Applied Mathematics and Computation 228 (2014) 598–605

Contents lists available at ScienceDirect

Applied Mathematics and Computation

journal homepage: www.elsevier.com/locate/amc

Multiple criteria dynamic programming and multiple knapsack problem

Sebastian Sitarz

Institute of Mathematics, University of Silesia, ul. Bankowa 14, 40-007 Katowice, Poland

ARTICLE INFO

Keywords:
Multiple criteria programming
Dynamic programming
Integer programming
Knapsack problem

ABSTRACT

The aim of the paper is to show relations between two different types of optimization problems: multiple criteria dynamic programming (MCDP) and integer linear programming (ILP) in a form of multiple knapsack (MK) problem. Moreover, the paper presents how to use MCDP methods in order to solve (MK) problems.

1. Introduction

The paper presents relations between two types of problems: multiple criteria dynamic programming and multiple knapsack problem.

The theory of the dynamic programming, started by Bellman, has developed in many directions over the last fifty years. One of the directions is the multiple criteria optimization. We are able to follow the development in the works by Brown and Strauch [1] – a model with criteria function values in lattices and Henig [4] – a model of infinite processes with criteria function values in a partially ordered set. Currently, multiple criteria dynamic programming models are used for solving specific types of problems, for instance, Warburton [20] and Getachew [3] consider multiple criteria shortest path problems, or Klamroth and Wiecek [6] apply the multiple criteria dynamic programming to a problem of loading with a variety of goals. A similar idea serves this paper where multiple criteria dynamic programming is used for integer linear programming.

Integer linear programming appeared in the late 1950s as a branch of the linear programming. The theory is widely described in the works of Garfinkel and Nemhauser [2]. Whereas, the work of Martello and Toth [7] is devoted to algorithmic aspects and their computer implementation, whereas Polatschek [8] considered the simulation theory used for the discrete programming. The latest results in the field of integer programming may be studied from the works of "The 15th Integer Programming and Combinatorial Optimization 2005" conference edited by Gunluk and Woeginger [5].

It is worth mentioning that Schrijver [9] describes the connections between dynamic programming (DP) and ILP with many references therein. All of these approaches use one criterion DP for ILP problems. However, this paper considers multiple criteria dynamic programming (MCDP) and uses the ordered structures to obtain the theorems.

The paper consists of six sections. Section 1 introduces the paper. Section 2 presents a dynamic programming model with cost function values in the ordered structure, which means a semi-group with a partial order. Such a general approach allows us to use the model for many types of problems, for instance a for multiple knapsack problem which is characterized in Section 3. Relations between two theories are shown in Section 4 where theorem for using dynamic optimization in integer problems is presented. The numerical application is illustrated in Section 5. The computational experiments are presented in Section 6. The last section summarizes the paper.

E-mail address: ssitarz@math.us.edu.pl

0096-3003/$ - see front matter
http://dx.doi.org/10.1016/j.amc.2013.12.022

2. Dynamic programming in ordered structures

We consider a multi-stage discrete dynamic process with Markovian property. A more detailed description of such processes may be found in the works by Trzaskalik and Sitarz [19] or Sitarz [15].

2.1. Description of dynamic process

The notation of the considered dynamic process and the graph interpretation are presented in Table 1.

2.2. Criteria function description

Now, we describe a criteria function in the dynamic process. Each decision in each stage has its rate which is an element of a certain ordered set $(W, \preccurlyeq)$, ($\preccurlyeq$ means a binary relation which is reflexive, antisymmetric and transitive). Values of the stage criteria function will be connected in each stage by an operator $\circ$: $W \times W \to W$ that satisfies the conditions:

$$\wedge_{a,b,c \in W}\ a \circ (b \circ c) = (a \circ b) \circ c, \quad \text{(associativity condition)},$$
$$\wedge_{a,b,c \in W}\ a \preccurlyeq b \Rightarrow (a \circ c \preccurlyeq b \circ c \wedge c \circ a \preccurlyeq c \circ b), \quad \text{(monotonicity condition)}.$$

The operator $\circ$ will be the same for each stage. Introduction of such a way of connecting the stage criteria function values includes a condition of multi-stage criteria function separability.

To be specific, we will consider a triple $(W, \preccurlyeq, \circ)$ that satisfies the above conditions and that will be called the ordered structure. In a special case, assuming $(W, \preccurlyeq) = (\mathbb{R}^m, \leqslant)$ we get a model of the multiple criteria optimization.

The relation $\preccurlyeq$ generates for each finite set $A \subset W$ maximal elements of the set denoted as max A, and defined as follows:

$$\max A = \{a \in A : \wedge_{x \in A}\ a \preccurlyeq x \Rightarrow a = x\}.$$

Let P stand for the process and $(W, \preccurlyeq, \circ)$ for the ordered structure. The stage criteria function in the stage t is the function:

$$f_t : R_t \to W$$

In other words, using the graph interpretation, the value of function assigns a certain element w to the given edge. Next, we assume the following simplification of the notation:

$$f_t(s_t, s_{t+1}) = f_t((s_t, s_{t+1}))$$

Below, we present definitions of the multi-stage criteria functions.

- Partial backward criteria function $\overleftarrow{F}_t : \overleftarrow{R}_t(S_t) \to W$ is defined as follows:

$$\overleftarrow{F}_t = f_t \circ f_{t+1} \circ \ldots \circ f_T$$

- Partial forward criteria function $\vec{F}_t : \vec{R}_t(S_t) \to W$ is defined as follows:

$$\vec{F}_t = f_t \circ f_2 \circ \ldots \circ f_{t-1},$$

- Multi-stage criteria function $F : R \to W$ is defined as follows:

$$F = f_1 \circ f_2 \circ \ldots \circ f_T.$$

Table 1
List of the introduced notations.

Symbol	Meaning	Elements description	Interpretation in the graph theory
T	The number of stages	$t\{1, \ldots, T\}$	The number of layers
S_t	The set of feasible states at the beginning of the stage t	s_t	The nodes at the point t
$D_t(s_t)$	The set of feasible decisions at the beginning of the stage t, in the state s_t	$d_t(s_t)$	The nodes obtained from edges starting at the state s_t
R	Realizations of the process-sequence of the succeeding states	$r = (s_1, \ldots, s_{T+1})$	Paths starting from the initial node and ending in the final node
R_t	The set of stage realizations at the beginning of the stage t	$r_t = (s_t, s_{t+1})$	The edges of the graph between the nodes at the point t and $t+1$
$\overleftarrow{R}_t(s_t)$	The set of partial realizations of the process beginning at the point t and the state s_t	$(s_t, \ldots, s_{T+1})$	Paths joining the given node s_t with the final nodes
$\vec{R}_t(s_t)$	The set of partial realizations of the process ending at the point t in the state s_t	$(s_1, \ldots, s_t)$	Paths joining the initial node with the given node s_t
$\overleftarrow{R}_t(S_t)$	The set of partial realizations beginning at the point t	$(s_t, \ldots, s_{T+1}) : s_t \in S_t$	Paths joining nodes at the point t with the final nodes
$\vec{R}_t(S_t)$	The set of partial realizations ending at the point t	$(s_1, \ldots, s_t) : s_t \in S_t$	Paths joining the initial nodes with the nodes at the point t
P	The process in which T, S_t, $D_t(s_t)$ were defined		The whole graph