A MILP model for an integrated project scheduling and multi-skilled workforce allocation with flexible working hours

Ahmed Karam, El-Awady Attia, Philippe Duquenne

A MILP model for an integrated project scheduling and multi-skilled workforce allocation with flexible working hours

Ahmed KARAM*, El-Awady ATTIA*
Philippe DUQUENNE**

**Assistant professor, Mechanical Engineering Department, faculty of engineering (Shoubra) Benha University, Cairo, Egypt.*
(e-mail: {ahmed.mostafa, elawady.attia}@ feng.bu.edu.eg)
***Industrial Engineering Department, Toulouse University, INPT, Toulouse cedex 4 France. (e-mail: Philippe.Duquenne@ensiacet.fr)*

Abstract: In this paper, we integrate two decision problems arising in various applications such as production planning and project management: the project scheduling problem, which consists in scheduling a set of precedence-constrained tasks, where each task requires executing a set of skills to be performed, and the workforce allocation problem which includes assigning workers as scarce resources to the skills of each task. These two problems are interrelated as the tasks durations are not predefined, but depend on the number of workers assigned to that task as well as their skill levels. We here present a mixed integer linear programming model that considers important real life aspects related to the flexibility in the use of human resources, such as multi-skilled workers whose skill levels are different and measured by their efficiencies. Hence, execution times of the same workload by different workers vary according to these efficiencies. Moreover, the model considers the flexible working time of employees; i.e. the daily and weekly workload of a given worker may vary from one period to another according to the work required. Furthermore, efficient team building is incorporated in this model; i.e. assigning an expert worker and one or more apprentice worker(s) together with the purpose of skill development thanks to knowledge transfer. A numerical example is provided to check the performance of the model.

Keywords: Workforce flexibility, multi-skilled workers, working time flexibility, scheduling, mixed integer linear programming.

1. INTRODUCTION

Manufacturing or service organizations seek the best use of their resources' capacities. One of the most important is the increase of human resources capacity through the development of workers flexibility. This can be enhanced by adopting some managerial practices e.g. multi-skilled workers or flexible working time. The multi-skilled flexibility could be developed by adopting job rotation, cross-training, or team working that allows knowledge transfer. As stated by Małachowski and Korytkowski (2016), one of the fundaments of lean management lays in multi-skilled employees, able to reduce the non-added values by reducing different types of wastes: for instance, the idle time can be reduced whenever multi-skilled employees are available. A multi-skilled worker can be able to carry out a set of specified skills, which help him to overcome many changes or many problems. The degree of this flexibility can be evaluated by the number of skills and their heterogeneous level. Recently, the consideration of multi-skilled workers attracts the attention of both researchers and practitioners. As example Yang *et al.*, (2007) examined the effect of cross training and flexible workdays on job-shops performance; their results showed that the effect of cross training is more significant than that of flexible workdays. Attia *et al.*, (2012) investigated the factors that affect the development of workforce multi-skilled flexibility. Małachowski and Korytkowski (2016) proposed a competence-based analytical model that determines the performance of multi-skilled workers in executing repetitive tasks. Qin *et al.* (2015) provided a detailed survey for the different aspects and considerations of workforce flexibility in operations management.

Responding to the importance of implementing workforce flexibility in project/production planning and scheduling, the allocation problem of multi-skilled workforce was presented. Li and Womer (2009); Heimerl and Kolisch (2010); Chen *et al.* (2016) and Almeida *et al.* (2016) considered simultaneously the scheduling of projects with multi-skilled workers allocation. In production, Mencía *et al.* (2016) considered the production scheduling with workers allocations. Most of literature works treat the problem as a generalized case of the multi-mode resources constrained project scheduling problem, in which every work package requires a set of skills to be performed, and there is a pool of workers in which each one can perform a set of skills with homogonous or heterogeneous performance levels. Consequently, the task execution mode depends on the performance level of the workforce allocated.

Moreover, the literature provides evidence for the interest given to flexibility of working time that can be implemented by adopting overtime, working during days-off, or annualized hours etc. As example, the annualized working hours have been considered by Hung (1999) in developing weekly staff scheduling, Filho and Marçola (2001) considered it in rough-cut capacity planning of the production of agriculture equipments, and Corominas and Pastor (2010) adopted it in a reactive re-planning approach for the short-term workforce allocation. In aggregate production planning, many kinds of working time flexibility can be used that enlarges workforce capacity in order to overcome fluctuations in production demand (Leung and Chan, 2009; Baykasoglu and Gocken, 2010; Zhang *et al.*, 2012; Tavaghof-Gigloo *et al.*, 2016). Other human factors were considered in the workforce allocation problem e.g. productivity evolution, learning by practice, or learning by knowledge transfer. Wu and Sun (2005) considered learning phenomenon during workforce allocation for multi-projects, in R&D department. Attia *et al.* (2014) considered it in project scheduling with multi-skilled workforce allocation. But the model they presented is a MINLP that is very hard to be solved optimally. Consequently, they proposed to solve it with genetic algorithms. Recently, Nembhard and Bentefouet (2015) considered the problem with learning by knowledge transfer.

The current paper introduces a mixed integer linear program that integrates project scheduling with tasks precedence relations and the allocation of multi-skilled workforce. This problem is known here as Project Scheduling with Multi-Skilled Workforce Allocation Problem "PSMWAP". Each task requires a set of skills to be performed with an associated workload for each skill. The task duration is not predefined in advance but it depends on the employee skill level as well as the number of workers assigned to this task. The paper presents a novel mixed integer linear programming model that considers important real life aspects related to the flexibility in the use of human resources such as the multi-skilled workers whose skill levels are different and measured by the workers' efficiencies. Hence, execution times for a given workload by different workers may vary, depending on workers' efficiencies. In addition, our model considers the flexible working time of employees; i.e. the daily and weekly workload of a specified worker can vary from one period to another according to the work required. Many objectives are considered that minimize penalty for project tardiness, for allocating a work-team without an expert in a specified skill, or for the excessive use of available working time for a given worker, and the direct cost associated to normal working hours and overtime.

The rest of this paper will be organized as the following: section 2 discusses the problem description. Section 3 represents the mathematical model of the problem. Section 4 represents the solution methodology, results and discussion. finally section 5 presents the conclusions and the future work.

2. PROBLEM DESCRIPTION

In practice, the project consists of a specified set of unique tasks and each task requires a set of skills, each of which has a predetermined workload (measured in hours). In addition, all skills' workloads for a given task must start at the same time and the task ends when all of these workloads are completely performed. Furthermore, there exists a set of workers available for the execution of tasks' skills. We consider multi-skilled workers whose skill levels are different and measured by the worker's efficiency ($0 \le \theta \le 1$) in each skill. Hence, execution times of the same skill by different workers vary and depend on number of workers allocated and their efficiencies. There is a minimum level of efficiency for each skill that must be respected when assigning workers to it. In other words, assignment of a worker to a given skill is valid only if the worker can ensure this minimum level of efficiency. Each worker can't be assigned to more than one skill at the same time. On the other side, it is possible for one worker to be assigned to more than one skill, which means that workers can be reallocated among skills during different allocation periods. In addition, assigning idle worker(s) to ongoing skill would accelerate its execution while removing worker(s) from it would interrupt it. The numbers of daily and weekly working hours assigned to each worker are restricted by specified limits. Overtime work is allowed to accelerate the project execution when needed. Of course, temporal relationships between tasks must be respected. Workforce multi-skilled flexibility can be enhanced by considering the allocation of experts with non-professional workers for performing a specified skill. This allocation permits to enhance the secondary skills of workers thanks to knowledge transfer. Learning by knowledge transfer requires, if two or more workers are assigned simultaneously to a skill, that at least one of them is an expert in that skill. The more the worker's efficiency in a given skill approaches or exceeds a standard value, the more the worker becomes expert with this skill. Moreover, the effectiveness of learning by knowledge transfer depends to a large extent on the efficiency level of this expert. It should be noted that the dynamic nature of workers' efficiencies as a result of learning by knowledge transfer is not considered in this work. The aim of the PSMWAP is to allocate specific workers to the skills of every task, with an objective of minimizing four indicators which are related to labour cost, execution delay of the project, expert assignment and workforce temporal flexibility.

3. MATHEMATICAL MODEL

This section presents the proposed model for the PSMWAP incorporating additional real-life aspects as illustrated above.

Sets:

$\boldsymbol{X}$: Set of working days (indexed by t), $X = \{1,\dots,h\}$.
$\boldsymbol{\Psi}$: Set of weeks (indexed by s), $\boldsymbol{\Psi} = \{1,\dots,v\}$.
$\boldsymbol{\Omega}$: Set of workers (indexed by a), $\boldsymbol{\Omega} = \{1,\dots,u\}$.
$\boldsymbol{\pi}$: Set of project tasks (indexed by i), $\boldsymbol{\pi} = \{1,\dots,m\}$.
$\boldsymbol{\omega}$: Set of skills (indexed by k), $\boldsymbol{\omega} = \{1,\dots,n\}$.
$\boldsymbol{\Phi}$: Set of tasks among which there is a precedence relationship; when task i must precede task j, $(i,j) \in \boldsymbol{\Phi}$.

Parameters:

PS : The starting day of the project or the work package.
g_k : The number of workers mastering skill k.

y_{ak} : 1 if worker *a* is able to perform skill *k*; 0 otherwise.

∂_{ij} : Delay or advance between task *i* and task *j*, (i,j) ϵ**Φ.**

ES_i : Early start of task *i*.

$td_i^{\max}$: Maximum duration of task *i* in days.

$\sigma_a^{\max}$: Maximum daily working hours for worker *a*.

$\sigma_a^{\min}$: Minimum daily working hours for worker *a*.

γ_{norm} : Standard number of working hours per week.

$\gamma_{\max}$: Maximum number of working hours per week.

L : Contractual duration of the project in days.

λ : Number of working days per week.

$\theta_k^{\min}$: Minimum efficiency to practice skill *k*, $\theta_k^{\min} \leq 1$.

θ_{ak} : Efficiency of worker *a* in practicing skill *k*, $\theta_{ak} \in [0,1]$.

θ_{st} : Standard efficiency level at which a worker is considered as expert in practicing skill *k*.

WL_{ik} : Workload of skill *k* for task *i* in hours.

M : A positively large constant.

c^1 : Penalty cost incurred if the actual duration of the project exceeds the contractual duration, *L* (unit: USD per day).

c^2 : Penalty cost incurred if two or more workers assigned to a skill without assigning an expert and low efficient worker together to that skill (unit: equivalent USD).

c_a^3 : Penalty cost incurred if the actual working hour of worker *a* exceeds the standard number of working hours per week, δ (unit: USD per hour).

c_a^4 : Working cost per hour for worker *a* (USD).

c_a^5 : Overtime hourly cost for worker *a* (USD).

Decision variables:

pf : Integer variable represents project completion date.

ts_i : Integer variable represents starting date of task *i*.

tf_i : Integer variable represents completion date of task *i*.

ss_{ik} : Integer variable represents starting date of skill *k* of task *i*.

sf_{ik} : Integer variable represents completion date of skill *k* of task *i*.

f_{as} : Integer variable represents number of working hours performed by worker *a* in week *s*.

q_{as} : Integer variable represents overtime hours performed by worker *a* in week *s*.

Δpd : Integer variable representing the deviation between the actual duration of the project and the contractual duration, $\Delta pd = (pf - ps + 1 - L)^+$.

p_{aikt} : A binary variable equal to 1, if worker *a* is assigned to skill *k* of task *i* in day *t*, 0 otherwise.

e_{ikt} : A binary variable equal to 1, if skill *k* of task *i* is practiced by at least one worker during day *t*, 0 otherwise.

d_{ikt} : A binary variable equal to 1, if skill *k* of task *i* is assigned to two or more workers at day *t*, 0 otherwise.

c_{at} : Integer variable representing the number of working hours assigned to worker *a* at day *t*. $c_{at} \in [\sigma_a^{\min}, \sigma_a^{\max}]$.

h_{ikt} : Continuous variable representing the difference between θ_{st} and the efficiency of the most efficient worker assigned to skill *k* of task *i* at day *t*. Note that $h_{ikt} = 0$ if $d_{ikt} = 0$.

s_{ikt} : Continuous variable representing the difference between $\theta_{\min}^k$ and the minimum efficiency of the workers assigned to skill *k* of task *i* at day *t*. Note that $S_{ikt} = 0$ if $d_{ikt} = 0$.

The project scheduling with multi-skilled workforce allocation problem considering flexible working hours is formulated as follows:

$$\begin{aligned} Minimize\, Z = c^1 \cdot \Delta pd + c^2 \sum_{i\in\pi}\sum_{k\in\omega}\sum_{t\in X}(h_{ikt} + s_{ikt}) \\ + \sum_{a\in\Omega}\sum_{s\in\Psi} c_a^3 \cdot |f_{as} - \delta| + \sum_{a\in\Omega}\sum_{s\in\Psi} c_a^4 \cdot f_{as} \\ + \sum_{a\in\Omega}\sum_{s\in\Psi} c_a^5 \cdot q_{as} \end{aligned} \tag{1}$$

Subject to:

$$\sum_{i\in\pi}\sum_{k\in\omega} p_{aikt} \leq 1 \qquad \forall a\in\Omega, \forall t\in X \tag{2}$$

$$p_{aikt} \leq y_{ak} \qquad \forall a\in\Omega, \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{3}$$

$$\sum_{a\in\Omega}\sum_{i\in\pi} p_{aikt} \leq g_k \qquad \forall k\in\omega, \forall t\in X \tag{4}$$

$$ts_j \geq tf_i + \partial_{ij} \qquad \forall i,j\in\Phi \tag{5}$$

$$\theta_k^{\min} \cdot p_{aikt} \leq \theta_{ak} \qquad \forall a\in\Omega, \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{6}$$

$$\sum_{t\in X}\sum_{a\in\Omega} \theta_{ak} \cdot c_{at} \cdot p_{aikt} \geq WL_{ik} \qquad \forall i\in\pi, \forall k\in\omega \tag{7}$$

$$\sum_{a\in\Omega} p_{aikt} \leq M \cdot e_{ikt} \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{8}$$

$$e_{ikt} \leq \sum_{a\in\Omega} p_{aikt} \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{9}$$

$$sf_{ik} \geq t \cdot e_{ikt} \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{10}$$

$$ss_{ik} \leq t \cdot e_{ikt} + M(1 - e_{ikt}) \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in \{ES_i, \dots, h\} \tag{11}$$

$$\sum_{t\in X} e_{ikt} = sf_{ik} - ts_i + 1 \qquad \forall i\in\pi, \forall k\in\omega \tag{12}$$

$$sf_{ik} - ss_{ik} + 1 \leq td_i^{\max} \qquad \forall i\in\pi, \forall k\in\omega \tag{13}$$

$$\sum_{t=\lambda(s-1)+1}^{\lambda s}\sum_{a\in\Omega}\sum_{k\in\omega} c_{at} \cdot p_{aikt} = f_{as} \qquad \forall a\in\Omega, \forall s\in\Psi \tag{14}$$

$$f_{as} \leq \gamma_{\max} \qquad \forall a\in\Omega, \forall s\in\Psi \tag{15}$$

$$q_{as} \geq f_{as} - \gamma_{norm} \qquad \forall a\in\Omega, \forall s\in\Psi \tag{16}$$

$$\sum_{a\in\Omega} p_{aikt} - 2 \leq M \cdot d_{ikt} \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{17}$$

$$\sum_{a\in\Omega} p_{aikt} - 2 \geq M(d_{ikt} - 1) \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{18}$$

$$h_{ikt} \geq \theta_{st}.d_{ikt} - \max_{a\in\Omega}(\theta_{ak} \cdot p_{aikt}) \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{19}$$

$$s_{ikt} \geq \min_{a\in\Omega}(\theta_{ak} \cdot p_{aikt}) - \theta_{\min}^k.d_{ikt} \qquad \forall i\in\pi, \forall k\in\omega, \forall t\in X \tag{20}$$

$$ts_i = ss_{ik|WL_{ik}\neq 0} \qquad \forall i\in\pi, \forall k\in\omega \tag{21}$$

$$tf_i \geq sf_{ik} \qquad \forall i\in\pi, \forall k\in\omega \tag{22}$$

$$pf \geq tf_i \qquad \forall i\in\pi \tag{23}$$

$$\Delta pd \geq (pf - L) \tag{24}$$

$$pf, ts_i, tf_i \in X,\ ss_{ik}, sf_{ik} \in \{0,\ldots,h\}\ \forall i \in \pi, \forall k \in \omega \quad (25)$$

$$p_{aikt}, e_{ikt} \in \{0,1\} \quad \forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X \quad (26)$$

$$h_{ikt}, s_{ikt} \in Z, \Delta pd, q_{as} \in Z_{\geq 0}$$
$$\forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X, \forall s \in \Psi \quad (27)$$

The proposed optimization model pursues the minimization of five components as in (1). The first component is a penalty cost charged when the computed duration of the project exceeds its contractual duration. The second component of the objective function penalizes the assignment of two or more workers to a specified skill without considering allocation of an expert among them. This objective has been adopted to assure high work quality, moreover to the benefits gained from knowledge transfer between workers. The third component is a penalty cost awarded when there is a deviation between the actual number of weekly working hours of each worker and the standard number of weekly working hours. The fourth and fifth components together represent the labor cost, respectively during normal working hours and overtime. Constraints set (2) is the assignment constraints that ensure that each worker is assigned to at most one task for performing only one of its skills at any working period. Constraints set (3) restricts the assignment of a worker to a skill that he or she can not practise. Constraints set (4) ensures that at any working day *t* the number of workers allocated to the skill *k* is always lower than or equal to the total number of workers practicing this skill. Constraints set (5) defines the temporal relationships among tasks. Constraints set (6) guarantees that worker *a* must provide the minimum level of efficiency required to practise skill *k*. Constraints set (7) ensures that the workload required for each skill should be satisfied considering the workers' efficiencies as well as the daily working hours of each worker. Constraints sets (8) and (9) ensure that the value of e_{ikt} is set to one if a skill *k* of task *i* is practiced by at least one worker at day *t,* while e_{ikt} is set to zero if a skill *k* of task *i* is not practiced by any worker at day *t*. Constraint (10) relates the completion time of the workload of skill sf_{ik} to e_{ikt}. Because e_{ikt} is one if at least one worker is assigned to skill *k* of task *i* at day *t*, the completion time of a skill sf_{ik} must be equal to maximum *t* at which $e_{ikt}=1$. Similarly, constraints set (11) relates the starting time of the workload of skill ss_{ik} to the variable e_{ikt}. Constraints set (12) ensure that once the skill *k* is started, the execution of this skill continues without preemption until it is completed. Constraints set (13) ensures that the duration of each skill *k* relevant to task *i* must not exceed the maximum allowed duration of task *i*. Constraints set (14) defines the variable f_{as}. Constraints set (15) ensures that the weekly working hours assigned to worker *a* must not exceed the maximum allowed amount of weekly working hours. Constraints set (16) defines the variable q_{as}. This constraint is formulated as an inequality to ensure that q_{as} does not drop beyond zero in case of $f_{as}=0$. Constraints sets (17) and (18) ensure that d_{ikt} is equal to one only when two or more workers are assigned simultaneously to the skill. As stated before, the model considers efficient team building for a skill when this skill is assigned two or more workers simultaneously. Efficient team building implies that when a team (two or more workers) is assigned to a skill, this team includes a highly efficient worker (expert) and one or more less efficient one(s). In order to model this requirements, constraints set (19) defines the variables h_{ikt} which measure the gap between the efficiency of the most efficient worker assigned to skill *k* and θ_{st}. Note that by including the variables h_{ikt} in the objective function, this ensures that at least one of the workers assigned to the skill is a highly efficient one. Similarly, constraint (20) ensures that one or more workers in the team assigned to skill *k* possibly has low efficiency. Constraints set (21) ensures that all skills relevant to task *i* must have the same starting time which itself is the starting time of task *i*. Constraints set (22) defines the completion of the task *i* as the maximum completion time of its relevant skills. Constraints set (23) defines the completion time of the project. Constraints set (24) defines Δpd. Constraints sets (25)-(27) define the domains for decision variables.

The previous model is a mixed integer nonlinear programming model because of the nonlinear terms in constraints (7) and (14), the absolute value in the objective function and maximum and minimum operators in constraints (19) and (20). In order to linearize these nonlinear terms of the model, we define the following auxiliary variables $aw_{aikt}, wf_{as}, st_{ak}$ and zt_{ak}. Thus, the following constraints should be added to the mathematical model:

$$aw_{aikt} \geq c_{at} - M(1-p_{aikt}) \quad \forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X \quad (28)$$

$$aw_{aikt} \leq c_{at} + M(1-p_{aikt}) \quad \forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X \quad (29)$$

$$wf_{as} \geq f_{as} - \gamma_{norm} \quad \forall a \in \Omega, \forall s \in \Psi \quad (30)$$

$$wf_{as} \geq \gamma_{norm} - f_{as} \quad \forall a \in \Omega, \forall s \in \Psi \quad (31)$$

$$st_{ak} \geq \theta_{ak} \cdot p_{aikt} \quad \forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X \quad (32)$$

$$zt_{ak} \leq \theta_{ak} \cdot p_{aikt} + M(1-p_{aikt}) \quad \forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X \quad (33)$$

$$aw_{aikt}, wf_{as} \geq 0 \text{ and integer} \quad \forall a \in \Omega, \forall i \in \pi, \forall k \in \omega, \forall t \in X \quad (34)$$

$$st_{ak}, zt_{ak} \geq 0 \quad \forall a \in \Omega, \forall k \in \omega \quad (35)$$

Therefore, the proposed linear mathematical programming model is as follows:

$$Minimize\, Z = c^1 \cdot \Delta pd + c^2 \sum_{i \in \pi} \sum_{k \in \omega} \sum_{t \in X} (h_{ikt} + s_{ikt}) + \sum_{a \in \Omega} \sum_{s \in \Psi} c_a^3 \cdot wf_{as} + \sum_{a \in \Omega} \sum_{s \in \Psi} c_a^4 \cdot f_{as} + \sum_{a \in \Omega} \sum_{s \in \Psi} c_a^5 \cdot q_{as}$$

Subject to constraints (2)-(6), (8)-(13), (15)-(18), (21)-(27) and (28)-(35) and the new version of constraints (7), (14), (19) and (20) are:

$$\sum_{t \in X} \sum_{a \in \Omega} \theta_{ak} \cdot aw_{aikt} \geq WL_{ik} \quad \forall i \in \pi, \forall k \in \omega \quad (36)$$

$$\sum_{t=\lambda(s-1)+1}^{\lambda s} \sum_{a \in \Omega} \sum_{k \in \omega} aw_{aikt} = f_{as} \quad \forall a \in \Omega, \forall s \in \Psi \quad (37)$$

$$h_{ikt} \geq \theta_{st}.d_{ikt} - st_{ak} \qquad \forall i \in \pi, \forall k \in \omega, \forall t \in X \qquad (38)$$

$$s_{ikt} \geq zt_{ak} - \theta_{\min}^{k}.d_{ikt} \qquad \forall i \in \pi, \forall k \in \omega, \forall t \in X \qquad (39)$$

4. NUMERICAL EXPERIMENTS

To illustrate the validity of this model, we applied it to a simple project provided in the work of Attia *et al.* (2014). The problem was solved using CPLEX™ 12.2 on a PC with 2.3 GHz processor and 4 GB RAM working under windows 7 operating system. This project includes ten tasks, four skills and ten workers. All data about tasks, skills and workers can be found in the paper of Attia *et al.* (2014). The parameters θ_{st} , $\sigma_{\min}^{a}$ and $\sigma_{\max}^{a}$ are set at θ_{st} =0.8, $\sigma_{\min}^{a}$ =4 hours and $\sigma_{\max}^{a}$ =10 hours. Moreover, the contractual duration of the project is 40 days and the values of c^1 , c^2 , c_a^3 , c_a^4 and c^4 are set to 0.1, 0.2, 0.6, 0.6 and 0.1 respectively. Figure 1 shows the optimal solution obtained by the solving process after a running time of about 18 minutes. In figure 1, the vertical and horizontal axes represent the project tasks and the time horizon respectively. The schedule as well as daily working hours of each worker can be read from the top part of the figure. For example, at the first day, skill 2 of task 1 is assigned to worker 1 and worker 9. In addition, working hours of worker 1 and 9 in that day are 4 and 10 hours respectively. As can be noted in figure 1, the actual duration of the project is 40 days, which means that there is no delay compared to the contractual duration (40 days). To go further with the model validation and illustrate its potential as a decision support tool, we solved the problem again with a contractual duration set at 25 days instead of 40 days. Figure 2 shows the optimal solution of the model for a contractual duration of 25 days.

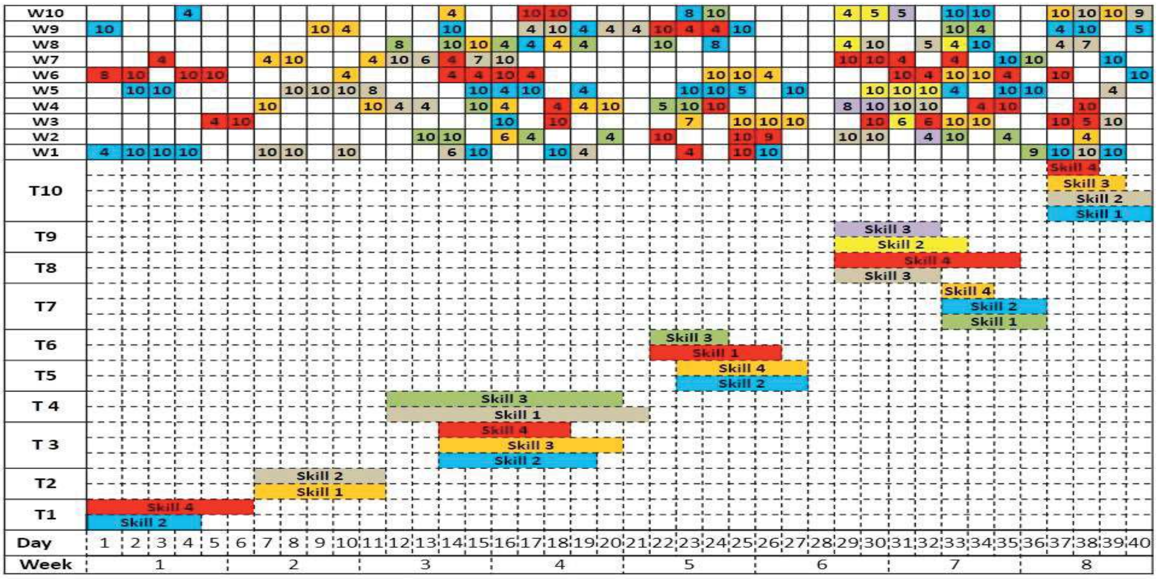

Figure1. Optimal results of the proposed model for a project duration of 40 days.

It can be noted that the project is executed in 25 days without any delay. A comparison between figure1 and figure 2 shows that number of working hours per week of each worker increases. On average, this number increased from 16.7 hours to 26.92 hours. We can also note that in figure1, at most two workers are assigned simultaneously to the same skill (see skill 1 of task 2 at first day) while in figure 2, they may be up to four (see skill 2 of task 7 at day 22). Thus, the average utilization of worker in figure 1 is 33.4% whereas it rises to 54% in figure 2.

As stated before, the model considers developing skills of workers through building teams in which an expert and non-professional worker(s) performing together a given skill. However, in case of 40-days project duration, teams gathering one expert and “non-professional” worker(s) are rarely built (i.e. workers 6 and 7 whose efficiencies with skill 4 are 1 and 0.6 respectively, are assigned to skill 4 of task 3 at day #14). On the other hand, in case of 25-days project duration, these teams are more frequently encountered (i.e. workers 7, 6 and 3 whose efficiencies with skill 4 are 1, 1 and 0.6 respectively, are assigned to skill 4 of task 1 at day 1). Before explaining reasons for these observations, we first illustrate the following points: assigning “non-professional” workers to a skill would increase the execution time of that skill. Therefore, constraint (13) assigns experts to a skill instead of non-professional workers in order to avoid increasing its execution time above $td_i^{\max}$. In addition to constraint (13), the objective function (minimizing total working hours) also prevents the assignment of non-professional workers to reduce total working hours. Based on these understandings, when contractual duration of project is relatively long (40 days), the model mostly assigns workers to skills in which they are experts in order to minimize the total working hours. On the contrary, when contractual duration of project is relatively short (25 days), the model assigns both expert and non-professional workers to the skill in order to accelerate its execution and avoid delaying the project completion. Relying on this discussion, integrating the activities scheduling with the flexibility of human-resources (multi-skilled workers and flexible working hours) enables decision makers to ovoid projects delay in an effective manner.

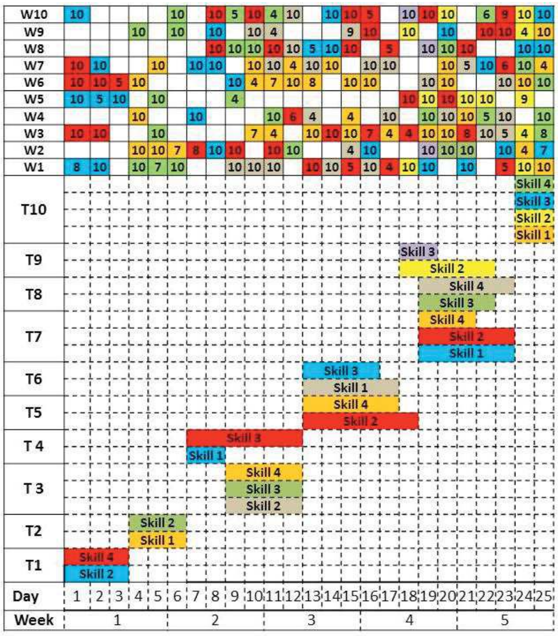

Figure 2. Optimal results of the proposed model for a project duration of 25 days.

5. CONCLUSIONS

The paper studied integrating project scheduling problem with multi-skilled workforce allocation problem. We present two mixed integer programming models: a mixed integer nonlinear programming model, and a linearization which reduces it to a mixed integer linear programming model. The model considers different aspects such as activities' precedence constraints, multi-skills required for each task, flexible working hours and efficient team building that enhances knowledge transfer. The proposed model is applied to a simple project taken from literature. The results show that the model could provide a relevant project schedule, including the starting and completion times of all the project tasks, and simultaneously assignment of workers to skills, with consideration of all operational constraints. In addition, the results indicated that efficient team building of multi-skilled workers is rarely achieved when the contractual duration of project is not too binding. Furthermore, a computing time of somewhat 18 minutes for this quite simple problem (10 tasks, 4 skills, 10 workers) illustrates the need for more rapid methods in case of more complex cases.

As a future expansion of this work, we will integrate the group learning curves that allows a dynamic nature of knowledge transfer between team members. In addition, uncertainty of some of the model parameters can be considered e.g. learning rates of workers, performance level of workers at the project start date, the estimated workloads of activities, or/and workers availability.

REFERENCES

Attia, E.-A., Dumbrava, V., Duquenne, P., 2012. Factors affecting the development of workforce versatility. *IFAC Proceedings Volumes*, 45(6), 1221-1226.

Attia, E.-A., Duquenne, P., Le-Lann, J.-M., 2014. Considering skills evolutions in multi-skilled workforce allocation with flexible working hours. *International Journal of Production Research.* 52(15), 4548–4573.

Baykasoglu, A., Gocken, T., 2010. Multi-objective aggregate production planning with fuzzy parameters. *Advances in Engineering Software*, 41(9), 1124-1131.

Chen, R., Liang, C., Gu, D., 2016. IT project portfolio scheduling and multi-skilled staff assignment with ant colony optimization algorithm. *WHICEB 2016 Proc. Pap. 9.*

Corominas, A., Pastor, R., 2010. Replanning working time under annualised working hours. *Int. J. Prod. Res.* 48, 1493–1515.

Filho, E.V.G., Marçola, J.A., 2001. Annualized hours as a capacity planning tool in make-to-order or assemble-to-order environment: An agricultural implements company case. Prod. Plan. Control 12, 388–398.

Heimerl, C., Kolisch, R., 2010. Scheduling and staffing multiple projects with a multi-skilled workforce. Spectr. 32, 343–368.

Hung, R., 1999. Scheduling a workforce under annualized hours. Int. J. Prod. Res. 37, 2419–2427.

Leung, S.C.H., Chan, S.S.W., 2009. A goal programming model for aggregate production planning with resource utilization constraint. Comput. Ind. Eng. 56, 1053–1064.

Li, H., Womer, K., 2009. Scheduling projects with multi-skilled personnel by a hybrid MILP/CP benders decomposition algorithm. J. Sched. 12, 281–298.

Małachowski, B., Korytkowski, P., 2016. Competence-based performance model of multi-skilled workers. Comput. Ind. Eng. 91, 165–177.

Mencía, R., Sierra, M.R., Mencía, C., Varela, R., 2016. Genetic algorithms for the scheduling problem with arbitrary precedence relations and skilled operators. Integr. Comput.-Aided Eng. 23, 269–285.

Nembhard, D.A., Bentefouet, F., 2015. Selection, grouping, and assignment policies with learning-by-doing and knowledge transfer. Comput. Ind. Eng. 79, 175–187.

Qin, R., Nembhard, D.A., Barnes II, W.L., 2015. Workforce flexibility in operations management. Surv. Oper. Res. Manag. Sci. 20, 19–33.

Tavaghof-Gigloo, D., Minner, S., Silbermayr, L., 2016. Mixed integer linear programming formulation for flexibility instruments in capacity planning problems. Comput. Ind. Eng. 97, 101–110.

Wu, M.-C., Sun, S.-H., 2005. A project scheduling and staff assignment model considering learning effect. Int. J. Adv. Manuf. Technol. 28, 1190–1195.

Yang, K.-K., Webster, S., Ruben, R.A., 2007. An evaluation of worker cross training and flexible workdays in job shops. *IIE - Trans*. 39, 735.

Zhang, R., Zhang, L., Xiao, Y., Kaku, I., 2012. The activity-based aggregate production planning with capacity expansion in manufacturing systems. Comput. Ind. Eng. 62, 491–503.