User project planning in social and medico-social sector: models and solution methods

Yinuo Li[a,b], Jin-Kao Hao[a,*], Brahim Chabane[b]

[a]*LERIA, Université d'Angers, 2 Boulevard Lavoisier, 49045 Angers, France*
[b]*GePI Conseil, Grand Maine, Allée du Grand Launay, 49000 Angers, France*

Expert Systems with Applications, 2021
https://doi.org/10.1016/j.eswa.2021.114684

Abstract

Social and medico-social centers are the main structures in France where different categories of vulnerable populations are hosted. In addition to the daily care, these centers have to ensure the implementation of the personalized project for each resident in response to his/her needs in comply with national legal provisions. This work deals with the main issue of elaborating feasible and thoughtful personalized projects, which aims to improve the whole efficiency of project implementation in social and medico-social centers. A personalized project is composed of a set of activities chosen among available activities proposed by a center. The creation of the personalized projects for the residents requires the satisfaction of a number of imperative constraints while optimizing some objectives concerning the residents and the center. We present a general formulation of this problem and investigate two solution approaches based on mathematical programming and greedy search. This work provides the basis for elaborating a decision support system for efficient user project planning in social and medico-social centers.
Keywords: Project planning; resource assignment; constrained optimization; mathematical programming; heuristics.

1. Introduction

As in all developed countries, the social and medico-social sector in France is experiencing a fast evolution due to the continuing growth of its aging and disabled population. Around 12 million French people (out of 65 million) are affected by disability, 1.7 million are visually impaired and 3.5 million have reduced mobility (INSEE, 2016). These people are looked after in more than

*Corresponding author.
Email addresses: `yinuo.li@etud.univ-angers.fr` (Yinuo Li), `jin-kao.hao@univ-angers.fr` (Jin-Kao Hao), `brahim.chabane85@yahoo.fr` (Brahim Chabane)

34,000 centers of different types (retirement homes, special education centers, work-based support centers, etc, called "structures" hereafter) offering 1.4 million places (French SAH Ministry, 2012).

In addition to the daily care, the main mission of the social and medico-social structures is to provide their residents (called "users" hereafter) with a personalized support in terms of a life project that takes into account professional, social, administrative and health needs of the residents. This is in accordance with legal provisions that reinforce the concept of "personalized user project" which aims to improve the person's life quality with different services.

Indeed, the law No.2002-2 ensures the right for the users of the social and medico-social structures to participate in elaboration and implementation of their "accompanying projects". The law No.2005-102 provides the users with the right of compensation for the disability consequences, in the form of a "personalized compensation plan" which is implemented according to the "life project" of the person, in the sense of the needs and desires that they express.

The personalized user project is defined by a specific document that describes the professional, social and medico-social supports for the user according to his/her needs, expectations and resources. This document also specifies the objectives to achieve, the personalized activities to carry on and the assignment of different roles among various interveners (professionals, family members, etc).

Related official organizations also provide useful guides to help the structures to elaborate their user projects, such as the "Guide to Developing a Personalized Support Project" (CEDIS, 2012)[1] and the "Good professional practice recommendations: User's expectations and personalized project" (ANESM, 2008)[2].

Thereby, the personalized user project aims to:

- Co-construct with the user his/her life project in accordance with his/her resources and the assessment of his needs and expectations.
- Ensure the individualization of each support. This results in the implementation of individual and collective services and activities in line with the resources of the structure and the user's wishes.
- Take into account several aspects of the user's life: social, health, professional and/or educational, etc.

However, the elaboration and the implementation of personalized user projects are complex. Social and medico-social structures receive heterogeneous population with different profiles and needs, which forces the structures to adapt their activities to each user profile and, consequently, to create numerous and varied activities requiring different types of materials and human resources. Moreover, since several years, there has been a sharp drop in resources, in particular staff, on the one hand, and an increase in the number of hosted users, on the other

[1] European committee for the development of social integration.

[2] National agency for the evaluation and quality of social and medico-social centers and services.

hand. As a result, the decision-makers of these structures have to consider various factors and constraints such as varied user needs, resource availability (staff, budget and materials), activity scheduling, assignment, etc. Especially, these structures are typically faced with a lack of decision support tools to help them to make the right decisions.

In the related medical care sector, several studies have been reported especially for routing problems. For instance, Melachrinoudis and Min (2011) developed an integer programming model and a tabu search heuristic to solve the dial-a-ride problem in a center for addictive behavior. Frifita and Masmoudi (2020) designed VNS methods for home care routing and scheduling with temporal dependencies. Fikar and Hirsch (2017) provided an overview of routing and scheduling problems for home care services. Shi et al. (2017) studied a hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand.

In the social and medico-social sector, even if the increased societal demands and the popularization of electronic health records (Zhang et al., 2012; Wang et al., 2019) enable new research opportunities, the current practice of informatics tools mainly concerns dashboard management software which is only useful for some simple decisions (Montoya, 2015; Anderson Fabian, 2019). One exception is the work reported by Chabane et al. (2017) where multiobjective optimization was used to elaborate efficient action plans for social and medico-social structures. However, their work does not concern user project planning. To sum, decision-making tools for personalized user projects are still missing and there is an urgent need for research in this area.

In this work, we fill the gap by presenting the first formulation of the user project planning problem. We also develop solution methods based on mathematical programming and greedy search, which can serve as the basis for building decision-making tools for effectively managing personalized user projects in social and medico-social structures. This work is part of the *"MSUsager"* software developed by the Company GePI[3] for the social and medico-social sector and concerns specially its decision support system.

The rest of the paper is organized as follows. Section 2 provides a literature review of related studies. Section 3 is dedicated to the description of the personalized user project problem, followed by its formulation in Section 3.2. Section 4 describes two solution approaches using mathematical programming and greedy search. Section 5 reports computational experiments on realistic test cases. Finally, we draw conclusions and identify future research perspectives.

2. Literature Review

In this section, we present a literature review on related topics concerning especially project scheduling and planning.

[3]http://www.gepi-conseil.com

From a very general perspective, the personalized user project planning problem studied in this work is related to general scheduling problems, which have various applications such as hospital patient scheduling (Decker and Li, 1998), transport scheduling (Laurent and Hao, 2007), course timetabling (Lü and Hao, 2010), and nurse rostering (Lü and Hao, 2012). For project management, several studies have been reported in different fields including IT-projects scheduling and assigning (Heimerl and Kolisch, 2010), assignment of projects optimizing the utilization of employees' time and expertise in chemical industry (Bassett, 2000) and software maintenance projects selection and scheduling (Ballou and Tayi, 1996). In particular, a classical model of project scheduling called the resource constrained project scheduling problem (RCPSP) (Brucker et al., 1999) has been widely investigated. Extended RCPSP models have also been proposed to cope with special needs (Hartmann and Briskorn, 2010) such as stochastic RCPSP (SCPSP) (Stork, 2001), proactive and reactive RCPSP (Davari and Demeulemeester, 2019), and scheduling and staffing multiple projects with a multi-skilled workforce (Heimerl and Kolisch, 2010). The models in this area are complex, highly constrained and notoriously difficult to solve, but still very attractive because of their wide range of applications in the industry and public services (see the comprehensive surveys on this topic in Hartmann and Briskorn (2010), Herroelen and Leus (2005) and Habibi et al. (2018)).

Within the context of project management, there are several planning levels such as project (task) selection, project scheduling and project staffing (Heimerl and Kolisch, 2010). Enea and Piazza (2004) and Chin et al. (2008) treated the problem of project selection with several critical factors based on the Analytic Hierarchy Process. Al-Harbi (2001) considered the contractor prequalification problem in project management. Moreover, some studies only focused on project scheduling (Servranckx and Vanhoucke, 2019; Davari and Demeulemeester, 2019), while other works considered both project (task) selection and project scheduling (Kosztyán, 2015, 2020; Kosztyán and Szalkai, 2020) and even a mix of portfolio selection, scheduling as well as staff assignment (Gutjahr et al., 2008). Generally, problems with multiple planning levels are more complex and computationally challenging than problems with only one planning level. To cope with the problem complexity, decomposition-based approaches were typically adopted to handle the subproblems in an independent manner.

Compared to the above problems and models, the user project planning problem studied in this work has several particularities, making the problem original in its definition and difficult in terms of problem solving. First, the social and medico-social sector is characterized by its diversity of activities. Since the admitted persons are disabled and to reduce the burden of their family, the structures usually provide a range of services and activities from daily life care to education even medical treatment. This high degree of diversity makes the underlying project planning problem a real challenge for the structures. Second, unlike other project planning problems, the main target of the user project planning problem is not really the projects but the users. Therefore, the objective here is not to arrange as many as possible activities nor to minimize the project duration, but to improve the overall quality of the projects of all users. This requires the decision-maker to consider the benefits that each activity can bring to the users when deciding which activities should be selected. Third, the user project planning problem needs to jointly consider activity selection, project scheduling and resource allocation. Indeed, due to the resource shortages in the social and medico-social sector, the decision results of scheduling and resource allocation could influence the selection of activities for each user. Finally, user projects need to be planned usually for a group of users at the same time, instead of independent users, adding another dimension of planning difficulty. All these features make user project planning a complex and challenging problem.

3. Problem Statement and Formulation

3.1. Personalized user project in the social and medico-social sector

In France, the social and medico-social structures receive people who are disabled, dependent or socially excluded, for a short or long period of time. During the stay period, the structures provide the residents with different kinds of supports to improve their life quality and satisfy their needs of different levels (McLeod, 2007). These supports cover all aspects of life, from daily care, accommodation, catering service, to medical examination and treatment, special education, career planning, even culture and entertainment activities. These comprehensive supports can help residents to improve their independence and reintegration. This diversity of service types implies that these structures have to call for multidisciplinary professionals (e.g., doctors, psychologists, educators, etc.) and resources (rooms, vehicles, medical facilities, etc.) that need to be coordinated to achieve the elaboration and implementation of users' projects.

Social and medico-social actions are based on the provision of services connected to a continuous assessment of the needs and desires of the users. These actions are conducted with respect to the equal dignity of all residents with the aim of responding in a suitable manner to the needs of each user. According to the population received, these structures are divided into different categories and provide very different services. To ensure a smooth operation of a structure, the actions and services to be implemented will be decided in the structure action plan which is assessed and improved continuously (Chabane et al., 2017).

Unlike other planning and scheduling problems such as nurse rostering (Lü and Hao, 2012), hospital patient scheduling (Decker and Li, 1998), university course planning (Lü and Hao, 2010) and project scheduling (Hartmann and Briskorn, 2010; Heimerl and Kolisch, 2010; Davari and Demeulemeester, 2019), a personalized project in the social and medico-social sector implies that the services and resources allocated to a particular user may be very different from the other users and depends on his/her vulnerability situation, family situation, personal desires, finance situation, etc. This particularity makes the elaboration of the personalized project of a user very complex, which becomes even more difficult when a group of users must be considered simultaneously.

According to the guide issued by the European Committee for the Development of Social Integration (CEDIS, 2012), there are three main phases in a personalized user project (Figure 1).

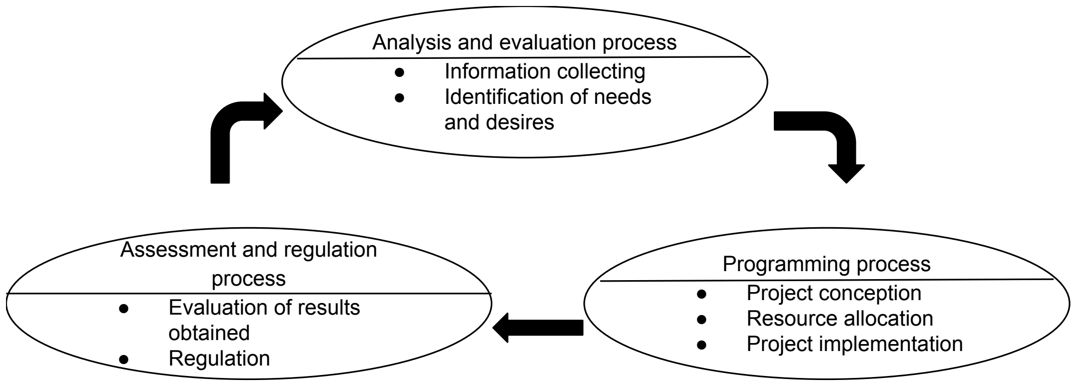


Figure 1: The three main phases of the personalized user project

- *Analysis and evaluation process* This phase aims to identify user's global desires, abilities and needs. The tasks of this phase are usually implemented by qualified professionals to perform special measures such as personal interview, physical and mental test, capacity evaluation, etc. This phase provides necessary information to guide the selection of services and activities for each user.

- *Programming process* This phase is the main phase of project. According to the information collected in the first phase, decision-making team has to draft user's project by specifying the objectives to achieve and the activities to carry on during a period, while ensuring that the project is feasible by considering resource and availability constraints of the structure and the user. Once the project is elaborated and validated, the user can start executing his/her project.

- *Assessment and regulation process* This phase is carried out at the end of the project. It aims to assess the progress of every activity involved in the project, then make the necessary adjustment and provide information for the next project.

This work focuses on the second phase, i.e., the programming process. From the first phase, we can obtain a preliminary estimation of activity preference for the user according to his/her needs and personal desires. The activity preference can be understood by the benefits for a user to participate in an activity, which is certainly user specific. This information is the most important indicator for decision-makers because the primary goal of the structure is to help users to improve their daily life. In addition, the differences between activities are reflected in many aspects such as cost, individual activity or group activity, professionals (doctors, educators, drivers, etc.), necessary facilities (bus, room, etc), etc. In general, during the project elaboration, decision-makers are confronted with three questions: i) which activities or services should be implemented for the user, according to his/her needs; ii) how to schedule these activities during the project period; iii) which resources should be allocated to satisfy implementation conditions of each activity. These questions are tightly coupled, and the decision for one of them may influence the others.

This complexity does bring difficulties to the daily work in these structures. In the structure organization, the design of an operational planning is typically realized by a qualified professional or a team who must take account into many factors and unexpected changes. Additionally, the communication cost of this work is very important especially when no digital tools can be used.

Thus, the goal of this work is to propose a model and a tool able to help the decision-maker to elaborate efficient user projects while respecting a set of constraints.

3.2. Constraints and objectives

To introduce the constraints and the objectives of the user project planning problem, we use the following notations that are summarized in Table 1.

- A set $\mathcal{U}$ of users ($U = |\mathcal{U}|$), each being characterized by his/her budget, availability and preference for each activity. According to our survey mentionned in Section 5.2, there are several tens of users in a typical structure.
- A set $\mathcal{A}$ of activities ($A = |\mathcal{A}|$), each being characterized by its price, duration (number of timeslots needed), capacity, availability and features required. Typical activities include medical tests and treatments, entertainment activities (sport, trip, museum visiting), professional training, etc.
- A set $\mathcal{R}$ of resources ($R = |\mathcal{R}|$), including human and facility resources, each being characterized by its availability and features. Typical resources include rooms, vehicles, medical professionals, educators, etc.
- A set $\mathcal{T}$ of timeslots ($T = |\mathcal{T}|$), during which users, activities and resources are to be scheduled. For example, if we take 8 timeslots per day (typically, timeslot equals one hour) and 5 working days per week, then a project of 4 weeks has 160 timeslots.

To formulate the user project planning problem, we represent a candidate solution (i.e., a user project schedule) S by a $U \times T$ matrix $\mathbf{X}$ and a $R \times T$ matrix $\mathbf{Y}$ where

$$x_{ut} = \begin{cases} a & \text{user } u \text{ participates in activity } a \text{ at time } t, \\ -1 & \text{user } u \text{ is not available at time } t \ (ua_{ut} = -1), \\ 0 & \text{user } u \text{ is available at time } t \end{cases}$$

Table 1: Settings of important parameters

Symbol	Description
$\mathcal{U}$	Set of users, $\lvert\mathcal{U}\rvert = U$
$\mathcal{A}$	Set of activities, $\lvert\mathcal{A}\rvert = A$
$\mathcal{R}$	Set of resources, $\lvert\mathcal{R}\rvert = R$
$\mathcal{T}$	Set of timeslots, $\lvert\mathcal{T}\rvert = T$
$\mathcal{F}$	Set of features, $\lvert\mathcal{F}\rvert = F$
TS	Number of timeslots per day
ua_{ut}	Whether user u is available at timeslot t, $ua_{ut} = 0$ if available, -1 otherwise
bud_i	Budget of user i
aa_{at}	Whether activity a is available at timeslot t, $aa_{at} = 0$ if available, -1 otherwise
dur_a	Duration of activity a
pri_a	Price of activity a
cap_a	Capacity of activity a
ra_{rt}	Whether resource r is available at timeslot t, $ra_{rt} = 0$ if available, -1 otherwise
pre_{ua}	Preference of user u to activity a
req_{af}	Whether activity a requires feature f , $req_{af} = \{0, 1\}$
fea_{rf}	Whether resource r contains feature f , $fea_{rf} = \{0, 1\}$
$num\_u_{at_k}$	Number of users starting activity a at timeslot t_k in a candidate solution S; $num\_u_{at_k} = \sum_{u\in\mathcal{U}} sta_{uat_k}$, where $sta_{uat_k} = \begin{cases} 1 & \text{if } x_{ut_k} = a \wedge (x_{ut_{k-1}} \neq a \vee k = 1), \\ 0 & \text{otherwise.} \end{cases}$
act_{ua}	Whether user u participates in activity a in a candidate solution S, $act_{ua} = 1$ if $\exists x_{ut} \in S$, $a = x_{ut}$, $act_{ua} = 0$ otherwise.
res_{at}	Set of resources used by activity a with start-time t in a candidate solution S $res_{at} = \{r \mid r \in \mathcal{R}, y_{rt} = a\}$

$$y_{rt} = \begin{cases} a & \text{resource } r \text{ used by activity } a \text{ started at time } t \text{ ,} \\ -1 & \text{resource } r \text{ is not available at time } t \ (ra_{rt} = -1), \\ 0 & \text{resource } r \text{ is available at time } t \end{cases}$$

Below, we describe the problem constraints and the optimization objectives, followed by their mathematical formulation.

3.2.1. Constraints

To establish a feasible user project, a number of constraints must be satisfied.

- *User availability* (C_1): A user cannot be scheduled when he/she is not available. Moreover, a user cannot participate in two activities at the same time.

  It is worth noting that the solution representation above ensures that a user can only participate in at most one activity per timeslot, and will not be assigned any activity when the user is not available (preset to -1). In other words, the first constraint *User availability* (C_1) is satisfied by the solution representation.

- *Activity availability* (C_2): An activity cannot be scheduled for users when it is not available and cannot cross more than one day. Indeed in the social and medico-social structures, it may happen that some activities can be arranged only at certain timeslots, especially when external resources are needed.

$$\forall u \in \mathcal{U}; \forall a \in \mathcal{A}; \forall t \in \mathcal{T}; (aa_{at} = 0) \vee (x_{ut} \neq a)$$

- *Resource availability* (C_3): A resource cannot be scheduled when it is not available and cannot be used by two activities at the same time.

$$\forall r \in \mathcal{R}; \forall t_i \in \mathcal{T}; \forall a \in \mathcal{A}; (y_{rt_i} \neq a) \vee (\bigcup_{d=i+1}^{i+dur_a-1} y_{rt_d} = \{0\})$$

- *Budget constraint* (C_4): The cost of the selected activities in a user's project cannot exceed user's budget.

$$\forall u \in \mathcal{U}; \sum_{a \in \mathcal{A}} act_{ua} \times pri_a \leq bud_u$$

- *Activity capacity* (C_5): For each activity starting at a given timeslot, the number of users attending this activity cannot bypass the capacity of this activity. This constraint is defined for group activities to ensure the activity quality.

$$\forall a \in \mathcal{A}; \forall t \in \mathcal{T}; num\_u_{at} \leq cap_a$$

- *Feature constraint* (C_6): An activity cannot be scheduled if one or more required resources are unavailable. In social and medico-social structures, coordination between various disciplines is required for most of activities, and an activity cannot be carried out when some required features are not satisfied. For example, if the painting workshop needs a room of at least 20 places and professional(s) with art skills and special education skills, then the resources allocated should cover all these features.

$$\forall a \in \mathcal{A}; \forall t \in \mathcal{T}; \forall f \in \mathcal{F}; (num\_u_{at} = 0 \vee req_{af} = 0) \vee \sum_{r \in res_{at}} fea_{rf} > 0$$

  This multi-feature constraint is specific to the social and medico-social structures, and to our knowledge, has not been considered in other project planning problems.

- *Preference constraint* (C_7): A user must indicate a preference for each intended activity. This constraint is defined especially for large structures which propose various services. Unlike classic project planning problems where activities are fixed or all selectable, in social and medico-social structures, some activities could be identified as unsuitable for some users after a first analysis and evaluation phase. So we add this non-selectable constraint to prevent selecting unsuitable activities for users.

$$\forall u \in \mathcal{U}; \forall a \in \mathcal{A}; (act_{ua} = 0) \vee (pre_{ua} > 0)$$

To keep our model compact, we assume that no relation exists between activities and a user can only participate in an activity at most once during the given period.

3.2.2. Objectives

The user project planning problem has one main objective and two secondary objectives.

Main objective: suitability

The main objective of a personalized user project is to provide the best possible response for each need or expectation of the user. The quality of each response is assessed by the professional and the user (or his/her family), according to the impacts the response could have on the life of the user. However, due to the constraints above, it would be impossible to satisfy all the preferable activities for every user. Consequently, the first goal is to find a solution that covers as many as possible preferable activities for the users.

Suitability maximization: Maximize user satisfaction in terms of the preferences of activities implemented for them.

$$f_1(S) = \sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{A}} act_{ua} \times pre_{ua}$$

Secondary objectives: costs

It is desirable to reduce the costs during project implementation for evident economic reasons and unexpected variation. For the structure, the cost corresponds to the utilization of its resources, while for each user, the cost corresponds mainly to the economic expenditure for the chosen activities.

Structure resource minimization: Minimize the use of resources to save costs for the structure, which is equivalent to maximize the timeslots of the resources that are not utilized.

$$f_2(S) = R \times T - \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} dur_{y_{rt}} + \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} ra_{rt}$$

User cost minimization: Minimize the cost of the planned activities to save expenses for the users, which is equivalent to maximize the unconsumed budget of each user.

$$f_3(S) = \sum_{u \in \mathcal{U}} (bud_u - \sum_{a \in \mathcal{A}} act_{ua} \times pri_a)$$

Even if different types of structures have distinctive operational modes, the above constraints and objectives are shared by all structures and constitute the basis for building our user project planning decision-making system.

4. Solution Approach

Based on the above problem description and formulation, we propose two mathematical programming models and run a general tool (CPLEX MIP Solver) to solve these models. We additionally experiment a greedy heuristic and compare it with the mathematical programming approach.

4.1. Integer programming

We firstly create an integer programming model where we use decision variables similar to those described in Section 3.2. The decision variables take their values in an integer interval representing the activity. This model has the advantage of requiring a reduced number of decision variables and easing the expression of the constraints.

Decision variables

$$x_{ij} = \begin{cases} k \in \{1, \dots, A\} & \text{activity } a_k \text{ started by user } i \text{ at timeslot } j, \\ -1 & \text{user } i \text{ is not available at time } j \ (ua_{ij} = -1), \\ 0 & \text{otherwise} \end{cases}$$

$$y_{ij} = \begin{cases} k \in \{1, \dots, A\} & \text{activity } a_k \text{ with resource } i \text{ started at timeslot } j, \\ -1 & \text{resource } i \text{ is not available at time } j \ (ra_{ij} = -1), \\ 0 & \text{otherwise} \end{cases}$$

$$Maximize \quad f(S) = w_1 f_1(S) + w_2 f_2(S) + w_3 f_3(S) \tag{1}$$

where

$$f_1(S) = \sum_{u \in \mathcal{U}} \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} (x_{ut} == a) \times pre_{ua} \tag{2}$$

$$f_2(S) = \sum_{u \in \mathcal{U}} bud_u - \sum_{u \in \mathcal{U}} \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} (x_{ut} == a) \times pri_a \tag{3}$$

$$f_3(S) = R \times T - \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} (y_{rt} == a) \times dur_a + \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} ra_{rt} \tag{4}$$

Subject to

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, x_{ut} = -1, if\, ua_{ut} = -1 \tag{5}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t+1, \ldots, t+dur_a-1\}, \\ x_{ud} = 0, if\, x_{ut} = a \tag{6}$$

$$\forall a \in \mathcal{A}, \forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall d \in \{t-dur_a+1, \ldots, t\}, \\ x_{ut} \neq a, if\, aa_{at} = -1 \tag{7}$$

$$\forall a \in \mathcal{A}, \forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall d \in \{t-dur_a+1, \ldots, t\}, \\ y_{rt} \neq a, if\, aa_{at} = -1 \tag{8}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t-dur_a+1, \ldots, t\}, \\ ((x_{ut} == a) \times ((t \div TS) == ((t+dur_a-1) \div TS)) > 0) \vee (x_{ut} \neq a) \tag{9}$$

$$\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t-dur_a+1, \ldots, t\}, \\ ((y_{rt} == a) \times ((t \div TS) == ((t+dur_a-1) \div TS)) > 0) \vee (y_{rt} \neq a) \tag{10}$$

$$\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, y_{rt} = -1, if\, ra_{rt} = -1 \tag{11}$$

$$\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t+1, \ldots, t+dur_a-1\}, \\ y_{rd} = 0, if\, y_{rt} = a \tag{12}$$

$$\forall u \in \mathcal{U}, \forall a \in \mathcal{A}, \sum_{t \in \mathcal{T}} (x_{ut} == a) \leq 1 \tag{13}$$

$$\forall u \in \mathcal{U}, \forall a \in \mathcal{A}, \sum_{t \in \mathcal{T}} (x_{ut} == a) \times pri_a \leq bud_u \tag{14}$$

$$\forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \sum_{u \in \mathcal{U}} (x_{ut} == a) \leq cap_a \tag{15}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall f \in \mathcal{F}, \\ (x_{ut} == a) \times req_{af} - \sum_{r \in \mathcal{R}} (y_{rt} == a) \times fea_{rf} \leq 0 \tag{16}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, x_{ut} \neq a \vee (x_{ut} == a) \times pre_{ua} > 0 \tag{17}$$

where the " $==$ " term is the equivalence linear constraint which behaves like a binary condition such that it returns 1 or 0 according to whether the given term is true and false.

Equation (1) is the objective function combining sub-objectives (2)-(4). Expressions (5) and (6) formulate constraint C_1. Expressions (7)-(10) model constraint C_2, Expressions (11) and (12) formulate constraint C_3. Expression (13) ensures that a user can only participate in an activity at most once. Expressions (14)-(17) formulate constraints C_4, C_5, C_6, C_7, respectively.

4.2. 0/1 programming

Defining decision variables with binary values is very popular in mathematical modeling. We also experiment this modeling method for our user project planning problem. Compared to the integer programming model, the 0/1 programming model requires more decision variables and needs more mathematical expressions to describe the problem constraints.

Decision variables

- x_{ijk} Binary variable taking value 1 if user i starts activity k at timeslot j, and 0 otherwise.
- y_{ijk} Binary variable taking value 1 if resource i starts to be used by activity k at timeslot j, and 0 otherwise.

$$Maximize \quad f(S) = w_1 f_1(S) + w_2 f_2(S) + w_3 f_3(S) \tag{18}$$

where

$$f_1(S) = \sum_{u \in \mathcal{U}} \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} x_{uta} \times pre_{ua} \tag{19}$$

$$f_2(S) = \sum_{u \in \mathcal{U}} bud_u - \sum_{u \in \mathcal{U}} \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} x_{uta} \times pri_a \tag{20}$$

$$f_3(S) = R \times T - \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} y_{rta} \times dur_a + \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} ra_{rt} \tag{21}$$

Subject to

$$\begin{aligned}&\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t - dur_a + 1, \ldots, t\},\\ &x_{uda} = 0, if\ ua_{ut} = -1\end{aligned} \tag{22}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \sum_{a \in \mathcal{A}} x_{uta} \leq 1 \tag{23}$$

$$\begin{aligned}&\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t + 1, \ldots, t + dur_a - 1\},\\ &x_{uda} = 0, if\ x_{uta} = 1\end{aligned} \tag{24}$$

$$\begin{aligned}&\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t - dur_a + 1, \ldots, t\},\\ &x_{uda} = 0, if\ aa_{at} = -1\end{aligned} \tag{25}$$

$$\begin{aligned}&\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t - dur_a + 1, \ldots, t\},\\ &y_{rda} = 0, if\ aa_{at} = -1\end{aligned} \tag{26}$$

$$\begin{aligned}&\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t - dur_a + 1, \ldots, t\},\\ &(x_{uta} \times ((t \div TS) == ((t + dur_a - 1) \div TS)) > 0) \vee x_{uta} = 0\end{aligned} \tag{27}$$

$$\begin{aligned}&\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t - dur_a + 1, \ldots, t\},\\ &(y_{rta} \times ((t \div TS) == ((t + dur_a - 1) \div TS)) > 0) \vee y_{rta} = 0\end{aligned} \tag{28}$$

$$\begin{aligned}&\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t - dur_a + 1, \ldots, t\},\\ &y_{rda} = 0, if\ ua_{ut} = -1\end{aligned} \tag{29}$$

$$\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall d \in \{t+1, \ldots, t+dur_a-1\}, \\ y_{rda} = 0, if\, y_{rta} = 1 \tag{30}$$

$$\forall r \in \mathcal{R}, \forall t \in \mathcal{T}, \sum_{a \in \mathcal{A}} y_{rta} \leq 1 \tag{31}$$

$$\forall u \in \mathcal{U}, \sum_{t \in \mathcal{T}} \sum_{a \in \mathcal{A}} pri_a \times x_{uta} \leq bud_u \tag{32}$$

$$\forall a \in \mathcal{A}, \forall t \in \mathcal{T}, \sum_{u \in \mathcal{U}} x_{uta} \leq cap_a \tag{33}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, \forall f \in \mathcal{F}, x_{uta} \times req_{af} - \sum_{r \in \mathcal{R}} y_{rta} \times fea_{rf} \leq 0 \tag{34}$$

$$\forall u \in \mathcal{U}, \forall t \in \mathcal{T}, \forall a \in \mathcal{A}, x_{uta} = 0 \vee x_{uta} \times pre_{ua} > 0 \tag{35}$$

$$\forall u \in \mathcal{U}, \forall a \in \mathcal{A}, \sum_{t \in \mathcal{T}} x_{uta} \leq 1 \tag{36}$$

Equation (18) is the objective function combining sub-objectives (19)-(21). Formulas (22)-(24) express constraint C_1, formulas (25)-(28) define constraint C_2, expressions (29)-(31) model constraint C_3. Constraint (36) ensures that a user can only participate in an activity at most once. Formulas (32)-(35) are expression of constraints C_4, C_5, C_6, C_7, respectively.

4.3. Constructive greedy heuristic

The constructive greedy procedure (Algorithm 1) generates always a feasible solution satisfying all constraints. Following the spirit of the constructive greedy initialization in (Lü and Hao, 2010) and (Laurent and Hao, 2007), activities with a high preference are given higher priorities to be selected. The construction of solution is implemented with several rounds of activity arrangements for users. In every round, users are traversed according to the increasing order of budgets, users with low budgets are more vulnerable to budget constraint so similar to the solidarity consideration in real-life, users with financial difficulties are given priority.

Specifically, for each user under consideration, the first step is to select an activity to arrange. To response to the main suitability objective, we always take the most preferred eligible activity, which means that this activity has not been arranged and satisfies the budget constraint and the preference constraint. Then, we consider scheduling and resource allocation for this activity. To arrange this activity for the user, we consider two cases. First, to ensure an efficient use of the resources, we verify whether the same activity has been already arranged for other users and still has available places. If the answer is affirmative, the user will be added directly to the activity in the first available timeslot during the period (lines 12-19). In this case, no new resources need to be allocated. If the first case is impossible, we allocate the required resources to this activity, and arrange the activity at the first timeslot having enough available resources (lines 20-32). The selection of resources is achieved by iterating through the available resources and taking the resources with the required features until all features are satisfied. If this activity cannot be arranged in either way, the algorithm skips this activity and moves to the next preferred activity. Finally, once an activity is successfully arranged for this user or all activities have been verified as unschedulable, the algorithm moves to the next user. We only arrange at most one activity for each user in a round in order to allocate resources more fairly and prevent users who are first considered from occupying too many resources. When the algorithm finishes the activity arrangement for the last user, it returns to the first user to start a new round of arrangements. The algorithm stops when all the activities of all users have been considered and no more activity can be arranged without violating constraints.

During the constructive greedy procedure, the feasibility of each decision is first verified and then executed. If the decision cannot satisfy all constraints, it will not be performed. As such, the greedy procedure always provides a feasible solution satisfying all the constraints of the problem. On the other hand, as explained above, the algorithm considers the users and assigns activities sequentially and according to some specific orders while ignoring alternative decisions. Consequently, the decisions made during the solution construction are not necessarily optimal, leading typically to sub-optimal solutions.

Finally, for each user and activity, since the decision of activity arrangement is verified by checking each timeslot and resource, the time complexity of this constructive greedy procedure is bounded by $\mathcal{O}(U*A*T*R)$ where U, A, T and R represent the number of users, activities, timeslots and resources, respectively.

Algorithm 1: Greedy heuristic

Input: Problem instance I

Output: Initial solution S

```
/* Sort the users of set $\mathcal{U}$ by increasing order of their budgets */
sort_increasing($\mathcal{U}$)
repeat
    for $u \in \mathcal{U}$ do
        /* Sort the activities of set $\mathcal{A}$ by decreasing order of preferences of $u$ */
        sort_decreasing($\mathcal{A}$)
        for $a \in \mathcal{A}$ do
            actArranged ← 0
            if a is not arranged for u and $(u, a)$ satisfies $C_4$, $C_7$ then
                for $t \in \mathcal{T}$ do
                    if a is already arranged at start-time t and $(u, a, t)$ satisfies $C_1$, $C_5$ then
                        /* Arrange a for u at t */
                        arrange$(u, a, t)$
                        actArranged ← 1
                        break
                    end
                end
                if actArranged = 0 then
                    for $t \in \mathcal{T}$ do
                        if $(u, a, t)$ satisfies $C_1$, $C_2$, $C_5$ then
                            /* Check if $C_6$ can be satisfied for $(u, a, t)$, if yes, return resources selected as $\mathcal{R}^*$ */
                            if select_resources($\mathcal{R}^*$) then
                                /* Arrange a for u at t */
                                arrange$(u, a, t)$
                                actArranged ← 1
                                break
                            end
                        end
                    end
                end
                if actArranged = 1 then
                    break
                end
            end
        end
    end
until no more activity can be arranged without violating problem constraints
```

5. Computational experiments and Results

5.1. Evaluation function

As mentioned above, the quality of a candidate solution S is evaluated by the weighted sum method as follows.

$$f(S) = \sum_{o \in \{1,2,3\}} w_o \times f_o(S)$$

The weight w_o associated to the oth objective function is computed by

$$w_o = \alpha_o \times \frac{1}{f_o^{max}}$$

with

- α_o the importance coefficient to the oth objective function specified by the decision-maker. To easy the use of the decision-making tool, we predefine five default levels for the decision-maker to choose (i.e., $\alpha \in [1, 5]$).
- $f_o^{max} = max\{f_o(S) : S \in \mathcal{S}\}$, which represents the maximum value of the oth objective in the whole solution space.

The purpose of this function transformation is to apply a normalization to the objectives which are different in nature and scale. Given that all sub-objective functions are positive, we choose the upper-bound approach (Proos et al., 2001) as the transformation method.

$$F_i^{trans} = \frac{F_i(x)}{|F_i^{max}|}$$

This transformation ensures that our transformed sub-objective function has a value between zero and one. To determine F_i^{max}, the most recommended method is to optimize separately each objective function (Grodzevich and Romanko, 2006). However, this approach is too computationally expensive. In our case, we estimate F_i^{max} as follows:

- $f_1^{max} = \sum_{u \in \mathcal{U}} \sum_{a \in \mathcal{A}} pre_{ua}$;
- $f_2^{max} = R \times T + \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} ra_{rt}$;
- $f_3^{max} = \sum_{u \in \mathcal{U}} bud_u$.

Other transformation methods are possible as discussed in (Marler and Arora, 2005, 2004). However, the adopted transformation has the advantage of being simple and efficient for our application.

5.2. Test instances

To our knowledge, there are no publicly available data for the user project planning problem studied in this work. For our experimental study, we created a set of benchmark instances as follows. We sent a large-scale survey to more than 3600 structures in the social and medico-social sector in France. From the responses received (more than 200), we created an instance generator that is parametrized to generate problem instances covering various real scenarios in different structures. For our experimental study, we generated 20 test instances with different sizes and characteristics to fit as much as possible typical real situations[4]. Table 2 shows the main features of these instances with their parameter values where

- Ins: Instance number;
- U: Number of users;
- R: Number of resources;
- A: Number of activities;
- F: Number of features;
- cap: Maximum capacity of activities.

Table 2: Parameters of the test instances

Ins	U	R	A	TS	W	F	act_cho%	ua%	ra%	aa%	cap	Structure size	Scheduling length
0	10	5	10	2	2	5	100	100	100	100	5	Small	Short
1	10	8	15	8	1	5	100	100	100	100	5	Small	Short
2	10	8	15	4	2	5	100	100	100	100	5	Small	Short
3	30	10	20	4	4	5	100	100	100	100	5	Medium	Medium
4	30	20	30	2	4	5	100	100	100	100	5	Medium	Medium
5	30	20	40	2	4	5	100	100	100	100	5	Medium	Medium
6	50	20	15	8	1	10	100	100	100	100	10	Medium	Short
7	50	30	20	4	2	10	100	100	100	100	10	Medium	Short
8	50	30	30	2	4	10	100	100	100	100	10	Medium	Medium
9	50	20	30	2	4	10	100	100	100	100	10	Medium	Medium
10	50	30	50	2	2	10	100	100	100	100	10	Medium	Short
11	50	30	40	4	4	10	100	100	100	100	10	Medium	Medium
12	50	30	60	2	8	10	100	100	100	100	10	Medium	Long
13	50	30	80	4	8	10	80	100	100	100	10	Medium	Long
14	50	30	100	4	12	10	80	95	95	95	10	Medium	Long
15	70	30	30	2	2	10	100	100	100	100	10	Large	Short
16	70	40	40	2	4	20	100	100	100	100	10	Large	Medium
17	70	40	50	4	4	20	80	95	95	95	10	Large	Medium
18	100	50	30	2	2	20	100	100	100	100	10	Large	Short
19	100	70	50	2	4	30	800	95	95	95	10	Large	Medium

[4]We make these instances publicly available at: `http://www.info.univ-angers.fr/pub/hao/userprojectplanning.html`

The values of the above parameters characterize the structure size. And larger structures have more users and of course more resources and activity types. We classify the instances into three categories according to the structure size: 10 users (small structure), 30/50 (medium structure) and 70/100 users (large structure). These instances cover different possibilities of resource adequacy levels and the distribution matches the realistic results according to the survey that most of structures are of medium size.

- TS: Number of timeslots per day;
- W: Number of weeks, hence $T = TS \times W \times 5$;

These two parameters determine the period length required for scheduling, which is different according to the structure type. The instances cover period lengths from one week to 12 weeks.

- act_cho%: Percentage of activities that can be selected;
- ua%: Percentage of user's available timeslots;
- ra%: Percentage of resource's available timeslots;
- aa%: Percentage of activitie's available timeslots;

These four parameters indicate that some activities and timeslots are not selectable. According to our survey, this may happen especially for large structures or long period projects. Even though in most cases, all activities and timeslots are considered selectable during the conception phase, to ensure our model can handle this real situation, we decide to add some unselectable cases for some instances with large structure size or long project period. The constraints C_1-C_3 ensure that users, resources and activities will not be arranged when they are unavailable and the constraint C_7 ensures that the non-selectable activities are never selected into users' projects.

The other parameters:

- Number of features per resource has: $1 \sim 3$ (uniform distribution)
- Number of features per activity needs: $1 \sim 5$ (uniform distribution)
- Activity price: $0 \sim 10$ (truncated normal distribution $\mu = maxPrice/2, \sigma = (maxPrice/2 - minPrice)/3$)
- User budget: $maxPrice/2 * activityNumber/2 \sim maxPrice * activityNumber$ (truncated normal distribution $\mu = maxBudget/2, \sigma = (maxBudget/2 - minBudget)/3$)
- Preference Level: $0 \sim 10$ (uniform distribution)

To cover as many situations as possible, no two instances have the same parameter settings.

Table 3: Experimental results and comparison

Instance	$\{\alpha\}$	IP			0/1P			GHA	
		LB	UB	GAP	LB	UB	GAP	Obj	Time (s)
0	{5,3,0}	4.1142	15.4908	276.52%	**4.8033**	4.8068	0.074%	4.0623	0.0003
1	{5,2,1}	2.0186	78.7249	3799.94%	**4.9300**	5.2124	5.73%	4.2060	0.0008
2	{5,1,2}	4.8469	10.1700	109.82%	**5.4966**	5.5445	0.87%	5.1804	0.0003
3	{5,1,1}	-	-	-	**4.7178**	5.4622	15.78%	4.2480	0.0079
4	{5,3,0}	-	-	-	**5.2055**	7.2272	38.84%	3.9318	0.0059
5	{5,1,2}	-	-	-	4.1436	5.9389	43.33%	**4.1916**	0.0123
6	{5,2,1}	-	-	-	**4.3155**	5.6144	30.10%	3.5143	0.0136
7	{5,3,0}	-	-	-	4.3039	130.5689	2933.77%	**5.0879**	0.0051
8	{5,2,1}	-	-	-	4.1035	119.8726	2821.22%	**4.7085**	0.0140
9	{5,2,1}	-	-	-	4.2911	123.7569	2783.99%	**4.7456**	0.0103
10	{5,1,1}	-	-	-	2.7762	4.0878	47.24%	**2.8983**	0.0110
11	{5,1,1}	-	-	-	-	-	-	**4.2377**	0.0879
12	{5,1,1}	-	-	-	-	-	-	**4.5686**	0.1007
13	{5,1,2}	-	-	-	-	-	-	**5.4005**	0.1621
14	{5,3,0}	-	-	-	-	-	-	**4.8303**	0.6189
15	{5,1,2}	-	-	-	3.4143	5.3128	55.60%	**3.6945**	0.0197
16	{5,3,0}	-	-	-	-	-	-	**3.8128**	0.3053
17	{5,1,2}	-	-	-	-	-	-	**4.6198**	0.2827
18	{5,2,1}	-	-	-	-	-	-	**3.5620**	0.0678
19	{5,1,1}	-	-	-	-	-	-	**4.2342**	0.5341

5.3. Computational Results

The experiments were carried on a computer with an Intel Core i7-8750H 2.20GHz processor and 32 GB RAM under Linux. The integer programming model and the 0/1 programming model are solved by running the CPLEX MIP solver (Version 12.8) with the MemoryEmphasis parameter as true and other default parameters under a time limit of 2 hours. The proposed greedy algorithm was implemented in C++ and compiled using the g++ compiler with the -O3 option. The greedy algorithm has no parameter to tune and stops when no more activities can be scheduled. For the importance coefficient α_o, we collected four combinations recommended by professionals in the social and medico-social sector and then distributed them randomly to 20 test instances.

Table 3 shows computational results on the 20 instances obtained by CPLEX with the two mathematical models and results from the greedy algorithm. Column 2 gives the $\{\alpha\}$ used for an instance. Columns 3 to 8 show the results of the integer model (IP) and the 0/1 model (0/1P) in terms of the lower bound, upper bound and gap achieved by the CPLEX solver. Columns 9 and 10 present the results and run time of the greedy algorithm (GHA). The best values among the results of the compared approaches are highlighted in boldface. Entries with "-" mean that CPLEX failed to output a result due to one of three reasons: i) out of memory; ii) unknown solution status; iii) the pre-processing phase of CPLEX cannot be finished in two hours.

Table 3 shows that the 0/1 programming model led to much better results than the integer programming model. This difference comes from the use of logical constraints in the integer programming model, as described in 4.1, to express certain constraints. Indeed, the equivalence relation "==" is used as a term in a numeric expression to behave like a binary value, these nonlinear expressions weakened the performance of the model.

The results of the 0/1 programming model is representative. Even if it does not give the optimal solution for any instance, for the small scale instances (such as instance 0, 1, 2, etc), the model obtains near optimal solutions, as shown by the tiny gaps between the lower bound and the upper bound. But as the instance scale increases, the solution quality is greatly worsen. Indeed, the solutions for the instances 7-10, 15 are feasible, but far from optimality with large gaps between the lower bound and the upper bound. For the remaining large instances, no feasible solution was found by CPLEX with the two mathematical models within 2 hours. We notice that the instance 15 contains 70 users and was solved by CPLEX, but the instances 11-14 with 50 users were not solved. This can be explained by the fact that even if the instance 15 has more users than the instances 11 to 14, it has a shorter scheduling length, fewer timeslots (TS) and fewer weeks (W) (see Table 2). This implies that the mathematical model for the instance 15 has much fewer decision variables (60000 against 256000 to 1920000 for the instances 11-14), making it easier to solve than the instances 11 to 14.

Compared to the CPLEX solver with the mathematical programming models, the greedy heuristic algorithm is more successful with a better balance between solution quality and run time. For all instances, the greedy algorithm reached its solutions under one second. Additionally, the solution quality of the greedy algorithm is much better than the mathematical programming models for medium and large scale instances.

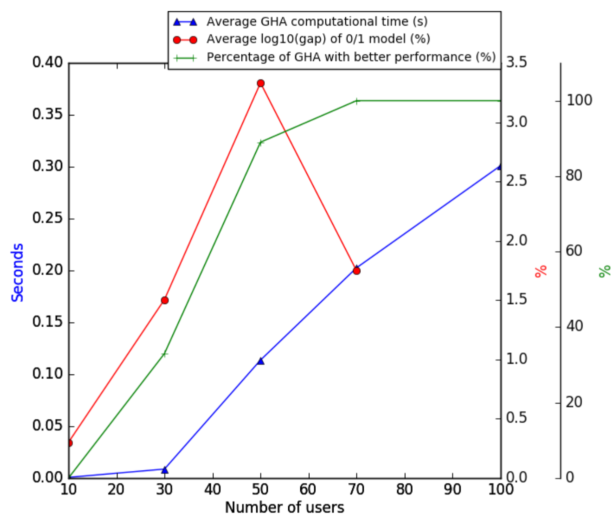
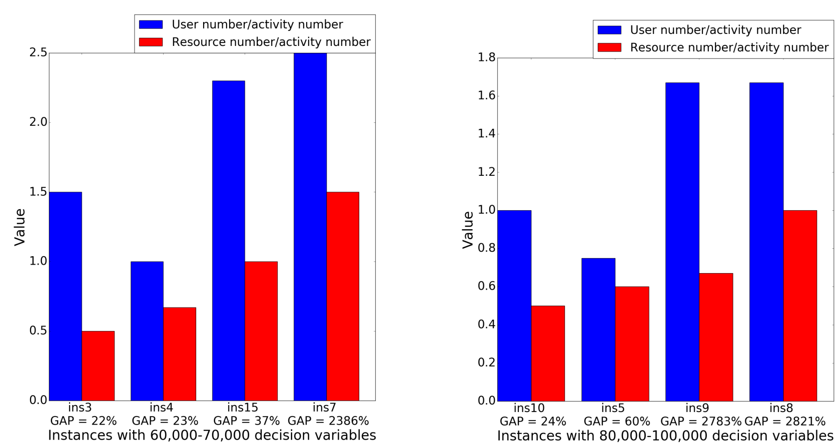


Figure 2: Summary of the computational results by the number of users

Figure 2 shows a visual presentation of the computational results. Given that the integer programming model solved just 3 out of the 20 instances, it is ignored in the figure. The curves in Figure 2 correspond to the average $\log_{10}$ GAP of the 0/1 programming model (red), the percentage of the instances for which the greedy algorithm gave the best result (green) and the average computational time (in seconds) of the greedy algorithm (blue). Overall, the results indicate that as the number of users increase, the problem becomes more difficult to solve for both methods. However the greedy algorithm becomes much more computationally efficient compared to the 0/1 programming model.

The plots of Figure 3 illustrate the relation between the computational results of the 0/1 programming model and more detailed characteristics of the instances. According to the type of structures, the magnitudes between users, activities and resources are different, which lead to different organizational difficulty levels. To analyze more accurately and avoid the influence of other factors, we tested all instances with a same importance coefficient and compared the results between the instances of similar sizes. The plots show two sets of instances with similar numbers of decision variables of the 0/1 programming model and ordered by increasing order of the gap given by CPLEX (calculated by $|UB - LB|/|LB|$). We observe that instances become harder to solve as the number of users per activity and the number of resources per activity increase. This means that when there are more users and resources to arrange for activities, it is more difficult to make the choice and obtain the best assignment. Indeed, this result corresponds to the decision making difficulties in reality, especially for large structures. For example, when most of the users are interested in several popular activities, the decision-maker has to make the choice to satisfy as many users as possible while guaranteeing the quality of activities at the same time.



(a) Instance set 1

(b) Instance set 2

Figure 3: Characteristics of instances and the gap of the 0/1 programming model

6. Conclusions and future research perspectives

We presented a real-life user project scheduling problem in the social and medico-social sector. The targeted application requires selection and scheduling of a set of activities for users and resources while satisfying some imperative constraints and optimizing different objectives. To tackle this problem, we introduced a formal description and devised two integer programming models as well as greedy algorithm.

The proposed approaches were tested on 20 simulated test instances (that we made publicly available) corresponding to real scenarios identified according to a large-scale survey. The 0/1 programming model (solved by the CPLEX software) showed a high competitive performance compared to the integer programming model and obtained close-to-optimality solutions for some small instances. However, as the instance scale increases, the performance of integer programming deteriorates significantly. The greedy algorithm was more successful and obtained more stable and better results for the middle and large instances. Consequently, the proposed greedy approach will serve as the basis for building the decision-making tool for the personalized projects planning.

This work can be further extended according to several directions. First, the results of the proposed greedy algorithm can be improved by other search strategies such as stochastic local search (Hoos and Stützle, 2004). For this purpose, it is worth investigating different neighborhood structures and constraint handling techniques. Second, this work used a weighted sum method to combine the three objectives. An interesting possibility would be to consider the multiobjective optimization approach to obtain multiple alternative non-dominated solutions from which the decision-maker can choose the most suitable solutions. Third, the activities in the proposed model are considered to be independent. It would be interesting to study possible relationships within the activities. Such an extension could be useful in structures where such situations can be encountered. Finally, the research outcomes will contribute to enrich the existing *"MSUsager"* software mentioned in the introduction with a powerful decision-making component for social and medico-social structures. The data collected from the system after its deployment will enable the development of new system features based on machine learning techniques. These new features can provide automatic generation of user preferences for activities by learning useful information from historical data such as activity evaluation and similar user experiences, which will definitively help to define the most suitable personalized user projects of high quality.

Acknowledgment

We are grateful to our reviewers for their insightful comments which helped us to improve the paper. This work was partially supported by the French Ministry for Research and Education through a CIFRE grant (number 2018/0225). The authors thank M. Rachid Naitali, the General Director of GePI, for his support.

References

Al-Harbi, K.M.A.S., 2001. Application of the AHP in project management. International Journal of Project Management 19, 19–27.

Anderson Fabian, M.V., 2019. Approche territoriale d'une gestion dynamique et multidimensionnelle de l'orientation des personnes en situation de handicap: D'une orientation handicap-établissement à une orientation besoins-prestations. Doctoral thesis. Institut National des Sciences Appliquées de Lyon. Lyon, France.

ANESM, 2008. Les attentes de la personne et le projet personnalisé. Technical Report. Agence nationale de l'évaluation et de la qualité des établissements et services sociaux et médico-sociaux.

Ballou, D.P., Tayi, G.K., 1996. A decision aid for the selection and scheduling of software maintenance projects. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans 26, 203–212.

Bassett, M., 2000. Assigning projects to optimize the utilization of employees' time and expertise. Computers & Chemical Engineering 24, 1013–1021.

Brucker, P., Drexl, A., Möhring, R., Neumann, K., Pesch, E., 1999. Resource-constrained project scheduling: Notation, classification, models, and methods. European Journal of Operational Research 112, 3–41.

CEDIS, 2012. Guide d'élaboration du projet d'accompagnement personnalisé. Technical Report. Le Comité Européen pour le Développement de l'Intégration Sociale.

Chabane, B., Basseur, M., Hao, J.K., 2017. R2-IBMOLS applied to a practical case of the multiobjective knapsack problem. Expert Systems with Applications 71, 457–468.

Chin, K.S., Xu, D.l., Yang, J.B., Lam, J.P.K., 2008. Group-based ER-AHP system for product project screening. Expert Systems with Applications 35, 1909–1929.

Davari, M., Demeulemeester, E., 2019. The proactive and reactive resource-constrained project scheduling problem. Journal of Scheduling 22, 211–237.

Decker, K., Li, J., 1998. Coordinated hospital patient scheduling, in: Proceedings of the International Conference on Multi Agent Systems (Cat. No. 98EX160), IEEE. pp. 104–111.

Enea, M., Piazza, T., 2004. Project selection by constrained fuzzy AHP. Fuzzy Optimization and Decision Making 3, 39–62.

Fikar, C., Hirsch, P., 2017. Home health care routing and scheduling: A review. Computers & Operations Research 77, 86–95.

Frifita, S., Masmoudi, M., 2020. VNS methods for home care routing and scheduling problem with temporal dependencies, and multiple structures and specialties. International Transactions in Operational Research 27, 291–313.

Grodzevich, O., Romanko, O., 2006. Normalization and other topics in multi-objective optimization, in: Proceedings of the Fields-MITACS Industrial Problems Workshop, pp. 89–101.

Gutjahr, W.J., Katzensteiner, S., Reiter, P., Stummer, C., Denk, M., 2008. Competence-driven project portfolio selection, scheduling and staff assignment. Central European Journal of Operations Research 16, 281–306.

Habibi, F., Barzinpour, F., Sadjadi, S., 2018. Resource-constrained project scheduling problem: review of past and recent developments. Journal of Project Management 3, 55–88.

Hartmann, S., Briskorn, D., 2010. A survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of Operational Research 207, 1–14.

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

Herroelen, W., Leus, R., 2005. Project scheduling under uncertainty: Survey and research potentials. European Journal of Operational Research 165, 289–306.

Hoos, H.H., Stützle, T., 2004. Stochastic Local Search: Foundations & Applications. Elsevier / Morgan Kaufmann.

Kosztyán, Z.T., 2015. Exact algorithm for matrix-based project planning problems. Expert Systems with Applications 42, 4460–4473.

Kosztyán, Z.T., 2020. An exact algorithm for the flexible multilevel project scheduling problem. Expert Systems with Applications 158, 113485.

Kosztyán, Z.T., Szalkai, I., 2020. Multimode resource-constrained project scheduling in flexible projects. Journal of Global Optimization 76, 211–241.

Laurent, B., Hao, J.K., 2007. Simultaneous vehicle and driver scheduling: A case study in a limousine rental company. Computers & Industrial Engineering 53, 542–558.

Lü, Z., Hao, J.K., 2010. Adaptive tabu search for course timetabling. European Journal of Operational Research 200, 235–244.

Lü, Z., Hao, J.K., 2012. Adaptive neighborhood search for nurse rostering. European Journal of Operational Research 218, 865–876.

Marler, R.T., Arora, J.S., 2004. Survey of multi-objective optimization methods for engineering. Structural and Multidisciplinary Optimization 26, 369–395.

Marler, R.T., Arora, J.S., 2005. Function-transformation methods for multi-objective optimization. Engineering Optimization 37, 551–570.

McLeod, S., 2007. Maslow's hierarchy of needs. Simply Psychology 1, 1–8.

Melachrinoudis, E., Min, H., 2011. A tabu search heuristic for solving the multi-depot, multi-vehicle, double request dial-a-ride problem faced by a healthcare organisation. International Journal of Operational Research 10, 214–239.

Montoya, G.O., 2015. Conception des outils pour le suivi des activités et l'aide au pilotage dans le secteur médico-social. Doctoral thesis. Institut National des Sciences Appliquées de Lyon. Lyon, France.

Proos, K., Steven, G., Querin, O., Xie, Y., 2001. Multicriterion evolutionary structural optimization using the weighting and the global criterion methods. AIAA Journal 39, 2006–2012.

Servranckx, T., Vanhoucke, M., 2019. A tabu search procedure for the resource-constrained project scheduling problem with alternative subgraphs. European Journal of Operational Research 273, 841–860.

Shi, Y., Boudouh, T., Grunder, O., 2017. A hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand. Expert Systems with Applications 72, 160–176.

Stork, F., 2001. Stochastic resource-constrained project scheduling. Doctoral thesis. Technische Universität Berlin, Fakultät II - Mathematik und Naturwissenschaften. Berlin, Germany.

Wang, Y., Cang, S., Yu, H., 2019. A survey on wearable sensor modality centred human activity recognition in health care. Expert Systems with Applications 137, 167–190.

Zhang, Y., Yu, P., Shen, J., 2012. The benefits of introducing electronic health records in residential aged care facilities: a multiple case study. International Journal of Medical Informatics 81, 690–704.