Planning and Scheduling of Business Processes in Run-Time: A Repair Planning Example

Irene Barba and Carmelo Del Valle

Abstract Over the last decade, the efficient and flexible management of business processes has become one of the most critical success aspects. Furthermore, there exists a growing interest in the application of Artificial Intelligence Planning and Scheduling techniques to automate the production and execution of models of organization. However, from our point of view, several connections between both disciplines remains to be exploited. The current work presents a proposal for modelling and enacting business processes that involve the selection and order of the activities to be executed (planning), besides the resource allocation (scheduling), considering the optimization of several functions and the reach of some objectives. The main novelty is that all decisions (even the activities selection) are taken in run-time considering the actual parameters of the execution, so the business process is managed in an efficient and flexible way. As an example, a complex and representative problem, the repair planning problem, is managed through the proposed approach.

1 Introduction

In the last years, the effective management of business processes (BP) in organizations became more important, since they need to adapt to the new commercial conditions, as well as to respond to competitive pressures, considering the business environment and the evaluation of their information systems. BP Management (BPM) can be seen as supporting BP using methods, techniques, and software to design, enact, control and analyze operational processes involving humans, organizations, applications, and other sources of information [2]. Similarly, Workflow Management Systems [1, 11] consist of methods and technologies for managing the flow of work in organizations. In a related way, BPM Systems (BPMS) are software tools that support the management of the BP. In some cases, they use temporal information and ignore, in some ways, the required resources, considering them unlimited. This may not be adequate in different situations, for example when limited resources can be required by different activities at overlapped periods of time.

The area of Scheduling [8] includes problems in which it is necessary to determine an execution plan for a set of activities related by temporal constraints. Moreover, the execution of each activity requires the use of resources so they may compete for limited resources. In general, the objective is to find a feasible plan so that temporal and resource constraints are satisfied, usually optimizing objective functions related to temporal measures. In a wider perspective, in Artificial Intelligence (AI) planning [12], the activities to be executed are not established a priori, so it is necessary to select and to order them from a set of alternatives. In most cases, the specification of planning problems includes the initial state of the world, the goal (a predicate representing to a set of possible final states) that must be reached, and a set of operators (actions) that can be applied to one state to reach another one, allowing the evolution of the system. In order to solve this kind of problems, it is necessary to select a suitable set of actions that must be executed in a correct order, in some cases allowing parallelism. Furthermore, in several planning problems, the optimization of some objectives functions is pursued.

Currently, there is an increasing interest in integrating the application of AI Planning and Scheduling (P&S) techniques, some of them are maintenance and repair planning [10], where there may be a cascading set of choices for actions, facilities, tools or personnel, which affect different features of the plan, such as duration or cost [24]. Also, connection and disconnection planning involve the identification, selection and sequencing of operations, which can be specified by their effects on the components. In other context, disconnection planning has been object of different studies, such as maintenance or repair purposes [17]. Many problems can involve multiple conflicting objectives at the same time (multi-objective optimization [9]).

The And/Or graphs [15] (Sect. 3) can be used as a basis of representing most problems that involve both P&S, including the repair planning problem (Sect. 3), that is studied in the current work. A similar representation for this kind of problems can be found in [6], that proposes an extension of temporal networks by parallel and alternative branching to represent some kinds of P&S problems. Also, in [3] it is proposed the concept of blocks that can classify BP flows into several patterns: iterative block, serial block and parallel block including *AND* and *OR* structures.

BPM and AI P&S are two disciplines with many parallels, but which have largely been pursued by disjoint communities. BPMS are responsible for goals specification, design, implementation, enactment, monitoring and evaluation of BP [18], so they must deal with the anticipated P&S of processes. In general, automated P&S has not been integrated in business environments due to, between others, the lack of friendly modelling tools. There are several points where AI P&S tools can be effectively applied to BPMS, as it is explained in Sect. 2. In the last years, there exists a growing interest in the application of AI P&S techniques to automate the production and execution of models of organization, as shown by the existence of a Technical Coordination Unit of the European research network on P&S, PLANET [20], and

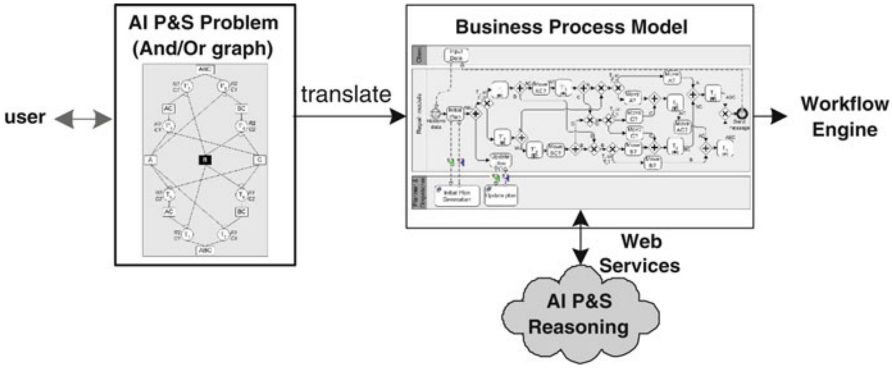


Fig. 1 AI P&S techniques for optimal business process enactment

several related research projects (SWIM system [7], the MILOS project [19], the TBPM project [16], etc). However, from our point of view, several connections between both disciplines remains to be exploited.

This work is based on modelling and enacting a problem involving both AI planning (selection and order of the activities) and scheduling (temporal resources allocation) through BP tools, using AI techniques in run-time for the optimization of several objective functions regarding the resulting execution plan, specifically duration and cost. The repair planning problem has been selected for being a complex and representative problem involving P&S. The most important contribution of this paper is the automatic management of the enactment of BP representing complex P&S problems, selecting and ordering the activities execution and allocating the resources in run-time, in order to avoid the drawbacks of taking these decisions during the design phase. Figure 1 shows a graphic representation of this work: first, the user introduces the information about the P&S problem, in this case, the repair problem. This information can be modelled through an And/Or graph or a BP model language, where all the execution alternatives are included, also containing the estimated values of several parameters, such as activity durations, resource availability, etc. This graph is translated to a BP model, that still represents a P&S problem since the selection and the order of the activities, together with the temporal resources allocation, are carried out in the enactment phase. The resulting BP model includes a pool composed by web services based on AI P&S techniques, that drives the execution considering the optimization functions and the actual values of the parameters. From this specification, the workflow engine can enact the plan in an optimized way.

In the BP enactment, the P&S decisions are taken considering the actual execution values instead of the estimated ones, optimizing the objective functions, so a flexible and dynamic BP is got. On the other hand, the decision-making in run-time provides the problem with a high spatial and temporal complexity. It is necessary to find a good balance between flexibility–optimality and complexity of the BP.

The main contributions of this paper can be summarized as:

- Optimal execution of BP that, in run-time, require: first, the selection and the order of the activities to be executed, due to the existence of several possible alternatives (planning) expressed through exclusive data-based gateways; and secondly, the resources allocation involving temporal reasoning due to the use of shared resources (scheduling), both considering the optimization of functions.
- Optimal execution of AI P&S problems that can be represented by And/Or graphs through BP tools, considering both P&S of the activities, and taking into account the optimization of objective functions.
- Translation from AI P&S problems that can be modelled through And/Or graphs, to BP models, focused on a complex and representative P&S problem: the repair planning problem.

In the following, Sect. 2 describes the BPM life cycle, Sect. 3 details the considered repair planning problem, Sect. 4 states the translation from the And/Or graph to a BP model, Sect. 5 summarizes the more related work and, finally, Sect. 6 presents some conclusions and future work.

2 AI Planning and Scheduling for BPM

The integration of AI P&S techniques with Workflow Management Systems can improve overall system functionality and can help to automate the definition of BP. As follows, the different stages of a typical BPM Life Cycle [2] are briefly presented:

- Process design: This stage involves designing, modelling, optimizing and simulating the organization processes. It is basically a human activity.
- System configuration: In this stage a schedule must be generated, taking into account the information given by the customer, such as the target end date, dependencies between activities or activity durations. The resources are assigned to activities for the appropriate time slots (scheduling), considering the finite capacity and/or non-sharable resources and, generally, taking into account the optimization of objective functions, such as process duration or cost.
- Process enactment: In this phase, the plan is carried out so that the activities have to be coordinated to ensure correct sequencing. At the same time, resources will be involved in enacting multiple processes and instances of the same process. When the same resource is required by several activities at the same time, generally, rules for prioritizing activities must act on the conflict.
- Diagnosis: As execution proceeds, the enactment information must be analyzed due to the possible appearance of unexpected incidents, such us unavailability of resources, actual activity durations different from expected ones, fails, etc. Minor incidents may require updating of the plan. More significant differences may require great changes in the plan, even a re-planning.

This work is based on automating the resource allocation and the selection and order of the activities, in both system configuration and process enactment phases, considering AI P&S techniques for optimizing several objective functions.

3 The Repair Planning Problem

Maintenance and repair planning is becoming a more important issue in many systems, according to the increasing problem of their complexity, the limitations of the technology used to maintain it and the problem of its ageing. In some studies, repair activities are integrated into the diagnosis process [10]. In order to repair some (previously diagnosed) faulty components, a sequence of disconnection activities must be executed to get it. After that, a repair action would substitute or repair the component, and then a set of connection activities must reconnect the system.

And/Or graphs have been used for the planning of assembly and repair processes [15]. This graph has been adapted for the representation of the proposed repair problem due to the existing similarities between both problems. The And/Or graph allows to represent the set of all feasible connection/disconnection plans in a natural way, depicting the system structure and including the alternative activities and the precedence constraints between them (Fig. 2a). It represents the disconnection (top part) and connection (bottom part) activities that can be selected in order to repair a faulty component (D in this case), in a system made of five components (*ABCDE*). In this representation two kinds of nodes can be distinguished:

- Or nodes: Correspond to subsystems, i.e., *AC* in Fig. 2. The top node corresponds to the complete system, and the leaf ones correspond to the components.
- And nodes: Correspond to the connection activities joining the subsystems of its two Or nodes below it producing the subsystem corresponding to the Or node above it, i.e., T_3; and the disconnection activities, that decomposes the subsystem above it to obtain the two subsystems below it, i.e., $T_1{}'$.

For the same Or node, there can be several And nodes (activities) below it, representing different alternatives to connect/disconnect the corresponding subsystem. In these graphs, each disconnection plan is associated to a tree, that is an And/Or path starting at the root node and ending at the nodes representing components (also, each connection plan is associated to a tree in a similar way). An important advantage is that the And/Or graph shows the activities that can be executed in parallel (Fig. 2). Furthermore, both precedence constraints and those related to the selection of activities, can be easily obtained from this representation.

A feasible repair plan can be seen as a (minimum) set of activities that starts with the disconnection of the complete system, repairs the faulty component, and finishes with the connection of the complete system. In this work, two kinds of activities are considered, each one presenting an associated duration and cost:

- Connection/disconnection activities: are executed on an established resource with a particular configuration to obtain the different subsystems.

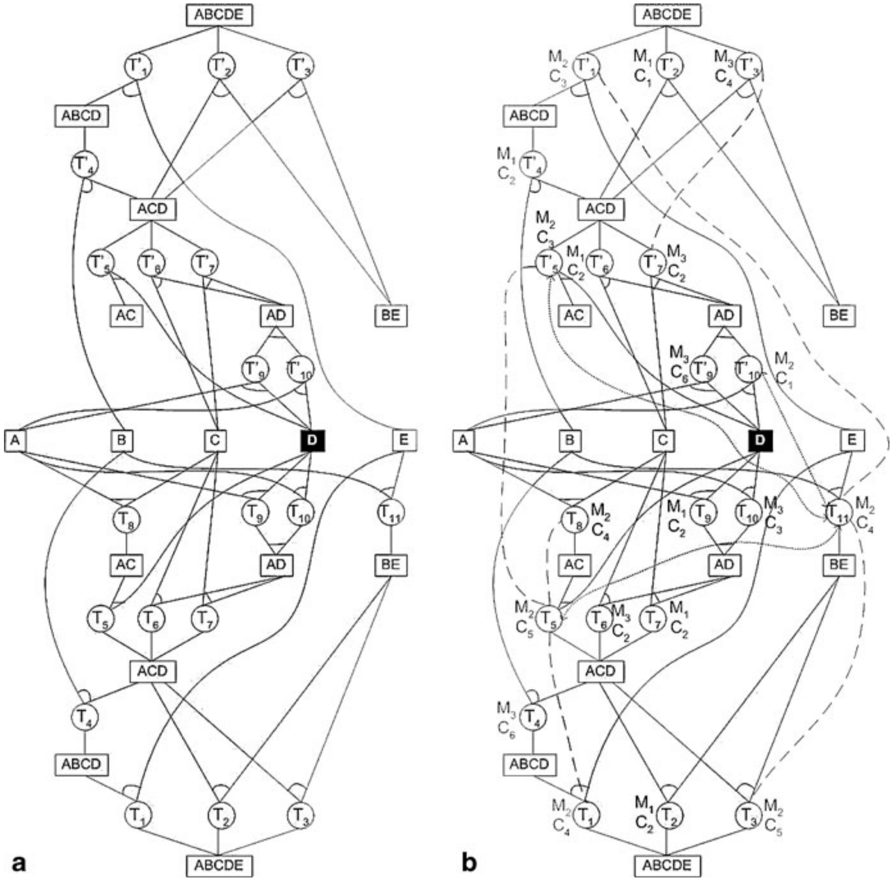


Fig. 2 Example of And/Or graph and an extension including resource constraints. **a** Original And/Or graph. **b** Extended And/Or graph

- Auxiliary activities: due to the use of shared and different resources. Two kinds of operations are considered: *set-up operations,* that change the configuration of a resource when two successive activities with different configuration use that resource; and *transportation operations,* that transport the subsystems between locations when the location where the subsystem is obtained after a connection/disconnection activity is different from the location where is required before a connection/disconnection activity.

In order to manage the shared resources with different configurations, the original And/Or graph has been extended, so that the new representation includes all the constraints involved in the problem, adding new types of links between the And nodes (Fig. 2b). The new links represent non-precedence constraints: due to the use of shared resources (i.e., T_5 and T_{11}) and due to the change of configurations in the resources (i.e., T_5 and T_1). The reasoning with resources, specifically temporal resources allocation, lead to NP completeness due to disjunctive constraints.

It is important to emphasize that the And/Or graph represents the system structure through components relations, that remains permanent despite of different required resources or faulty components. Consequently, for the same graph there can be several repair problems varying the faulty components or the required resources.

The And/Or graph can be used as a base representation of most AI P&S problems, since it allows to represent important P&S aspects, such as alternative activities or precedence relations. In general, this representation can be easily extended in order to include other aspects, i.e., shared resources in the repair planning problem.

4 The Repair Planning Problem as a Business Process

Once the problem in modelled as an And/Or graph, it can be translated to a BP language model, as instance BPMN [26], in order to be deployed and executed. It is important to emphasize that the And/Or graph contains all the possible alternative activities to be selected, and during the enactment phase, AI P&S methods are used to select and to order the suitable activities automatically, in order to obtain an optimal execution plan. As stated before, the And/Or graph represents the system structure through components relations, so different required resources and faulty components leads to different problems.

The BP model for the repair planning problem (Fig. 4) contains three pools:

1. *Client:* Abstract pool that acts as intermediary between the user and the BP. The user must specify the faulty components, that starts the BP enactment. At the end of the enactment, a message containing the selected activities together with their temporal allocation, is sent to the client.
2. *Repair module:* Executable pool that contains all the activities of the problem, from the And/Or graph specification.
3. *Planner and dispatcher:* Abstract pool that contains the web services. These web services are based on AI P&S techniques and are responsible for the optimal P&S of the activities. Each time that the repair plan execution *(Repair Module)* reaches OR branches, the *Planner and Dispatcher* pool decides the way that must be followed (planning). Furthermore, for the tasks requiring the same resources to be executed, this pool establishes the execution order (scheduling).

During the execution of the repair plan, the *Repair Module* and the *Planner and Dispatcher* pool are interchanging information. When the *Repair Module* reaches an exclusive data-based gateway, the decision to take is established by the information previously received from the *Planner and Dispatcher* pool. Different AI P&S techniques can be used in order to obtain an optimal execution plan. In previous works, we have developed proposals for solving the optimal P&S for the proposed repair planning problem, taking into account the minimization of both duration and cost. Specifically, in [4] it is explained a constraint based proposal [23] and in [5] it is shown a PDDL specification for generic planners [12], both considering multimode

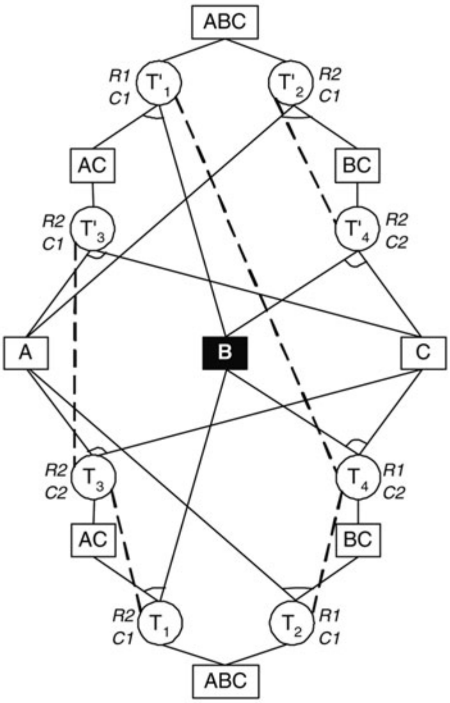


Fig. 3 Example of extended And/Or graph for a system made of three components when faulty component is B

activities and multi-objective optimization (duration and cost). In both, several experimental results are shown, demonstrating that these proposals are successful at solving this kind of problems. These tools have been used for developing the web services related to P&S in the current work.

As follows, the BP model generation from the And/Or graph is explained. As an example, the BPMN corresponding to the repair problem of Fig. 3 can be seen in Fig. 4. For the sake of clarity, in Fig. 4a the activities corresponding to connection/disconnection tasks appear collapsed, while in Fig. 4b its expanded representation is shown. Regarding to the proposed BP model, first of all, a message containing the faulty components is sent from the *Client* pool to the *Repair Module*. After that, an initial plan is generated trying to obtain a good start point (optimizing the objective functions). This initial plan, in most cases, will be iteratively improved during the BP enactment, as explained later. After the initial plan generation, two branches are executed in parallel in the *Repair Module*:

- Optimization process (Branch 1 in Fig. 4a): The initial plan can be non-optimal. Also, the actual parameters can be different from the estimated ones. Considering both reasons, it is proposed a loop updating activity that is continuously trying to improve the current solution, taking the actual values of the parameters. In order to do this, a web service based on AI techniques *(Update plan)* is invoked. Each time a better solution is found, the current plan is updated and these information

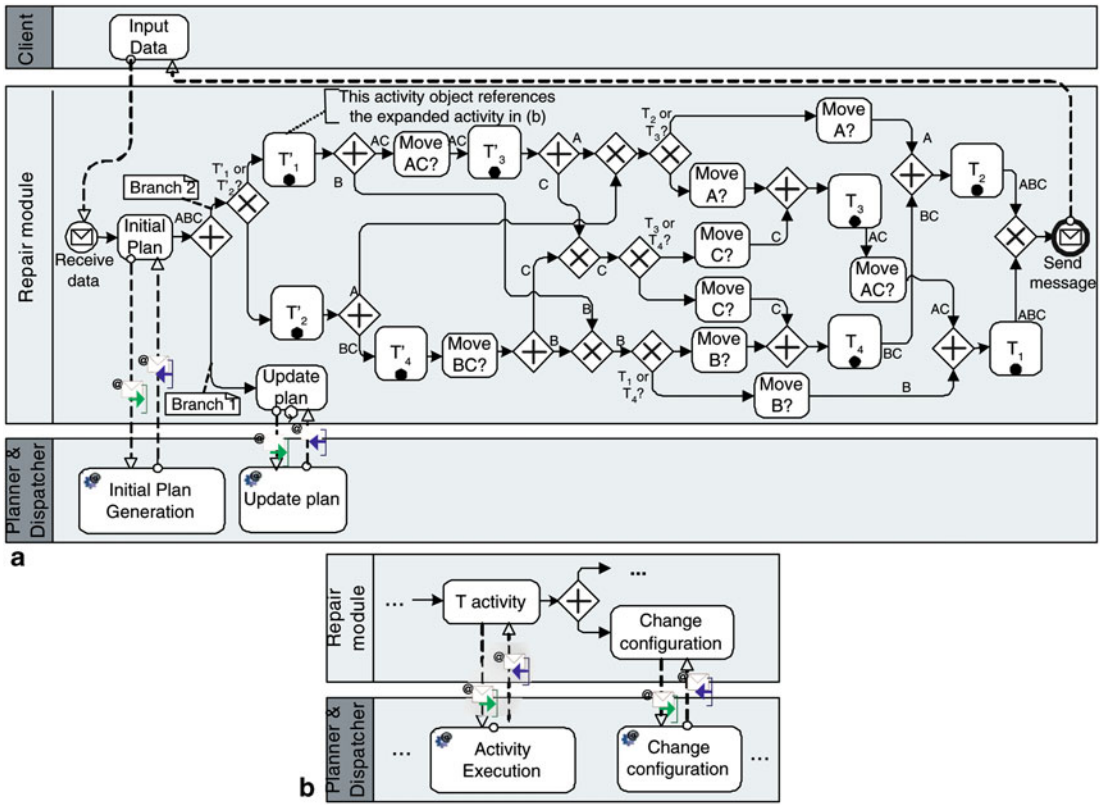
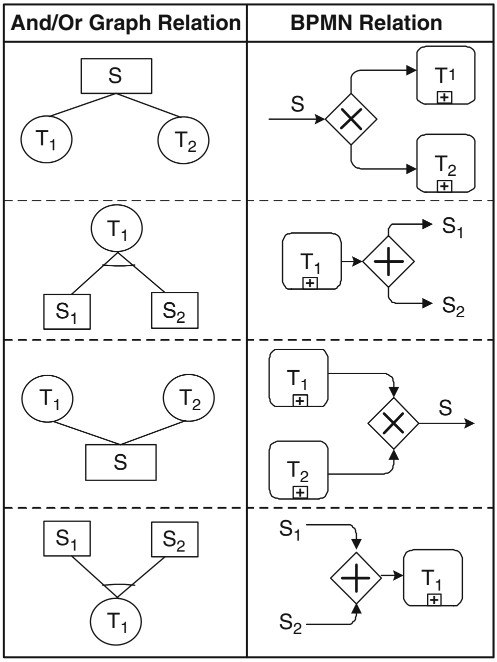


Fig. 4 BPMN diagram of the repair problem of Fig. 3. **a** BPMN diagram with collapsed activities. **b** BPMN diagram for the expanded activity

Fig. 5 Translation from And/Or graph to BPMN relations



is used to decide the way to follow in the exclusive data-based gateways. The loop finishes when the repaired complete system is successfully obtained.

- Repair plan (Branch 2 in Fig. 4a): the activities are executed considering the P&S established by the *Planner and Dispatcher,* as follows:
  - Each relation that exists in the And/Or graph between several components is translated to another relation in the BPMN model (Fig. 5).
  - The location where the subsystem is obtained after a connection/disconnection activity can be different from the one where it is required before a connection/disconnection activity, so it is necessary to consider the activity *Move subsystem?* between two successive activities that act on the same subsystem. The *Move?* activity has to be executed only when the successive activities take place in different locations (in the current work, different resources are considered different locations). For the *Move?* execution, it is not necessary to call web services of *Planner and Dispatcher* pool since the information about the location where the subsystem was obtained can be consulted in the plan execution, and the information about the location where it is required depends on the selected branch. If a subsystem will be moved or not during the enactment phase, it depends on the plan execution because of different activities can obtain a subsystem, and it can be required also for different ones.
  - Regarding to Fig. 4b, *T Activity* is the actual execution of the connection/disconnection task. A web service must manage it, since it must establish

the start time of the task, its duration, etc. Taking into account the task execution information, the actual values are stored, and used for the plan update and the rest of the BP enactment. The *Change of configuration* activity, provided that it is necessary, must be done in parallel with the treatment of the subsystems obtained after the activity execution (Fig. 4b). The *Change of configuration* activity is also controlled by the *Planner and Dispatcher* module since its execution causes the resource lock, that influences the BP execution. Also, if this activity is executed or not, and when it is executed, depend on several factors that must be analyzed by the *Planner and Dispatcher* pool.

Finally, when the repair plan has been successfully executed, a message with the plan details is sent to the client.

5 Related Work

Several research groups have integrated AI techniques with BPM systems. As follows, some of the similar works are briefly described, explaining the main differences between them and the current approach.

Some related works integrate P&S tools in BPM systems for the modelling phase, such as [13], where a user specifies the information related to the BP through an execution/interchange standard language, XPDL, that is translated to HTN-PDDL language (an extension of PDDL) in order to obtain suitable timeframe for each activity and resource allocation. In a similar way, in [22], planning tools are used to generate BP models, taking into account the knowledge introduced through BP Reengineering languages. This knowledge is translated into predicate logic terms in order to be treated by a planner that integrates P&S, and obtains an updated BP model. Both works generate the BP model during the build-time, without taking into account the actual values of the parameters obtained in run-time. Consequently, the initial optimization plan can be obsolete due to the non-updating aspect. On a different way, the current proposal, in order to update the plan in run-time, maintains all the possible alternatives in the BP model, taking into account the actual values of the parameters for updating the P&S of the activities.

On the other hand, most of the related works integrate scheduling tools in BPM systems for the enactment phase, in order to take dispatching decisions as to which activity should be executed using a resource when it becomes free (dynamic scheduling). As follows, a representative set of them are briefly summarized. One of the first works, [27], developed a framework for temporal workflow management, including turnaround time predication, time allocation, and task prioritization. In a related way, [25] proposes a schema for maximizing the number of workflow instances satisfying a predetermined deadline, based on a method to find out the critical activities. Also, [14] proposes a set of process execution rules based on individual worklists, and it develops algorithms for the task assignment in order to maximize the overall process efficiency, taking into account the resources capacities. Recently, [21] presents a Theory of Constraints (TOC)-based method for improving the efficiency of BP. There are several differences between these works and the current approach. First, in the current proposal the selection of the activities to be executed (planning) is developed in run-time, besides the resource allocation, while the previously summarized works only consider the prioritization of the tasks using the same resource (scheduling). On the other hand, the current approach considers several objective functions, while the other works only focus on the temporal aspects of the processes for allocating the resources.

To the best of our knowledge, there is not any proposal for selecting the adequate activities in run-time, in order to optimize some functions in BP environments and considering reaching several objectives (planning).

6 Conclusions and Future Work

The current work is based on modelling and enacting a representative and complex problem through BP tools, the repair planning problem, involving both planning and scheduling aspects. The enacting of the BP is managed by AI techniques in run-time, so that the activities to be executed are selected and ordered, and the resources are temporarily allocated. The main novelty of the work is the planning of the activities in run-time, taking into account the optimization of several functions and the reach of some objectives. On the other hand, the proposed architecture and AI techniques can be used for the optimal execution of different BP (expressed through a model language) that, in run-time, require: first, the selection and the order of the activities to be executed, due to the existence of several possible alternatives (planning), and secondly, the temporal resources allocation due to the use of shared resources (scheduling), both taking into account the optimization of objective functions.

As future work, it is intended to analyze another kinds of BP problems involving planning and scheduling aspects. Also, a bidirectional translation between planning languages, as PDDL, and BP model or execution languages, is intended to be explored. Furthermore, another objective functions can be considered.

Acknowledgements This work has been partially funded by the Consejeria de Innovacion, Ciencia y Empresa of Junta de Andalucia (P08-TIC-04095) and by the Spanish Ministerio de Ciencia e Innovacion (TIN2009-13714) and the European Regional Development Fund (ERDF/FEDER).

References

1. van der Aalst WMP, van Hee KM (2002) Workflow management: models, methods, and systems. MIT Press, Cambridge
2. van der Aalst WMP, ter Hofstede AHM, Weske M (2003) Business process management: a survey. In: Business process management: international conference, BPM 2003, Eindhoven, The Netherlands, 26–27 June 2003 (Proceedings 1–12)
3. Bae J, Bae H, Kang S, Kim Y (2004) Automatic control of workflow processes using ECA rules. IEEE Tran Knowl Data Eng 16(8):1010–1023
4. Barba I, Del Valle C, Borrego D (2009) A constraint-based model for multi-objective repair planning. In: 14th IEEE international conference on emerging techonologies and factory automation

5. Barba I, Del Valle C, Borrego D (2009) PDDL specification for multi-objective repair planning. In: Caepia 2009 workshop on planning, scheduling and constraint satisfaction, pp 21–33
6. Bartak R, Cepek O (2007) Temporal networks with alternatives: complexity and model. In: Proceedings of the twentieth international florida artificial intelligence research society conference, FLAIRS 2007, pp 641–646
7. Berry PM, Drabble B (2000) SWIM: an AI-based system for organizational management. In: The 2nd NASA international work shop on planning and scheduling for space, NASA, Washington, DC
8. Brucker P, Knust S (2006) Complex scheduling (GOR-Publications). Springer, New York
9. Coello CA (2006) Evolutionary multi-objective optimization: a historical view of the field. IEEE Comput Intell Magaz 1(1):28–36
10. Friedrich G, Gottlob G, Nejdl W (1994) Formalizing the repair process—extended report. Ann Math Artif Intell 11:187–201
11. Georgakopoulos D, Hornick M, Sheth A (1995) An overview of workflow management: from process modeling to workflow automation infrastructure. In: Distributed and parallel databases, pp 119–153
12. Ghallab M, Nau D, Traverso P (2004) Automated planning: theory and practice. Morgan Kaufmann, Amsterdam
13. Gonzalez-Ferrer A et al (2008) Towards the use of XPDL as planning and scheduling modeling tool: the workflow patterns approach. In: 5290 LNAI, pp 52–61
14. Ha B, Bae J, Park Y, Kang S (2006) Development of process execution rules for workload balancing on agents. Data Knowl Eng 56(1):64–84
15. Homem de Mello LS, Sanderson AC (1988) Planning repair sequences using the And/Or graph representation of assembly plans. IEEE international conference on robotics and automation, pp 1861–1862
16. Stader J, Moore J, Chung P, McBriar I, Ravinranathan M, Macintosh A (2000) Applying intelligent workflow management in the chemicals industries. In: Fisher L (ed) The workflow handbook 2001, Published in association with the Workflow Management Coalition (WfMC), Oct 2000, pp 161–181, ISBN 0-9703509-0-2
17. Li W, Zhang C (1995) Design for disassembly analysis for environmentally conscious design and manufacturing. Proc 1995 ASME Int Mech Eng Cong 2:969–976
18. Muehlen MZ, Ting-Yi Ho D (2005) Risk management in the BPM lifecycle. In: Business process management workshops, pp 454–466
19. Münch J, Goldmann S, Holz H (2000) Distributed process planning support with MILOS. Int J Softw Eng Knowl Eng 10(4):511–525
20. Network home page: European Network of Excellence in AI planning. http://planet.dfki.de/index.html. Accessed 27 June 2011
21. Rhee S, Cho NW, Bae H (2008) Increasing the efficiency of business processes using a theory of constraints. Inf Syst Front 12(4), 1–13. doi:10.1007/s10796-008-9145-9
22. R-Moreno MD, Borrajo D, Cesta A, Oddi A (2007) Integrating planning and scheduling in workflow domains. Expert Syst Appl 33(2):389–406
23. Rossi F, Van Beek P, Walsh T (2006) Handbook of constraint programming. Elsevier, Amsterdam
24. Smith D, Frank J, Jonsson A (2000) Bridging the gap between planning and scheduling. Knowl Eng Rev 15(1):47–83
25. Son JH, Kim MH (2001) Improving the performance of timeconstrained workflow processing. J Syst Softw 58(3):211–219. doi:10.1016/S0164-1212(01)00039-5
26. White SA et al (2004) Business process modeling notation (BPMN), working draft, Version 1.0. BPMI.org
27. Zhao JL, Stohr EA (1999) Temporal workflow management in a claim handling system. SIGSOFT: Softw Eng Notes 24(2):187–195. doi:10.1145/295666.295687