A Modified Branch and Cut Procedure for Resource Portfolio Problem under Relaxed Resource Dedication Policy

Umut Beşikci[a], Ümit Bilge[a], Gündüz Ulusoy[b,*]

[a]*Boğaziçi University, İstanbul*
[b]*Sabancı University, İstanbul*

First draft, November, 2012.

Abstract

Multi-project scheduling problems are characterized by the way resources are managed in the problem environment. The general approach in multi-project scheduling literature is to consider resource capacities as a common pool that can be shared among all projects without any restrictions or costs. The way the resources are used in a multi-project environment is called resource management policy and the aforementioned assumption is called Resource Sharing Policy in this study. The resource sharing policy is not a generalization for multi-project scheduling environments and different resource management policies maybe defined to identify characteristics of different problem environments. In this study, we present a resource management policy which prevents sharing of resources among projects but allows resource transfers when a project starts after the completion of another one. This policy is called the Relaxed Resource Dedication (RRD) Policy in this study. The general resource capacities might or might not be decision variables. We will treat here the case where the general available amounts of resources are decision variables to be determined subject to a limited budget. We call this problem as the Resource Portfolio Problem (RPP). In this study, RPP is investigated under RRD policy and a modified Branch and Cut (B&C) procedure based on *CPLEX* is proposed. The B&C procedure of *CPLEX*

---
[*]Sabanci University, Orhanli, Tuzla, 34956 Istanbul, Turkey Tel.: ++(90)216-483-9503.
E-mail address: gunduz@sabanciuniv.edu (G. Ulusoy)

is modified with different branching strategies, heuristic solution approaches and valid inequalities. The computational studies presented demonstrate the effectiveness of the proposed solution approaches.

1. Introduction

Resource management policy is an important issue for the multi-project scheduling problems, since it characterizes the problem environment by defining the coupling among different projects. The common resource policy applied in the multi-project scheduling literature is Resource Sharing (RS) policy where resources can be shared among projects without any restrictions. Even though RS policy is employed in the multi-project scheduling literature frequently, it does not cover all possible cases. We have proposed Resource Dedication (RD) policy for the cases where resources cannot be shared among projects because of various reasons. The most common ones can be listed as follows:

- The project characteristics may not allow RS. This can be seen in various Research and Development (R&D) projects where the development process is highly technology intensive.

- The resource characteristics may force RD. For example, in software development projects, it is not desired to allocate developers to different projects because of the learning curve concept.

- Another example can be given as the heavy machinery equipment for certain cases where it becomes too costly to share them among projects because of installation.

- Another reason can be geographical limitations, where projects are distributed across the world such that it becomes unpractical to share the resources.

RD policy is extensively investigated in Besikci et al. (2012a,b). In this study, RD policy is extended further to Relaxed Resource Dedication (RRD) policy. In RRD policy, a resource can be transferred among projects, when

2

the project using that resource is completed and another project is to start which requires that same resource.

The general resource capacity will have different meanings according to the valid resource management policy in the problem environment. In RS policy, the general resource capacity corresponds to the size of the shared pool for each renewable resource. Under RD policy, on the other hand, it corresponds to the total amount of each renewable resource to be dedicated among the projects. In other words, resource management policy defines the way of coupling projects via general resource capacities. The general resource capacities might or might not be decision variables. We will treat here the case where the general resource capacities are to be determined subject to a limited budget. We call this problem the Resource Portfolio Problem (RPP), which extends the multi-project scheduling problem under RRD policy to a higher decision level.

The problem environment in this paper has the following additional characteristics; the projects are ready to start at time zero but their start time is also a decision variable to enable resource transfers. Each project has an assigned due date which are soft in the sense that any completion time exceeding the due date is reflected to the objective function as tardiness cost. The activities are non-preemptive and have zero finish-to-start time lag. The resource set contains both renewable and nonrenewable resource types. For each activity, there is one or more modes of execution. The objective is selected as the total weighted tardiness cost over all projects which is a common objective function in scheduling problems, but any regular objective function can be integrated into the model.

The remainder of the paper starts with a brief literature review of studies that focus on resource related decisions in project planning. In Section 3, RPP under RRD policy is defined and its mathematical formulation is presented. In Section 4 the modified B&C solution procedure employing *CPLEX* is explained in detail. The computational study and results are given in Section 5. Finally conclusions and further research suggestions are given in Section 6.

2. Literature Review

Project scheduling problem has been studied in various forms. Resource constrained project scheduling problem (RCPSP) is the root for different

3

extensions of project scheduling problem. Extensive information regarding RCPSP can be found in Ozdamar and Ulusoy (1995), Herroelen et al. (1998), Brucker et al. (1999) and Kolisch and Padman (2001). Multi-mode resource constrained project scheduling problem (MRCPSP) is an extension for RCPSP where activities can be executed with a set of different alternatives which are called modes. Interested readers can refer to Hartmann and Briskorn (2010) for a survey of RCPSP and its extensions. Additionally, the study of Zhu et al. (2007) is worth mentioning, where the authors employ *CPLEX* to solve MRCPSP with different modifications. First of all, authors use different approaches to reduce the number of decision variables in the model. Additionally, branching strategy of *CPLEX* is modified by giving corresponding time period of the decision variable as a weight. Finally a local branching strategy is used to search the close neighborhood of an incumbent solution. Note that the authors also used a genetic algorithm (GA) to provide an initial solution to the B&C procedure of *CPLEX*. In this section, apart from the studies mentioned above, the studies addressing the determination of resource requirements of projects and other resource related concerns such as rough-cut capacity planning or resource availability cost problem are presented in detail.

Resource related decisions are an important part of the multi-project scheduling problems since the coupling of individual projects in the problem environment is generally achieved with the resource related constraints.

Mohring (1984) investigates the resource requirement problem for project scheduling with a graph theoretical approach. The author defines two important problem classes for project scheduling. The first problem, *problem A*, is defined as a problem of scarce resources where the objective is finding the shortest project duration with a given amount of resources. The second problem, *problem B*, is defined as a problem of scarce time where the objective is finding the least cost resource requirements within a given project duration. The author uses graph theory to represent and solve these project scheduling problems. Apart from the solution methodologies for these problems, the important part of the study is the realization of the interaction between the two different problems. Since solution procedures for scarce resource problem have a better algorithmic behavior, one can find better solution procedures for scarce time problem, if a way of transformation can be found among the problems. The author notes that the objective of scarce resource problem is presented as a constraint in scarce time problem and vice versa. Using this rationale, a special "dual" relation between these two problems is defined

and used in the solution procedures proposed. Note that, the problem under study in this study is of the form of scarce resource problem.

Another study in the form of scarce time problem is presented in Demeulemeester (1995) and is defined as the resource availability cost problem (RACP). The general idea is determining the least cost resource requirements for a single project scheduling problem. The proposed formulation for RACP is similar to the formulations for resource constrained project scheduling problem (RCPSP) with the basic differences being in the objective function and in the constraint for project duration. The solution methodology proposed for RACP is based on decision problems defined with resource limits. These definitions serve as cuts for the feasible region for the problem, and the algorithm employs efficient capacity points (resource availability value in the solution that is greater than the other resource availabilities) to obtain the optimal solution. The experiments have demonstrated that the solution procedure is very effective.

An approach dealing with the determination of the resource requirements of projects is the rough-cut capacity planning (RCCP) for a multi project environment. A recent example is provided in Gademann and Schutten (2004). It is assumed that a project can be broken into jobs that stand for activity groups, which are not known yet. For each of these estimations, a precedence relation structure is constructed. In addition to this, a multi-project environment is modeled as a single project problem with all the estimated activity groups. Two variants of the RCCP problem are defined in line with scarce resource and scarce time problems, namely resource driven and time driven. In the resource driven RCCP, only the regular capacity can be used and the objective is minimizing the total lateness. In the time driven RCCP, a project delivery time is to be met by determining the non-regular capacity (i.e., the additional capacity beyond the already available capacity) and the objective is minimizing the cost of this non-regular capacity. The authors propose a linear program for the time driven RCCP problem. The model allows for precedence infeasible solutions and preemptions and heuristic procedures are proposed to repair such solutions. To overcome the precedence infeasibility, the mathematical model is changed slightly to determine the time windows for activities to ensure feasibility. The solutions of the modified linear program are improved by local search heuristics based on dual relations between the time windows and the decision variables.

Yang and Sum (1997) presented one of the first studies questioning the generalizing RS policy by pointing out the assumption of sharing resources

without any costs. Kruger and Scholl (2008) introduce further resource management schemes. They classify different resource transfers for different multi-project environments. Authors define three different approaches for dealing with resource transfers; transfer neglecting, transfer reducing and transfer using approach. Transfer neglecting approach is conceptually RS policy where there is not any restriction on resource flow between projects whereas transfer reducing approach does not allow RS and resources are dedicated to projects. And finally in transfer using approach resources can be transferred between projects with explicit costs. Note that the individual projects in the problem environment inherit the characteristics of an RCPSP. The transfer using approach is further investigated in Kruger and Scholl (2009). Authors propose different heuristic rules for the parallel and serial scheduling schemes modified for the resource transfer concept.

RD policy is first introduced in Besikci et al. (2012a) leading to the so-called Resource Dedication Problem (RDP), where the objective is minimizing the total weighted tardiness over all projects. The authors present a mathematical formulation for RDP and two different solution methodologies. The first solution approach is a GA employing a new improvement move called the combinatorial auction for RD (CA for RD), which is based on preferences of projects for resources. Preference of a project for a resource can be defined as the relative value of a resource for a project or the marginal gain in the objective function when one unit of the corresponding resource is made available. Two different methods for calculating the projects' preferences are proposed as linear relaxation based and Lagrangian relaxation based preferences. The second solution approach is a Lagrangian relaxation based heuristic employing subgradient optimization. The proposed solution approaches are compared with exact or best solutions obtained employing *CPLEX* and are shown to be efficient.

The study reported in Besikci et al. (2012a) is extended to deal with RPP under RD policy by including general resource capacity decisions in Besikci et al. (2012b). The objective is again the minimization of the total weighted tardiness over all projects. A two-phase GA is proposed as a solution approach, where in the first phase, RD space is searched and in the second phase resource portfolio (RP) space is searched along with the RD space. For the RP space search, the authors propose a new improvement move called the combinatorial auction for RP (CA for RP). CA for RP employs general resource preferences to improve a given initial solution. Numerical results demonstrate the effectiveness of the proposed solution approach.

6

3. Resource Portfolio Problem under Relaxed Resource Dedication Policy

RPP in a multi-project environment covers the determination of general resource capacities for a given total resource budget, dedication of these resources to a set of projects in certain amounts such that the general resource capacities are obeyed, and determinination of the project sequence relations and resource transfers between feasible projects (according to the project sequence relations) in such a way that individual project schedules would result in an optimal solution for a predetermined regular objective function. The objective for the problem environment is taken here as the minimization of the total weighted tardiness of the projects. Mathematical programming formulation for the RPP under the RRD policy is given below.

Sets:

V	Set of projects, $v \in V$
J_v	Set of activities of project v, $j \in J_v$
vN	Last activity (indexed with N) of project v
P_v	Set of all precedence relationships of project v
M_{vj}	Set of modes for activity j of project v, $m \in M_{vj}$
K	Set of renewable resources, $k \in K$
I	Set of nonrenewable resources, $i \in I$
T_v	Set of time periods of project v, $t \in T_v$

Parameters:

E_{vj}	Earliest finish time of activity j of project v
L_{vj}	Latest finish time of activity j of project v
d_{vjm}	Duration of activity j, operating on mode m
r_{vjkm}	Renewable resource k usage of activity j of project v, operating on mode m
w_{vjim}	Nonrenewable resource i usage of activity j of project v, operating on mode m
dd_v	Assigned due date for project v
c_v	Relative weight of project v
cr_k	Unit cost of renewable resource k
cw_i	Unit cost of nonrenewable resource i
tb	Total resource budget
Ω	A big number calculated as the sum of time periods of all projects

Decision Variables:

$$X_{vjmt} = \begin{cases} 1 & \text{if activity } j \text{ of project } v, \text{ operating on mode } m \text{ is finished} \\ & \text{at period } t \\ 0 & \text{otherwise} \end{cases}$$

BR_{vk} = Amount of renewable resource k dedicated to project v

BW_{vi} = Amount of nonrenewable resource i dedicated to project v

TC_v = Weighted tardiness cost of project v

R_k = Total amount of required renewable resource k

W_i = Total amount of required nonrenewable resource i

F_v = Release time of project v

$SR_{vv'k}$ = Amount of renewable resource k transferred to project v' from project v

$$Y_{vv'} = \begin{cases} 1 & \text{if project } v' \text{ is released after project } v \text{ is finished} \\ 0 & \text{otherwise} \end{cases}$$

Mathematical Model RPP-RRD

$$min. \; z = \sum_{v \in V} TC_v \tag{1}$$

Subject to

$$\sum_{m \in M_{vj}} \sum_{t=E_{vj}}^{L_{jv}} X_{vjmt} = 1 \quad \forall \, j \in N_v \text{ and } \forall \, v \in V \tag{2}$$

$$\sum_{m \in M_{vj}} \sum_{t=E_{vb}}^{L_{vb}} (t - d_{vbm}) X_{vbmt} \geq \sum_{m \in M_{vj}} \sum_{t=E_{va}}^{L_{va}} t X_{vamt}$$
$$\forall \, (a,b) \in P \text{ and } \forall \, v \in V \tag{3}$$

$$F_{v'} - F_v - \sum_{t=E_{vN}}^{L_{vN}} \sum_{m \in M_{vN}} t X_{vNmt} \leq \Omega(Y_{vv'}) \quad \forall \, v, v' \in V \tag{4}$$

$$F_v + \sum_{t=E_{vN}}^{L_{vN}} \sum_{m \in M_{vN}} t X_{vNmt} - F_{v'} \leq \Omega(1 - Y_{vv'}) \quad \forall \, v, v' \in V \tag{5}$$

$$SR_{vv'k} \leq \Omega(Y_{vv'}) \quad \forall \, v, v' \in V \text{ and } \forall \, k \in K \tag{6}$$

$$\sum_{j \in N_v} \sum_{m \in M_{vj}} \sum_{q=min\{t,E_{vj}\}}^{max\{t+d_{vjm}-1,L_{vj}\}} r_{vjkm} X_{vjmq} \leq BR_{vk} + \sum_{v' \in V} SR_{v'vk}$$
$$\forall \, k \in K \; \forall \, t \in T_v \; \forall \, v \in V \tag{7}$$

$$\sum_{j \in N_v} \sum_{m \in M_{vj}} \sum_{t=E_{vj}}^{L_{vj}} w_{vjim} X_{vjmt} \leq BW_{vi} \quad \forall \, i \in I \text{ and } \forall \, v \in V \tag{8}$$

8

$$BR_{vk} + \sum_{v' \in V} SR_{v'vk} \geq \sum_{v' \in V} SR_{vv'k} \quad \forall \, k \in K \text{ and } \forall \, v \in V \tag{9}$$

$$\sum_{v \in V} BR_{vk} \leq R_k \quad \forall \, k \in K \tag{10}$$

$$\sum_{v \in V} BW_{vi} \leq W_i \quad \forall \, i \in I \tag{11}$$

$$\sum_{i \in I} cw_i W_i + \sum_{k \in K} cr_k R_k \leq tb \tag{12}$$

$$TC_v \geq C_v(F_v + \sum_{t=E_{vN}}^{L_{vN}} \sum_{m \in M_{vN}} tX_{vNmt} - dd_v) \quad \forall \, v \in V \tag{13}$$

$$X_{vjmt} \in \{0,1\} \quad \forall \, j \in J, \forall \, t \in T_v, \forall \, m \in M_{vj} \text{ and } \forall \, v \in V \tag{14}$$

$$BR_{vk}, BW_{vi}, R_k, W_i, TC_v \in Z^+ \quad \forall \, v \in V, \forall \, k \in K \text{ and } \forall \, i \in I \tag{15}$$

$$Y_{vv'} \in \{0,1\}, F_v \text{ and } SR_{vv'k} \in Z^+ \quad \forall \, v, v' \in V \text{ and } \forall \, k \in K \tag{16}$$

The objective function (1) is the minimization of the total weighted tardiness cost for all projects. Constraint sets (2) and (3) impose activity finish and precedence relations. Constraint sets (4) and (5) set the decision variable $Y_{vv'}$ to 1, if project v is finished before project v' is released, and to 0 otherwise. Thus, the $SR_{vv'k}$ values will only have positive values, if project v is finished before project v' is released (constraint set (6)). Constraint set (7) limits the renewable resources employed for each project with the dedicated renewable resources and the transferred renewable resources from the other projects. Constraint set (8) calculates the nonrenewable resource dedication values available for each project. In constraint set (9), the total resource that can be transferred by a project is limited with the total resource dedicated to this project and the total resource it gained from transfers. Constraint sets (10) and (11) calculate the total renewable and nonrenewable resource requirements, respectively. Constraint (12) limits the sum of the total renewable and nonrenewable resource costs with the general resource budget. And finally, constraint set (13) calculates the weighted tardiness value for each project. Constraint sets (14)-(16) define the feasible ranges for the decision variables. Note that in the above formulation every project has an independent time index. This feature is enabled by the structure of the following constraints. Firstly, the coupled resource constraints for projects (constraint sets (4) and (5)) do not include a time period index. Furthermore, to calculate the start and finish times of projects while including project sequence relations, constraint sets (10) and (11) are used. In these constraints, if there

is a sequence relation between two projects, then the successor project's start time decision variable ($F_{v'}$) is given a lower bound of finish time of the predecessor project (the finish time of the predecessor project is also corrected with the corresponding start time decision variable, F_v). Note that having independent time indices for each project decreases the number of activity finish decision variables (X_{vjmt}) since every project will have its individual upper bound for time periods.

4. A Modified B&C Approach for RPP under RRD Employing CPLEX

The given mathematical model for RPP under the RRD policy is a complex scheduling problem. It is even very difficult to reflect all the aspects of the problem to a heuristic approach. Thus to solve RPP-RRD, B&C procedure of *CPLEX* version 11.2, which will be referred to as *CPLEX* from now on, is modified with different branching strategies, feasible solution approaches and valid inequalities. Firstly, the general procedure of *CPLEX* will be briefly discussed and then modifications will be explained in detail.

4.1. General Branch and Cut Procedure of CPLEX

CPLEX employs the well known B&C procedure to solve integer problems. The general B&C procedure starts with preprocessing of the given problem and proceeds with the given steps below (Users Manual for CPLEX, 2009):

1. A node from the tree is selected.
2. The linear programming (LP) relaxation of the problem at the selected node is solved and a lower bound is generated.
3. Cutting planes are generated to cut off the current solution.
4. An heuristic procedure is invoked to generate an integer feasible solution close to the current relaxation solution and an upper bound is generated.
5. A branching variable is chosen
6. Two nodes are generated and included to the tree.

All these steps can be modified by the user by employing special purpose algorithms developed specially for the problem at hand. The control of the above procedures is enabled with callback functions in the *CPLEX*. At every

node, when a feasible LP relaxation solution is found for the problem, the associated callback functions for each of the above procedures can be used to control the details of the B&C procedure. Three basic steps will be modified to control the details of the B&C procedure in this study; branching, feasible solution generation and local cut generation. Below explanations of these modifications are given in detail.

4.2. Branching Strategy for the RPP under RRD

The selection of branching variables can greatly improve the execution time of the B&C procedure. Model RPP-RRD has two sets of binary and seven sets of integer decision variables which can be used to generate branches. In *CPLEX*, the branching can be modified by giving priorities to decision variables or directly selecting a decision variable for branching on a viable node. In this study, we select the branching variable explicitly or leave it to *CPLEX*.

The proposed branching strategy is based on the structure of the mathematical model RPP-RRD. Note that when $Y_{vv'}$ variables are known, the problem reduces to RPP with available resource transfer options according to the values of $Y_{vv'}$ decision variables. In other words, when $Y_{vv'}$ variables are set, the remaining decisions are general resource capacities, resource dedications, resource transfers between feasible projects (for projects finish before the start of another one) and finally scheduling of individual projects. This structure of the problem not only facilitates the search but also allows for different feasible solution generation approaches. To branch on $Y_{vv'}$ decision variables, the branch callback function of *CPLEX* is used and branches are generated from integer infeasible variables from the linear relaxation solution on the node explicitly and fed into B&C procedure of *CPLEX*.
The selection rule for $Y_{vv'}$ variable among the integer infeasible ones is given below:

Initialization: Select all the projects that have integer infeasible $Y_{vv'}$ decision variables in a node and add selected projects to set *InP*.

Step 1: Select the projects from *InP* that do not have any sequence relations (i.e., all $Y_{vv'}$ and $Y_{v'v}$ variables are integer infeasible for project v) and add selected projects to set *SP*. If set *SP* is empty, then go to *Step 2*; else go to *Step 4*.

Step-2: Select the projects from *InP* that are the predecessor projects in their sequence relations (i.e. all $Y_{v'v}$ variables are integer feasible and 0 for

project v) and add selected projects to set SP. If set SP is empty, then go to *Step 3*; else go to *Step 4*.

Step-3: Select the projects from InP that have integer infeasible $Y_{vv'}$ decision variables and add selected projects to set SP, go to *Step-4*.

Step-4: Select the project v from SP with maximum weight and select the project v' with minimum weight among the projects that have integer infeasible $Y_{vv'}$ values. Generate two branches with the selected $Y_{vv'}$ decision variable ($Y_{vv'} \leq 0$ and $Y_{vv'} \geq 1$).

This selection procedure gives first priority to projects without any sequence relations, and then to the projects that are predecessor in their sequence relations. For tie breaking project weights are used. Another selection procedure based on ratio of due date and weight of the projects which prioritizes projects with earlier due dates and higher weight values is also considered. However, in the preliminary test runs it has been observed that there is not a significant difference between two approaches.

When all $Y_{vv'}$ decision variables are integer feasible in a node, then BR_{vk} decision variables are selected as the next variables for branching. The selection rule for BR_{vk} variable among the integer infeasible ones is given below: The selection rule for BR_{vk} variable among the integer infeasible ones is given below:

Initialization: Select all the projects that have integer infeasible BR_{vk} decision variables in a node and add selected projects to set InP.

Step-1: Select the projects from InP that do not have any predecessors (i.e., all $Y_{v'v}$ variables are 0 for project v) and add selected projects to set SP. If set SP is empty then go to *Step 2* else go to *Step-4*.

Step:2 Select the projects from InP that have at least one successor (i.e., at least one $Y_{vv'}$ variable is 1 for project v) and add selected projects to set SP. If set SP is empty, then go to *Step 3* else go to *Step-4*.

Step-3: Select the projects from InP that have integer infeasible BR_{vk} decision variables and add selected projects to set SP, go to *Step-4*.

Step:4 Select the project v from SP with maximum weight and select corresponding renewable resource k with integer infeasible BR_{vk} decision variable and maximum resource cost. Generate two branches with the selected BR_{vk} decision variable.($BR_{vk} \leq \lfloor LinearRelaxationSolutionValue \rfloor$ and $BR_{vk} \geq \lceil LinearRelaxationSolutionValue \rceil$).

This selection procedure similarly favors predecessor projects in their sequence relations. Note that, when all BR_{vk} variables are integer feasible the branching procedure is left to *CPLEX*.

4.3. Feasible Solution Generation Procedure

At each viable node, *CPLEX* attempts to generate a feasible solution close to the solution of the LP relaxation. If good (in some cases any) feasible solutions can be generated, *CPLEX* can use these solutions to facilitate the execution of the B&C algorithm. The proposed branching strategies prioritize branching on $Y_{vv'}$ variables which results in integer feasible values for these variables in the early stages of B&C procedure. This structure can be used to generate feasible solutions, since when $Y_{vv'}$ variables are known, the remaining problem reduces to RPP with possible resource transfers according to the values of $Y_{vv'}$ decision variables. The solution approaches developed for RDP in Besikci et al. (2012a) and RPP in Besikci et al. (2012b) can be modified and used to generate feasible solutions at these stages of the B&C procedure. In Besikci et al. (2012a), an improvement heuristic based on preferences of projects for resources, CA for RD, is proposed which moves the current resource dedication state of the problem to a more "preferable" resource state within the given general resource capacities, such that the objective function would be improved (the objective function value will at least remain the same). Similarly, in Besikci et al. (2012b), an improvement heuristic based on general resource preferences, CA for RP, is proposed which moves the current general resource capacity state to a more "preferable" resource state within the given resource budget.

The proposed feasible solution generation procedure basically determines the general resource capacities, resource dedication values and resource transfers (according to the values of $Y_{vv'}$ variables) and generates an initial solution (not necessarily feasible). And then, this initial solution is improved with CA for RD and CA for RP improvement heuristics. The procedure for feasible solution generation is given below:

Initial Solution Generation

Step-1: Calculate no-delay renewable and nonrenewable resource requirements of all projects and determine total required budget for these values. Dedicate nonrenewable resource values proportional to the no-delay requirements of the projects and total required budget as follows:

$$BW'_{vi} = \frac{NoDelayRequirement_{vi} \times GeneralBudget}{TotalRequiredBudget} \qquad (17)$$

Calculate remaining resource budget after nonrenewable resource dedication.

Step-2: Select all projects that does not have any predecessor projects in their sequence relations, such that for project v, for all v', $Y_{v'v}$ values are 0. Add these projects to NP set. Add projects v' to set SP_v for each project v in NP, such that for project v', $Y_{vv'} = 1$.

Step-3: Calculate no-delay renewable resource requirements of projects in NP set. Check if these resource amounts are sufficient for no-delay resource requirements of successor projects of v in set SP_v. If not, calculate necessary amounts as additional renewable resources. Determine desired general renewable resource capacities with these calculated values and then determine total required budget for these desired values. Calculate realized general renewable resource capacities from remaining budget, proportional to desired values and remaining resource budget as follows:

$$R'_k = \frac{DesiredResourceCapacity_k \times RemainingBudget}{TotalRequiredBudget} \qquad (18)$$

Step-4: Calculate total no-delay resource requirement of a project as the sum of no-delay resource requirement of corresponding project and additions required for successor projects. Calculate renewable resource dedication values for projects in NP proportional to their total no-delay resource requirements using general renewable resource capacities calculated in *step-3* as follows:

$$BR'_{vk} = \frac{TotalNoDelayRequirement_{vk} \times R'_k}{TotalNoDelayRequirement_k} \qquad (19)$$

Calculate the renewable resource transfer values for projects in SP with the following mathematical model:

$$min.z = \sum_{v}^{V} \sum_{k}^{K} (ND_{vk} - \sum_{v'}^{V} SR_{v'vk}) \qquad (20)$$

Subject to
$$SR_{vv'k} \leq BR'_{vk}Y_{vv'} \qquad \forall\ v, v' \in V \text{ and } \forall\ k \in K \qquad (21)$$

$$\sum_{v'}^{V} SR_{vv'k} \leq BR'vk \qquad v \in V \text{ and } \forall\ k \in K \qquad (22)$$

$$\sum_{v'}^{V} SR_{v'vk} \leq ND_{vk} \qquad v \in V \text{ and } \forall\ k \in K \qquad (23)$$

14

where BR'_{vk} is the renewable resource dedication values for projects in *NP* set calculated in previous *step-4* and ND_{vk} is the no-delay renewable resource requirement for projects in *SP* set. This mathematical model tries to determine resource transfers among available projects such that the deviation between total resource transfers to a project and its no-delay resource requirement is minimized over all projects.

Step-5: With renewable and nonrenewable resource dedication values, resource transfer values and precedence relations of projects at hand, the problem reduces to MRCPSP for each individual project. Solve each single project scheduling problem and obtain an initial solution by calculating start times and finish times of the projects considering sequence relations and single project scheduling results.

Note that the special structure of the mathematical model RPP-RRD allows the calculation in *step-5*. The individual projects have their time periods independent from the other projects in the multi-project environment. Thus when their individual schedules and sequence relations are given the resulting schedule for multi-project problem can be easily calculated.

Solution Improvement with CA for RP

Step-1: Apply CA for RD to the initial solution till no improvement is seen.

    Step 1.1 Determine preferences of projects for resources for each project

    Step 1.2 Determine total slack resources for each project (total difference between allocated resources and given resource capacities for each project)

    Step 1.3 Employ a continuous knapsack model to maximize the total preference value gained by dedicating the spare resources to projects.

    Step 1.4 Calculate schedules for each project with the given resource state.

Step-2: Apply CA for RP to the initial solution till no improvement is seen.

    Step 2.1 Calculate preferences for general resources summing up preferences of projects for resources over all projects.

    Step 2.2 Determine the slack resource budget for each resource (total difference between used budget and total budget)

    Step 2.3 Employ a continuous knapsack model to maximize the total preference values gained by transferring slack budget to general resource capacities.

    Step 2.4 Apply CA for RD to the new general resource capacity state

The preferences of projects for resources are calculated with a modified Lagrangian relaxation formulation of the single project scheduling problem. The MRCPSP formulation of Talbot (1982) is modified with a constraint which limits the completion time of the project with its no-delay completion time. The basic rationale behind this approach is to capture the infeasibility tolerated for the corresponding resource to reach the no-delay completion time with corresponding Lagrangian coefficients. The values of the Lagrangian coefficients are taken as preferences of projects for resources. Then using these preferences, the resource dedication state of the solution is moved to a more preferable state and a new schedule for each project is calculated. Detailed information related with CA for RD can be found in Besikci et al. (2012a)

Similarly CA for RP uses preferences of projects for resources to calculate general resource preferences simply summing up project preferences for each resource. Then the general resource capacity state is moved to a more preferable state and CA for RD is applied for this new solution to modify resource dedication values accordingly and to obtain a new solution as described in Besikci et al. (2012b).

4.4. Valid Inequality Generation

Valid inequalities can improve the B&C procedure and can have an important impact on the efficiency of the procedure. *CPLEX* generates various valid inequalities based on polyhedral theory. One additional valid inequality, namely sequence valid inequality, will be generated at every viable node whenever a $Y_{vv'}$ decision variable is set to 1:

$$F_{v'} \geq F_v + EP_v Y_{vv'} \qquad \forall (v, v') \in V : Y_{vv'} = 1 \tag{24}$$

where EP_v is the possible earliest completion time of project v calculated from CPM.

Note that these valid inequalities can be applied a priori in the problem formulation with the following constraint set:

$$F_{v'} \geq F_v + EP_v + \Omega(y_{vv'} - 1) \qquad \forall \ v, v' \in V \tag{25}$$

This approach will result with different solutions for the LP relaxation at each node which can improve the B&C procedure.

5. Experimental Study

To test the efficiency of the proposed modification for *CPLEX* two different problem sets are used. The first problem set is a control set, which is used to roughly identify the effects of different modifications and rapidly observe the project sequence relations among the projects. The second problem set is used for an extensive study.

The problems in the first set are taken from the PSPLIB (http://129 .187.106.231/ psplib/) Kolisch and Sprecher (1996). For each test problem, two projects from each *j16*, *j20* and *j30* sets are taken (a total of six projects in each multi-project scheduling problem) and one of the projects in each problem set has a weight value significantly higher than the other one. This modification enabled a rapid observation of project sequence relations when combined with a tight resource budget. The modes of the activities are modified such that the total resource cost of modes and their corresponding durations are proportional with diminishing returns for mode durations. The due dates of the projects are set to their calculated no-delay due dates. A total number of 20 problems are generated in this way. The results are given in Table 1 below. The first row shows the best combination of the proposed modifications according to absolute gap $((UB-LB)/LB)$ and objective value of the best feasible solution which turns out to be the strategy where $Y_{vv'}$ *Branching - Heuristic Solution - Sequence Valid Inequality* are combined. *AWT* and *AG* columns show the average weighted tardiness and average gap (distance of the objective value to the lower bound), respectively. Note that for both of the columns, only the common problems that have resulted in a feasible solution are used. Finally *ART* and *NS* columns show the average run time for the procedure in minutes and the number of problems for which no feasible solutions can be found with the given combination, respectively. The total run time limit is set to 240 minutes.

As can be seen in Table 1, *unmodified CPLEX* and BR_{vk} *Branching* approaches can only find two feasible solutions out of 20 problems. Thus for these cases a comparison cannot be made with the other solution approaches and they will be excluded from the analysis. When branching strategy of *CPLEX* is modified with $Y_{vv'}$ *Branching*, an important improvement can be obtained such that only three out of 20 problems are infeasible. Furthermore, when *heuristic solution approach* is included to the B&C procedure of *CPLEX*, these remaining three problems also yield some feasible solutions. The same improvements cannot be seen when branching strategy is

17

Table 1: The results for different combinations of the proposed modifications for the initial 20 test problems with 6 projects each.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**33.00**	**1.83**	**240**	**0**
Unmodified CPLEX	39.50	3.81	240	18
$Y_{vv'}$ Branching	74.71	6.65	240	3
Sequence Valid Inequality	39.5	3.43	240	18
A priori Valid Inequality	39.5	3.47	240	18
$Y_{vv'}$ Branching - Heuristic	60.30	4.75	240	0
$Y_{vv'}$ Branching - Heuristic Solution- A priori Valid Inequality	34.95	1.85	240	0
BR_{vk} Branching	51.50	4.51	240	18
BR_{vk} Branching - Heuristic	203.62	25.56	240	4
$Y_{vv'}$ and BR_{vk} Branching - Heuristic - Seq. Val. Ineq.	89.00	34.09	240	2

changed to BR_{vk} *Branching*. Even though the B&C procedure is improved with the addition of *heuristic solution approach*, the results are inferior to $Y_{vv'}$ *Branching* when compared according to solution quality.

Note that locally applied sequence as well as *a priori valid inequality* additions do not contribute to solution quality when solely applied. They cannot find a feasible solution for 18 problems out of 20. This shows that the valid inequality modification only improves the procedure when there is an upper bound provided with $Y_{vv'}$ Branching and heuristic solution approach as reported in Table 1. Note that there is not a significant difference between a priori applied and locally applied sequence valid inequalities. This trend is not expected to change when solving test problems, since there was not an improvement in B&C procedure with the different LP relaxations including *a priori valid inequalities*. Hence, we will not employ *a priori valid inequalities* further.

When modifications are compared according to the solution quality it is observed for which $Y_{vv'}$ *Branching - Heuristic Solution - Sequence Valid Inequality* combination gives the best results overall. We can observe that all $Y_{vv'}$ branching, heuristic solution approach and sequence valid inequality modifications have a cascading effect, such that the solution quality always increases significantly when modifications are used on top of each other. In Table 2, the AWT for problems that all of these three different levels of modifications have found feasible solutions are given.

Even though the modifications improve the UB values, the gap between LB and UB values are still relatively high. The $Y_{vv'}$ Branching and heuristic

Table 2: The results for cases such that all approaches found feasible solutions.

Combination	AWT
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**32.06**
$Y_{vv'}$ Branching - Heuristic	62.88
$Y_{vv'}$ Branching	74.71

solution approach modifications do not improve the average gap but sequence valid inequality has a significant effect on gap values when used with heuristic solution approach and $Y_{vv'}$ Branching strategy. Note that, despite the improvement in UB values, the gap is still very high such that a termination by reaching optimality cannot be observed.

To extensively test the solution approaches proposed here for RPP under RRD another set of test problems are used. These problems are grouped according to the maximum utilization factor (MUF) and the number of activities in a project Kurtulus and Narula (1995). MUF is the ratio of the resource requirement of no-delay schedule of the project to available resource, i.e., resource capacity. Thus, if MUF is less than or equal to one, the project can be scheduled without any delays. Since the resources cannot be shared, combining the projects and calculating the MUF values are not suitable for the proposed multi-project problem environment. In order to take into account resource dedication and resource budget concepts, the no-delay resource requirement of the multi-project problem is calculated as the budget value of the sum of no-delay resource requirements of the individual projects Besikci et al. (2012b). With this approach, if MUF value is less than or equal to one, the multi-project problem has necessary budget that will allow a no-delay schedule. Similarly when MUF value is increased, resource budget becomes tight for the multi-project problem.

Multi-project problems each consisting of 6 projects are created using activity-on-node representation. There are two levels for the number of activities for each of 6 projects in the test problems: 22 and 32 which are taken from PSPLIB (http://129 .187.106.231/psplib/) (Kolisch and Sprecher, 1996). Three different levels of MUF (1.5, 1.6, and 1.7) are selected and a full factorial design with 10 problems in each combination is created. In order to compare different methods, for each combination a total of 10 base problems are used. There are two levels for number of activities in a project: 22 and 32 activities.

The results for the problem sets consisting of 6 projects with 22 activities each are given in Tables 3, 4, 5; and for 32 activities in Tables 6, 7, 8, respectively. Note that when a solution procedure cannot find a reasonable number of feasible and/or optimal solution, it is not included in the subsequent tables with higher MUF values since the problems become tighter with respect to resource constraints.

Table 3: The results for different combinations of the proposed modifications for problems consisting of 6 projects each with 22 activities and 1.5 MUF.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**41.00**	**0.04**	**158.34**	**0**
$Y_{vv'}$ Branching - Heuristic	42.50	0.12	190.62	0
$Y_{vv'}$ Branching	44.40	0.17	195.97	0
Unmodified CPLEX	44.75	0.23	216.70	6

Table 4: The results for different combinations of the proposed modifications for problems consisting 6 projects each with 22 activities and 1.6 MUF.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**44.33**	**0.17**	**200.50**	**0**
$Y_{vv'}$ Branching - Heuristic	51.17	0.33	240	0
$Y_{vv'}$ Branching	44.40	0.28	240	4

Table 5: The results for different combinations of the proposed modifications for problems consisting of 6 projects each with 22 activities and 1.7 MUF.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**79.50**	**0.39**	**200.50**	**0**
$Y_{vv'}$ Branching - Heuristic	82.30	0.47	240	0

Considering the results for cases in Table 3 it can be seen the AWT for each modification combination and *unmodified CPLEX* are not significantly different but *unmodified CPLEX* cannot find feasible solution for most of the cases (6 out of 10 problems). Even though the AWT values do not differ significantly for solution approaches, the AG for Y_{vv} *Branching - Heuristic Solution - Sequence Valid Inequality* modification combination is significantly

lower than the other cases. This shows that when MUF values are not relatively tight project sequence valid inequalities can improve the lower bound for the B&C procedure. For the results obtained in Table 4 Y_{vv} *Branching - Heuristic Solution - Sequence Valid Inequality* modification combination has a significant AWT and AG advantage when compared to the remaining modification combinations. Note that $Y_{vv'}$ *Branching* modification starts to fall behind and cannot find feasible solutions for 4 cases out of 10. The lower AWT values for $Y_{vv'}$ *Branching* modification is the result of this limited feasible result base where it can find competitive results with the other approaches. The final test cases reported in Table 5 have some interesting implications. The two remaining modification combinations, namely Y_{vv} *Branching - Heuristic Solution - Sequence Valid Inequality* and Y_{vv} *Branching - Heuristic Solution*, do not have a significant AWT difference and the difference between AG values further diminishes with respect to cases with lower MUF values. This points out that the effectiveness of sequence valid inequality is highly correlated with MUF values and the improvement decreases when resource budget is tight.

Table 6: The results for different combinations of the proposed modifications for problems consisting of 6 projects each with 32 activities and 1.5 MUF.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**35.30**	**0.01**	**49.34**	**0**
$Y_{vv'}$ Branching - Heuristic	37.10	0.06	148.72	0
$Y_{vv'}$ Branching	35.20	0.01	49.34	0
Unmodified CPLEX	237.83	5.79	219.06	4

Table 7: The results for different combinations of the proposed modifications for problems consisting of 6 projects each with 32 activities and 1.6 MUF.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**47.33**	**0.25**	**240**	**1**
$Y_{vv'}$ Branching - Heuristic	50.78	0.35	240	1
$Y_{vv'}$ Branching	94.75	1.58	240	6

Table 8: The results for different combinations of the proposed modifications for problems consisting of 6 projects each with 32 activities and 1.7 MUF.

Combination	AWT	AG	ART	NS
$Y_{vv'}$ Branching - Heuristic Solution - Seq. Val. Ineq.	**96.62**	**0.48**	**240**	**2**
$Y_{vv'}$ Branching - Heuristic	96.62	1.06	240	2

The results for the cases with 32 activities are quite parallel to the results for the cases with 22 activities. $Y_{vv'}$ Branching - Heuristic Solution - Sequence Valid Inequality modification combination has a significant advantage for AWT and AG performance measures for all cases with diminishing returns for increasing MUF values.

6. Summary and Further Research Topics

In this study, an extension of Resource Dedication policy, namely Relaxed Resource dedication policy is investigated extensively. In RRD policy, resources are dedicated to individual projects, but they can be transferred among projects when one of the projects starts after the finish of the other one. The problem environment for this study is defined in such a way that general resource capacities are included as a new decision which is called Resource Portfolio Problem. The mathematical formulation for RPP under RRD policy is proposed and different features of the B&C procedure of *CPLEX* are modified and presented as a solution approach for the RPP under RRD policy.

The B&C procedure of *CPLEX* is modified with different branching strategies, valid inequalities and feasible solution approaches. We have used insights gained from our previous studies and applied modified versions of our previous heuristics for similar problems. When all these modifications are combined and used in B&C procedure of *CPLEX* the results for the test problems show the effectiveness of the modifications.

A new research direction would be defining new resource management policies that apply to different characteristics of multi-project environments that are not investigated in this study. One case can be considering resource transfers among projects with a time and/or value cost. This case can be considered as an extension of the studies by Kruger and Scholl (2008) and Kruger and Scholl (2009), where the individual projects have characteristics of RCPSP and general resource capacities are given as parameters. Another

research direction can be combining shared resources, dedicated resources and transferable resources and generating a general resource management setting.

Changing the objective criterion would open new research directions. Even though makespan based objective functions are common in multi-project scheduling, cost/revenue based objective functions also have an important place. A cost/revenue based objective function with an upper bound on makespan and/or due dates for certain milestones can be an interesting research direction.

References

Besikci, U., Bilge, Ü., Ulusoy, G., (2012). Resource dedication problem in a multi-project environment. *Flexible Services and Manufacturing Journal.* DOI: 10.1007/s10696-012-9140-9.

Besikci, U., Bilge, Ü., Ulusoy, G., (2012). Multi-mode resource constrained multi-project scheduling and resource portfolio problem. *Submitted to European Journal of Operation Research.*

Brucker, P., Drexl, A., M?hring R., Neumann, K., Pesch, E., (1999). Resource constrained project scheduling: notation, classification, models and methods. *European Journal of Operational Research*, 112, 3-41.

Demeulemeester, E., (1995). Minimizing resource availability costs in time-limited project networks. Management Science, 41(10), 1590-1598.

Gademann, N., Schutten, M., (2004). Linear programming based heuristics for project capacity planning. *IIE Transactions*, 37, 153-165.

Hartmann, S., Briskorn, D., (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*, 207, 1-14.

Herroelen, W., DeReyck, B., Demeulemeester, E., (1998). Resource constrained project scheduling: A survey of recent developments. *Computers and Operations Research*, 25, 279-302.

Kolisch, R., Sprecher, A., (1996). PSPLIB - a project scheduling problem library. *European Journal of Operational Research*, 96, 205-216.

Kolisch, R., Padman, R., (2001). An integrated survey of deterministic project scheduling. *OMEGA*, 29, 249-272.

Kruger, F., Scholl, A., (2008). Managing and modeling general resource transfers in (multi-)project scheduling. *OR Spectrum*, 32, 369-394.

Kruger, F., Scholl, A., (2009). A heuristic solution framework for the resource constrained (multi-)project scheduling problem with sequence dependent transfer times. *European Journal of Operations Research*, 197, 492-508.

Kurtulus, I.S., Narula, S.C., (1985). Multi-project scheduling: analysis of project performance. *IIE Transactions*, 17(1), 58-66.

Mohring, R.H., (1984). Minimizing costs of resource requirements in project networks subject to a fixed completion time. *Operations Research*, 32(1), 89-120.

Ozdamar, L., Ulusoy, G., (1995). A survey on the resource-constrained project scheduling problem. *IIE Transactions*, 27, 574-586.

Yang, K.K., Sum, C.C., (1997). An evolution of due date, resource allocation, project release and activity scheduling rules in a multiproject environment. *European Journal of Operation Research*, 103, 139-154.

Talbot, F.B., (1982) Resource-constrained project scheduling with time-resource tradeoffs: the nonpreemptive case. *Management Science*, 28(10), 1199-1210.

Users Manual for CPLEX 2009; ftp://public.dhe.ibm.com/ software/websphere/ilog/docs/optimization/cplex/ps_usrmancplex.pdf: ILOG, May 2012.

Zhu G., Bard J.F., Yu G., (2007). A branch-and-cut procedure for the multimode resource-constrained project-scheduling problem. *INFORMS Journal on Computing*, 18(3), 377-390.