PhD Thesis Abstract
Doctoral School of Business Administration

A New Bi-Objective Hybrid Metaheuristic Algorithm for the Resource-Constrained Hammock Cost Problem (RCHCP)

Oren Eliezer

University of Pécs
Faculty of Business and Economics

Pécs, 2011

Title of the Doctoral School: Doctoral School of Business Administration

School Leader: Prof. Dr. Iván Bélyácz

Title of the Doctoral Program:
International PhD program

Program Leader: Dr. Edit Bányai

Discipline: Project Management and Scheduling

Supervisors: Prof. Dr. György Csébfalvi

and Prof. Roni Levi

Contents

1. Introduction

Project management has become a mainstream discipline in every institute in the world, and a challenge that every manager has to deal with during his/her career.

Project management is sometimes considered an “art of management”, since it demands a variety of skills, abilities, and the implementation of many other disciplines. It covers wide areas of management like: budget, human resources management, quality control, and risk and time management.

People recognize that most of the simplest activities, like traveling from one city to another, require prior planning. This planning entails resource management. A project is defined as: any process whereby its activities and sub-activities need resources and so a preset goal is reached. However since ancient times people are in fact dealing with projects on a daily basis; only in the 1950's the formal tools and definitions of project management were developed. Firstly, techniques were developed to deal with time allocations. These techniques ensured that every activity would get its time window (its beginning and ending times), and that the whole project would be concluded by the dictated deadline. Next, another time technique was developed which dealt with resource allocation under time limits. All of these techniques were collectively termed: ***scheduling.*** Scheduling is a theory which deals with ***time constraints*** and in particular ***resource constraints.***

Thus scheduling became one of the most important issues in project management ─ both in research and practice. Resource scheduling, if done correctly, can make the difference between successful and unsuccessful projects. Many projects had a lot of activities and many kinds of resources, so scheduling becomes a NP-hard problem in project management. Many kinds of heuristics have been developed as a result of projects becoming more and more complex.

This dissertation addresses time scheduling, which includes the problem of resource allocation, and deals mainly with the scheduling of regular and hammock activities. The main objectives of this research are:

1. Minimization of a project's makespan as a primary criterion.
2. Minimization of hammock cost as a secondary criterion.

This research is part of a larger project, conducted under supervision of Csébfalvi, G., and is geared toward developing a new model, in order to solve the Resource Constrained Hammock Problem (RCHCP).

The latter combines three innovative algorithms: the Sound of Silence (SoS) algorithm (Csébfalvi, G., 2007) and conflict repairing harmony search metaheuristic (Csébfalvi, G., Eliezer O., Láng B., Levi, R., 2008b), and the hybrid algorithm (Eliezer, O. and Csébfalvi, G., 2009). This new approach aims towards getting a minimal makespan as the primary objective and a minimal Hammock Cost (HC) as the secondary one.

Hammock activities play a significant role in project management. Most of the research conducted in this area concentrated on unconstrained cases. Although pioneering attempts to address the constrained case were made by Vanhoucke M. and Van Osselaer K. (2004), and Csébfalvi, G. and Csébfalvi, A.(2005), the literature does not offer a general and useful method to compute the constrained hammock activities' durations. This dissertation presents a new metaheuristic algorithm to address this kind of issue; this new approach will calculate the hammock's duration by minimizing the hammock cost and makespan of the whole project. In keeping with this new approach, minimization of the hammock cost should reduce the total project's cost; this should be done by fixing the hammock's hangers according to the optimal solution.

The main objectives of the research are:

- To present a new approach towards dealing with hammock activities under resource constraints.
- To introduce a new algorithm, one that is based on: solving the MILP problem, and the forbidden set principle. This algorithm cannot provide an exact and optimal solution owing to the NP-hard of the problem.
- To implement a heuristic of Sound of Silence algorithm (SoS) in order to get a near-optimal solution for the MILP model above.
- To emphasize new results, and prove that those results are better than most other known scheduling approaches.

Research's stages:

- We present the traditional approach to solving RCPSP problems.

- We present a scheduling of hammock activities without constraints – according to the Harhalakis G. (1990) paper.
- We describe research that was done on the RCHCP (scheduling hammock activities under resource constraints) – mainly on small-to-medium sized problems.
- We present SoS heuristic, in order to treat large-scale problems.
- We present new results and conclusions.

The following booklet is organized as follows:

- ***Section no.2*** is divided into sub-sections:

  2.1 define the hammock activity and its importance.

  2.2 describe the constraint hammock problem and the research was done in treating it.

2.3 describe the Harmony Search algorithm, which makes an analogy between music world and RCPSP problem. This section presents an interesting attitude of dealing with RCPSP problem solving.

- ***Section no. 3*** is divided into three sub-sections:

3.1 describe the SoS algorithm, as a heuristic method that is used to solve conflicts and "relief" the resource constraint hammock problem into unconstraint model, and the hybrid algorithm.

3.2 describe the NP-Hard RCHC model.

3.3 describe the unconstraint Hammock Cost Problem (HCP)

Section no. 4 describes the new results.

- ***Section no. 5*** contains the references list.

2. Theoretical model

2.1 Definition of a hammock activity

Hammock activity is an activity that we schedule between “regular” activities, since its duration cannot be estimated or calculated at the initial time of project planning. A hammock activity contains a collection of "regular" activities which have the same starting and ending points and/or need special equipment and/or special resource(s). Since hammock activity has the same starting and ending points similar to a standard project, hammock activity could be considered as a project itself, except that there is no significance in the order between its inner activities. Hammock activities can play a useful role in project management. Typically, they have been used to denote usage of equipment needed for a particular chain of activities (e.g. a load lifting device), without predetermining the estimated time the equipment must be present on site. Similarly, it may be required to pick up the cost of a complete section of the project, or more usually some background cost related to a section. Such background costs could arise from storage, supervision etc. and can be allocated to hammock activity. Also for upper management reporting hammocks are used to collectively represent a sequence of consecutive normal activities, all of which form the task of one department, or relate to the same cost center.

The use of hammock activity has become more and more popular; computer programs are developed to handle project management dilemmas. Software helps treat hammocks as a part of a whole project. However there is some confusion about estimating the hammock's duration. Consequently every hammock activity is connected on both sides to regular activities; it is connected in such a way that all its activities have the same starting and ending points.

Hammock and the scheduling process — hammock activity constitutes a group of activities which are not concerned with regular precedence and resource constraint; they therefore do not apparently affect the scheduling process and the project makespan.

Hammock cost describes the expense of every hammock member for operation time unit, so as a result, managers can use it as a subproject which have a starting and ending times and a known total expense, but without managing it directly.

2.2 The constraint hammock problem

Vanhoucke and Van Osselaer (2004) solved the tunnel problem in the Netherlands by applying the hammock theory**.** It was the first time that hammock activities were scheduled under resource constraints; however their model was quite restricted, because it was applied as a repetition procedure. Csébfalvi G. and Csébfalvi A. (2005) proposed an algorithm which deals with the hammock's scheduling under resource constraints. Thus a new approach was developed: RCHCP that is a solution to the scheduling problem under resource constraints and hammock cost consideration. The algorithm is based on solving the MILP problem and IE (implicit enumeration), which are both based on: the *forbidden set principle*.

Forbidden set: A forbidden activity set F is identified such that:

(1) All activities in the set may be executed concurrently.

(2) The usage of some resources by these activities exceeds the resource availability.

(3) The set does not contain another forbidden set as a recognized subset.

A resource conflict can be repaired explicitly by inserting a network feasible precedence relation between two forbidden set members, which will guarantee that not all members of a forbidden set can be executed concurrently. At the same time, an inserted explicit conflict repairing relation (as its side effect) might be able to repair one or more of the other conflicts implicitly. Let $i \to \cdots \to j$ denote that activity j is a direct (indirect) successor of activity i. An $i \to j$ explicit repairing relation might be replaced by an $p \to q$ implicit relation, where $i \to \cdots \to p$ and $q \to \cdots . \to j, i \neq p \cup p \neq j$ if there is another forbidden set for which $p \to q$ is an explicit repairing relation. Let ER(F) (IR(F)) denotes the set of implicit (explicit) repairing relations for forbidden set F.

The existing algorithms are exact algorithms which provide a solution only for small to medium sized problems. These algorithms do not provide a solution in polynomial for a large scale problem, and so a heuristic algorithm is needed. As a result of this limitation, a new heuristic (and efficient) tool has to be developed. The heuristic algorithm in this research is SoS, which should discuss later.

2.3 Harmony search (HS)

The HS process is an analogy between project management scheduling under resource constraints (RCPSP) and the music world. Lee K.S. and Geem Z.W. (2005) developed a useful tool: a metaheuristic algorithm which efficiently solves a RCPSP problem. Every activity i $(i \in \{1,2, \ldots\ldots N\})$ is represented by a player in the orchestra. Every player makes a single sound from his/her initial repertoire. This single sound has a known time-range and is chosen randomly by the player who determines its timing. All those sounds are collected into a melody which contains N sounds from N players. This melody is compared to the worst one in the known repertoire to date. When the new melody is better than the worst one in the repertoire it replaces it. Thus the repertoire's quality steadily improves with every step, as does the solution, and there is a better chance to get the near-optimal solution to the second criteria of the problem (RCHCP).

Every player is an activity that has to be scheduled in the RCPSP world. The timing of entering into the melody is represented by a time variable that is limited to the activities' time duration. In the end the N-players melody is a new schedule that was created by improvisation. This algorithm's purpose is to find the best schedule by improvisation, where 'best' means the shortest and most feasible makespan from the improvisation with a near-optimal, to the hammock cost as a second objective. During all this process, the aesthetic value of the sounds is important, meaning that in every scheduling period, no resource violation is allowed. In conclusion, 'best scheduling' means: the shortest and most feasible scheduling. The main advantage of HS over and above other methods is that there is no need for a complicated mathematical calculation to get initial values of time variables. Instead, HS uses stochastic variables to get random values. HS solves quite complicated problems, not only scheduling problems, but also problems in other areas of engineering.

In the language of music, the RCPSP can be summarized as follows:

- the band consists of N musicians;
- the polyphonic melody consists of R phrases and N polyphonic sounds;
- each $i \epsilon \{1,2......n\}$ musician is responsible for exactly one polyphonic sound;
- each $i \epsilon \{1,2......n\}$ polyphonic sound is characterized by the set of the following elements: $\left\{X_i, D_i, \left\{R_{ir} | r \in \{1,2, \dots \dots R\}\right\}\right\}$; the polyphonic sounds (musicians) form a partially ordered set

according to the precedence (predecessor-successor) relations;

- each $r \in \{1,2......R\}$ phrase is additive for the simultaneous sounds;

- in each phrase only the high sounds are audible: $\{U_{tr} | U_{tr} > R_r, t = 1,2, \ldots\ldots T\}$;

- in each repertoire’s uploading (improvisation) step, each $i \in \{1,2......n\}$ musician has the right to present (modify) an idea $IP_i \in [-1, +1]$ about X_i where a large positive (negative) value means that the musician wants to enter the melody as early (late) as possible;

- in the repertoire's uploading phase the "musicians" improvise freely, $IP_i \leftarrow RandomGauss(0,1)$

  The function $\eta \leftarrow RandomGauss(\mu, \sigma)$ generates random numbers from a truncated

$-1 \leq \eta \leq 1$ normal distribution with mean μ and standard deviation σ;

- During the improvisation phase the "freedom of imagination" is monotonically decreasing step by step, $IP_i \leftarrow RandomGauss(0,1)$ where standard deviation σ is a decreasing function of the progress. The conductor reduces the freedom of players by an asymmetric normal-distribution with a monotonic growth towards 1 (meaning that most of schedules should be timed as early as possible);

- each of the possible decisions of the HS process (melody selection and idea-driven melody construction) is the conductor's responsibility;

- the band tries to find the shortest "SoS" melody by improvisation.

3. New Model

3.1 The Sound of Silence (SoS) metaheuristic algorithm

The central element of the “SoS algorithm” can be classified as a pure forward-backward list scheduling without improvement. In this element the only, but extremely important novelty, is the developed activity list generator, which is practically independent of the applied metaheuristic frame, and able to provide near-optimal solutions for large “academic” instances very quickly. The fast and effective activity list generator is based on an “unusual” integer linear programming problem which can be solved in polynomial time by relaxing the integrality assumption. The computational results show that the SoS is a fast and high quality algorithm.

The SoS algorithm is an improvement version of Harmony Search. At the beginning of the iterations the conductor collects the ideas. The ideas are random sounds which are produced by the players in the orchestra which is the initial basis of the repertoire. All ideas are activities which have to be scheduled, and the schedule itself is an LP problem.

The conductor solves the inductive logic programming (ILP) problem as an LP using a fast interior point method. Usually the result of the optimization is a “very interesting” schedule (melody).

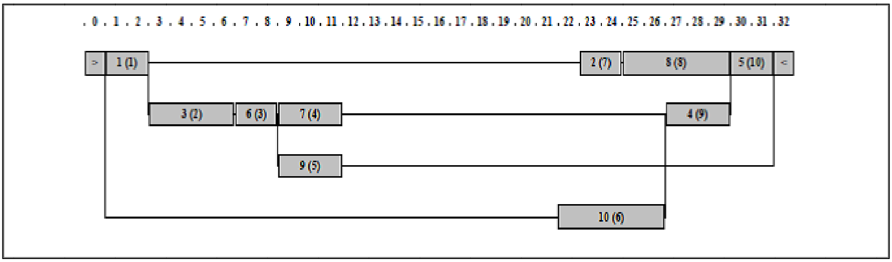

Figure 1: Activity list generation ─ two loud parts with a "long break" between them (Csébfalvi et al., 2008a)

In the case of a simple project the melody consists of two (loud) parts (themes) with a very long break between them (as in Figure 1 above). According to our objective function, in the schedule, activity 4 and activity 8 border one another and push activity 5 forward as much as possible, so activity 5 does not have enough force to avoid the aggression.

This procedure is used by the conductor for determining an initial schedule which defines the final starting (entering) order of the sounds (musicians).

When two or more activities have the same starting time (e.g. activity 7 and 9), the conductor solves the problem by random permutation.

The essence of the activity list oriented (serial) forward-backward scheduling is well known.

The schedule generation schema transforms activity list L into a schedule $S(L)$ by taking the activities one by one in the order of the activity list and scheduling them at the earliest (latest) precedence and resource-feasible starting time. In the example above, the order is dictated by the following vector $I = \{1,3,6,7,9,10,2,8,4,5\}$. This vector is created as a result of tossing random numbers between $[-1, 1]$. Those numbers are tossed by the players in the orchestra. When a positive (negative) number is tossed the activity should be scheduled earlier (later). After that, in the “SoS algorithm” the conductor selects the best (makespan minimal) schedule. When two or more schedules have the same makespan, the conductor chooses the forward one (earlier) between the two.

As previously mentioned, the vector I created as a decision of the players, e.g. *I={0.3,-0.1,0.5,-0.3,0.4,-0.2,0.3,-0.6 ,0.7,-0.6}.* This means that activities no. 1,3,5,7 and 9 should be scheduled at their earliest starting times, however the rest at their latest. Therefore, the conductor controls the players' freedom by solving an LP problem.

The conflict repairing version of SoS is based on the forbidden set concept. In the conflict repairing version the primary variables are conflict repairing relations; a solution is a makespan minimal resource-feasible solution set in which every movable activity can be shifted without affecting the resource feasibility. In the traditional “time oriented” model the primary variables are starting times, therefore an activity shift may be able to destroy the resource feasibility. The makespan minimal solutions of the conflict repairing model are immune to the activity movements; therefore we can introduce a secondary performance measure to select the “best” makespan minimal resource feasible solution from the generated solution sets. In the “SoS algorithm”, according to the applied replacement strategy (whenever the algorithm obtains a solution superior to the worst solution of the current population, the worst solution will be replaced by the better one), the quality of the population gradually improves. The larger the makespan minimal subset size becomes, the greater the chance to get a good solution for the secondary criterion. It is well-known that the crucial point of the conflict repairing model is the forbidden set computation. In the "SoS algorithm" the conductor uses a simple (but fast and effective) "thumb rule" to decrease the time requirement of the forbidden set computation. In the forward-backward list scheduling process the conductor (without explicit forbidden set computation) inserts a precedence relation $i \rightarrow j$ between an already scheduled activity i and the currently scheduled activity j whenever they are connected without lag ($S_i + D_i = S_j$). The result is a schedule without "visible" conflicts. Thereafter the conductor (in exactly one step) repairs all of the hidden (invisible) conflicts, and always inserts the "best" conflict repairing relation for each forbidden set. In this context "best" means a relation $i \rightarrow j$ between two forbidden set members for which the lag ($S_i - S_j - D_i$) is maximal. In the language of music, the result of the conflict repairing process will be a robust (flexible) “SoS” melody, in which the musicians have some freedom to join the performance without affecting the aesthetic value of the composition.

Theoretically the resource-constrained case can be formulated as a MILP problem which can be solved for small-scale projects within a reasonable time. It is important to note that in the HS algorithm the RCHC measure of a “repaired” schedule can be evaluated in polynomial time by using the presented unconstrained LP formulation.

A resource profile of the scheduling problem out of a forward and backward procedure presented hereunder, clarifies the above mentioned example:

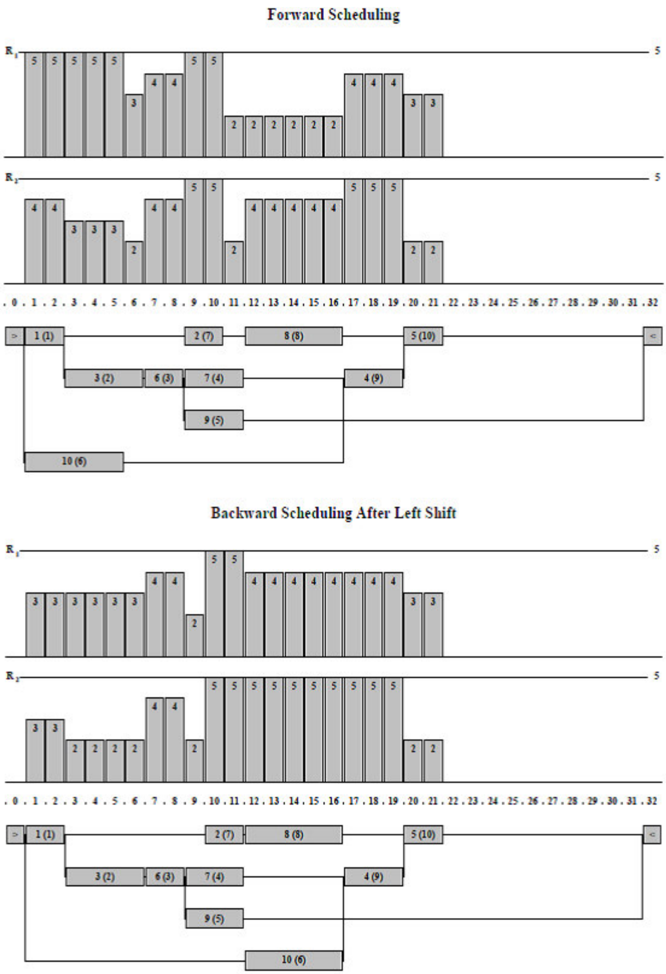

Figure 2: Activity list oriented serial forward-backward scheduling

(Csébfalvi, et al., 2008a)

The solution is a feasible schedule with a makespan of 21. Later, this idea was expanded to deal with hammock activities.

3.2 The NP-Hard RCHC problem

In order to model hammock activities in projects, we consider the following resource constrained project-scheduling problem: a single project consists of N real activities $i \in \{1,2,\ldots.N\}$ with a non-preemptable duration of D_i periods. Furthermore, activity $i=0$ ($i=N+1$) and is defined as the unique dummy source (sink). The activities are interrelated by precedence and resource constraints. Precedence constraints prevent an activity from being started before all its predecessors are finished.

Let:

$$PS = \begin{Bmatrix} i \to j, i \neq j, i \in \{0,1, \dots .N\}, \\ j \in \{1,2, \dots \dots N+1\}, FS_{ij} = 0 \end{Bmatrix} \qquad (1)$$

denote the set of immediate predecessor-successor relations. Resource constraints arise as follows: in order to be a processed activity i require R_{ir} units of resource type $r \in \{1,2, \dots R\}$ during every period of its duration. Since resource $r, r \in \{1,2, \dots R\}$ is only available with the constant period availability of R_r units for each period, activities might not be scheduled at their earliest (network-feasible) start time but later.

Let $\bar{T}$ denote the resource-constrained project's makespan (or its upper bound) and fix the position of the unique dummy sink in period $\bar{T}+1$. Let H denote the number of hammock activities.

A hammock activity: $\overleftrightarrow{H}_{h,}$ $h \in \{1,2, \dots \dots H\}$ can be represented by a dummy activity pair: $\{\vec{H}_h, \overleftarrow{H}_h\}$ with zero duration (hammock hangers), and a membership function: $M_h(i)$ to identify the hammock members. Following the constraint

(4) below, the membership function $M_h(i)$ is a Boolean one, which returns the true value when an activity i is found to be the inner activity of the hammock.

The NP-hard RCHC problem can be written as follows:

$$min\left[\sum_{h=1}^{H} C_h \cdot \overleftrightarrow{H}_h\right] \tag{2}$$

$$\overleftrightarrow{H}_h = \overleftarrow{H}_h - \overrightarrow{H}_h + 1, h \in \{1,2,\ldots..H\} \tag{3}$$

$$H_h = \{i | M_h(i) = true, i \in \{1,2,\ldots.,N\}\}, \tag{4}$$

$$|H_h| \geq 2, h \in \{1,2,\ldots.,H\}$$

$$\overrightarrow{H}_h \leq X_i, i \in H_h, h \in \{1,2,\ldots.,H\} \tag{5}$$

$$X_i \leq \overleftarrow{H}_h, i \in H_h, h \in \{1,2,\ldots.,H\} \tag{6}$$

$$X_i + D_i \leq X_j, i \rightarrow j \in PS \tag{7}$$

$$X_{N+1} = \bar{T} + 1 \tag{8}$$

$$X_i = \sum_{t \in T_i} X_{it} \cdot t, T_i = \{EST_{i,} EST_i + 1, \ldots., LST_i\}, \tag{9}$$

$$i \in \{1,2, \ldots. N\}$$

$$\sum_{t \in T_i} X_{it}, X_{it}=1, X_{it} \in \{0,1\}, i \in \{1,2, \ldots. N\} \tag{10}$$

$$A_t = \{i | X_i \le t \le X_i + D_i, i \in \{1,2, \ldots. N\}\}, \tag{11}$$

$$t \in \{1,2, \ldots. T\}$$

$$U_{tr} = \sum_{i \in A_t} R_{ir,}\ t \in \{1,2, \ldots. T\}, \tag{12}$$

$$r \in \{1,2, \ldots. R\}$$

$$U_{tr} \le R_r t \in \{1,2, \ldots. T\}, r \in \{1,2, \ldots. R\} \tag{13}$$

Objective (2) minimizes the total hammock cost subject to the precedence relations and the limited resource availabilities. Constraints (3-6) define the distance of the hammock hangers, identify the hammock members, and describe the relations between the hammock hangers and members for each hammock activity $h,\ h \in \{1,2,\ldots.,H\}$. Constraints (7) represent the precedence relations. Constraint (8) defines the deadline (latest completion time) of the project. Constraints (9-10) define the activity starting times in the function of the binary indicator variables, and ensure that each activity has exactly one starting time within its time window $T_i = \{EST_{i,}EST_i + 1,\ldots.,LST_i\}$ where $EST_{i\,,}(LST_i)$ is the early (late) starting time for activity, $i \in \{1,2,\ldots.N\}$ according to the precedence constraints and the latest project completion time $\overline{T}$. Constraints (11-13) ensure that resources allocated to activities at any time during the project do not exceed the resources' availability.

3.3 The unconstrained Hammock Cost Problem (HCP)

The unconstrained hammock cost problem (HCP) can be formulated as a LP problem therefore it can be solved in polynomial time. Although the heuristic above is resource constrained, the analysis of the hammock is carried out in the unconstrained case. The reason is: by using the metaheuristics of SoS that was developed by Csébfalvi et al. (2008a, 2008b), and Eliezer and Csébfalvi G. (2009) and which is based on the forbidden set principle, all resource conflicts are solved in advance, as a result we get a robust (flexible) schedule in such a way that all activities can be moved without a negative effect to its feasibility. The following algorithm can be applied on polynomial time which facilitates matters, and a hammock activity can be analyzed by using the unconstrained case below:

$$HC^* = min\left[\sum_{h=1}^{H} C_h \cdot \overleftrightarrow{H}_h\right] \qquad (14)$$

$$\overleftrightarrow{H}_h = \overleftarrow{H}_h - \overrightarrow{H}_h + 1, h \in \{1,2,\ldots..H\} \qquad (15)$$

$$H_h = \{i | M_h(i) = true, i \in \{1,2, \ldots., N\}\}, \qquad (16)$$

$$|H_h| \geq 2, h \in \{1,2, \ldots., H\}$$

$$\vec{H}_h \leq X_i, i \in H_h, h \in \{1,2, \ldots., H\} \qquad (17)$$

$$X_i \leq \overleftarrow{H}_h, i \in H_h, h \in \{1,2, \ldots., H\} \qquad (18)$$

$$X_i + D_i \leq X_j, i \to j \in PS \qquad (19)$$

$$X_{N+1} = \bar{T} + 1 \qquad (20)$$

$$EST_i \leq X_i \leq LST_i \qquad (21)$$

The model (14)-(21), is similar to (2)-(13) however there are no resource constraints at all. The heuristic produces (2-13) resource-feasible schedules (where every possible activity's movement is resource-feasible). In order to investigate its hammock cost we have to use only the unconstrained model. The early (late) model with fixed hammock durations is very important from a managerial point of view, because it describes the flexibility of the hammock activities (a hammock activity should move without affecting the resource feasibility at any time period).

In order to illustrate the RCHCP we present hereunder (figure no.3) a simple non-resource feasible project instance with 20 activities, while activities no. 5,6,9,10,13 and 14 are hammock members. (Activities no. 2 and no. 17 are the hammock hangers, with zero duration each):

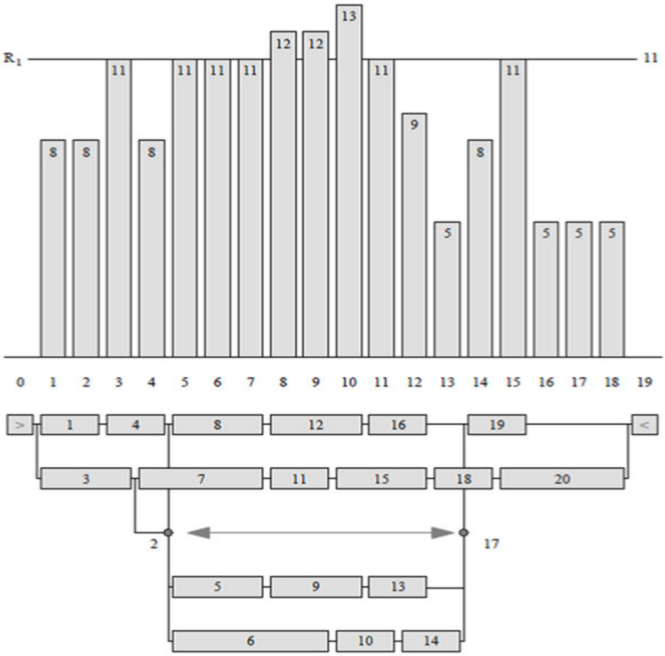

Figure 3: The early CPM of a simple example (Csébfalvi G.and Csébfalvi A., 2005)

Following figure 3, the total makespan of the project equals 18 time units and the hammock cost equals (HC) equals 9 time units, but it is not feasible.

Figure 4 illustrates the optimal solution of the project above:

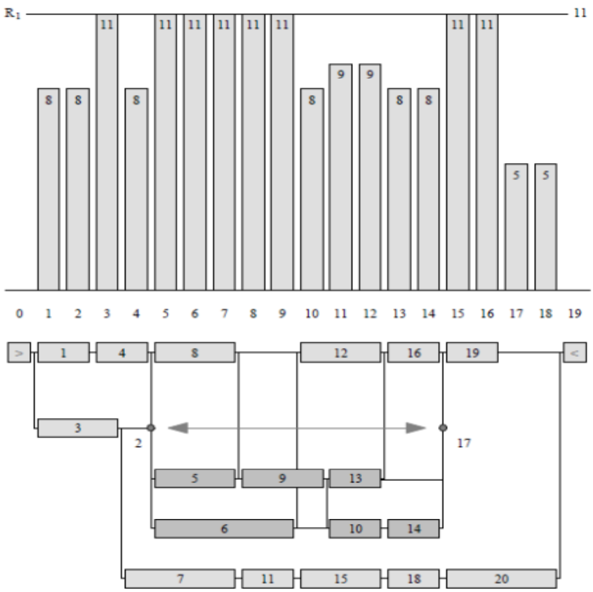

Figure 4: The optimal solution of a simple resource-constrained example

As a result of conflict repairing process, the original structure of the project had been changed, however we got a feasible and optimal solution which contains a wider hammock activity: the hammock cost is changed from 9 time units into 10, however the total makespan wasn't changed at all.

In the ternary phase, after fixing the position of the finishing milestone according to the optimal makespan and the hammock durations according to the optimal durations and inserting the optimal conflict repairing relations, I introduced a ternary criterion as a MILP to smooth out the resource usage histograms on the set of possible activity movements.

Herein the MILP model of leveling procedure:

$$\min\ LM = \sum_{r=1}^{R}\sum_{t=1}^{T} CU_{rt}^{+} = LM^{*} \tag{22}$$

$$\overleftrightarrow{H}_h = \overleftarrow{H}_h^{\ *}, h \in \{1,2,\ldots\ldots H\} \tag{23}$$

$$H_h = \{i|M_h(i) = true, i \in \{1,2,\ldots.,N\}\}, |H_h| \geq 2, h \in \{1,2,\ldots.,H\} \tag{24}$$

$$\vec{H}_h \leq X_i, i \in H_h, h \in \{1,2, \dots ., H\} \quad (25)$$

$$X_i \leq \overleftarrow{H}_h, i \in H_h, h \in \{1,2, \dots ., H\} \quad (26)$$

$$X_i + D_i \leq X_j, i \to j \in PS \quad (27)$$

$$X_i = \sum_{t \in T_i} X_{it} \cdot t, \;\; T_i = \{\underline{X_i}, \underline{X_i} + 1, \dots .., \overline{X}_i\}, i \in \{1,2, \dots N\} \quad (28)$$

$$\sum_{t \in T_i} X_{it} = 1, \;\; X_{it} \in \{0,1\}, \quad i \in \{1,2, \dots N\} \quad (29)$$

$$A_t = \{i | X_i \leq t \leq X_i + D_i, i \in \{1,2, \dots N\}\}, t \in \{1,2, \dots . T\} \quad (30)$$

$$U_{tr} - CU_{tr}^{+} + CU_{tr}^{-} = U_{t-1,r}\;, t \in \{1,2, \dots . T\}, r \in \{1,2, \dots . R\} \quad (31)$$

$$U_{1r} - CU_{1r}^{+} = 0, r \in \{1,2, \dots . R\}$$

$$U_{tr} \leq R_r, t \in \{1,2, \dots . T\}, r \in \{1,2, \dots . R\} \quad (32)$$

The objective (22) minimize the total amount of resource using r, at time t, while $r \in \{1,2, \dots . R\}$ and $t \in \{1,2, \dots . T\}$. (23) determines the fact that the hammock is scheduled in its optimal position, (24-26) defines the hammock's inner activities, (27) determines the precedence relations , (28-30) define exactly one starting point of every activity, following its time window, (31) and (32) define the smoothness of usage resources, in such a way that the heights difference of each adjacent histograms should be minimized. In order to illustrate the leveling process, herein a simple example, which emphasizes the non-feasible starting state and the optimal solution with a balanced (smooth) usage of resources:

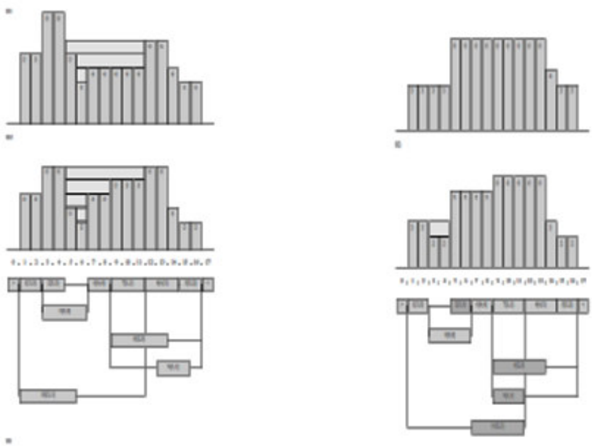

Figure 5a: An early schedule Figure 5b: optimal solution to the leveling problem (Konstantinidis, 2002)

The applied leveling/smoothing criterion (developed by Csébfalvi and Konstantinidis (2002)) can be replaced by any other ternary criterion. The MILP approach can be replaced by a problem-specific heuristic algorithm to manage larger problems.

4. New results

1. I developed a new multi-phase MILP approach to manage the RCHCP, where the hammock hangers are represented by dummy activities with zero duration. Starting base: Harhalakis (1990) (HCP) and Csébfalvi and Csébfalvi (2005) (RCHCP):

    a. In the first phase, I solved the RCPSP to get the resource-feasible makespan.

    b. In the second phase, after fixing the position of the finishing milestone according to the optimal makespan, I solved the RCHCP on the set of the possible conflict repairing relations, to get the optimal hammock durations. According to the applied forbidden-set-oriented approach, the optimal schedule of the second phase is totally invariant to the activity movements; therefore an activity move is unable to destroy the resource-feasibility.

c. In the ternary phase, after fixing the position of the finishing milestone according to the optimal makespan and the hammock durations according to the optimal durations and inserting the optimal conflict repairing relations, I introduced a ternary criterion as a MILP to smooth out the resource usage histograms on the set of possible activity movements. The applied leveling/smoothing criterion (developed by Csébfalvi and Konstantinidis (2002)) can be replaced by any other ternary criterion.

The MILP approach can be replaced by a problem-specific heuristic algorithm to manage larger problems.

2. I developed a harmony search metaheuristic algorithm to manage RCHCP. The algorithm is based on the time-oriented Sounds of Silence (SoS) algorithm developed by Csébfalvi for RCPSP. The new problem-specific forbidden-set-oriented SoS algorithm manages the primary and secondary criteria simultaneously using a FBI approach. The new forbidden-set-oriented elements are invariant to the SoS algorithm; therefore they may be inserted to any other population-based metaheuristic algorithms which generate resource-feasible solutions using a variant of FBI. The hammock-oriented secondary criterion can be replaced by any other criterion which can be optimized on the set of the resource-feasible activity movements. The applied ternary criterion (Konstantinidis (2002)) can be replaced by any other criterion which is able to rearrange the schedule according to a well-specified managerial

goal without affecting the primary (secondary) optimality. The introduced ternary MILP approach can be replaced by a problem-specific heuristic (metaheuristic) algorithm, which generates resource-feasible activity movements and maintains the ternary criterion time to time.

5. References

- Csébfalvi, G., and Csébfalvi A. (2005). "Hammock activities in project scheduling***", in Proceedings of the Sixteenth Annual Conference of POMS***, ***S. Panwalkar, J. Li, (Editors)***, POMS, USA, . http://www.poms.org/ConfPapers/003/003-0572.pdf

- Csébfalvi, G. (2007). "Sounds of Silence: a harmony search metaheuristic for the resource-constrained project scheduling problem." ***European Journal of Operational Research***, (under review).

- Csébfalvi, G., Csébfalvi, A., and Szendrői E. (2008a). "A harmony search metaheuristic for the resource-constrained project scheduling problem and its multi-mode version", ***in Project management and scheduling 2008,*** F. S. Serifoglu, and Ü. Bilge (Editors), 56-59, Istanbul, Turkey.

- Harhalakis G. (1990). "Special features of precedence network charts." ***European Journal of Operational Research*** 49 (1), 50-60.

- Konstantinidis, P. (2002). "A model to optimize project resource allocation by construction of a balanced histogram". Unpublished Ph.D. dissertation, University of Pécs, Department of Business Administration and Economics.

- Lee K.S. and Geem Z.W. (2005). "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice." ***Computer Methods in Applied Mechanics and Engineering,*** 194, 3902-3933.

- Vanhoucke M., and Van Osselaer K. (2004). "Work continuity in a real-life schedule: the Westerschelde Tunnel", ***Working Papers of Faculty of Economics and Business Administration, Ghent University, Belgium Faculty of Economics and Business Administration ,*** October, wp_04_271.

Publication list

- Csébfalvi, G., Eliezer O., Láng B., Levi, R. (2008b)."A conflict repairing harmony search metaheuristic and its application for bi-objective resource-constrained project scheduling problems", ***in Project managementand scheduling 2008***, Serifoglu, F.S., Bilge,Ü. (Editors), 60–63. Istanbul, Turkey.

- Eliezer O., and Csébfalvi G. (2009). "A hybrid method for the resource-constrained project scheduling problem with hammock activities", **in *Proceedings of the first international conference on soft computing technology in civil, structural and environmental engineering*,** B.H.V. Topping, & Y. Tsompanakis, (Editors), Civil-Comp Press, Stirlingshire, United Kingdom, paper: 1, 2009. doi:10.4203/ccp.92.1

- Eliezer O. and Levi R. (2010)."A hybrid method for the resource-constrained project scheduling problem with hammock activities and strip packing like resource constraints", in ***Proceedings of the seventh international conference on engineering computational technology***, B.H.V. Topping, J.M. Adam, F.J. Pallarés, R. Bru, and M.L. Romero, (Editors), Civil-Comp Press, Stirlingshire, UK, Paper 91, 2010. doi:10.4203/ccp.94.91.

- Eliezer, O., and Levi, R. (2011). "An improved hybrid algorithm for the resource-constrained project scheduling problem with hammock activities." ***Ref: CSC2011/2010/000020.***