OPERATIONS RESEARCH AND DECISIONS

No. 1 2011

Marcin ANHOLCER*
Helena GASPARS-WIELOCH*

EFFICIENCY ANALYSIS OF THE KAUFMANN AND DESBAZEILLE ALGORITHM FOR THE DEADLINE PROBLEM

Time-cost tradeoff analysis is a very important issue in project management. The Kaufmann–Desbazeille algorithm is considered by numerous authors to be an exact algorithm to solve this problem. In the paper, we prove that this claim is not true. In particular, we perform a worst-case analysis. The accuracy of the KDA is the worst when: the network has many critical and subcritical paths with a lot of common arcs (i), shortening costs are constant (ii), the level of shortening costs for a given activity depends on its type.

Keywords: *time-cost trade-off analysis, network, critical path, accuracy of an algorithm, project compression time, time-cost curves*

1. Introduction

The Kaufmann and Desbazeille algorithm is a procedure used in time-cost trade-off project analysis (TCTP). We assume that the structure of each project may be represented by a network. The best known network presentation techniques are called AON (activities-on-nodes) and AOA (activities-on-arcs). In the first approach, nodes (vertices) show activities (tasks) and arcs – the precedence relationship. In the second one, arcs indicate tasks and nodes – events, i.e. the beginning and/or the end of an activity. In the TCTP problems a trade-off occurs between the project completion time (T) and the amount of non-renewable resources, i.e. money, which constitute the total cost of the project (C).

*Department of Operations Research, Poznań University of Economics, al. Niepodległości 10. E-mail addresses: m.anholcer@ue.poznan.pl, helena.gaspars@ue.poznan.pl

The total cost (C) includes direct (DC) and indirect costs (IC). The first category concerns costs related directly to the completion of activities (e.g. labor, raw materials) and accelerating (compressing) them. The durations of activities are bounded from below (crash duration) and from above (normal duration). The shortening of a given task requires the allocation of more resources. Indirect costs are assigned to the project as a whole. They include, among other things, taxes, insurance, management and penalty costs. The project manager bears penalty costs when the project completion time is delayed. When the project completion time increases, indirect costs increase as well, but direct costs decrease (Fig. 1).

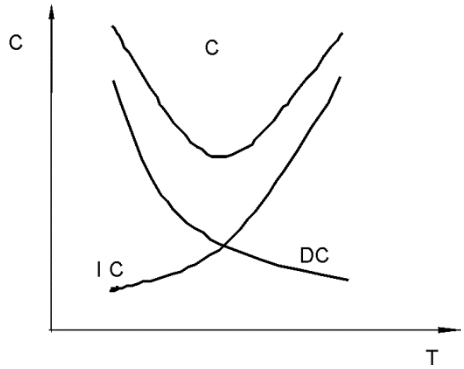

Fig. 1. Time-cost curves

The main goals considered in TCTP analysis are as follows ([37], p. 910):

1) minimizing time-dependent project costs within a specified project deadline or target time T^d (the deadline problem)

$$C(X) \rightarrow \min \tag{1}$$

$$T(X) \leq T^d \tag{2}$$

2) minimizing project completion time within a specified budget C^b (the budget problem)

$$T(X) \rightarrow \min \tag{3}$$

$$C(X) \leq C^b \tag{4}$$

where X signifies the vector of the durations of activities.

The Kaufmann and Desbazeille algorithm (KDA) is one of the oldest procedures applied in TCTP analysis. Helena Gaspars-Wieloch presented a detailed survey and analysis of other optimization methods applicable to time-cost analysis for projects in the thesis [15] and paper [16]. Here we just show a comparative breakdown of these algorithms, which takes into consideration their idea, possible applications and accuracy, see Table 1 (t_i^c – the shortest possible time for the *i*-th activity i.e. crash time, t_i^n – the normal time for the *i*-th activity).

Table 1. Comparative breakdown of existing project time-cost trade-off algorithms

Criterion		Authors of procedures
1. Type of model used	primal model	1) Berman, 2) Bladowski, 3) Crowston and Thompson, 4) De, Dunne, Gosh, and Wells, 5) Falk and Horowitz, 6) Fondhal, 7) Goyal, 8) Hindelang and Muth, 9) Kaufmann and Desbazeille, 10) Liu, Burns and Feng, 11) Moder and Phillips, 12) Moussourakis and Haksever, 13) Panagiotakopoulos, 14) Siemens
	dual model	1) Fulkerson, 2) Gedymin, 3) Kelley, 4) Phillips and Dessouky, 5) Prager
2. Project presentation technique	network AOA	1) Berman, 2) Bladowski, 3) Falk and Horowitz, 4) Fulkerson, 5) Gedymin, 6) Goyal, 7) Kaufmann and Desbazeille, 8) Kelley, 9) Moder and Phillips, 10) Panagiotakopoulos,11) Phillips and Dessouky, 12) Siemens
	network AON	1) Fondhal, 2) Crowston and Thompson (DCPM networks), 3) De, Dunne, Gosh and Wells, 4) Hindelang and Muth (DCPM networks), 5) Liu, Burns and Feng, 6) Moussourakis and Haksever
	Gantt chart	1) Prager
3. Part of the network considered	the whole network	1) Berman, 2) De, Dunne, Gosh and Wells, 3) Fondhal, 4) Fulkerson, 5) Gedymin, 6) Hindelang and Muth, 7) Kelley, 8) Phillips and Dessouky, 9) Prager
	critical subnetwork	1) Bladowski, 2) Crowston and Thompson, 3) Kaufmann and Desbazeille
	subcritical subnetwork	1) Moder and Phillips
	inadmissible subnetwork*	1) Goyal, 2) Siemens
	activities unlabelled	1) Falk and Horowitz, 2) Panagiotakopoulos
4. Potential set of activities to shorten in a given iteration	cut established for the whole network	1) Fulkerson, 2) Gedymin, 3) Kelley, 4) Phillips and Dessouky, 5) Prager
	cut established for the critical subnetwork	1) Bladowski, 2) Kaufmann and Desbazeille
	single activity	1) Crowston and Thompson, 2) Goyal, 3) Siemens

*An inadmissible subnetwork contains paths whose duration is longer than the deadline time.

Table 1 continued

<table>
<tr><th colspan="2">Criterion</th><th colspan="2">Authors of procedures</th></tr>
<tr><td rowspan="3">5. Possibility of prolonging an activity</td><td>no</td><td colspan="2">1) Bladowski, 2) Kaufmann and Desbazeille</td></tr>
<tr><td rowspan="2">yes</td><td>in a given iteration</td><td>1) Fulkerson, 2) Gedymin, 3) Goyal, 4) Kelley, 5) Moder and Phillips, 6) Phillips and Dessouky, 7) Prager</td></tr>
<tr><td>at the end of the procedure</td><td>1) Berman, 2) Crowston and Thompson, 3) Fondhal, 4) Hindelang and Muth, 5) Panagiotakopoulos, 6) Siemens</td></tr>
<tr><td rowspan="2">6. Intermediate solutions constituting solutions for other T^d values</td><td>yes</td><td colspan="2">1) Bladowski, 2) Fulkerson, 3) Gedymin, 4) Kaufmann and Desbazeille, 5) Kelley, 6) Moder and Phillips, 7) Phillips and Dessouky, 8) Prager</td></tr>
<tr><td>no</td><td colspan="2">1) Berman, 2) Falk and Horowitz, 3) Fondhal, 4) Goyal, 5) Panagiotakopoulos, 6) Siemens</td></tr>
<tr><td rowspan="3">7. Number of critical paths during compression</td><td>non-decreasing</td><td colspan="2">1) Bladowski, 2) Fulkerson, 3) Gedymin, 4) Kaufmann and Desbazeille, 5) Kelley, 6) Phillips and Dessouky, 7) Prager</td></tr>
<tr><td>non-increasing</td><td colspan="2">–</td></tr>
<tr><td>various</td><td colspan="2">1) Falk and Horowitz, 2) Fondhal, 3) Moder and Phillips, 4) Panagiotakopoulos, 5) Siemens</td></tr>
<tr><td rowspan="3">8. Type of optimization problems solved</td><td>$C(X) \to \min$, $T(X) \le T^d$</td><td colspan="2">1) Berman, 2) Bladowski, 3) De, Dunne, Gosh and Wells, 4) Falk and Horowitz, 5) Fondhal, 6) Fulkerson, 7) Gedymin, 8) Goyal, 9) Hindelang and Muth, 10) Kaufmann and Desbazeille, 11) Kelley, 12) Liu, Burns and Feng, 13) Moder and Phillips, 14) Moussourakis and Haksever, 15) Panagiotakopoulos, 16) Phillips and Dessouky, 17) Prager, 18) Siemens</td></tr>
<tr><td>$T(X) \to \min$, $C(X) \le C^d$</td><td colspan="2">1) Bladowski, 2) De, Dunne, Gosh and Wells, 3) Fulkerson, 4) Gedymin, 5) Kaufmann and Desbazeille, 6) Kelley, 7) Liu, Burns and Feng, 8) Moder and Phillips, 9) Moussourakis and Haksever, 10) Phillips and Dessouky, 11) Prager</td></tr>
<tr><td>$C(T^{ad}) \to \min$*</td><td colspan="2">1) Bladowski, 2) Crowston and Thompson, 3) De, Dunne, Gosh and Wells, 4) Fulkerson, 5) Gedymin, 6) Hindelang and Muth, 7) Kaufmann and Desbazeille, 8) Kelley, 9) Liu Burns and Feng, 10) Moder and Phillips, 11) Moussourakis and Haksever, 12) Phillips and Dessouky, 13) Prager</td></tr>
</table>

* T^{ad} denotes an advisable but not mandatory project completion time. When the advisable time is exceeded, then the total project cost must include a penalty. When the project is finished before the advisable time, the total cost is reduced by a bonus.

Table 1 continued

Criterion			Authors of procedures
9. Type of costs considered	shortening costs		1) Bladowski, 2) Fulkerson, 3) Gedymin, 4) Goyal, 5) Kaufmann and Desbazeille, 6) Kelley, 7) Panagiotakopoulos, 8) Phillips and Dessouky, 9) Prager 10) Siemens
	shortening costs and other direct project costs		1) Berman 2) Bladowski 3) De, Dunne, Gosh and Wells 4) Falk and Horowitz 5) Fondhal 6) Kaufmann and Desbazeille 7) Liu, Burns and Feng 8) Moussourakis and Haksever
	all costs (direct and indirect)		1) Crowston and Thompson, 2) Hindelang and Muth, 3) Moder and Phillips
10. Growth speed of direct costs	constant (linear time-cost curve)		1) Fulkerson, 2) Goyal, 3) Kelley, 4) Phillips and Dessouky 5) Prager, 6) Siemens
	increasing (convex curve)	linear approximation	1) Kelley, 2) Goyal, 3) Liu, Burns and Feng, 4) Phillips and Dessouky, 5) Prager 6) Siemens
		no approximation	1) Berman
	decreasing (concave curve)	linear approximation	1) Falk and Horowitz 2) Gedymin
		no approximation	–
	various (ex. concave -convex curve)	linear approximation	1) Falk and Horowitz, 2) Gedymin, 3) Moussourakis and Haksever
		no approximation	1) Bladowski, 2) Crowston and Thompson, 3) De, Dunne, Gosh and Wells, 4) Fondhal, 5) Hindelang and Muth, 6) Kaufmann and Desbazeille, 7) Moder and Phillips, 8) Panagiotakopoulos
11. Feasible activity durations	any real number from $\langle t^c_i, t^n_i \rangle$		1) Berman, 2) Falk and Horowitz, 3) Fondhal, 4) Fulkerson, 5) Gedymin, 6) Goyal, 7) Kelley, 8) Phillips and Dessouky, 9) Prager, 10) Siemens
	any integer from $\langle t^c_i, t^n_i \rangle$		1) Bladowski, 2) Kaufmann and Desbazeille
	discrete options from $\langle t^c_i, t^n_i \rangle$		1) Crowston and Thompson, 2) De, Dunne, Gosh and Wells 3) Hindelang and Muth, 4) Liu, Burns and Feng, 5) Moder and Phillips, 6) Moussourakis and Haksever, 7) Panagiotakopoulos
12. Accuracy of solutions obtained	optimum solutions		1) Berman 2) De, Dunne, Gosh and Wells, 3) Falk and Horowitz, 4) Fulkerson,5) Gedymin, 6) Kelley, 7) Liu, Burns and Feng, 8) Moussourakis and Haksever 9) Phillips and Dessouky 10) Prager
	optimum or suboptimum solutions		1) Bladowski 2) Crowston and Thompson 3) Fondhal 4) Goyal 5) Hindelang and Muth, 6) Kaufmann and Desbazeille, 7) Moder and Phillips, 8) Panagiotakopoulos, 9) Siemens

Source: [15] and [16]

Exact algorithms, i.e. those which guarantee finding an optimum solution, to discrete problems (DTCTP – discrete time-cost trade-off problems) are characterized by exponential complexity and are NP-hard. Procedures designed for continuous cases are solvable in polynomial time (see [5], [6], ([16], p. 22), ([31], p. 308), ([36], p. 398), [37]).

As one can notice from the table, KDA enables solution of the problems (1)–(2) or (3)–(4), see crit. 8. This procedure focuses on the minimization of direct costs (crit. 9). Theoretically, the algorithm may be used in time-cost trade-off problems with any types of time-cost curves for project activities (crit. 10)*. Nevertheless, in practice it is applied only when the unit shortening cost is constant or non-decreasing, because in other cases the solutions obtained are extremely bad (i.e. far from the optimum ones). Most of the existing exact algorithms for TCTP are designed just for linear or convex time-cost curves.

Kaufmann and Desbazeille assume that the parameters representing the normal activity durations (t_i^n), the project target time (T^d) and the difference ($t_i^n - t_i^c$) are integers. They take into consideration only integer realizations of the project time, even if for a given C^b (see the budget problem (3)–(4)) the optimum solution (i.e. with the shortest time within a specified budget) requires non-integer times for some tasks (compare crit. 11).

The original version of the KDA consists in iteratively shortening each critical path in the network by exactly one time unit. The target is to find the cheapest way of compressing the project time by one unit. We stop when constraints (1)–(2) or (3)–(4) are satisfied. In order to apply this procedure, the user should know:

- the structure of the project network,
- the normal and the crash duration of the activities within the project,
- the unit shortening cost for each task,
- the deadline or budgetary constraint.

The above version of the method in question is well known by project management trainers, academic teachers and students. The algorithm is presented, among other places, in [2], [13], [14], [19], [21], [25], [29], [30], [36] and [38]. Authors use different names for this method (e.g. CPM-COST analysis, time-cost analysis, network compression algorithm, MCX – minimum cost expediting). That is why Kaufmann and Desbazeille are little-known surnames in TCTP analysis. Nevertheless, their book, published in 1964 [22], is the oldest one containing a description of the algorithm considered. Therefore, the procedure analyzed in this paper is conventionally called the Kaufmann and Desbazeille algorithm (for convenience we also use the abbreviation KDA).

*For instance, the time-cost curves may be concave. This case occurs, among other things, when the compression of the time for a given activity by its first unit requires the application of a totally new technology. When the technology is changed, further compression requires only additional resources within the same technology.

Many people claim that this method is an exact algorithm, i.e. an algorithm which always indicates the cheapest way of project compression. The KDA is however only a heuristic procedure (see Table 1, crit. 12). Examples of problems for which the method gives quasi-optimum solutions are presented in [13] and [14].

Kaufmann and Desbazeille were only interested in integer realizations of the project time. Therefore, empirical research will just be related to the deadline problem (see problem 1–2), as the discrete solutions obtained simultaneously constitute solutions for different values of the budgetary parameter in the budget problem. Solutions generated for one of the chosen problems (deadline or budget) suffices to establish the whole time-cost project curve *DC* (see Fig. 1).

Gaspars-Wieloch [13], [14] came to the conclusion that the main (but not the only!) factor affecting accuracy is the way in which variants for compressing the completion time are found. In the original version of the KDA, the best set of activities to shorten is chosen from a list of sets containing exactly one activity on each critical path. This approach exposes us to overlook the cheapest way of accelerating the project by one time unit.

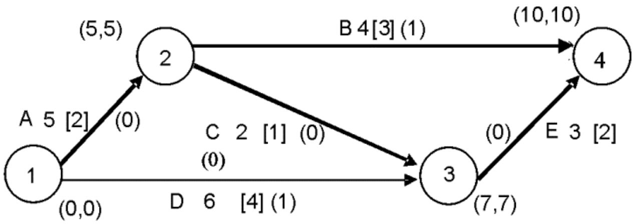

Fig. 2. Structure of a simple project. Source ([14], p. 17)

Figure 2 illustrates the structure of a simple project. This structure is presented according to the AOA technique. There are 5 activities (A, B, C, D and E) and four events (1–4). Normal completion times are 5, 4, 2, 6 and 3 days, respectively. Crash durations are not given in the figure. Assume that they are equal to 2, 1, 1, 3 and 1 day, respectively. The numbers in round brackets next to the nodes indicate the earliest and the latest times of events and the numbers in round brackets situated close to the arcs show the slack time for each activity (i.e. the so-called total float). The unit shortening cost is given in square brackets. There is only one critical path (A–C–E). Suppose that the target time is 8 days. Therefore, it is necessary to compress the project by 2 days.

According to the original version of the KDA, it is recommended to compress activity C in the first iteration, because shortening activity A or E entails higher costs. In the second step the project completion time is equal to 9 days and the whole network is critical. There are 3 potential sets containing exactly one activity on each critical path: I. A + D, II. B + C + D, III. B + E. The total compression cost for these sets amounts to:

$$2 + 4 = 6, \qquad 3 + \infty + 4 = \infty, \qquad 3 + 2 = 5.$$

In the second iteration, the best solution consists in accelerating activities B and E. Shortening the project from 10 to 8 days will cost 1 + (3 + 2) = 6.

In [13] and [14], it has been emphasized that the easiest way of improving the results, i.e. reducing the compression costs, is to find the cheapest solution among sets containing at least (but not exactly) one activity on each critical path, but this is not sufficient to find the optimum solution.

If we apply this guideline to the KDA, first we shorten activity C by one day and then we will choose the set A + E which costs only 2 + 2 = 4 and is cheaper than the sets considered before (i.e. A + D, B + C + D, B + E). Accelerating activities A and E means shortening paths A–B and D–E by one day and shortening the path A–C–E by two days. This set has not been studied before, because it contains as many as two tasks belonging to the path A–C–E. Nevertheless, it is much more profitable. Thus, the total compression cost may amount to 1 + (2 + 2) = 5. Thanks to this modification of the algorithm, it was possible to reduce the cost by 1. Notice that the cost of 5 is not the lowest possible cost – the optimum solution consists in shortening just tasks A and E (without C) and costs only 4!

The modification suggested in papers [13] and [14] is so natural and obvious that in the next section we will analyze the efficiency of this modified version of the KDA.

2. The project network – notation

In all of this paper we use the AOA version of the project network, i.e. activities are represented by arcs of the network, while vertices represent events. To be more exact, the set E of arcs consists of m arcs $e_1, \ldots, e_m$, while the set of nodes V consists of n nodes $v_1, \ldots, v_n$. Each arc e_i, $i = 1, \ldots, m$ is labelled by some positive natural number t_i, i.e. the duration of the respective activity. For convenience, we often index the arc by e_{jk}, where $j = 1, \ldots, n - 1$ and $k = 2, \ldots, n$ are the indices of the starting and end node of the arc, respectively.

In addition, for each arc we define a non-decreasing sequence $C^{(i)} = \left(c_{k_i}^{(i)}\right)_{k_i=1}^{K_i}$ of real numbers representing the shortening cost, where $c_{k_i}^{(i)}$ is the cost of reducing the duration of the i-th activity by the k_i-th unit and $K_i \le t_i$ (one may shorten an activity at most $(t_i^n - t_i^c)$ times).

The earliest time of the event j (the earliest time at which node j can be reached such that all its preceding activities have been finished) is denoted by t_j^I (t_n^I being equal to the minimum completion time of the project, i.e. T^*). The latest time of the event j (the latest time that node j can be left such that it is still possible to finish the overall project in the minimum completion time) is denoted by t_j^{II}. Finally, the slack time of activity e_{jk} (also called the total time reserve or the float time) is denoted by f_{jk}.

3. The Kaufmann and Desbazeille algorithm – description and implementation

Let us recall the details of the Kaufmann and Desbazeille algorithm for the deadline problem in TCTP. Assume that we are given the project network as defined in the previous section and the desired completion time (the deadline) T^d. The goal is to reach the desired time in the cheapest way possible. To achieve this, one has to perform two steps in every iteration: one is to implement the CPM method for the actual durations of activities and the other is the shortening of the project by one time unit. The algorithm stops when the desired time has been reached.

To compress the project duration, it is necessary to shorten each critical path by one time unit. It is not necessary to shorten one activity from each critical path as the critical paths do not have to be disjoint. In order to reduce the project duration, all the cuts of the critical subnetwork are considered and the cheapest one is chosen. Then, all the activities belonging to the minimum cut are shortened by one time unit. In the original Kaufmann and Desbazeille algorithm, the cut P is defined as the set of arcs such that:

1. After removing all the arcs belonging to P, the project network is no longer connected.

2. The set of nodes V splits into two disjoint subsets: V_1 containing the starting event and V_2 containing the end event of the project, such that the starting nodes of all the arcs from P belong to V_1, while the end nodes belong to V_2.

As mentioned above, the natural way of increasing the exactitude of the algorithm is to consider a more general concept of a cut, as Ford and Fulkerson do (see e.g. [3], [10], [17], [24]). More exactly, they only use the second point of the last definition. This implies in turn that in the case of Kaufmann and Desbazeille's definition, the cut contains exactly one activity from each path, while in the Ford and Fulkerson variant it contains at least one task from each path. This slight modification allows us to use the FFEK* algorithm (see e.g. [7]) for finding the minimum cut, instead of an exhaustive search over all the cuts,

*Edmonds–Karp algorithm, one of the adaptations of the Ford–Fulkerson algorithm.

which makes the algorithm much faster. To illustrate the difference between these two definitions of cuts, see Fig. 3 below. In the case of the original Kaufmann–Desbazeille algorithm the cuts allowed are $\{e_1, e_2\}$, $\{e_2, e_3, e_4\}$ and $\{e_4, e_5\}$, while in the case of the Ford–Fulkerson algorithm there is one more cut allowed, namely $\{e_1, e_5\}$.

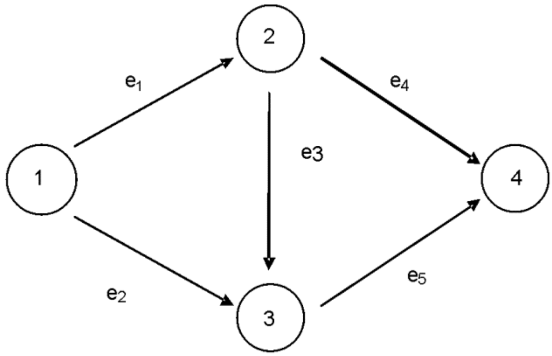

Fig. 3. Sample network

Finally, the tested version of the Kaufmann–Desbazeille algorithm can be defined as follows:

1. Initialization. Set the total shortening cost $TC := 0$ and go to step 2.
2. CPM. Perform CPM analysis of the project network. If $t_n^I \leq T^d$, then STOP. The current solution is the optimum one*. Otherwise go to step 3.
3. Time reduction/FFEK. Define the following maximum flow problem. The set of considered arcs in the network are all the critical arcs in the project network and the capacity of every arc e_i belonging to this network is equal to the first element of the sequence $C^{(i)}$ if this sequence has at least one element and ∞ otherwise. Using the FFEK method, find the minimum cut. If the value CV of the minimum cut MC is equal to ∞, then STOP. The deadline problem cannot be solved**. Otherwise go to step 4.
4. Update. Set $TC := TC + CV$. For each arc $e_i \in MC$ set $t_i := t_i - 1$ and remove the first element of $C^{(i)}$. Go back to step 2.

Observe that each time step 4 is performed, the total completion time of the overall project decreases by one. Thus the version of the KDA defined above always stops after a finite number of steps: it either finds the optimum solution or reports the problem to be inconsistent.

---

*It is optimum in the sense of the Kaufmann–Desbazeille algorithm, while it does not have to be the global minimum of the defined deadline problem.

**It cannot be solved using the KDA method. It can be easily proved this means that the deadline problem is contradictory (inconsistent).

The accuracy (A) of the KDA may be calculated in percentage terms as the relative difference between the total project compression costs obtained using this algorithm, i.e. C^{KDA}, and the minimum project compression costs $C^{\min}$ (the average is calculated over P, i.e. all the test problems, see Eq. (5)).

$$A = \frac{1}{P}\sum^{P}\left(\frac{C^{\mathrm{KDA}} - C^{\min}}{C^{\min}}\right) \tag{5}$$

The values $C^{\min}$ constitute the optimum solutions of the following deadline problem:

$$\sum_{i=1}^{m}\sum_{k=1}^{K(i)} c_{ik} y_{ik} \to \min \tag{6}$$

$$-x_{p(i)} + x_{q(i)} + \sum_{k=1}^{K(i)} y_{ik} - z_i = t_i^n, \qquad i = 1, ..., m \tag{7}$$

$$x_1 = 0, \qquad x_j \ge 0, \qquad j = 1, ..., n, \qquad x_n \le T^d \tag{8}$$

$$0 \le y_{ik} \le 1, \qquad i = 1, ..., m, \qquad k = 1, ..., K(i) \tag{9}$$

$$z_i \ge 0, \qquad i = 1, ..., m \tag{10}$$

with parameters: m – number of activities (arcs), n – number of events (nodes), t_i^n, t_i^c – the normal and crash times of activity e_i, $1 \le i \le m$, $K(i)$ – maximum number of units that activity e_i can be shortened by, $K(i) = t_i^n - t_i^c$, c_{ik} – cost of shortening activity e_i by k^{th} unit, $c_{ik} \le c_{i,k+1}$, $1 \le k < K(i)$, $p(i)$, $q(i)$ – indices of the starting and end node of arc e_i, $1 \le p(i) < q(i) \le n$, T^d – the desired project completion time, and variables: x_j – time of event v_j, $1 \le j \le n$, y_{ik} – binary variable equal to 1 when activity e_i is shortened by k^{th} unit and 0 otherwise*, z_i – final time of activity e_i, $1 \le i \le m$.

In connection with the assumptions made by Kaufmann and Desbazeille (the time parameters are integer) and the fact that we only analyze the deadline problem, the variables in the model (6)–(10) are always integer, although there are no additional constraints requiring integer solutions, see [34]. This problem is not NP-hard.

4. Worst case analysis

Let us consider a network defined as follows: The set of nodes consists of n nodes. The set of arcs (activities) consists of all the pairs of nodes of the form (v_j, v_{j+2}),

---

*We can assume, however, that the variables y_{ik} are continuous – the properties of the constraint matrix (total unimodularity) imply that in the case of integer times t_i^n and T^d, the optimum values of all the variables are integer as well. Thus every y_{ik} finally equals either 0 or 1.

$j = 1, \ldots, n - 2$, and all the pairs of nodes of the form $(v_j, v_{j+1}), j = 1, \ldots, n - 1$ (Fig. 4). We claim that such a network is the worst case for the KDA, provided the duration times and shortening costs are chosen in the right way.

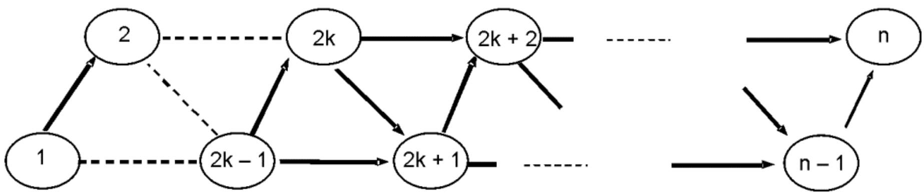

Fig. 4. Network of type 1

Proposition 1. The worst case for the KDA (possibly not the only one) is the situation where:

1. The network has the structure given in Fig. 4 and the number of nodes (n) is even.

2. Normal completion times of the activities <k, $k + 1$> equal $t^n_{k,k+1}$ and the normal completion times of the activities <k, $k + 2$> equal $2t^n_{k,k+1} - 1$ (Fig. 4).

3. Unit shortening costs of the activities <$2k$, $2k + 1$>, <$2k - 1$, $2k$> and <k, $k + 2$> are equal to a, b and c, respectively, where $b > a$ and $c > b((n/2) - 1)$ (Fig. 4).

4. Desired project completion time T^d is equal to $T^n - ((n/2) - 1) - 1$, where T^n is the normal project time (i.e. $T^n = (n-1)t^n_{k,k+1}$) and $(n/2) - 1$ is the number of all activities <$2k$, $2k + 1$>. This means that

$$T^d = T^n - \frac{n}{2} = (n-1)t^n_{k,k+1} - \frac{n}{2}$$

Proof. As can be easily checked, in such a situation the solution given by the KDA can be arbitrarily far from the optimum one. Indeed, the difference between the total shortening cost given by KDA and the optimum one amounts to $C^{\mathrm{KDA}} - C^{\min} = a((n/2) - 1)$ and thus cannot be bounded from above, even if we fix either the shortening cost a or the number of vertices n.

5. Conclusion

Although numerous authors (see e.g. [2], [12], [21], [25], [29], [30], [38], [39]) describe the KDA as an exact algorithm, our analysis has disproved this. Worst-case networks for the KDA are characterized by an extremely specific structure and level of shortening costs for each type of activity, together with a huge number of critical and subcritical paths with a lot of common edges. We should also take into account that our analysis focused on the modified version of the KDA. The accuracy of the original method is certainly worse.

References

[1] BERMAN E.B., *Resource allocation in a PERT network under continuous activity time-cost functions*, Management Science, 1964, 10 (4), 734–745.

[2] BLADOWSKI S., *Metody sieciowe w planowaniu i organizacji pracy*, PWE, Warszawa, 1970.

[3] CORMEN T.H., LEISERSON C.E., RIVEST R.L., *Introduction to Algorithms*, MIT Press, Cambridge, 2001.

[4] CROWSTON W.B., THOMPSON G.L., *Decision CPM: A method for simultaneous planning, scheduling, and control of projects*, Operations Research, 1967, 15, 407–426.

[5] DE P., DUNNE E.J., GOSH J.B., WELLS C.E., *The discrete time-cost tradeoff problem revisited*, European Journal of Operations Research, 1995, 81, 225–238.

[6] DE P., DUNNE E.J., GOSH J.B., WELLS C.E., *Complexity of the discrete time-cost trade-off problem for project networks*, Operations Research, 1997, 45, 302–306.

[7] EDMONDS J., KARP R.M., *Theoretical improvements in the algorithmic efficiency for network flow problems*, Journal of the ACM, 1972, 19, 248–264.

[8] FALK J.E., HOROWITZ J.L., *Critical path problems with concave cost–time curves*, Management Science, 1972, 19, 446–455.

[9] FONDHAL J.W., *A Non-Computer Approach to the Critical Path Method for the Construction Industry*, Construction Institute Technical Report No. 9, Construction Institute, Stanford University, 1961.

[10] FORD L.R., FULKERSON D.R, *Flows in Networks*, Princeton University Press, Princeton, New York, 1962.

[11] FULKERSON D.R., *A network flow computation for project cost curves*, Management Science, 1961, 7 (2), 167–178.

[12] FUSEK A., NOWAK K., PODLESKI H., *Analiza drogi krytycznej (CPM i PERT). Instrukcja programowana*, PWE, Warszawa, 1967.

[13] GASPARS H., *Analiza czasowo-kosztowa* (*CPM-COST*). *Algorytm a model optymalizacyjny*, Badania Operacyjne i Decyzje, 2006, nr 1, 5–19.

[14] GASPARS H., *Propozycja nowego algorytmu w analizie czasowo-kosztowej przedsięwzięć*, Badania Operacyjne i Decyzje, 2006, nr 3–4, 5–27.

[15] GASPARS-WIELOCH H., *Metody optymalizacji czasowo-kosztowej przedsięwzięcia*, Thesis, Poznań, University of Economics, Poznań, 2009.

[16] GASPARS-WIELOCH H., *Przegląd wybranych metod skracania czasu realizacji przedsięwzięcia*, [w:] *Metody i zastosowania badań operacyjnych*, Prace Naukowe Akademii Ekonomicznej w Katowicach, D. Kopańska-Bródka (red.), Wydawnictwo Akademii Ekonomicznej w Katowicach, 2008.

[17] GEDYMIN O., *Metody optymalizacji w planowaniu sieciowym*, PWN, Warszawa, 1974.

[18] GOYAL S.K., *A note on a simple CPM time-cost tradeoff algorithm*, Management Science, 1975, 21 (6), 718–722.

[19] HENDRICKSON C., AU T., *Project Management for Construction. Fundamental Concepts for Owners, Engineers, Architects and Builders*, Prentice Hall, 1989.

[20] HINDELANG T.J., MUTH J.F., *A dynamic programming algorithm for Decision CPM networks*, Operations Research, 1979, 27 (2), 225–241.
[21] IDŹKIEWICZ A.Z., *PERT. Metody analizy sieciowej*, PWN, Warszawa, 1967.
[22] KAUFMANN A., DESBAZEILLE G., VENTURA E., *La méthode du chemin critique*, Dunod, Paris, 1964.
[23] KELLEY J.E., *Critical path planning and scheduling: mathematical basis*, Operations Research, 1961, 9 (3), 296–320.
[24] KORZAN B., *Elementy teorii grafów i sieci. Metody i zastosowania*, Wydawnictwa Naukowo-Techniczne, Warszawa, 1978.
[25] *Badania operacyjne w przykładach i zadaniach*, K. Kukuła (red.), PWN, Warszawa, 1996.
[26] LIU L., BURNS S.A., FENG C.-W., *Construction time-cost tradeoff analysis using LP/IP hybrid method*, Journal of Construction Engineering and Management, 1995, 121 (4), 446–453.
[27] MODER J.J., PHILLIPS C.R., *Project Management with CPM and PERT*, Reinhold Publishing Corporation, New York, 1964.
[28] MOUSSOURAKIS J., HAKSEVER C., *Flexible Model for Time/Cost Tradeoff Problem*, Journal of Construction Engineering and Management, 2004, 130 (3), 307–314.
[29] MULLER Y., *Exercices d'organisation et de la recherche opérationnelle*, Eyrolles, Paris, 1965.
[30] MULLER Y., *Organisation et recherche opérationnelle*, Eyrolles, Paris, 1964.
[31] PANAGIOTAKOPOULOS D., *A CPM time-cost computational algorithm for arbitrary activity cost functions*, INFOR, 1977, 15 (2), 183–195.
[32] PHILLIPS S.J., DESSOUKY M.I., *Solving the time/cost tradeoff problem using the minimum cut concept*, Management Science, 1977, 24 (4), 393–400.
[33] PRAGER W., *A structural method of computing project cost polygons*, Management Science, 1963, 9 (3), 394–404.
[34] SCHRIJVER A., *Combinatorial Optimization*, Springer, Berlin, Vol. A, 2003, p. 76, Corollary 5.20.B.
[35] SIEMENS N., *A simple CPM time-cost tradeoff algorithm*, Management Science 1971, 17 (6), 354–363.
[36] SIUDAK M., *Badania operacyjne*, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa, 1998.
[37] SKUTELLA M., *Approximation algorithms for the discrete time-cost tradeoff problem, Mathematics of Operations Research*, 1998, 23 (4), 909–928.
[38] TROCKI M., GRUCZA B., OGONEK K., *Zarządzanie projektami*, PWE, Warszawa, 2003.
[39] WATERS D., *A Practical Introduction to Management Science*, Second Ed., Addison-Wesley Longman, 1998.