hard (Bylander, 1994; Erol, Nau & Subrahmanian, 1995). Given the complexity limitations, most of the previous work on domain-independent planning has focused on finding any solution plan without careful consideration of plan quality. Usually very simple cost functions, such as the length of the plan, have been used. However, for many practical problems plan quality is crucial. In this chapter we present Planning by Rewriting (PbR), a planning paradigm that addresses both planning efficiency and plan quality while maintaining the benefits of domain independence. The framework is fully implemented and we present empirical results in several planning domains.

Two observations guided the present work. The first one is that there are two sources of complexity in planning:

- **Satisfiability**: the difficulty of finding any solution to the planning problem (regardless of the quality of the solution).
- **Optimization**: the difficulty of finding the optimal solution under a given cost metric.

For a given domain, each of these facets may contribute differently to the complexity of planning. In particular, there are many domains in which the satisfiability problem is relatively easy and their complexity is dominated by the optimization problem. For example, there may be many plans that would solve the problem, so that finding one is efficient in practice, but the cost of each solution varies greatly, thus finding the optimal one is computationally hard. We will refer to these domains as optimization domains. Some optimization domains of great practical interest are query optimization and manufacturing process planning.

The second observation is that planning problems have a great deal of structure. Plans are a type of graph with strong semantics determined by both the general properties of planning and each particular domain specification. This structure should and can be exploited to improve the efficiency of the planning process.

Prompted by the previous observations, we developed a novel approach for efficient planning in optimization domains: Planning by Rewriting (PbR). The framework works in two phases:

1. Generate an initial solution plan. Recall that in optimization domains this is efficient. However, the quality of this initial plan may be far from optimal.
2. Iteratively rewrite the current solution plan improving its quality using a set of declarative plan-rewriting rules, until either an acceptable solution is found or a resource limit is reached.

As motivation, consider the optimization domains of distributed query processing and manufacturing process planning. Distributed query processing (Yu & Chang, 1984)
Technology Fears: A Study of e-Commerce Loyalty Perception by Jordanian Customers
www.igi-global.com/chapter/technology-fears-study-commerce-loyalty/61574?camid=4v1a