Nonserial Dynamic Programming and Tree Decomposition in Discrete Optimization

Oleg Shcherbina

Institut fuer Mathematik, University of Vienna, Austria
oleg.shcherbina@univie.ac.at

1 Introduction

Solving discrete optimization problems (DOP) can be a rather hard task. Many real DOPs contain a huge number of variables and/or constraints that make the models intractable for currently available solvers. There are few approaches for solving DOPs: tree search approaches (e.g., branch and bound), relaxation and decomposition methods. Large DOPs can be solved due to their special structure. Among decomposition approaches we can mention poorly known local decomposition algorithms using the special block matrix structure of constraints and half-forgotten nonserial dynamic programming algorithms which can exploit sparsity in the dependency graph of a DOP.
One of the promising approaches to cope with NP-hardness in solving DOPs is the construction of decomposition methods [7]. Decomposition techniques usually determine subproblems, whose solutions can be combined to create a solution of the initial DOP problem. Usually, DOPs from applications have a special structure, and the matrices of constraints for large-scale problems have a lot of zero elements (sparse matrices).
This paper reviews main results for local decomposition algorithms in discrete programming and establishes some links between them, tree decompositions and nonserial dynamic programming.

2 Nonserial Dynamic Programming

One of the promising ways to exploit sparsity in the dependency graph of an optimization problem is nonserial dynamic programming (NSDP) (Bertele, Brioschi [2], Hooker [5]), which allows to compute a solution in stages such that each of them uses results from previous stages.
Classical dynamic programming has been proposed by R.Bellman as discrete analogue of the optimal control theory and is widely used in applications of operations research.
NSDP [2] appeared in 1960th but is poorly known to the optimization community.

This approach is used in Artificial Intelligence under the names "Variable Elimination" or "Bucket Elimination" [4]. Nonserial dynamic programming being a natural and general decomposition approach, considers a set of constraints and an objective function as recursively computable function. This allows to compute a solution in stages such that each of them uses results from previous stages.
An efficiency of this algorithms crucially depends on the interaction graph structure of a DOP. If the interaction graph is rather sparse or, in other words, has a relatively small induced width, then the complexity of the algorithm is reasonable.
We consider a DOP in the following form

$$F(x_1, x_2, \dots, x_n) = \sum_{k \in K} f_k(Y^k) \rightarrow \max \tag{1}$$

subject to the constraints

$$g_i(X^i)\ R_i\ 0,\ \ i \in M = \{1, 2, \dots, m\}, \tag{2}$$

$$x_j \in D_j,\ \ j \in \{1, \dots, n\}, \tag{3}$$

where

$$Y^k \subseteq \{x_1, x_2, \dots, x_n\}, X^i \subseteq \{x_1, x_2, \dots, x_n\}, R_i \in \{\leq, =, \geq\}. \tag{4}$$

Introduce an interaction graph [2] (dependency graph by HOOKER [5]) representing in natural way a structure of the DOP. The interaction graph will be denoted $G = (V,E)$ where a vertex $j \in V$ corresponds to a variable x_j of the DOP. Vertices 1 and 2 are adjacent (denoted $1 \sim 2$) if the variables x_1 and x_2 (corresponded to these vertices) appear together in the same member f_k of the objective function or in the same constraint i (in other words, if variables x_1 and x_2 are in the set Y^k or in the set X^i). Further, we shall use the notion of vertices that correspond one-to-one to variables. Let S_1, S_2, and S be vertex sets of the graph, and j be a vertex of the graph. Introduce the following notions:

1. Adjacency of sets: $S_1 \sim S_2$ (read as "S_1 is adjacent to S_2"), if there exists a vertex $j_1 \in S_1$ and a vertex $j_2 \in S_2$ such that $j_1 \sim j_2$.
2. $j \sim S$ means $\{j\} \sim S$.
3. Neighborhood of a set $S \subseteq V$, $Nb(S) = \bigcup_{v \in S} Nb(v) - S$.
4. Closed neighbourhood of a set $S \subseteq V$, $Nb[S] = Nb(S) \cup S$.

Consider a DOP and suppose without loss of generality that variables are eliminated in the order $x_1, \dots, x_n$. Using the NSDP scheme eliminate a first variable x_1. This x_1 is in a set K_1 of objective function members:

$$K_1 = \{k \mid x_1 \in Y^k\}$$

and in a set of constraints with the indices U_1:

$$U_1 = \{i \mid x_1 \in X^i\}.$$

Simultaneously with x_1 in objective function members Y^k, $k \in K_1$ and in constraints U_1 are variables from $Nb(x_1)$.
To the variable x_1 corresponds the following subproblem P_1 of DOP: