

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Scheduling of Construction Projects with a Hybrid Evolutionary Algorithm's Application

Wojciech Bożejko¹, Zdzisław Hejducki¹,
Magdalena Rogalska² and Mieczysław Wodecki³

¹*Wrocław University of Technology, 50-370 Wrocław, Wybrzeże Wyspiańskiego 27,*

²*Lublin University of Technology, 20-950 Lublin, Nadbystrzycka 40,*

³*Wrocław University, 50-383 Wrocław, Joliot-Curie 15,
Poland*

1. Introduction

In the works [5,16] there is presented a possibility of application of critical chain scheduling /buffer management methodology (CCS/BM) in investment enterprise and construction planning. The completion of a civil structure or a building complex is an undertaking consisting of the following factors: fulfilling the requirements (quality), cost, time of execution, range and resources [7]. In the article there are presented the outcomes of the tests concerning an improvement of methods used for investment scheduling and construction project with the implementation of memetic algorithm (i.e. hybrid evolutionary, HEA, [1]). The aim of the work was finding an optimal (for a taken goal function) level of workers' employment that is minimization of a divergence from the average level of employment with the implementation of CCS/BM methodology [5].

Scheduling of investment and construction project is connected to an optimization task. It is related to finding the best solution fulfilling the constraining conditions and taking into consideration the goal function. There are many known methods of optimization applied in specific cases.

Among others, there can be mentioned, for instance, for continuous tasks – methods of linear simplex, tasks of global optimization – when a goal function in the field of accepted solutions has more than one local minimum, discrete tasks with a greater complexity of calculations based most commonly on division or constraint methods, non – determined methods using random generating of solutions, a simulated annealing method as a modification of random walk with the improvement of quality of goal function, tabu search, that is with a list of revised variants and others.

There are also techniques applied with the use of biological systems – evolutionary algorithms, genetic, evolutionary strategies, evolutionary programming and genetic programming ([3],[18]). The general scheme of evolutionary algorithm's operation resides in creating a loop embracing reproduction, genetic operations, evaluation and succession. The classic scheme of operation of evolutionary algorithm is presented below according to [3].

This paper is a continuation of the topic presented in work [15], concerning the application of genetic algorithms [7] to steering of a level of an employment in investment and construction projects. Treating the evolutionary algorithm as a typical method of proceeding concerning

searching for better solutions, taking into consideration the closest environment, there was a slightly different approach proposed than the one applied in modeling of evolution with a application of a binary genetic code. The new one consists in replicating of randomly chosen variants (individuals) with a possibility of multiple copying of the same solution, in which the random choice takes into account a better adaptation of a variant (individual).

Solving practical issues of operational tasks, including scheduling of investment and construction project requires a selection of such an optimization method which leads to the best solution with the minimal cost. It is assumed [3], that one of the most efficient optimization algorithms is the evolutionary algorithm with strategy, constituting frameworks for an idea of a memetic algorithm HEA, [1].

2. Defining the problem

An optimization task refers to planning of regularity of workers' employment level [8,17,14,11,10] during realization of construction undertaking with regard towards CCM/BM methodology [5,16]. Time buffers introduced to scheduling task increase space of accepted variations of realizing the project and an extent of a task.

In order to solve an optimization task a hybrid evolution algorithm HEA created by Bożejko and Wodecki [1] was applied. The idea of a algorithm HEA consists of creation of a start-up population in which for every individual there is a permutation applied in order to find a local minimum. Then, there is a passing towards separate populations with defining a number and position in a set of individuals.

Theory of Constrains (TOC) by Goldratt [2,3,4,5,6] and its practical application in managing of projects known as Critical Chain Scheduling (CCS) and Buffer Management (BM), in short defined as CCS/BM Method [7], stays in the center of interest of many scientific groups. Precursors of this methods were Giffler and Thomson [8] together with Wiest [17] who introduced the concept of critical sequences determined not only by technological sequencing and adjusting the time of tasks' completion but also by constraints of resources in creation of schedules. Creating and steering of schedules is one of the aims of managing building projects. High level of simplification of the two systems: Critical Path Method (CPM) together with Program Evaluation and Review Technique (PERT) led to their popularizing and common use all over the world. Changing standards, growing expectations, cost cutting, minimizing fines and the extent of investment projects, clearness of expressions led to undertaking of many works aiming at increasing the effectiveness of scheduling treated in relation to traditional CPM/PERT methods.

2.1 Theory of constrains framework

Theory of Constraints (TOC) can be applied in all projects which aim at reaching revenue. TOC is based on five basic steps:

1. Identification system constraints.
2. Decision of maximum utilization of constrained resources.
3. Subordination of all processes to above decision.
4. Increasing number of constraints created as a result of a liquidation of a constraint defined in the first step as a bottleneck of the system.
5. Identification of new constraints of resources created as a result of bottleneck elimination in point 4; if constraint in point 4 is eliminated then return to step one in order not to allow any internal factors to constraint the whole system.

The proposed system aims at ongoing production improvement in order to gain bigger profit in current and future undertakings through identifying and eliminating bottlenecks in production. TOC can be used in realization of building projects. Let us assume that in the first step a tower crane is identified as a constrained resource. Thus, in the second step we take a decision of maximum utilization of crane's capacity shifting work schedule from 8 to 16 working hours [14]. In the third step, as a consequence of the taken decision, we must subordinate all teams and machines which cooperate with the crane. In the fourth step we analyze whether the resources pose a threat of becoming bottlenecks of the system. In case when internal factors constrain the whole system, we must return to step one and identify constraints one more time. While implementing a production improvement in accordance with TOC we can expect an increase of profits. The example of practical application of Theory of Constraints is a project management method – scheduling method using critical chain and buffer management CCS/BM.

2.2 CCS/BM methodology

Chronic problems appearing in realization of construction tasks which cannot be removed even with the use of advanced technologies became a reason for development of CCS/BM methods [11]. Classic scheduling methods used in construction projects CPM/PERT are characterized by utilization of time used for completing separate activities with regard to time limits resulting from valid, approximate, standard data. Goldratt perceives such reasoning as inappropriate and in his work [2] even calls this process 'a thief of time', whereas Turner [16] analyzes the influence of time reserve dispersion on revenue gained from undertaking. He assumes that a project consists of n tasks and each of them is afflicted with a contingency (randomness). Moreover, on the assumption that there is an even scheduling of time needed for task completion, it has a standard σ deviation. Thus, an overall contingent reserve equals $n\sigma$. The rights to manage it are dispersed. Every participant of the project can manage their time reserve. From the point of view of a project manager an optimal situation would be if he could manage the time himself for the following reasons: reserves not utilized by separate participants of activities would not get wasted and total time reserve could be smaller than in the first case. From the calculations presented by Turner it results that a total time reserve should equal $(n)^{1/2}\sigma$ which is a much smaller value than $n\sigma$.

Implementation of a method of time reserve aggregation results in a significant reduction of project costs. Moreover, lack of time reserve for a project manager often leads to a failure in meeting the deadlines. Goldratt recalls some known examples: a tunnel under British Channel or drilling towers in the North Sea. The size of time reserves and duration of separate activities (not taking into consideration its internal reserves) is one of the basic problems of CCS/BM. During developing of CCS/BM method Goldratt was basing on the following psychological assumptions:

- Student Syndrome – 'do not begin work before all possibilities and time limits are used';
- Parkinson's Law – 'every work will be done in a assigned time or longer';
- roadrunner mentality – 'real race against time', [7];
- Conkling's Roadunner – *Geococcyx californianus* is the quickest runner among its species reaching the speed of 30 km/h, it never moves slowly;
- Murphy's Law – 'anything that can go wrong, will go wrong'.

Goldratt proposes time reduction for a completion of an individual processes together with informing tasks executors only about due dates. A schedule including time buffers is accessible exclusively for a project manager. Participants, not possessing any time reserves, try to complete their tasks as quick as possible (roadrunner mentality), start their job with a full capacity (Student Syndrome), in case of threat of falling behind the deadline (Murphy's Law & Parkinson's Law) the project manager has time reserves to modify and steer the course of works.

Goldratt solves the problem of time reserves in an arbitrary manner. Namely, he introduces reduction of an activity duration by half (50%) and creates time buffers of a 50% value of a new, shortened activity duration. In reference to sub-critical chains Goldratt calls it a feeding buffer FB, whereas behind a critical chain he places a project buffer PB. According to the following assumptions:

- **Project buffer PB** is a time reserve placed at the end of critical chain, staying to project managers disposal, introduced in order to protect completion of a project, calculated on the basis of critical chain time duration (according to Goldratt a project buffer is 25% of critical chain duration), there is only one project buffer in scheduling.
- **Feeding Buffer FB** is a time reserve placed at the end of non-critical chain (feeding chain) staying to managers disposal in order to protect tasks placed in a critical chain, introduced as a protection of deadline of critical tasks, calculated on the basis of non-critical chain duration (in Goldratt's method a feeding buffer equals 25% of non-critical chain duration), there are as many feeding buffers as the number of non-critical chains (FB appears always when a non-critical chain links with a critical chain).
- **Resource buffer RB** is a time reserve placed in schedule before entering of a new resource into a critical chain, calculated on the basis of logistic dependences and possibilities, introduced in order to ensure initiation of a process in a critical chain, in a planned due date.
- **Critical chain CC** is a set of processes appearing in front of a project buffer, determined by: activities' duration time,, their technical sequence of completion, accessibility and resource requirements; processes in a critical chain are deprived of time reserves taken into consideration in CPM/PERT (according to Goldratt the length of a critical chain is calculated as 50% of critical path CP); in a critical chain cannot appear any processes not having a direct influence on due task time (owing to time, technical or resource constraints); in case of appearing of more than one critical path one must choose one of them and transform a critical chain (there can be only one critical chain).

Figure 1 presents a graph of activities 1-2-3-4-5 lasting adequately 16,4,8,8,4 time units. There are dependences between activities shown and their sequence has been marked in figure 1.

According to CCS/BM methodology and applying Goldratt's assumption concerning a duration of time of processes and length of feeding (FB) and project buffer (PB) we obtain a graph depicted in figure 2.

An individual assessment, based on system analysis of shortening separate processes, creation of critical chain and size of feeding and project buffers allows for creation of rational schedule including not only technical but also resources, organizational, financial constraints. Dependencies between process duration time, critical chain buffer and optimal utilization of resources with application of mementic algorithm (HEA) to calculations have been thoroughly examined. Expansion of implemented in financial calculations method of contingency onto area of scheduling of construction projects with application of TOC and CCS/BM methods accompanied by HEA might lead to shortening of time and costs of a

building process. Works concerning this problem will be continued in the future by the authors.

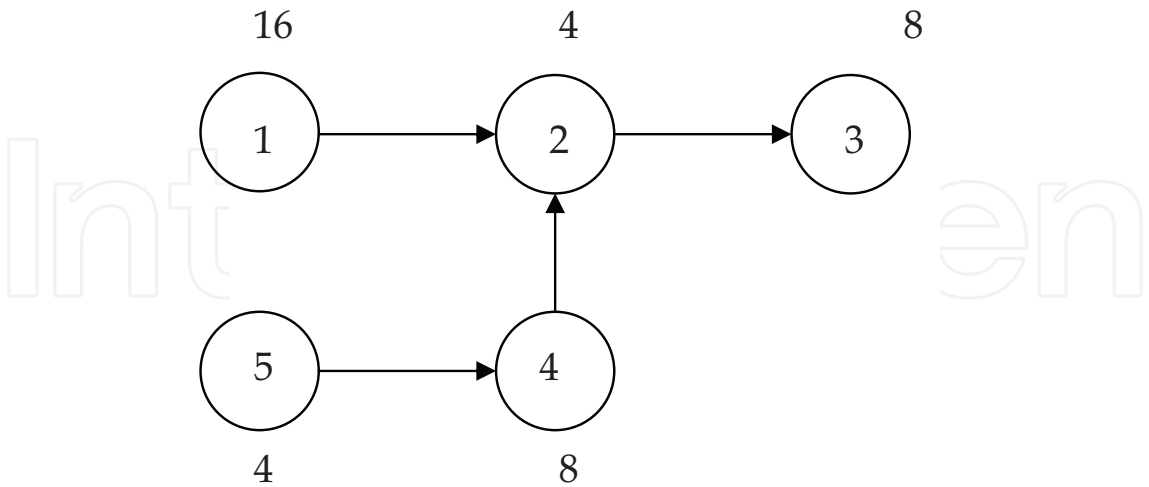


Fig. 1. Graph of activities and times of their duration in a classic formulation.

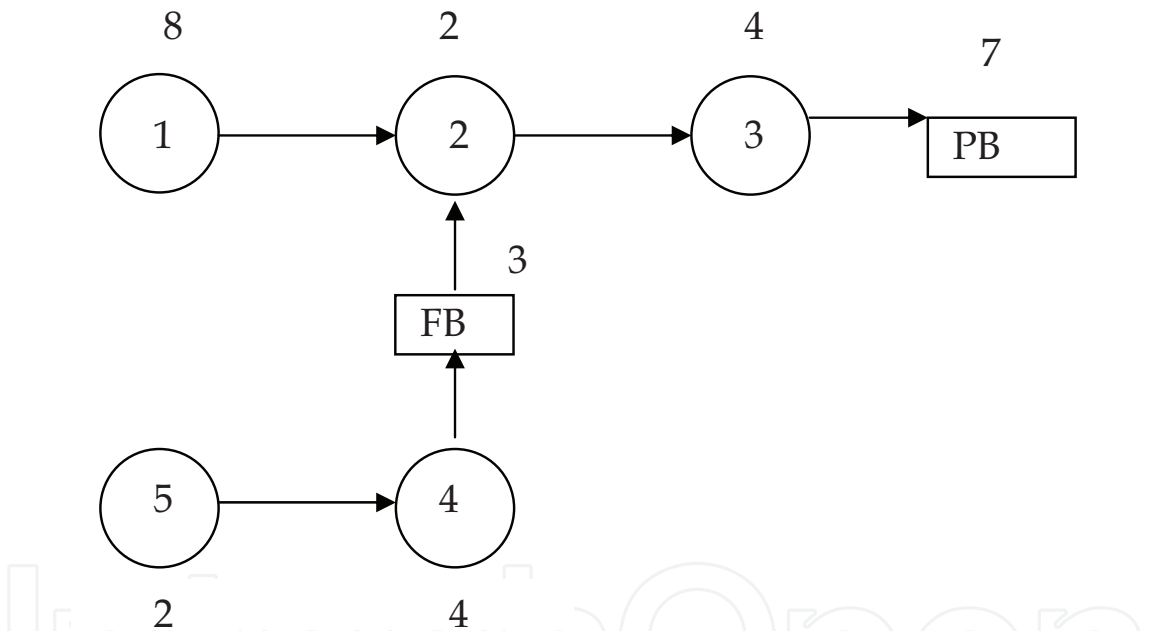


Fig. 2. Graph of activities and time of their duration according to CCS/BM including project (PB) and feeding buffers (FB).

3. Hybrid Evolutionary Algorithm

Hybrid evolutionary algorithm was proposed by Bożejko and Wodecki [1] and it is a general method of solving discrete optimization problems. Therefore some elements of the algorithm have to be addressed in detail to use it for solving automation problems in construction, especially the method of the problem's code for HEA, determining of set of fixed elements and local optimization approach. The algorithm starts by forming residual population P^0 (which can be randomly formed). The best element of population P^0 is adopted as suboptimum solution π^* . Let i be the algorithm iteration number. New population $i+1$ (i.e. set P^{i+1}) is generated as follows. For

current population P^i a set of local minima LM^i is fixed (by carrying out procedure $LocalOpt(\pi)$ for each element $\pi \in P^i$). Elements occupying the same positions in the local minima are fixed (procedure $FixeSet(LM^i, FS^i)$), forming a set of fixed elements and positions FS^{i+1} . Each permutation of new population P^{i+1} has fixed elements (in fixed positions) from set FS^{i+1} . Free elements are randomly assigned to the remaining (unoccupied) positions. If there is a permutation $\beta \in LM^i$ and $F(\beta) < F(\pi^*)$, then β is adopted as permutation π^* . The algorithm stops when it has generated a predetermined number of generations.

We apply following notation:

- π^* : sub-optimal solution determined by the algorithm,
- η : number of elements in population (the same in each generation),
- P^i : population in the iteration i of algorithm, $P^i = \{\pi_1, \pi_2, \dots, \pi_\eta\}$,
- $LocalOpt(\pi)$: local optimization algorithm to determining local minimum, where π is a starting solution of the algorithm,
- LM^i : a set of local minima in iteration i , $LM^i = \{\hat{\pi}_1, \hat{\pi}_2, \dots, \hat{\pi}_\eta\}$ where $\hat{\pi}_j = LocalOpt(\pi_j)$, $\pi_j \in P^i$, $j = 1, 2, \dots, \eta$.
- FS^i : a set of fixed elements and position in permutations of population P^i ,
- $FixSet(LM^i, FS^i)$: a procedure which determines a set of fixed elements and positions in next iteration of evolutionary algorithm,
- $NewPopulation(FS^i)$: a procedure which generates a new population in next iteration of the algorithm, $P^{i+1} = NewPopulation(FS^i)$.

The code of the proposed hybrid evolutionary algorithm is given below and figure 3.

Algorithm (HEA)

Initialization:

randomly formed population $P^0 = \{\pi_1, \pi_2, \dots, \pi_\eta\}$;

π^* = the best element of population P^0 ;

Iteration number $i=0$;

$FS^0 = \emptyset$;

repeat

Determine a set of local minima $LM^i = \{\hat{\pi}_1, \hat{\pi}_2, \dots, \hat{\pi}_\eta\}$, where

$\hat{\pi}_j = LocalOpt(\pi_j)$, $\pi_j \in P^i$;

for $j:=1$ **to** η **do**

if $F(\hat{\pi}_j) < F(\pi^*)$ **then** $\pi^* \leftarrow \hat{\pi}_j$;

$FS^{i+1} = FixSet(LM^i, FS^i)$; {fix set}

$P^{i+1} := NewPopulation(FS^i)$; {generate new population}

$i=i+1$;

not until Stop Criterion (exceeding time or a number of iterations).

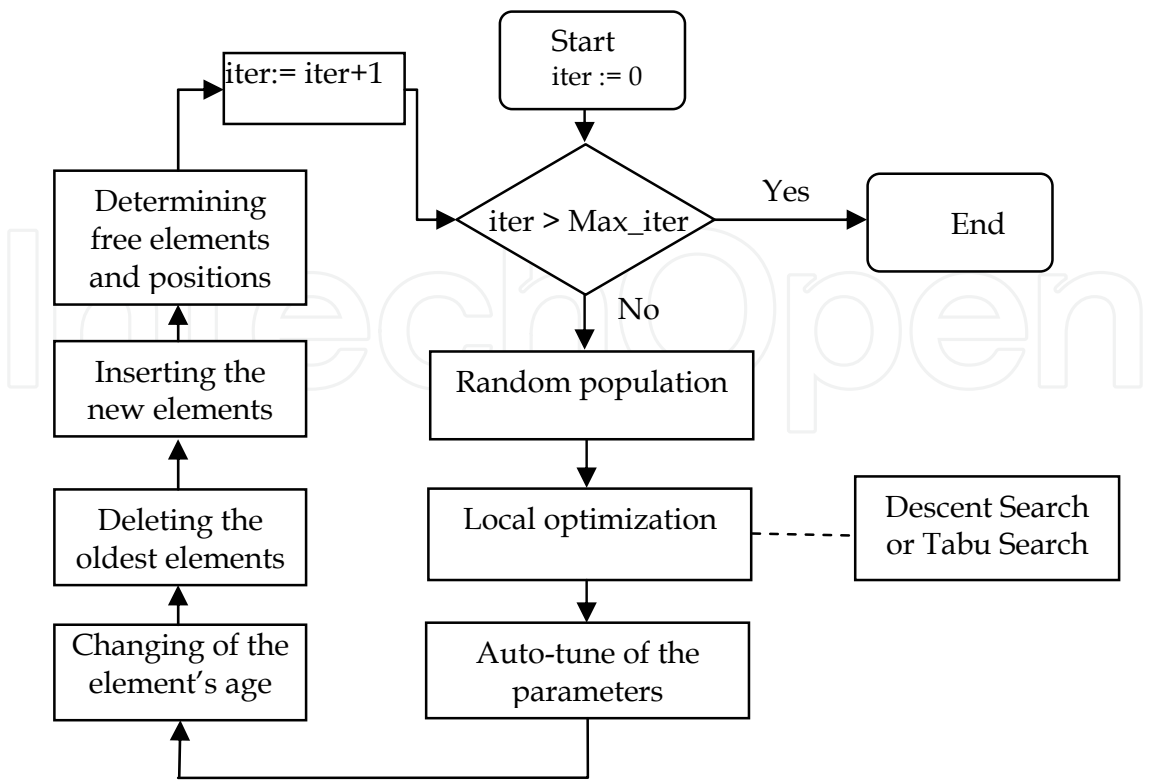


Fig. 3. Hybrid evolutionary algorithm.

4. Problem coding and notation

The problem can be defined as follows. There are: a set of n jobs $J=\{1,2,...,n\}$, a set of m machines $M=\{1,2,...,m\}$. Job $j\in J$, consists of a sequence of m operations $O_{j1}, O_{j2},..., O_{jm}$. Operation O_{jk} corresponds to the processing of job j on machine k during an uninterrupted processing time p_{jk} . We want to find a schedule such that the maximum completion times is minimal. Such a problem is known as a flow shop problem in literature. Let $\pi=(\pi(1), \pi(1),..., \pi(n))$ be a permutation of jobs $\{1,2,...,n\}$ and Π be the set of all permutations. Each permutation $\pi\in\Pi$ defines a processing order of jobs on each machine. Completion time of job $\pi(j)$ on machine k can be found using the recursive formula:

$$C_{\pi(j)k}=\max\{C_{\pi(j-1)k}, C_{\pi(j)k-1}\}+p_{\pi(j)k},$$

where $\pi(0)=0, C_{0k}=0, k=1,2,...,m$ and $C_{0j}=0, j=1,2,...,n$.

Local optimization (procedure LocalOpt)

The local search (LS) method is a metaheuristic approach designed to find a near-optimal solution of combinatorial optimization problems. The basic version of LS starts from an initial solution x^0 . The elementary step of the method performs, for a given solution x^i , a search through the neighborhood $N(x^i)$ of x^i . The neighborhood $N(x^i)$ is define by move performed from x^i . A move transforms a solution into another solution. The aim of this elementary search is to find in $N(x^i)$ a solution x^{i+1} with the lowest cost functions. Then the search repeats from the best found, as a new starting solution.

Local search algorithm

Select a starting point: x ;

$x_{best} := x$;

repeat

Select a point $y \in N(x)$ according to the given criterion
based on the value of the goal function $F(y)$;

$x := y$;

if $F(y) > F(x_{best})$ **then** $x_{best} := y$;

until some termination condition is satisfied.

A fundamental element of the algorithm, which has a crucial influence on quality and time of computation, is a neighborhood. A neighborhood is generated by the insert moves in the best local search algorithms with the permutation representation of the solution.

A set of fixed elements and position (procedure *FixSet*)

A set of fixed elements and positions FS^i (in i -th iteration of the algorithm) consists of quads (a, l, α, φ) , where a is an element of the set N ($a \in N$), l is a positions in a solution ($1 \leq l \leq n$) and α, φ are attributes of a pair (a, l) . Parameter α means „fitness“ and decides on belonging to the set, φ is an „age“ – element is removed from the set after exceeding some number of iterations (here: 3 iterations).

In every iteration of the algorithm, after determining the local minima (procedure *LocalOpt*), a new set $FS^{i+1} = FS^i$ is established. Next, a *FixSet*(LM^i, FS^i) procedure is called, in which there are executed the following operations :

- changing of the age of each element,
- deleting the oldest elements,
- inserting the new elements.

Inserting elements

Let $P^i = \{\pi_1, \pi_2, \dots, \pi_\eta\}$ be a population of η elements in iteration i . For each permutation π_j from P^i , applying the local search algorithm (*LocalOpt*(π_j) procedure), a set of local minima $LM^i = \{\hat{\pi}_1, \hat{\pi}_2, \dots, \hat{\pi}_\eta\}$ is determined. Each permutation

$$\hat{\pi}_j = (\hat{\pi}_j(1), \hat{\pi}_j(2), \dots, \hat{\pi}_j(n)), \quad j = 1, 2, \dots, \eta.$$

Let

$$nr(a, l) \geq \left| \{ \hat{\pi}_j \in LM^i : \hat{\pi}_j(l) = a \} \right|.$$

It is a number of permutations from the set LM^i , in which there is an element a in the position l . If $a \in N$ is a free element and

$$\alpha = \frac{nr(a, l)}{\eta} \geq \Phi(i)$$

then the element a is fixed in the position l .

A new population (procedure *NewPopul*)

To generate a new population P^{i+1} , randomly drawn free elements are inserted in remaining free positions of the elements of population P^i .

NewPopulation(FL^i)

$P^{i+1} \leftarrow \emptyset$;

Determine a set of free elements:

$FE = \{a \in N : \exists (a, l, \alpha, \varphi) \in FS^{i+1}\}$

and a set of free positions

$FP = \{l : \exists (a, l, \alpha, \varphi) \in FS^{i+1}\}$;

for $j:=1$ **to** η **do** {Inserting of fixed elements}

for each $(a, l, \alpha, \varphi) \in FS^{i+1}$ **do**

$\pi_j(l) := a$;

$W \leftarrow FE$;

for $s:=1$ **to** n **do** {Inserting of free elements}

if $s \notin FP$ **then**

$\pi_j(s) = w$, **where** $w = \text{random}(W)$ **and** $W \leftarrow W \setminus \{w\}$;

$P_{i+1} \leftarrow P_{i+1} \cup \{\pi_j\}$.

Function *random* generates from a uniform distribution an element of the set W . Computational complexity of the algorithm is $O(\eta n)$.

5. Case study

The subject matter of the analysis is a network model of an investment and construction project (ICP) according to [9,13], containing of $n = 16$ of linked building processes. There has been a computational test carried out in order to prove the possibility of time buffers influence on regularity of workers' employment in an enterprise schedule. With regard to taken constraints the values of a objective function F were adopted as follows:

$$f(x) = \sum_{j=1}^T \left(\left| q_j(x) - \frac{1}{T} \sum_{i=1}^n d_i r_i \right| \right) \quad (1)$$

where:

$x \in R^n$,

$x=(x_1, x_2, \dots, x_n)$ – vector of moments of initiating tasks' execution,

$x_i \in [a_i, b_i]$,

a_i – earliest moment of task beginning,

b_i – latest moment initiating tasks' execution,

$q_j(x)$ – number of workers employed on a j day, $j = 1, 2, \dots, T$,

T – time horizon,

d_i – time of process duration,

r_i – number of workers employed in order to carry out the process.

Below there is presented a network model of an exemplary enterprise taking into consideration time buffers; feeding FB [12] and project buffer PB. There is information concerning a number of activities, time of activities according to Goldratt method [4] and forecasted resources (number of employees).

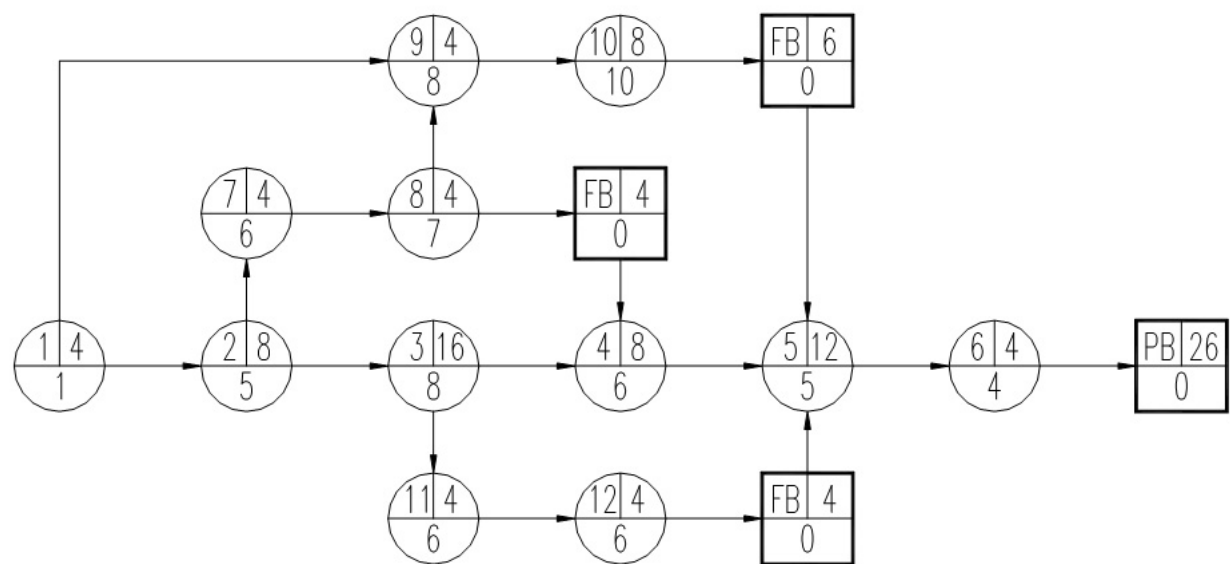


Fig. 4. The graph of ICP

Process number	Duration time	Earliest term of initializing	Earliest term of finishing	Latest term of begginning	Latest term of finishing	Number of workers
1	4	0	4	0	4	2
2	8	4	12	4	12	5
3	16	12	28	12	28	8
4	8	28	36	28	36	6
5	12	36	48	36	48	5
6	4	48	52	48	52	4
PB	26	52	78	52	78	0
7	4	12	16	16	20	6
8	4	16	20	20	24	7
FB	4	20	24	24	28	0
9	4	12	16	18	22	8
10	8	16	24	22	30	10
FB	6	24	30	30	36	0
11	4	28	32	40	44	6
12	4	32	36	44	48	6
FB	4	36	40	48	52	0

Table 1. The data of activities

Table 1 includes basic information concerning a construction project modeled by network. Basic time parameters forecasted to carry out construction works were counted. Numerical data necessary in order to carry out optimization calculations are presented below:

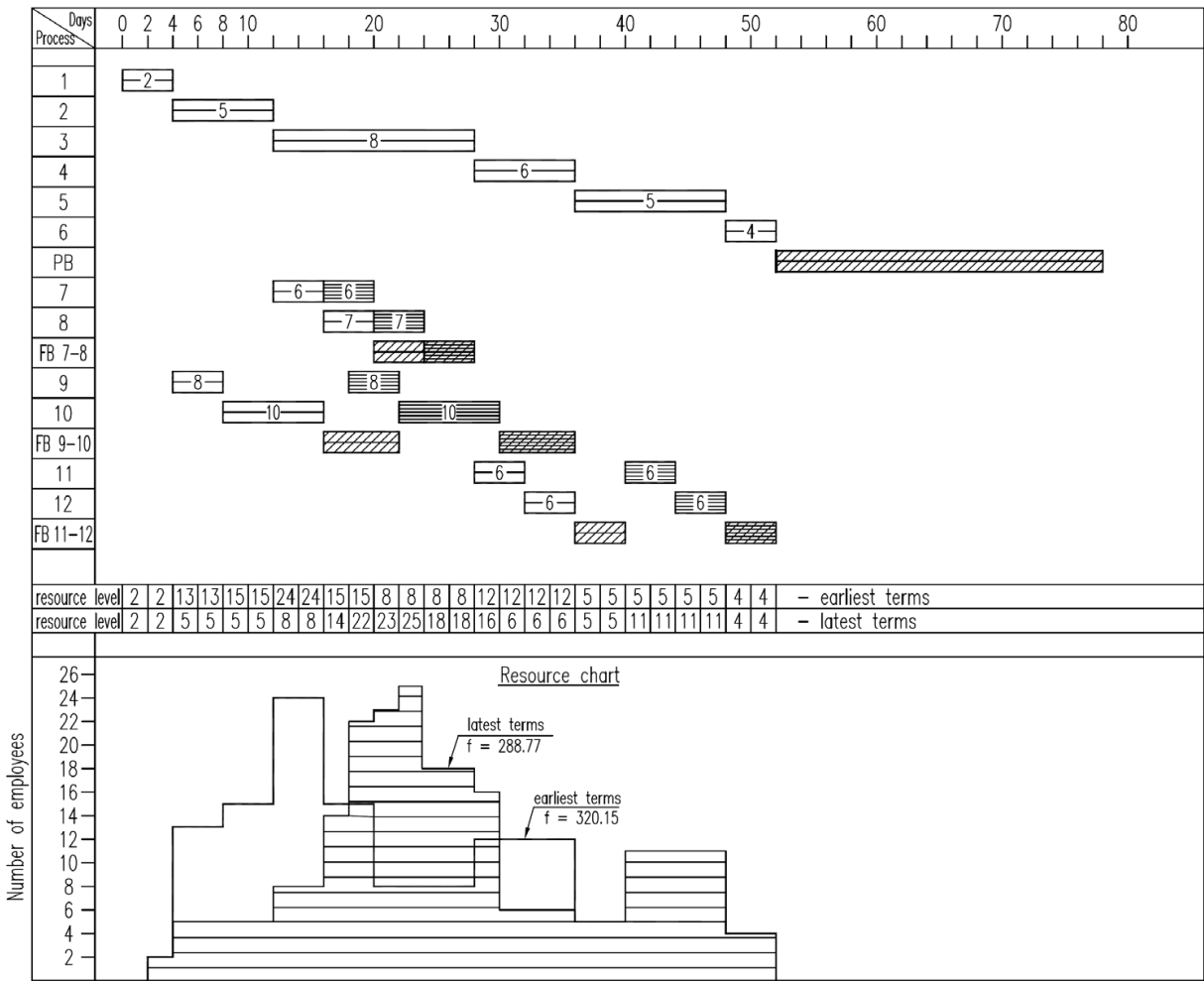


Fig. 5. The Gantt Chart, value of objective function for earliest terms $f=320.15$ for latest terms $f=288.77$.

Figure 5 depicts The Gantt Chart together with a graph of workers' employment for earliest and latest terms of works' beginning. For these extreme terms the values of goal function were calculated, i.e. $f=320.15$ and $f=288.77$ adequately.

Figure 6 shows Gantt's linear graph presenting an optimal schedule of a construction enterprise, for a taken goal function. Placement of a critical path does not change, whereas non-critical activities were put on a time scale taking into account time buffers (FB), in an optimal way with regard to a minimal value of a goal function.

Below Gantt's linear graph there is a graph of workers' employment corresponding to planned tasks. It shows the best solution concerning a minimal average divergence from an

average level of workers’ employment. A striped line shows time buffers which are in fact a time reserve enabling its utilization for an optimal placement of tasks on a time axis of construction works, so that the possible level of workers’ employment can be kept regular.

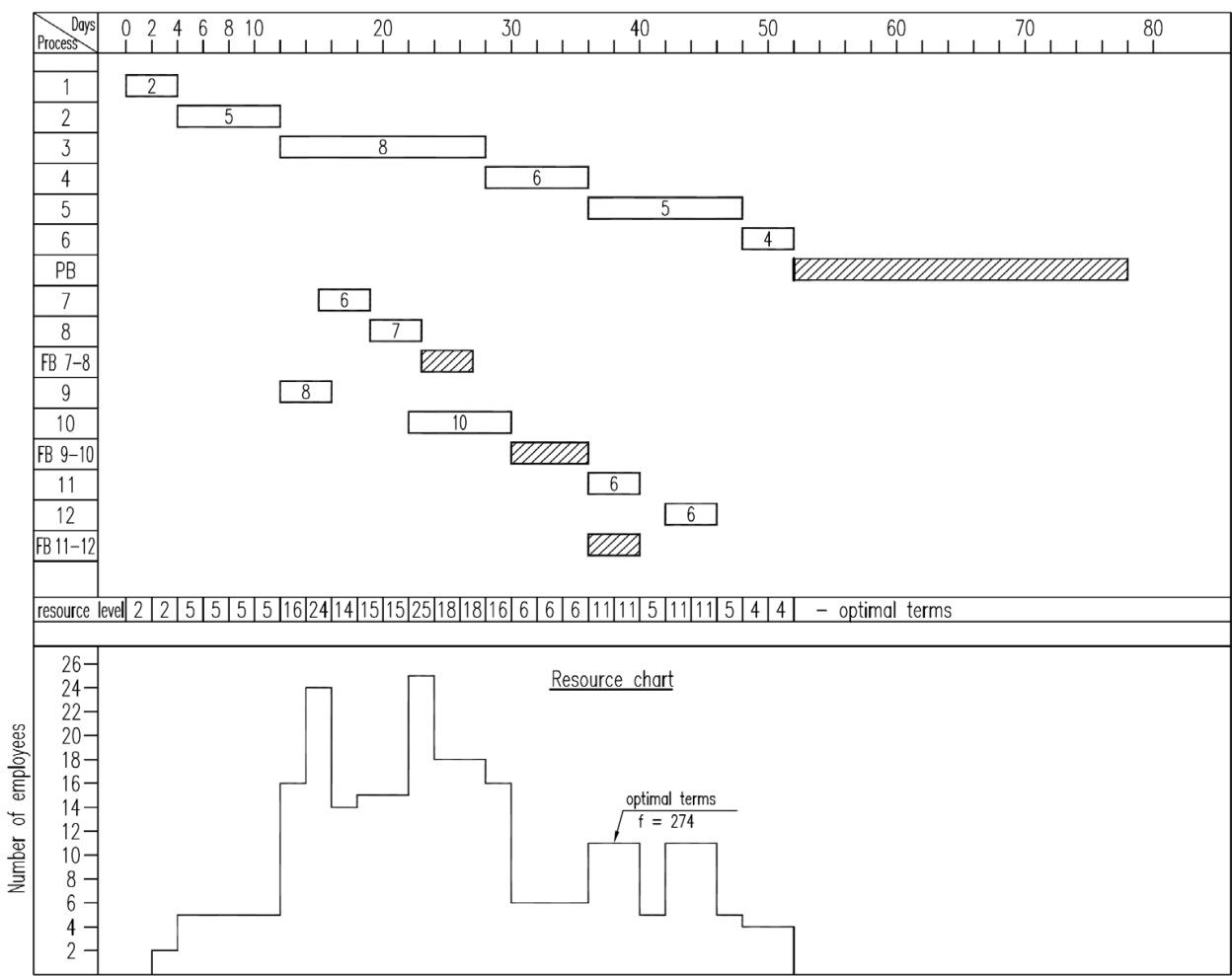


Fig. 6. The Gantt Chart after optimization, minimal value of objective function $f=274.00$. In an analyzed example there has been a genetic algorithm (GA) applied ensuring after 1000000th iteration the result of $f= 274.00$, whereas after application of hybrid evolutionary algorithm (HEA) the result of $f=274.00$ just after 100 iterations. The calculations were carried out on a Pentium IV computer with a clock of 3 GHz. The time of calculations in the first case was 2 seconds whereas in the second – below 1 millisecond, i. e. over 2000 times faster.

6. Summary

The calculations have been based on an elaborated optimization programs with application of a genetic algorithm GA a hybrid evolutionary algorithm. For a presented optimization task of $n = 16$ size, achieved values of a goal function equal: in case of application of a

genetic algorithm (GA), $f_{mi}=274.00$ whereas for a hybrid evolutionary algorithm (HEA) also $f_{min}=274.00$. While analyzing the results in case a genetic algorithm (GA) from an example [13], hybrid evolutionary algorithm (HEA) and after introduction of time buffers according to CCS/BM methodology one can state that application of time buffers of a zero load of resources (team workers) increases the extend of a optimization task and ensures the smallest value of a total divergence from an average level of employment.

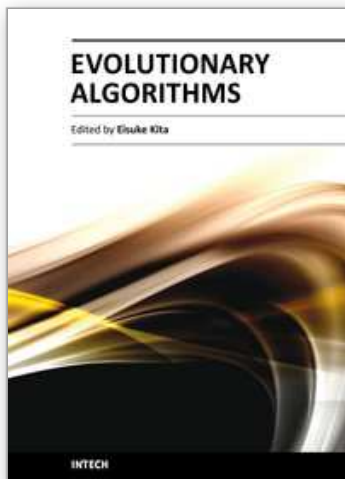
7. References

- [1] Bożejko W., Wodecki M., A hybrid evolutionary algorithm for some discrete optimization problems, IEEE Computer Society, 2005, 326-331.
- [2] Fox R., Goldratt E.M., The Race. [Croton-on-NY]: North River Press, 1986.
- [3] Goldratt E.M., The Goal. [Great Barrington, MA]: North River Press, 1 st ed. 1984, 2nd revised ed, 1992.
- [4] Goldratt E.M., Critical chain. [Great Barrington, MA]: North River Press, 1997.
- [5] Goldratt E.M., The Haystack Syndrome: sifting information out of the data ocean, New York, North River Press, 1990.
- [6] Goldratt E.M., It is no luck. [Great Barrington, MA]: North River Press, 1994.
- [7] Herroelen W., Leus R., On the merits and pitfalls of critical chain scheduling. Journal of Operations Management, 19, 2001, 559-577.
- [8] Giffler, B., Thompson, G.L., Algorithms for solving production-scheduling problems. Operations Research, 8, 1960, 487-503.
- [9] Leu S.S., Yang C.H., Huang J.C., Resource leveling in construction by genetic algorithm-based optimization and its decision support system application. Automation in Construction, 10, 2000, 27-41.
- [10] Radovilsky Z.D., A quantitative approach to estimate the size of the time buffer in the theory of constraints. International Journal of Production Economics 55, 1998, 113-119.
- [11] Rand G.K., Critical Chain: theory of constraints applied to project management. International Journal of Project Management, 18, 2000, 173-177.
- [12] Rogalska M., Hejducki Z., Shortening the realisation time of building project with application of theory of constraints and critical chain scheduling. Journal of Civil Engineering and Management, vol. X, Suppl 2, 2004, 99-105.
- [13] Rogalska M., Bożejko W., Hejducki Z., Employment level control using genetic algorithms (in Polish). 51st KILiW PAN and KN PZITB Scientific Conference Gdańsk-Krynica, 2005, 185-192.
- [14] Scavino N.J., Effect of Multiple Calendars on Total Float and Critical Path. Cost Engineering, vol.45, No.6, 2003.
- [15] Steyn, H., Project management applications of the theory of constraints beyond critical chain scheduling. International Journal of Project Management, 20, 2002, 75-80.
- [16] Turner JR., Controlling progress with planned cost or budgeted cost. International Journal of Project Management, 18(3), 2000, 153-4.
- [17] Wiest, J.D., Some properties of schedules for large projects with limited resources. Operations Research, 12, 1964, 395-418.

- [18] Yang J.B., Applying the theory of constraints to construction scheduling. Proceedings of Second International Structural Engineering and Construction Conference (ISEC 02), Vol. 1, 2003, 175-180.

IntechOpen

IntechOpen



Evolutionary Algorithms

Edited by Prof. Etsuke Kita

ISBN 978-953-307-171-8

Hard cover, 584 pages

Publisher InTech

Published online 26, April, 2011

Published in print edition April, 2011

Evolutionary algorithms are successively applied to wide optimization problems in the engineering, marketing, operations research, and social science, such as include scheduling, genetics, material selection, structural design and so on. Apart from mathematical optimization problems, evolutionary algorithms have also been used as an experimental framework within biological evolution and natural selection in the field of artificial life.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Wojciech Bożejko, Zdzisław Hejducki, Magdalena Rogalska and Mieczysław Wodecki (2011). Scheduling of Construction Projects with a Hybrid Evolutionary Algorithm's Application, Evolutionary Algorithms, Prof. Etsuke Kita (Ed.), ISBN: 978-953-307-171-8, InTech, Available from: <http://www.intechopen.com/books/evolutionary-algorithms/scheduling-of-construction-projects-with-a-hybrid-evolutionary-algorithm-s-application>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2011 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen