Part II
Neoclassical Planning

Neoclassical planning, like classical planning, is also concerned with restricted state-transition systems. We will be using here the classical representations developed in Chapter 2. However, at a time when classical planning appeared to be stalled for expressiveness as well as for complexity reasons, the techniques discussed in this part, that we qualify as *neoclassical*,[1] led to a revival of the research on classical planning problems. The development of neoclassical techniques brought new search spaces and search algorithms for planning that allowed directly (or indirectly through improvement of classical techniques) a significant increase on the size of classical problems that could be solved.

The main differences between classical and neoclassical techniques are the following.

- In classical planning, every node of the search space is a partial plan, i.e., a sequence of actions in the state space, or a partially ordered set of actions in the plan space; any solution reachable from that node contains *all* the actions of this partial plan.
- In neoclassical planning, every node of the search space can be viewed as a set of several partial plans. This set is either explicit or implicit in the data structures that make a search node, but it is evident in the fact that in the neoclassical approaches, not every action in a node appears in a solution plan reachable from that node.[2]

1. *Neoclassic: of or relating to a revival or adaptation of the classical style, especially in literature, art, or music* [Webster New Collegiate Dictionary]. Neoclassical has referred to slightly different meanings in planning literature depending on one's views of where and when the revival took place.
2. Because of this property, neoclassical planning approaches have sometimes been called *disjunctive-refinement approaches* [306].

We will come back to this common feature in Part III, once the reader has become familiar with the neoclassical techniques. Three such techniques will be studied here.

- *Planning-graph* techniques are based on a powerful reachability structure for a planning problem called a *planning graph*, which is used to efficiently organize and constrain the search space.
- *Propositional satisfiability* techniques encode a planning problem into a SAT problem and then rely on efficient SAT procedures for finding a solution, among which are complete methods based on the Davis-Putnam procedure and pseudorandom local search methods.
- *Constraint satisfaction* techniques similarly enable encoding a planning problem into a constraint satisfaction problem and also bring to the field a variety of efficient methods, in particular filtering and constraint propagation for disjunctive refinements in the plan space or within the planning-graph approaches.

These three techniques are described in Chapters 6, 7 and 8, respectively.