Research Article Open Access

Tadej Valenko and Uroš Klanšek*

An integration of spreadsheet and project management software for cost optimal time scheduling in construction

DOI 10.1515/otmcj-2016-0028
Received November 14, 2017; accepted November 24, 2017

Abstract: Successful performance and completion of construction projects highly depend on an adequate time scheduling of the project activities. On implementation of time scheduling, the execution modes of activities are most often required to be set in a manner that enables in achieving the minimum total project cost. This paper presents an approach to cost optimal time scheduling, which integrates a spreadsheet application and data transfer to project management software (PMS). At this point, the optimization problem of project time scheduling is modelled employing Microsoft Excel and solved to optimality using Solver while organization of data is dealt by macros. Thereupon, Microsoft Project software is utilized for further managing and presentation of optimized time scheduling solution. In this way, the data flow between programs is automated and possibilities of error occurrence during scheduling process are reduced to a minimum. Moreover, integration of spreadsheet and PMS for cost optimal time scheduling in construction is performed within well-known program environment that increases the possibilities of its wider use in practice. An application example is shown in this paper to demonstrate the advantages of proposed approach.

Keywords: optimization, project scheduling, spreadsheet application, project management software

***Corresponding author: Uroš Klanšek,** University of Maribor, Faculty of Civil Engineering, Transportation Engineering and Architecture, Smetanova ulica 17, 2000 Maribor, Slovenia, E-mail: uros.klansek@um.si
Tadej Valenko, University of Maribor, Faculty of Civil Engineering, Transportation Engineering and Architecture, Smetanova ulica 17, 2000 Maribor, Slovenia, E-mail: tadej.valenko@student.um.si

1 Introduction

Cost-effective time scheduling is widely recognized as an important topic in management of construction projects. Namely, execution of project activities needs engagement of certain resources and direct costs. Especially, the amount of working time consumed by labour and machines represents an influential indicator that requires to be considered in preparation of construction schedules (Petlíková and Jarský 2017).

Acceleration of project activities from their normal execution modes often demands additional resources, and the related direct costs are consequently amplified. On the other hand, indirect costs usually decrease when project implementation is accelerated. Both aforesaid facts encourage the achievement of project execution in an optimal duration and under the minimum total cost. Another critical success factor that can also come to the fore is keeping available project budget within boundaries and not to surpass contracted deadline.

In conventional construction practice, a cost-effective time schedule of the project is usually attained through a time-consuming cost-duration analysis of different feasible alternatives. However, optimization has proven to be a better way to achieve cost-effective schedules in comparison with traditional approaches. Over the years, a number of different approaches have been successfully applied in cost optimization of project schedules, mostly based either on approximate heuristic approaches or on exact mathematical programming.

In regard to heuristic methods, research works on the topic of optimal project scheduling can be found in recent literature. For example, cost optimization of project schedules has been effectively carried out by genetic algorithms (Eshtehardian et al. 2009), simulated annealing (He et al. 2009), tabu search (Hazir et al. 2011), neural networks (Adeli and Karim 1997), ant colony optimization (Kalhor et al. 2011), particle swarm optimization (Yang 2007), differential evolution (Nearchou 2010), harmony search (Geem 2010) and hybrid methods, such as genetic algorithm and dynamic programming (Ezeldin and Soliman 2009), cutting plane method and Monte Carlo simulation (Mokhtari et al. 2010), genetic algorithm and simulated annealing (Sonmez and Bettemir 2012) among others. Certainly, there are also various extensions of aforesaid techniques that can be found in the literature.

In search of high-quality solutions, different exact techniques have also been proposed to be used for reaching cost-optimized project schedules alongside the mentioned heuristic approaches. Linear and nonlinear programming models have been handled by continuous optimization algorithms. In regard to discrete scheduling, there exist two branches of optimization models that have been mostly suggested to be dealt by exact methods: (1) mixed-integer linear programming (LP) models such as those proposed by Sakellaropoulos and Chassiakos (2004), Vanhoucke (2005), Hazir et al. (2010) and others and (2) mixed-integer nonlinear programming models such as those recently introduced by Al Haj and El-Sayegh (2015) and Klanšek (2016).

Projects often consist of a complex system of activities that should be coordinated and managed to achieve their goals (Crnković and Vukomanović 2016). The dynamic and stochastic environment of construction projects further aggravates the decision-making processes and renders them more complex (Galić et al. 2016a, 2016b). Hence, usage of commercial software packages can significantly contribute to optimally solve project scheduling problems in practice. Core of the software packages is solver engines with different search algorithms that are capable of solving only a certain class or a number of different classes of optimization models.

For instance, spreadsheets are a popular tool for dealing with optimization problems among users. Also, WinQSB and Excel (Microsoft computer package), with add-ins like What'sBest, Evolver or Solver can be exposed as those spreadsheet software that are often employed for optimization. The latter one has proved to be a powerful spreadsheet tool with robust programming capabilities known as Visual Basic for Applications (VBA). Application of Solver in the field of project scheduling was suggested by researchers like Silva Filho et al. (2010), Trautmann and Gnägi (2015), and others.

Project management software (PMS) has been frequently considered for graphical representation and control of time schedules. In this context, Microsoft Project has been widely recognized as a useful tool for project management (Kostalova and Tetrevova 2014), and its successful applications were documented in a number of articles (Kažović and Valenčić 2013; Von Laszewski and Dilmanian 2008).

This paper presents an approach to cost optimal time scheduling, which integrates a spreadsheet application and data transfer to PMS. At this point, the optimization problem of project time scheduling is modelled employing Microsoft Excel and solved to optimality using Solver, while organization of data is dealt by macros. Thereupon, Microsoft Project software is utilized for further managing and presentation of optimized time scheduling solution. An application example is shown in this paper to demonstrate the advantages of proposed approach.

The rest of this article is organized as follows. Section 2 introduces optimization problem formulation. Modelling and solving optimization problems with spreadsheets are presented in Section 3, which is followed by Section 4 that deals with PMS. Section 5 demonstrates integration of spreadsheet and PMS on an application example related to cost optimal time scheduling in construction, and Section 6 provides some conclusions at the end of the paper.

2 Optimization problem formulation

Optimization methods are capable of solving various engineering problems. Although optimization problems can come from different areas and may deal with totally incomparable systems, they could be formulated in a similar manner. Generally, the problem of optimization may be determined as minimize $f(\boldsymbol{x})$, subject to $\boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}$ and $\boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0}$, where $f(\boldsymbol{x})$ denotes the objective function, which is required to be minimized over the vector of decision variables $\boldsymbol{x}$, $\boldsymbol{h}(\boldsymbol{x}) = \mathbf{0}$ covers equality constraints while $\boldsymbol{g}(\boldsymbol{x}) \leq \mathbf{0}$ indicates inequality conditions.

Here, the objective function formulates the criterion for identification of the optimum solution while constraints set up boundaries for the feasible space. It needs to be exposed here that the objective function can also be maximized within the space of feasible solutions if required. As far as decision variables are concerned, they are commonly calculated among their lower and upper boundaries, $\boldsymbol{x}^{\mathbf{LO}} \leq \boldsymbol{x} \leq \boldsymbol{x}^{\mathbf{UP}}$. Note that the variables can be continuous, $\boldsymbol{x} \in \boldsymbol{R}$, where $\boldsymbol{R}$ is the set of real numbers, or integer, $\boldsymbol{x} \in \boldsymbol{Z}$, where $\boldsymbol{Z}$ is the set of integers. Integer variables may also act as binary 0-1 decisions, i.e. $\boldsymbol{x} \in \{0,1\}^m$.

Certainly, it is necessary to consider that the selection of an adequate search algorithm for solving a particular optimization problem should be implemented attentively in order to acquire valuable output results. Therefore, prior to choosing the solution technique, the optimization problem needs to be analyzed in terms of its functions, conditions and decision variables.

3 Modelling and solving optimization problem with spreadsheets

Modelling advantages of spreadsheet software have been widely recognized quite sometime ago and such program tools are now broadly spread into numerous areas of human activities. One of the leading reasons that support their success lies in their intuitive cell-based structure and reasonably simple interface. The usable features of the application, such as data entry and manipulation, functions, graphs, word processing capacities, workgroup sharing, programmability alternatives, and a wide range of add-in programs, make them one of the essential and often employed tools by many computer users who need to process and deal with a large number of information. Along with their basic usage, spreadsheets have been applied as software tools for developing mathematical models in many different areas. Herein, a spreadsheet program Microsoft Excel 2016 is used for the purpose of modelling the optimization problem of project time scheduling.

Excel includes an add-in tool known as Solver that can be suitably used for the aim of solving the problem of project time scheduling to optimality. Solver tool basically represents a software program where solver engines cover more search algorithms for solving a certain type of optimization problems (Frontline Systems 2017). Here, Simplex LP engine is selected and applied to optimally solve project time scheduling problem, which was presented in the application example of this article. This engine possesses capacity of solving smooth linear optimization problems and represents one of the three engines that are covered by Solver software, others being covered by GRG Nonlinear engine, for solving smooth nonlinear optimization problems, and Evolutionary engine, for nonsmooth ones.

As soon as the project schedule is optimized, the obtained output data should be processed by PMS in which their graphical representation and control during project execution can be performed. Most PMS are capable of importing spreadsheets but only in a specific data arrangement. Excel programming capabilities, VBA, are suggested to be used as a very powerful additional feature at that phase.

VBA represents an implementation of Microsoft's event-driven programming language Visual Basic which allows building of automated processes. At this point, it is applied to automate the process of designing spreadsheet form, recognizable to selected PMS, from optimized time scheduling data reported in Microsoft Excel file.

4 Project management software

PMS is required to support organizing, planning and managing resources in project management. They are capable of dealing with estimation, project scheduling, budget management, resource assignment, cooperation, communication, decision-making, quality management and administration systems. Various PMS solutions are now available in the market. However, trends in the development of recent planning and optimization models are also simulation based (Galić et al. 2017) and this article intends to contribute to these areas.

For successful project management, it is important that all project activities are performed within a deadline, which means that the financial plan is set in accordance with the estimated budget as well as that the project beneficiaries are satisfied with project implementation and benefits derived from it (Biafore 2013; Harris 2016). In this particular paper, Microsoft Project is selected to manage project schedules. As is well-known, it is primarily intended for planning, monitoring and control of project realization and represents one of the most commonly applied software packages for project management in construction. Microsoft Project may be used in (Marmel 2013):

- monitoring of all gathered project information,
- visualization and demonstration of project schedules,
- efficient assignment of tasks and resources,
- exchange of information among the project team,
- communication between parties involved in the project, etc.

Mentioned program is applied here in the context of graphical demonstration of optimal project schedules. At the beginning, basic information about the project are required to be inserted in Project options dialog such as calendar type, currency, start date, working days, hours of work per day, and others as demonstrated in Figure 1.

Project information about activities, their durations and costs as well as precedence relations need to be

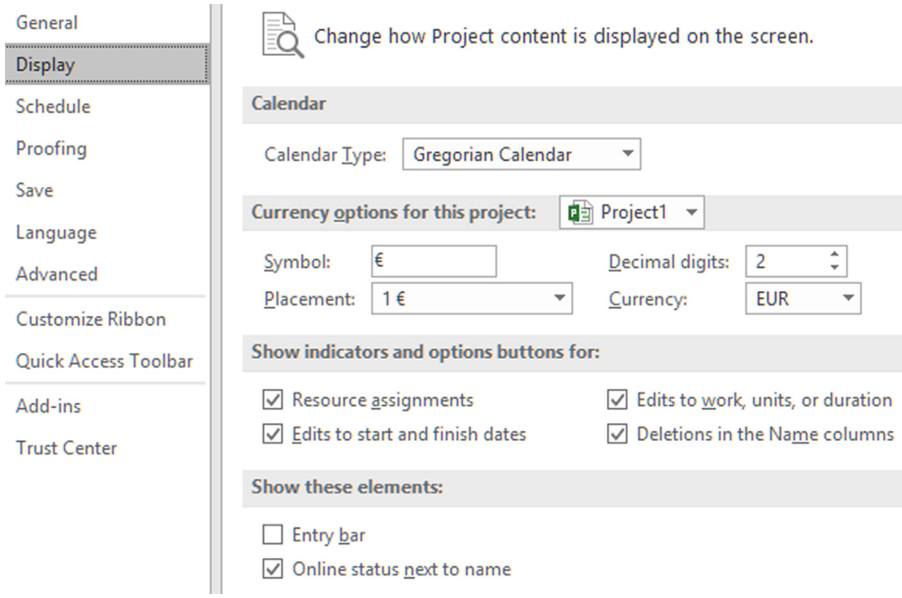


Fig. 1: Project options.

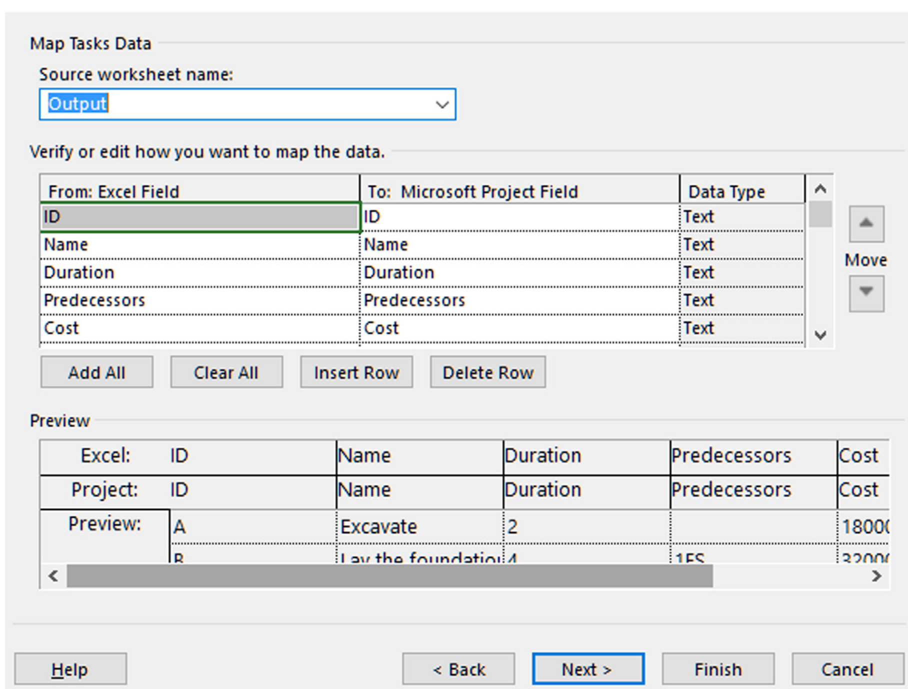


Fig. 2: Import wizard.

generated into PMS (Valenko and Klanšek 2017). Since all required information are gathered in Microsoft Excel, macro written in VBA programming language is implemented to create a new spreadsheet that can be recognized by Microsoft Project. New spreadsheet should contain columns with the following information about project activities for input recognition: ID, name, duration, predecessors and costs. Import wizard should be executed for data transfer as soon as all the data are nested within a new spreadsheet (Figure 2).

Visualization of optimization results is attained by Microsoft Project where Gantt chart tool is used for graphical illustration of optimal time schedule for the project. As is known, Gantt chart enables user-friendly identification of starting and finishing times for activities on a time line.

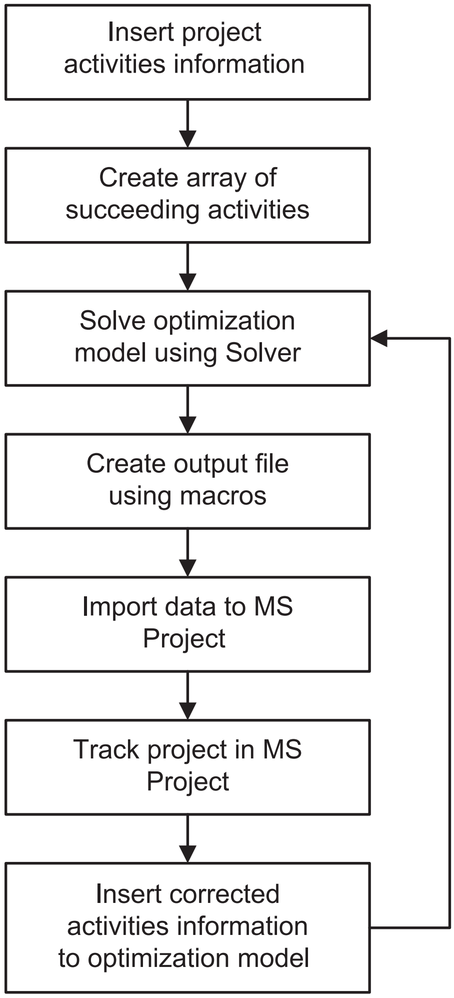


Fig. 3: Usage steps of proposed integration of spreadsheet and PMS.

The flowchart for using the proposed integration of spreadsheet and PMS for cost optimal time scheduling in construction is shown in Figure 3.

It should be noted here that the proposed approach enables optimal scheduling in project planning phase as well as tracking and optimal update of schedules during project execution. The use of proposed approach is supported by an application example shown in the following section.

5 Application example

In order to show advantages of solving cost optimization problems of project scheduling using Solver and data transfer from Microsoft Excel to Microsoft Project, a modified example project, originally introduced by Hillier and Lieberman (2014), is considered in this section. Example project incorporates 14 activities. Activities and their predecessors are given as follows:

- A – excavate,
- B – lay the foundations, preceding activity: A,
- C – put up the rough wall, preceding activity: B,
- D – put up the roof, preceding activity: C,
- E – install the exterior plumbing, preceding activity: C,
- F – install the interior plumbing, preceding activity: E,
- G – put up the exterior siding, preceding activity: D,
- H – do the exterior painting, preceding activities: E, G,
- I – do the electrical work, preceding activity: C,
- J – put up the wallboard, preceding activity: F, I,
- K – install the flooring, preceding activity: J,
- L – do the interior painting, preceding activity: J,
- M – install the exterior fixtures, preceding activity: H,
- N – install the interior fixtures, preceding activities: K, L.

Tab. 1: Duration and direct costs of activities.

Activity ID	Duration [weeks]		Direct cost		Direct cost–duration function
	Option 1	Option 2	Option 1	Option 2	
A	1	2	280000	180000	$(380–100T_A) \times 10^3$
B	2	4	420000	320000	$(520–50T_B) \times 10^3$
C	7	10	860000	620000	$(1420–80T_C) \times 10^3$
D	4	6	340000	260000	$(500–40T_D) \times 10^3$
E	3	4	570000	410000	$(1050–160T_E) \times 10^3$
F	3	5	260000	180000	$(380–40T_F) \times 10^3$
G	4	7	1020000	900000	$(1180–40T_G) \times 10^3$
H	6	9	380000	200000	$(740–60T_H) \times 10^3$
I	5	7	270000	210000	$(420–8300T_L) \times 10^3$
J	6	8	490000	430000	$(670–30T_J) \times 10^3$
K	3	4	200000	160000	$(320–40T_K) \times 10^3$
L	3	5	350000	250000	$(500–50T_L) \times 10^3$
M	1	2	200000	100000	$(300–100T_M) \times 10^3$
N	3	6	510000	330000	$(690–60T_N) \times 10^3$

Cost-duration alternatives and direct cost-duration functions for project activities are given in Table 1. Linear approximation of direct cost-duration relations is developed for project activities based on given cost-duration options. First option denotes crashed activity duration while the second one is representation of activity duration in normal mode.

Targeted project duration is set at 47 working weeks and indirect project cost is determined to be 31.000 € per week. The purpose of optimization was to find the project time schedule with optimal durations and start times of activities at minimum total project costs.

The problem of cost optimal time scheduling was set here as LP task. Optimization model formulation contained objective function, consisted of direct costs

Activity ID	Description	Preceding activities
A	*Excavate*	-
B	*Lay the foundations*	A
C	*Put up the rough wall*	B
D	*Put up the roof*	C
E	*Install the exterior plumbing*	C
F	*Install the interior plumbing*	E
G	*Put up the exterior siding*	D
H	*Do the exterior painting*	E, G
I	*Do the electrical work*	C
J	*Put up the wallboard*	F, I
K	*Install the flooring*	J
L	*Do the interior painting*	J
M	*Install the exterior fixtures*	H
N	*Install the interior fixtures*	K, L

Targeted project duration [weeks]
47

Indirect project cost [€/week]
31000

Activity	Duration [weeks]		Direct cost [€]		Direct cost-duration function
ID	Option 1	Option 2	Option 1	Option 2	[× 1000 €]
A	1	2	280000	180000	$380\text{-}100T_A$
B	2	4	420000	320000	$520\text{-}50T_B$
C	7	10	860000	620000	$1420\text{-}80T_C$
D	4	6	340000	260000	$500\text{-}40T_D$
E	3	4	570000	410000	$1050\text{-}160T_E$
F	3	5	260000	180000	$380\text{-}40T_F$
G	4	7	1020000	900000	$1180\text{-}40T_G$
H	6	9	380000	200000	$740\text{-}60T_H$
I	5	7	270000	210000	$420\text{-}8300T_L$
J	6	8	490000	430000	$670\text{-}30T_J$
K	3	4	200000	160000	$320\text{-}40T_K$
L	3	5	350000	250000	$500\text{-}50T_L$
M	1	2	200000	100000	$300\text{-}100T_M$
N	3	6	510000	330000	$690\text{-}60T_N$

Fig. 4: Input data in Microsoft Excel.

Suc\Pre	A	B	C	D	E	F	G	H	I	J	K	L	M	N
A														
B	1													
C		1												
D			1											
E			1											
F					1									
G				1										
H					1		1							
I			1											
J						1			1					
K										1				
L										1				
M								1						
N											1	1		

Fig. 5: Array of activities.

of activities and indirect project cost, which was subjected to precedence relationship constraints and project duration restraints, including bounds on durations and start times of activities as well as limitations on project as a whole. Project activities were generated following the principle of activity-on-node approach. Note here that the optimization problem formulation for the example project can be found in the study by Hillier and Lieberman (2014).

After the LP model formulation was set, a spreadsheet application Excel from Microsoft was employed for modelling for data inputs/outputs. Input data for the example project are shown in Figure 4.

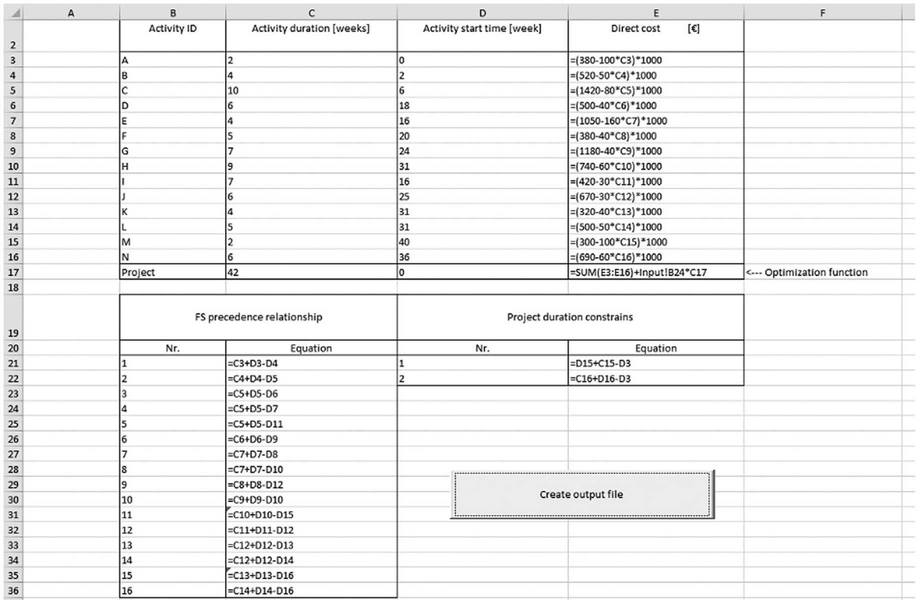
First spreadsheet covers input data of the addressed example project such as IDs of activities, their descriptions and predecessors, and normal and crashed durations supported by related direct costs accompanied with direct cost-duration functions, including targeted project duration as well as indirect project cost.

For the aim of further data manipulation, an array of preceding and succeeding project activities was created. Preceding activities were determined as columns and succeeding activities were set as rows of the array. As soon as precedence relation between connected activities was established, the intersecting cell included the value of 1. Array was defined by second spreadsheet as shown in Figure 5.

Durations and start times of project activities were determined as changing cells in third spreadsheet as introduced in Figure 6. Objective function and (in)equality constraints were also incorporated into the optimization model.

Excel's add-in Solver was allowed to execute the cost optimization of example project time schedule applying the developed model. Settings in Solver were done as shown in Figure 7. Objective was appointed, where minimal value was desired and changing variable cells were described. Objective function was subjected to blocs of constrains and Simplex LP was employed to solve the optimization problem.

Optimization problem was solved by a personal computer (Intel Core2 Duo P8600, 2.40 GHz, 4GB RAM DDR3 and 250 GB SSD disc). LP approach applied on considered project scheduling problem represents continuous



	A	B	C	D	E	F
2		Activity ID	Activity duration [weeks]	Activity start time [week]	Direct cost [€]	
3		A	2	0	=(380-100*C3)*1000	
4		B	4	2	=(520-50*C4)*1000	
5		C	10	6	=(1420-80*C5)*1000	
6		D	6	18	=(500-40*C6)*1000	
7		E	4	16	=(1050-160*C7)*1000	
8		F	5	20	=(380-40*C8)*1000	
9		G	7	24	=(1180-40*C9)*1000	
10		H	9	31	=(740-60*C10)*1000	
11		I	7	16	=(420-30*C11)*1000	
12		J	6	25	=(670-30*C12)*1000	
13		K	4	31	=(320-40*C13)*1000	
14		L	5	31	=(500-50*C14)*1000	
15		M	2	40	=(300-100*C15)*1000	
16		N	6	36	=(690-60*C16)*1000	
17		Project	42	0	=SUM(E3:E16)+Input!B24*C17	<--- Optimization function
18						
19		FS precedence relationship		Project duration constrains		
20		Nr.	Equation	Nr.	Equation	
21		1	=C3+D3-D4	1	=D15+C15-D3	
22		2	=C4+D4-D5	2	=C16+D16-D3	
23		3	=C5+D5-D6			
24		4	=C5+D5-D7			
25		5	=C5+D5-D11			
26		6	=C6+D6-D9			
27		7	=C7+D7-D8			
28		8	=C7+D7-D10			
29		9	=C8+D8-D12			
30		10	=C9+D9-D10			
31		11	=C10+D10-D15			
32		12	=C11+D11-D12			
33		13	=C12+D12-D13			
34		14	=C12+D12-D14			
35		15	=C13+D13-D16			
36		16	=C14+D14-D16			

Fig. 6: Changing cells.

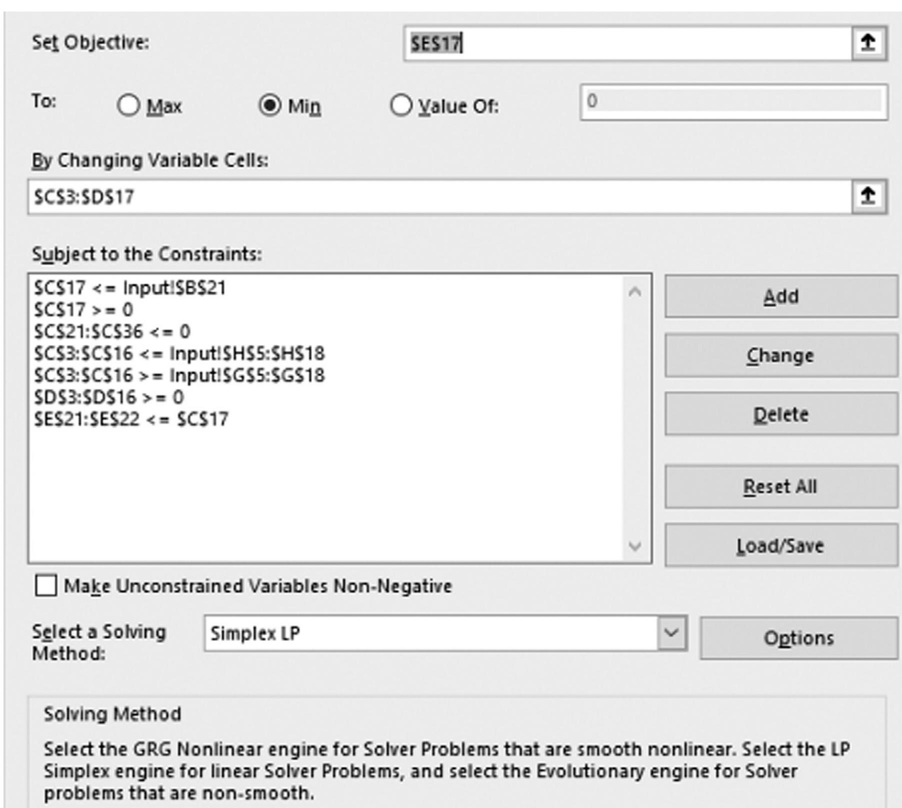


Fig. 7: Settings in Solver.

optimization technique by which optimal values of variables were calculated between their upper and lower bounds.

Minimum total cost necessary to complete the example project, gained by selected solution method, was 5.912.000 €, and the optimal project duration was found to be 42 weeks. Here, minimum total cost contained 4.610.000 € of direct cost and 1.302.000 € of indirect cost. The obtained optimal durations and direct costs of project activities are shown in Figure 8.

	A	B	C	D	E	F
1	ID	Name	Duration	Predecessors	Cost	
2	A	Excavate	2		180000	
3	B	Lay the foundations	4	1FS	320000	
4	C	Put up the rough wall	10	2FS	620000	
5	D	Put up the roof	6	3FS	260000	
6	E	Install the exterior plumbing	4	3FS	410000	
7	F	Install the interior plumbing	5	5FS	180000	
8	G	Put up the exterior siding	7	4FS	900000	
9	H	Do the exterior painting	9	7FS,5FS	200000	
10	I	Do the electrical work	7	3FS	210000	
11	J	Put up the wallboard	6	9FS,6FS	490000	
12	K	Install the flooring	4	10FS	160000	
13	L	Do the interior painting	5	10FS	250000	
14	M	Install the exterior fixtures	2	8FS	100000	
15	N	Install the interior fixtures	6	12FS,11FS	330000	

Fig. 8: Output file.

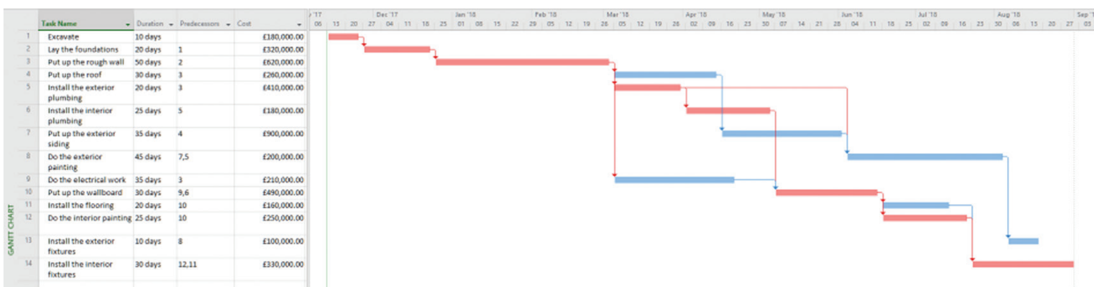


Fig. 9: Optimal time schedule for example project.

After the optimization process was executed and optimal solution for the example project time schedule was found, macro written in VBA was applied to create a new output file. Using the activation of the command button labelled "Create output file," demonstrated in Figure 6, macro was performed.

Output file included a spreadsheet in which array of project data was settled as displayed by Figure 8. Macro transferred input data of project activities from first spreadsheet are shown in the first two columns of the array of Figure 4. Durations of activities were rearranged from spreadsheet containing decision variables, introduced in Figure 6, to third column of the new array. Similar process was executed to direct cost values of project activities that are included in the fifth column.

Precedence relations among example project activities were set as finish-to-start (FS) ones, and therefore, predecessors in the fourth column were labelled with their ID numbers and FS abbreviations. In this connection, precedence relationships were rearranged from array of activities presented in Figure 5. Afterwards, the created output file was applied to import optimized data into Microsoft Office Project where subsequent data management were feasible. The graphical representation of optimal project time schedule, shown in Figure 9, was created using the Gantt chart tool.

In Gantt chart, non-critical activities were blue coloured while critical activities were labelled with red colour and their order comprised the critical path of optimized project schedule. As well-known, durations of critical activities directly affect the project duration and, consequently, its total costs. Thus, when execution times of critical activities are crashed, the associated direct costs are increased while indirect costs are reduced on account of shorter project duration. Note here that the acceleration of non-critical activities cannot affect the project duration but may only raise the direct costs.

For further workflow presentation, a delay in project activity was created. While tracking the progress of the project, the delay in activity J, i.e. putting up of the wallboard, has occurred. Aforesaid activity was 1 week behind the schedule, and therefore, the actualization of the project time schedule was necessary. Input data in Excel were changed in such a manner that durations of finished activities were fixed and the activity of putting up the wallboard was changed to a new duration of 7 weeks. Updated optimal duration of the project was rescheduled for a week since delayed activity represented a critical one (Figure 10).

Consequently, the total project costs increased to a value of 5.913.000 €. With transfer of data to MS Project, a new time schedule was created (Figure 11).

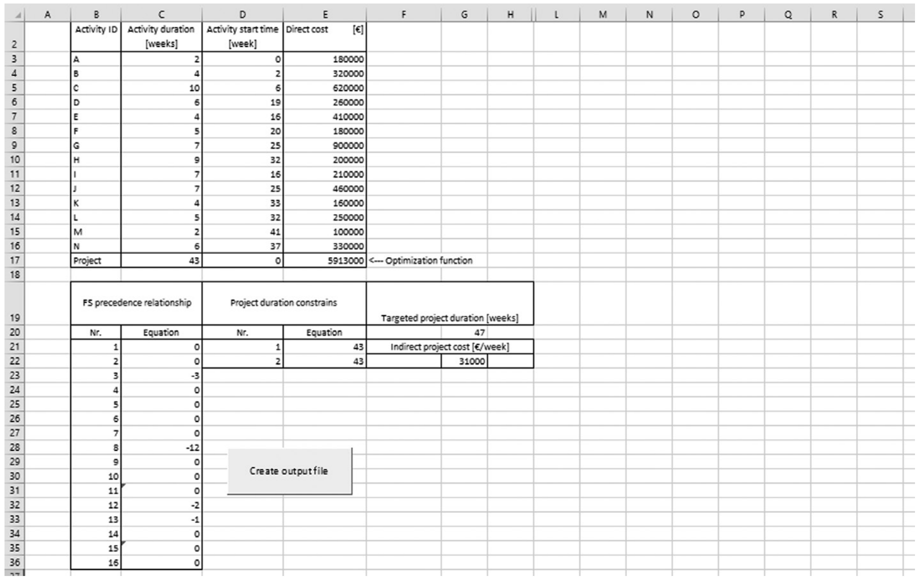


Fig. 10: Updated optimal solution for example project.

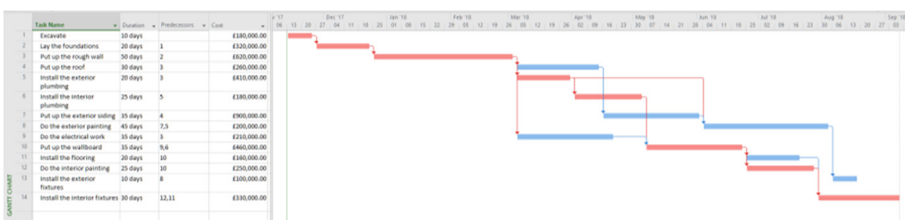


Figure 11: Updated time schedule for example project.

6 Conclusions

Construction projects are frequently recognized as challenging tasks where numerous activities should be performed in a certain mode before the production is finished. For that purpose, production resources like labour, machinery, materials and equipment need to be utilized in construction activities. Furthermore, resources can be assigned to different activities in such a way that activity durations are crashed, and therefore, time schedule is much important for cost-effective realization of construction project.

Significance of tools for project management was widely acquainted in construction business since their applications allow more efficient time scheduling. It is in parallel with the development of various programs in computers, which came to the fore and enabled quicker and easier introduction of management tools to workflow. Nowadays, contemporary PMS are employed for monitoring and visualization of project tasks, information exchange among engaged team, communication with project parties, etc. Cost optimization of time schedules, on the other hand, is commonly achieved by specialized software for solving mathematical models and thus additional information flow to project management programs is required.

This paper presented an approach to cost optimal time scheduling in construction, which integrated a spreadsheet application and data transfer to PMS. Optimization problem of project time scheduling was modelled applying Microsoft Excel and solved to optimality by Solver while data management was dealt by macros. Afterwards, Microsoft Project software was employed for further organizing and presentation of optimized time scheduling solution. Mentioned software was selected here since Microsoft Office suite is broadly used among construction companies and it incorporates all tools required for process implementation.

An application example was presented to expose the advantages of proposed approach. The example demonstrated that the project time schedule modelled and optimized in Microsoft Office Excel can be conveniently transferred to Microsoft Project, where visualization and further treatment of gained results can be executed.

Data flow between programs is thus automated and possibilities of error occurrence during scheduling process are reduced to a minimum. Finally, the example introduced integration of spreadsheet and PMS for cost optimal time scheduling in construction within well-known program environment, which increases the possibilities of its wider use in practice.

Acknowledgment

The authors acknowledge the financial support from the Slovenian Research Agency (Research Core Funding No. P2-0129).

REFERENCES

Adeli, H., & Karim, A. (1997). Scheduling/cost optimization and neural dynamics model for construction. *Journal of Construction Engineering and Management, 123*(4), pp. 450-458. doi: 10.1061/(ASCE)0733-9364(1997)123:4(450).

Al Haj, R., & El-Sayegh, S. (2015). Time-cost optimization model considering float-consumption Impact. *Journal of Construction Engineering and Management, 141*(5), 04015001. doi: 10.1061/(ASCE)CO.1943-7862.0000966.

Biafore, B. (2013). *Microsoft Project 2013: The Missing Manual*. O'Reilly Media, Sebastopol, CA.

Crnković, D., & Vukomanović, M. (2016). Comparison of trends in risk management theory and practices within the construction industry. *E-GFOS, 7*(13), pp. 1-11. doi: 10.13167/2016.13.1.

Eshtehardian, E., Afshar, A., & Abbasnia, R. (2009). Fuzzy-based MOGA approach to stochastic time-cost trade-off problem. *Automation in Construction, 18*(5), pp. 692-701. doi: 10.1016/j.autcon.2009.02.001.

Ezeldin, A. S., & Soliman, A. (2009). Hybrid time-cost optimization of nonserial repetitive construction projects. *Journal of Construction Engineering and Management, 135*(1), pp. 42-55. doi: 10.1061/(ASCE)0733-9364(2009)135:1(42).

Frontline Systems. (2017). *Frontline Solvers Optimization and Simulation User Guide*. Available at https://www.solver.com/user-guides-frontline-systems-excel-solvers/ on 07 November, 2017).

Galić, M., Završki, I., & Dolaček-Alduk, Z. (2016a). Scenario simulation model for optimized allocation of construction machinery. *Građevinar, 68*(2), pp. 105-112. doi: 10.14256/JCE.1462.2015.

Galić, M., Završki, I., & Dolaček-Alduk, Z. (2016b). Methodology and algorithm for asphalt supply chain optimization. *Tehnicki Vjesnik-Technical Gazette, 23*(4), pp. 1193-1200. doi: 10.17559/TV-20150623140015.

Galić, M., Barišić, I., & Ištoka Otković, I. (2017). Route reliability based simulation model for HMA delivery in urban areas. *Procedia Engineering, 187*, pp. 378-386. doi: 10.1016/j.proeng.2017.04.389.

Geem, Z. W. (2010). Multiobjective optimization of time-cost trade-off using harmony search. *Journal of Construction Engineering and Management, 136*(6), pp. 711-716. doi: 10.1061/(ASCE)CO.1943-7862.0000167.

Harris, P. E. (2016). *Planning and Control Using Microsoft Project 2013 and 2016*. Eastwood Harris Pty Ltd, Doncaster Heights, VIC.

Hazir, Ö., Haouari, M., & Erel, E. (2010). Robust scheduling and robustness measures for the discrete time/cost trade-off problem. *European Journal of Operational Research, 207*(2), pp. 633–643. doi: 10.1016/j.ejor.2010.05.046.

Hazir, Ö., Erel, E., & Günalay, Y. (2011). Robust optimization models for the discrete time/cost trade-off problem. *International Journal of Production Economics, 130*(1), pp. 87-95. doi: 10.1016/j.ijpe.2010.11.018.

He, Z., Wang, N., Jia, T., & Xu, Y. (2009). Simulated annealing and tabu search for multi-mode project payment scheduling. *European Journal of Operational Research, 198*(3), pp. 688-696. doi: 10.1016/j.ejor.2008.10.005.

Hillier, F. S., & Lieberman, G. J. (2014). *Introduction to Operations Research,* 10th edn. McGraw-Hill Higher Education, New York, NY.

Kalhor, E., Khanzadi, M., Eshtehardian, E., & Afshar, A. (2011). Stochastic time-cost optimization using non-dominated archiving ant colony approach. *Automation in Construction, 20*(8), pp. 1193-1203. doi: 10.1016/j.autcon.2011.05.003.

Kažović, D., & Valenčić, D. (2013, May). Using Microsoft Project for project management in non-governmental organisations. In: *Information & Communication Technology Electronics & Microelectronics (MIPRO), 2013 36th International Convention* on 20-24 May 2013, Opatija, Croatia. IEEE, pp. 681-684.

Klanšek, U. (2016). Mixed-Integer nonlinear programming model for nonlinear discrete optimization of project schedules under restricted costs. *Journal of Construction Engineering and Management, 142*(3), 04015088. doi: 10.1061/(ASCE)CO.1943-7862.0001074.

Kostalova, J., & Tetrevova, L. (2014). Project management and its tools in practice in the Czech Republic. *Procedia-Social and Behavioral Sciences, 150*, pp. 678-689. doi: 10.1016/j.sbspro.2014.09.087.

Marmel, E. (2013), *Project 2010 Bible*. John Wiley & Sons, Indianapolis, IN.

Mokhtari, H., Aghaie, A., Rahimi, J., & Mozdgir, A. (2010). Project time-cost trade-off scheduling: A hybrid optimization approach. *International Journal of Advanced Manufacturing Technology, 50*, 5-8(2010), pp. 811-822. doi: 10.1007/s00170-010-2543-4.

Nearchou, A. C. (2010). Scheduling with controllable processing times and compression costs using population-based heuristics. *International Journal of Production Research, 48*(23), pp. 7043-7062. doi: 10.1080/00207540903433874.

Petlíková, K., & Jarský, Č. (2017). Modeling of the time structure of construction processes using neural networks. *Organization, Technology and Management in Construction: An International Journal, 9*(1), pp. 1559-1564. doi: 10.1515/otmcj-2016-0018.

Sakellaropoulos, S., & Chassiakos, A. P. (2004). Project time-cost analysis under generalised precedence relations. *Advances in Engineering Software, 35*(10-11), pp. 715-724. doi: 10.1016/j.advengsoft.2004.03.017.

Silva Filho, O. S., Cezarino, W., & Ratto, J. (2010). Aggregate production planning: Modeling and solution via Excel spreadsheet and solver. *IFAC Proceedings Volumes, 43*(17), pp. 89-94. doi: 10.3182/20100908-3-PT-3007.00020.

Sonmez, R., & Bettemir, Ö. H. (2012). A hybrid genetic algorithm for the discrete time-cost trade-off problem. *Expert Systems with Applications. 39*(13), pp. 11428-11434. doi: 10.1016/j.eswa.2012.04.019.

Trautmann, N., & Gnägi, M. (2015, December). On an application of Microsoft Excel's evolutionary solver to the resource-constrained project scheduling problem RCPSP. In: *Industrial Engineering and Engineering Management (IEEM), 2015 IEEE International Conference on* 6-9 Dec. 2015, Singapore. IEEE, pp. 646-650. doi: 10.1109/IEEM.2015.7385727.

Valenko, T., & Klanšek, U. (2017, September). Cost optimal time scheduling integrating spreadsheet and project management software. In: *13th International Conference Organization, Technology and Management in Construction, 2017*. Croatian Association for Construction Management: University of Zagreb, Faculty of Civil Engineering, pp. 42-53.

Vanhoucke, M. (2005). New computational results for the discrete time/cost trade-off problem with time-switch constraints. *European Journal of Operational Research, 165*(2), pp. 359–374. doi: 10.1016/j.ejor.2004.04.007.

Von Laszewski, G., & Dilmanian, L. E. (2008, November). e-Science project and experiment management with Microsoft Project. In: *Grid Computing Environments Workshop, 2008. GCE'08*. IEEE, pp. 1-8. doi: 10.1109/GCE.2008.4738449.

Yang, I.-T. (2007). Using elitist particle swarm optimization to facilitate bicriterion time-cost trade-off analysis. *Journal of Construction Engineering and Management, 133*(7), pp. 498-505. doi: 10.1061/(ASCE)0733-9364(2007)133:7(498).