Chapter 5

A Branch-and-Bound Algorithm

In this chapter, we describe a branch-and-bound algorithm for the DSP in which to embed the constraint propagation techniques that we have derived in the last chapter. A general introduction to branch-and-bound has been given in section 2.3. As mentioned there, one of the most important components of a branch-and-bound solution method is a branching scheme which specifies how to decompose the search space into smaller subspaces that are hopefully easier to explore.

Defining efficient branching schemes is not a simple task and depends on the model formulation of the problem given. One of the advantages of the disjunctive scheduling model is the fact that it naturally implies a branching scheme which is based on the resolution of the disjunctive constraints of a DSP instance. The fundamental observation is almost trivial: if two operations $i$ and $j$ are in disjunction then there only exist two possible scheduling sequences, namely, $i \rightarrow j$ or $j \rightarrow i$. This induces a simple branching scheme which continually decomposes a search space into two subspaces, enforcing the processing sequence $i \rightarrow j$ in the first and $j \rightarrow i$ in the second subspace for a given disjunction $i \leftrightarrow j$.

Many solution methods for subclasses of the DSP, among others the algorithms of Carlier and Pinson [CP89], Applegate and Cook [AC91] and Caseau and Laburthe [CL95] for solving the JSP, apply this branching scheme which in spite of its simple description has proven to be astonishingly powerful in the past. It still remains one of the favourite choices in recent scheduling research, see e.g. Błażewicz et al. [BPS98]. The disjunctive branching scheme, however, has the drawback that at each level of the branching process only one single edge is oriented which leads to a tree of considerable depth.
The block branching scheme which has been developed by Grabowski et al. [GNZ86] for the single-machine scheduling problem and by Barker and McMahon [BM85], Brucker et al. [BJS94, BHJW97] and Krämer [Krä97] for the JSP and various other shop scheduling problems overcomes this disadvantage by orienting sets of edges in each node of the branching tree. Through an intelligent choice of the edges to be fixed, a large number of non-improving solutions can be excluded which further reduces the size of the branching tree.

A generalization of the block branching scheme for the DSP is discussed in section 5.1. Sections 5.2 and 5.3 deal with the "bounding" part of the branch-and-bound algorithm. Computing tight upper and lower bounds considerably influences the efficiency of a branch-and-bound algorithm, since it allows to prune uninteresting parts of the search tree that cannot improve upon a given makespan. We will, on the one hand, compute clique oriented and constraint propagation based lower bounds and, on the other hand, apply a simple priority rule based heuristic for computing upper bounds. Section 5.4 combines the branch-and-bound solution method with the constraint propagation algorithms derived in the last chapter and examines how to improve the efficiency through a careful choice of the search order. Section 5.5, finally, tests the branch-and-bound algorithm and constraint propagation techniques on both standard and newly generated problem instances.

5.1 The Block Branching Scheme

In this section, we will define a class of block branching schemes for the DSP. Subsection 5.1.1 introduces the concept of block decompositions. Each block decomposition scheme implies a particular block branching scheme which is described in subsection 5.1.2 in its general form. Various block decomposition schemes are then defined in subsection 5.1.3.

5.1.1 Block Decompositions

For the rest of this section, let $P = (O, C, D, P)$ be an instance of the DSP and $G$ the corresponding disjunctive graph. Further, given a partial or complete selection $S$, let $G_S$ denote the associated directed graph. Extracting the common properties of the definition of blocks of operations and block decompositions [GNZ86, BJS94, Krä97, BHJW97] leads to the following generalized concepts.

**Definition 5.1 (Blocks of Operations)**

Let $B = u_1 \rightarrow \ldots \rightarrow u_l$ be a path in $G_S$ of length $l \geq 2$. 