Planning & Scheduling of Crude Oil Distribution in a Petroleum Plant

Tiago Stegun Vaquero[1,2] and **Fernando Sette**[1] and **José Reinaldo Silva**[1] and **J. Christopher Beck**[2]
[1] Department of Mechatronics, Universidade de São Paulo, Brazil
[2] Department of Mechanical & Industrial Engineering, University of Toronto, Canada
tiago.vaquero@poli.usp.br, fernando.sette@poli.usp.br, reinaldo@usp.br, jcb@mie.utoronto.ca

Abstract

Since the end of the 1990s there has been a great deal of interest in the application of AI Planning techniques to solve real-life problems. Most of the problems identified as suitable to be solved with a planning approach are considered complex and require intense knowledge management and reasoning about actions. Indeed, these real problems bring challenges not only for the designers and Knowledge Engineering tool during design processes but also for the automated planners during the planning process itself. In this paper we present the experience and results from designing a real planning application in the petroleum industry. In this work we investigate the planning and scheduling of the daily activities of a petroleum plant for docking, storing and distributing oil from a planning perspective. Due to the complexity of this domain, the KE tool itSIMPLE was used in order to support the design processes of such application. We present the construction of domain model and the experimental results while testing the automatically generated PDDL model with a modern planner. Two semi-realistic scenarios are studied in order to evaluate the approach.

Introduction

Over the last five years, an increasing interest of applying all achievements on AI planning have lead researchers and industry to investigate such technology in real world domain. In fact, the recent efficiency improvement of planning systems and the development of Knowledge Engineering (KE) tools have become a great motivation to investigate and experience real design process of planning applications. From the challenges faced by researches and experts in these applications new requirements and roadmaps emerge for the planning & scheduling community.

The main purpose of this work is to share the experience of designing and investigating a real application in the petroleum industry that can challenges both innovative KE tools and modern planning algorithm. The real problem presented in this paper deals with the planning of the daily activities of a petroleum plant for docking, storing and distributing oil. The planning of these operations is very important to the functioning of refineries and constitutes a complex problem of difficult mathematical modeling (Dahal et al. 2003). When planning over this problem engineers must deal with tankers allocation, docking scheduling, tank volume control, crude oil storage with price maximization (avoiding mixing certain types of crude oils) and minimization of costs. In fact, this problem presents many challenges such as resource allocation, sequencing, scheduling and optimization.

The requirements phases and domain modeling process was performed using the KE tool itSIMPLE - Integrated Tools Software Interface for Modeling PLanning Environment (Vaquero et al. 2007) - in which all the model was built utilizing the UML - Unified Modeling Language (OMG 2005). Due to the size and complexity of this real problem it is indeed necessary to use tools that provide support for the phases of a design process such as requirements acquisition, modeling, testing and plan analysis. Actually, designing the domain in PDDL (Fox and Long 2003) from scratch would have proven extremely difficult and time consuming. Also, we believe that using such a KE tool it contributes to finding better modeling solutions, as well as to identifying relevant domain issues and features that otherwise could not be recognized by a totally action driven specification. The design process described in this paper extends our previous work on modeling this application (Sette et al. 2008) by developing and introducing a model that considered time constraints and also quality-metrics for plan analysis. In addition, two semi-realistic study cases are investigated in order to validate the approach.

This paper is structured as follows. First, we present the domain, its restrictions and requirements. Then we present the design process, including mainly the modeling process, performed the KE tool itSIMPLE. Next, we provided experimental results obtained by using SGPlan (Hsu et al. 2006) to solve two challenging planning problems. This paper ends with some conclusions.

Oil Supply as a Planning/Scheduling Problem

Operations with crude oil involve the unloading of tankers in docking stations into distribution tanks, and the supply of refineries. Since the refineries are constantly consuming oil, the plan must guarantee that, at all moments, the amount of oil in the refineries remains above a minimum level, while minimizing the cost of distribution.

Nowadays, most research work done in this area has utilized mathematical programming in which the models are adapted to mixed-integer linear programming (MILP) or mixed-integer non-linear programming (MINLP) to find solutions to this problem. However, current methods have failed to show feasible solutions or require a great amount of time to solve these problems. Furthermore, MILP methods require the use of linearization, which leads to flaws in the final solutions, while the discretization necessary in MINLP methods greatly increases the size of the problem (Li et al. 2005). Therefore, as described in (Li et al. 2005), there is no reliable efficient and robust algorithm for this real and very important problem in current literature.

In this work, a real problem encountered in one of the main oil supply distribution complexes of Brazil will be investigated under the automated planning perspective. The domain description and requirements was based on the work of Mas and Pinto (2003) and the information provided by Petroleo Brasileiro S.A. (Petrobras), the main petroleum producer and distributor in Brazil.

Crude oil is processed in four refineries in the State of Sao Paulo (Brazil): Paulinia (REPLAN), Sao Jose dos Campos (REVAP), Cubatao (RPBC) and Capuava (RECAP), which are supplied through a pipeline network that leaves the Sao Sebastiao terminal (GEBAST). The system also contains two intermediate substations (SEBAT, in Cubatao, and SEGUA, in Guararema), as well as pumping stations in Rio Prado and Guaratuba. All the crude oil that is consumed by the State of Sao Paulo comes through GEBAST and is distributed by two pipelines: OSVAT and OSBAT. This system is detailed in Figure 1.

In this work we consider only the planning of three main operations daily performed in Sao Sebastiao terminal: docking of oil tankers; oil storage; and distribution of crude oil to refineries. These operations are held in a infrastructure of distribution that consists of a port, refineries and pipelines that carry the oil to the refineries (where the oil will be processed). The port is compound by piers, tanks, and an internal pipeline that connects the two structures. This pipeline system has already been subject of study in the planning community, having appeared in the International Conference on Automated Planning and Scheduling ICAPS'04 as a domain in the fourth International Planning Competition IPC'04. However, while this problem is operational in nature, this paper is concerned with a more strategic issue: the planning and scheduling of crude oil distribution in order to maximize profit leaving the internal pipeline issue apart.

The planning and scheduling of port operations involves several activities such as assignment of tanks to piers, unloading of the tankers to the tanks in the terminal, and unloading of the terminal tanks to the pipelines (Mas and Pinto 2003). The main requirements associated to these activities are directly related to four main elements: tankers, tanks, pipelines and refinery. The main requirements for these elements are described below.

Tanker requirements: The crude oil arrives at GEBAST through oil tankers, which are unloaded at the docking stations and stored in the tanks of GEBAST. Each docking station has a limitation regarding the size of the tankers it can receive.

Furthermore, this unloading operation has to be done quickly and efficiently, since there are severe overstay costs in this operation. Each tanker has a limited time that it can stay docked in the pier and unload without paying overstay costs. Therefore, the planning of this operation should respect this period whenever possible.

Finally, every tanker takes a certain time to dock and to leave the port. In practice, this means that, after the order to dock is given, a period must be waited before unloading operations begin and that a docking station will only be able to receive another vessel a certain period after the exit order is issued to the tanker currently occupying it.

Tank requirements: Petrobras processes several different types of oil in its refineries. Since reserving a tank for each oil type is not practical, the oils are grouped into classes. The crude oil types that belong to the same class can be mixed together without losing value (Mas and Pinto 2003).

At a given moment, a tank can be in either one of three states: inoperative, loading, or unloading. Under no circumstance can a tank be unloading and loading simultaneously. Furthermore, there are some restrictions concerning the presence of brine in the tankers inventories. Since every oil type unloaded at Sao Sebastiao contains brine (even after separation in petroleum production platforms), the tanks must undergo a settling period (during which the tank remains inoperative), having received crude oil from a tanker, before it can send oil to the refineries. During this period, the brine settles in the bottom of the tank. This is done in the tanks of GEBAST because it is not desirable to transport brine through the pipelines or send it the refineries.

In order to prevent the accumulation of volatile components, the tanks operate using a floating roof system. Because a minimum safety level is required in order to avoid damage to these structures, the tanks can not be fully unloaded (Mas and Pinto 2003). This hard restriction is, usually, about two meters, which represents about 15% of the total capacity. Therefore, each tank has a maximum and minimum capacity that must be respected in the planning process of their operation.

Pipeline requirements: The pipelines are used to send oil from the terminal to the refineries that will process it. They are able to transport, simultaneously, more than one crude oil type. During this transport operation, an interface forms between two different oil types resulting in a loss of their properties (Mas and Pinto 2003). Moreover, a pipeline must not be used to unload distinct tanks simultaneously.

Refinery requirements: The refineries have maximum and minimum capacity restrictions that must be respected throughout their operation. However, a model of the refinery operation will not be considered and they will be modeled as continuous consumers of crude oil. It will be assumed, in the short term, that the refineries will have established an average rate of consumption of crude oil.

The Design Process with itSIMPLE

Since the investigated planning application requires a careful design process that involves intensive knowledge acqui-

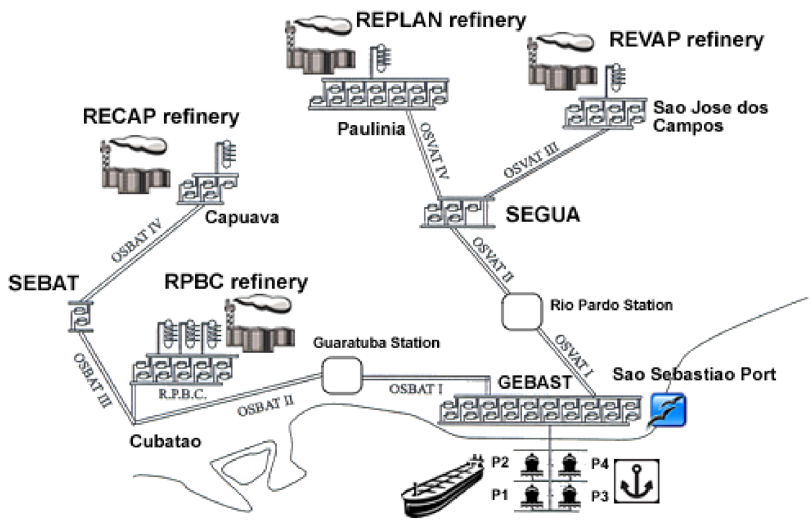

Figure 1: Crude oil distribution infrastructure of Petrobras in the State of Sao Paulo

sition and modeling, a Knowledge Engineering tool called itSIMPLE was used to support the construction and development of a domain model. The itSIMPLE$_{3.0}$ (Vaquero et al. 2007) environment aims to support designers in overcoming problems encountered during the life cycle of planning application projects, mainly at crucial initial phases such as requirements specification, modeling, and analysis phases (Vaquero et al. 2007). This Knowledge Engineering (KE) tool, addressing real-life planning problems, allows users to follow a disciplined design process to create knowledge intensive models of planning domains, from the informality of real world requirements to formal domain models. The suggested design processes for building planning domains models, shown in Figure 2, follows a cycle of phases inherited from Software Engineering and Design Engineering fields combined with real planning domain modeling experiences.

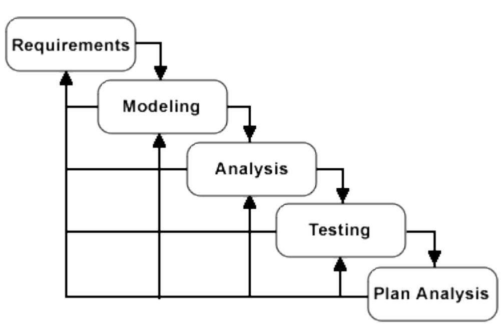

Figure 2: Planning domain design processes in itSIMPLE$_{3.0}$

In this work we are going to focus on four of the main phases of such design process: requirements gathering, modeling, testing with planners and plan analysis.

Gathering requirements

Requirements are gathered and represented using use case diagrams. These diagrams model the domain in the highest abstraction level in which the domain scope is firstly defined. The diagrams usually facilitate the unification of the different viewpoints involved. The Use Case diagram for the Sao Sebastiao terminal oil distribution activities is showed in Figure 3. The domain requirements were represented in the elements in the diagram where each use case receives a full description, pre and post condition, constraints, invariants, flow events and other relevant information.

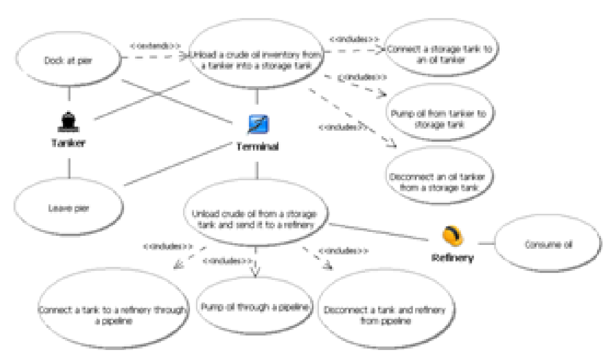

Figure 3: Use case diagram of the Oil Supply domain

As seen in Figure 3, this oil distribution system, which is centered at the terminal, possesses three independent agents (actors in UML): tanker, the terminal itself and the refineries. The actors interact together to perform the tasks required to take the oil from the tankers and deliver it to the refineries.

Domain Modeling

Modeling in itSIMPLE uses UML diagrams such as *class diagram*, *state machine diagram*, and *object diagram*. The class diagram represents the static structure of the planning domain. This diagram shows the existing entities, their relationships, their features, operators (actions) and constraints. Class attributes and associations between classes give a visual notion of the model semantic.

Figure 4 shows the class diagram designed for the oil distribution problem ate Sao Sebastiao terminal. The diagram consists of nine classes that model all the entities relevant to the real problem being modeled. The Domain_Metrics class is a utility class that stores variables that are relevant to all other classes in the model such as interface costs. In this particular case, these variables (corresponding to costs, revenue and time) are used as quality-metrics for the optimization of profit.

The refinery class controls the volume of oil that must be sent to them (properties *volumeNeed* and *volumeSent*). In fact, the refinery would need to be able to deal with continuous variable regarding the volume of oil; however the availability of general planners that handle such domain characteristic constrained the model in this direction.

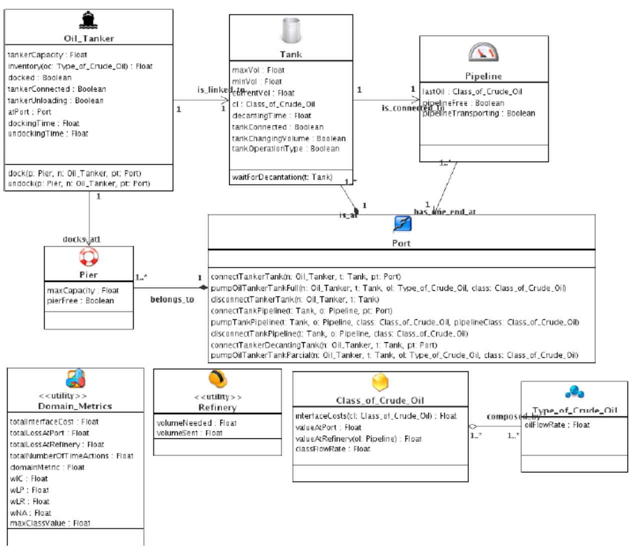

Figure 4: Class diagram of the Oil Supply domain

The dynamics of actions are specified in the state machine diagram, in which it is possible to define the pre and post conditions for the operators defined in the class diagram. In the itSIMPLE tool all the pre and post condition are defined by using the formal constraint language called *Object Constraint Language* (OCL) (OMG 2003), a predefined language of the UML.

Usually every class in the class diagram (especially those that change states during a plan execution) has its own state machine diagram. One state machine diagram does not intend to specify all changes caused by an action. Instead, it details only the changes that it causes in an object of a specific class. Figure 5 shows the state machine diagram for the class Tanker.

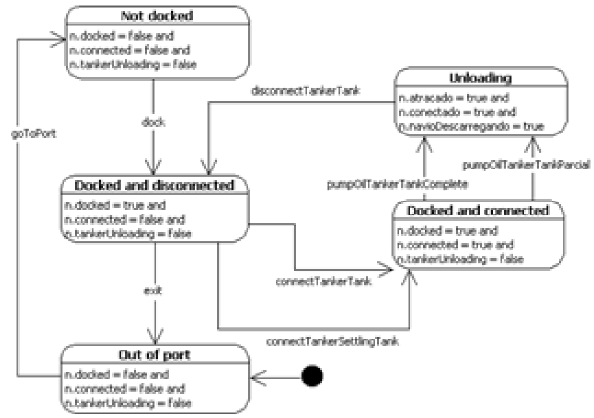

Figure 5: State machine diagram of the Oil Supply domain

In itSIMPLE, problems are modeled using the object diagrams. All the planning problems for the current Oil Supply domain were modeled in these diagrams.

A problem statement in a planning domain is characterized by a situation where only two points are known: the initial and goal state. The diagram itSIMPLE uses to describe these two states is called Object Diagram or Snapshots. A snapshot is a picture of the system at a specific state. It is also seen as an instantiation of the domain structure defined in previous diagrams. The instantiation defines four main aspects: how many objects are in the problem; which are their classes; what are the values of each object attribute; and how they are related with each other.

A planning problem is composed by two Object Diagrams: one describing an initial state and, another, the goal state (partial or complete). Figure 6, shows the initial snapshot for a planning problem example used in this work during the test with planners stage, while Figure 7 shows the goal state for the same problem. Indeed, it would be an arduous task to provide a complete definition of the goal state for the problem studied in this paper and, therefore, partial goal stated were used. The partial goal states consisted of all tankers unloaded (zero inventory) and undocked. The states of all other objects in the problem were left undefined.

Besides the objects diagrams for defining initial and goal states, we also define the metric function to be optimized in every planning situation. A optimized plan must considered four main aspects: (1) the cost of oil interfaces, which measures the losses at the interface between different quality oils in a pipeline; (2) the profit generated by storing oil in the refinery which measure the value of the oil sent (low or high quality); (3) the profit resulted from storing oil at the port among the tanks in the terminal; and, (4), the total time (minimizing the number of temporized actions). In this application, the optimization approach of these aspects considers four domain variables, with their respective weights, in a linear equation to be minimized during planning process. This four variables, from the Domain_Metrics class, are the following:

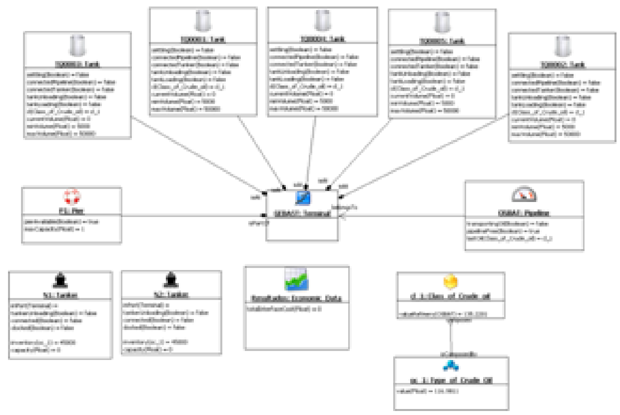

Figure 6: Initial state of a sample problem

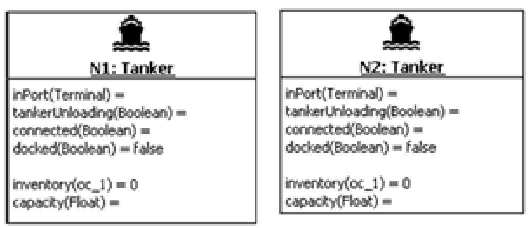

Figure 7: Goal state of a sample problem

- totalInterfaceCost: this variable holds the sum of interface cost during a plan.
- totalLossAtRefinery: represents the losses related to sending low quality oils to the refineries.
- totalLossAtPort: represents the losses related to storing low quality oils in high quality classes of tanks. Classes of oils must be properly mixed in order to maintain their high quality properties.
- totalNumberOfTimeActions. this variable measures the number of durative-actions as an alternative to totaltime.

Neither the cost of operations at the pier (which depend on docking time of each tanker), nor the cost of overtime docking (costs applied when tankers exceed the established period) were considered in this work due to limitation on available general planners while dealing with continuous properties/time.

Model Testing with Planners and Plan Analysis

The itSIMPLE can automatically generate a PDDL model from the UML representation based on the set of diagrams. Besides the automated translation process, the tool can communicate with most recent planning techniques in order to test planning domain models in a integrated design environment. In this application the planners must selected based on the resulting PDDL model requirements that go beyond the classical approaches. Next section discuss the chosen planners.

In order to analyze the output of these plans, itSIMPLE provides two main support tools for plan analysis: plan simulation and plan validation. Plan simulation is performed by observing a sequence of snapshots (UML object diagrams), state by state, generated by applying the plan from the initial state to the goal state. The tool highlights every change in each state transition as described in (Vaquero et al. 2007).

For the plan validation, itSIMPLE provide XY chart that represents the evolution of selected variables of the domain such as those that affect the quality of a plan (metrics).

The following section describes the experiments done with the domain model during test with planners and plan analysis.

Experimental Results

In this work, two planners were selected, among 10 available planners, based on the minimal requirements of the domain model such as numeric properties, durative actions, and metrics. The investigated planners were MIPS-XXL (Edelkamp, Jabbar, and Nazih 2006) and SGPlan (Hsu et al. 2006), two outstanding planning algorithms that participated in recent planning competitions. In order to evaluate if both planners could generate valid plans for the model, simplified problems were simulated and tested. During these initial simulations, some plans provided by MIPS-XXL had an invalid sequence of actions in which action duration were not respected. Considering these initial results for simplified problems, only SGPlan was used in the experiments of this work. Since SGPlan does not treat optimization functions we used the following approach: in each experiment we run SGPlan multiple times and in each iteration we try to find a better solution by adding a request for lower value of the optimization function.

In order to evaluate the domain described in this paper and the generated plans from SGPlan, we created two case studies. These cases reflect real scenarios (even with some simplifications) of the daily activities in the port of Sao Sebastiao, Sao Paulo. The scenarios were based and inspired by those presented and studied in (Mas and Pinto 2003). The first case investigate a simple situation and the second presents a more realistic scenario.

Case Study 1

This case study represents a simplified scenario in the port. The planning problem contains one port that receives three tankers: Reboucas, Front_Brea, and Pedreiras. These tankers can be docked in two available piers. The tankers are unloaded using five tanks in the port. The oil stored in the tanks must be sent (respecting the settling period) to a refinery throughout one pipeline.

The three tankers are loaded with crude oil that must be delivered at the port. The Reboucas tanker carries a type o crude oil called oc38 and Front_Brea tanker carries an oil called oc05. The Pedreiras tanker carry two oil types, oc08 and oc27. These tankers can dock in two piers P1 and P2.

The available tanks are available to receive the crude oil. Each one of these tanks store a distinct class of oil. These classes must be considered to maintain the quality of the oil while mixing different types. When loaded, the tanks must remain inoperative for 24 hours waiting the brine to reach the bottom. Finally the oil must be send to the refinery taking the interface cost into account. This problem is illustrated in Figure 8, along with the main oil flow.

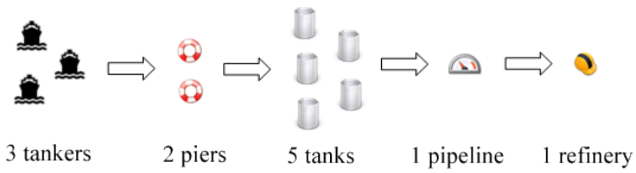

Figure 8: Case study 1 illustration

In fact, this case study uses real data regarding the volumes of oils, types and classes of crude oils, tankers, tanks, costs and pipelines. Figure 9 illustrates a partial view of the initial state for this scenario.

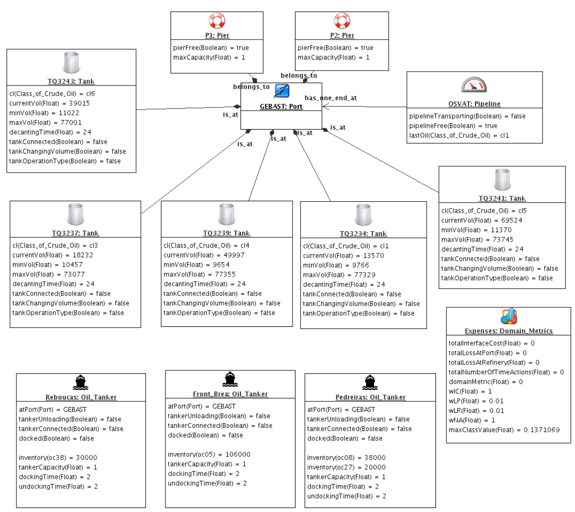

Figure 9: Initial state of case study 1

The described scenario was sent to SGPlan that generated a satisfactory plan in the second iteration. Table 1 shows that in the second iteration, SGPlan generated a shorter plan with a better quality. Figure 10 shows the resulting plan that brings a better solution.

	1st iteration	2nd iteration
time	0.17s	233.5s
Number of actions	27	24
Metric value	30.636	26.427

Table 1: Data of the solution for Case 1

In order to evaluate the solution, XY chats were used to check the changes of oil level in the tankers, tanks, and refinery. The charts in Figure 11 represents the evolution of oil levels (m^3) in some of these domain elements. This Figure

```
0.001: (DOCK P2 FRONT_BREA GEBAST) [2.0000]
0.002: (CONNECTTANKPIPELINE TQ3241 OSVAT GEBAST) [0.0000]
0.003: (PUMPTANKPIPELINE TQ3241 OSVAT CL5 CL1) [13.0771]
0.004: (DOCK P1 PEDREIRAS GEBAST) [2.0000]
2.005: (CONNECTTANKERTANK PEDREIRAS TQ3239 GEBAST) [0.0000]
2.006: (PUMPOILTANKERTANKFULL PEDREIRAS TQ3239 OC27 CL4) [3.0874]
5.094: (DISCONNECTTANKERTANK PEDREIRAS TQ3239) [0.0000]
5.095: (CONNECTTANKERTANK PEDREIRAS TQ3237 GEBAST) [0.0000]
5.096: (PUMPOILTANKERTANKFULL PEDREIRAS TQ3237 OC08 CL3) [5.6205]
10.718: (DISCONNECTTANKERTANK PEDREIRAS TQ3237) [0.0000]
10.719: (UNDOCK P1 PEDREIRAS GEBAST) [2.0000]
12.720: (DOCK P1 REBOUCAS GEBAST) [2.0000]
13.090: (DISCONNECTTANKPIPELINE TQ3241 OSVAT CL5) [0.0000]
13.091: (CONNECTTANKERTANK FRONT_BREA TQ3241 GEBAST) [0.0000]
13.092: (PUMPOILTANKERTANKPARCIAL FRONT_BREA TQ3241 OC05 CL5) [8.5457]
14.724: (CONNECTTANKERTANK REBOUCAS TQ3243 GEBAST) [0.0000]
14.725: (PUMPOILTANKERTANKFULL REBOUCAS TQ3243 OC38 CL6) [4.2589]
18.985: (DISCONNECTTANKERTANK REBOUCAS TQ3243) [0.0000]
18.986: (UNDOCK P1 REBOUCAS GEBAST) [2.0000]
21.643: (DISCONNECTTANKERTANK FRONT_BREA TQ3241) [0.0000]
21.644: (CONNECTTANKERTANK FRONT_BREA TQ3234 GEBAST) [0.0000]
21.645: (PUMPOILTANKERTANKFULL FRONT_BREA TQ3234 OC05 CL1) [5.9768]
27.623: (DISCONNECTTANKERTANK FRONT_BREA TQ3234) [0.0000]
27.624: (UNDOCK P2 FRONT_BREA GEBAST) [2.0000]
```

Figure 10: The plan for case study 1

shows the viability of the plan in which all storage level constraints are respected. It is possible to see that the refinery maintain its reservatory in an adequate level for a reliable supply.

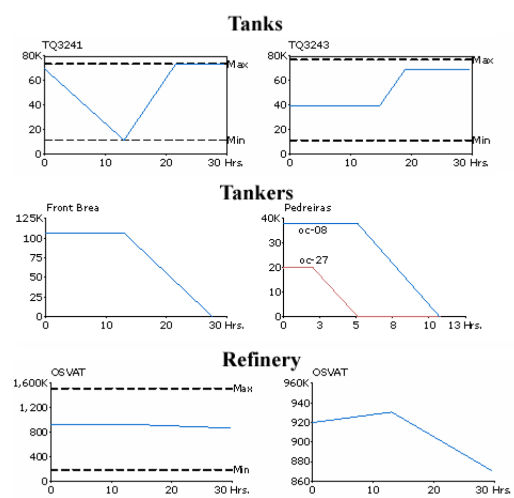

Figure 11: Oil levels evaluation

As mentioned before, the current model does not considered the docking period cost and the cost of overtime docking. However, we have analyzed the solution based on the metrics. Figure 12 shows the period order for each tanker (48 hours, blue bars) and the de facto time used by tankers in the solution given by SGPlan. The figure shows that the unloading activities of tankers were perform efficiently; in any case the time was exceeded, what would increase the costs. It also shows that the approach of using *totalNumberOfTimeActions* as a metrics worked as an preliminary alternative for dealing with the continuous restriction of planners. As a result, case study 1 shows a promising result for a more realistic case. Next section presents a more realistic case study.

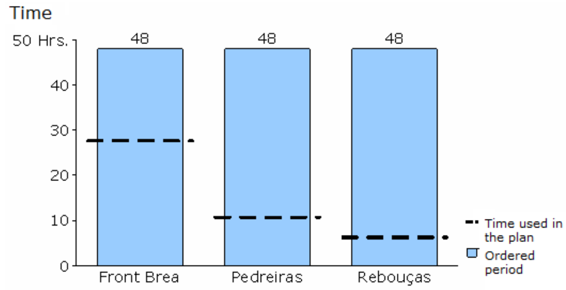

Figure 12: Time used vs. Ordered period of tanker operation

Case Study 2

This case aims to evaluate the model based on a realist problem encountered daily in the Sao Sebastiao port. In this scenario, seven tankers are considered: Reboucas, Front_Brea, Pederiras, Muriae, Vergina_II, North_Star, and Presidente. These tankers can unload the oil into ten tanks such as TQ3243, TQ3237, and TQ3238. For this problem, four pier are made available (P1, P2, P3, and P4). The delivered oil must supply a refinery considering the constraints on the tanks (24h inoperative period) and also the quality-metrics. Figure 13 illustrates the new scenario.

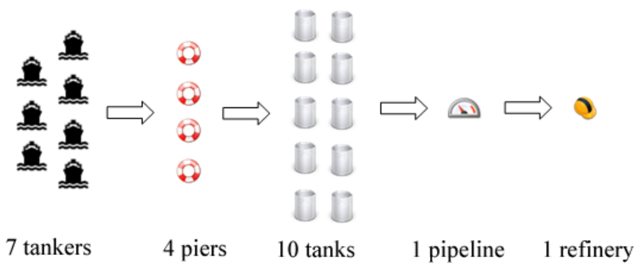

Figure 13: Case study 2 illustration

The initial and goal states are represented as object diagrams that contains all definition for the elements in the planning problem. Figure 14 shows a partial view of the initial state for the current scenario.

As in the previous case, the problems is automatically translated to a PDDL model that was sent to the SGPlan. As opposed to case 1, SGPlan generated a valid plan in the first interaction but it was unable to find a better solution in a second iteration. Table 2 shows general data about the plan generated by SGPlan. A partial list of plan's actions is shown in Figure 15.

	1st iteration
time	210.09s
Number of actions	88
Metric value	73.789

Table 2: Data of the solution for Case 2

In order to investigate the oil levels in tankers, tanks and refinery, XY charts were also analyzes to check the SGPlan

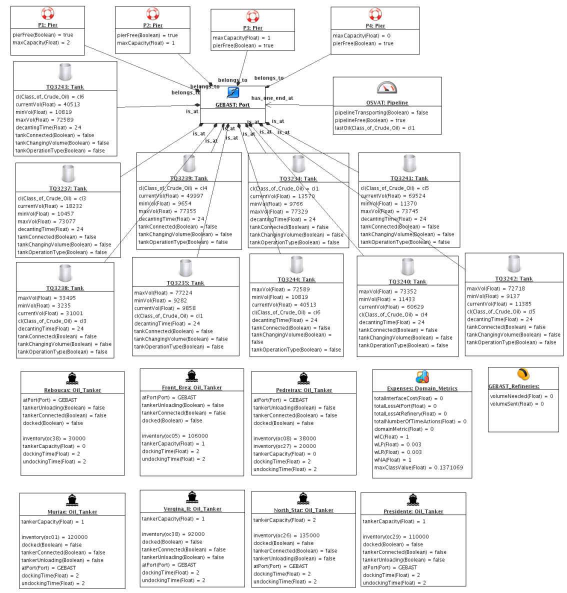

Figure 14: Initial state of case study 2

```
0.001: (CONNECTTANKPIPELINE TQ3240 OSVAT GEBAST) [0.0000]
0.002: (DOCK P1 NORTH_STAR GEBAST) [2.0000]
0.003: (DOCK P2 PRESIDENTE GEBAST) [2.0000]
0.004: (PUMPTANKPIPELINE TQ3240 OSVAT CL4 CL1) [11.1077]
2.005: (CONNECTTANKERTANK NORTH_STAR TQ3242 GEBAST) [0.0000]
2.006: (CONNECTTANKERTANK PRESIDENTE TQ3237 GEBAST) [0.0000]
2.007: (PUMPOILTANKERTANKPARCIAL NORTH_STAR TQ3242 OC26 CL5) [9.5833]
2.008: (PUMPOILTANKERTANKPARCIAL PRESIDENTE TQ3237 OC29 CL3) [8.4663]
10.475: (DISCONNECTTANKERTANK PRESIDENTE TQ3237) [0.0000]
...
```

Figure 15: The plan for case study 2

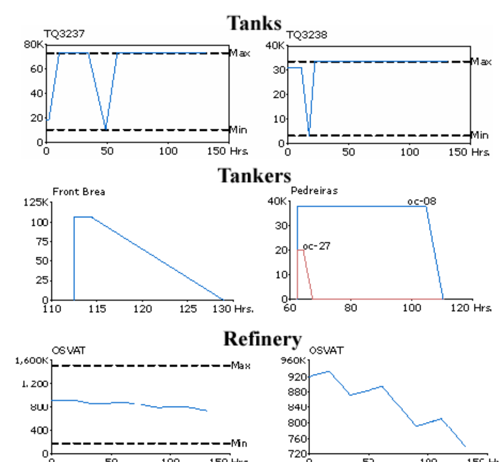

Figure 16: Oil levels evaluation in case 2

solution. Figure 16 illustrates some of the charts used in this approach.

We also have analyzed the solution to this realistic scenario based on the described metrics. Figure 17 shows the period order for each tanker compared to the time used by the tankers in the generated plan. This analysis chart also emphasizes the efficiency of SGPlan's solution, i.e., the tankers operations remained below the ordered period. In fact, not all tankers finished their activities at the pier during the estimated 48 hour period. Presidente tanker was the only one that had to remain docked overtime. This indeed affects the final cost of the whole port operation; however, since the other tankers do not used fully the established period, an optimization and revision process on the regular hired docking time would, on the other hand, decrease the cost of docking operations.

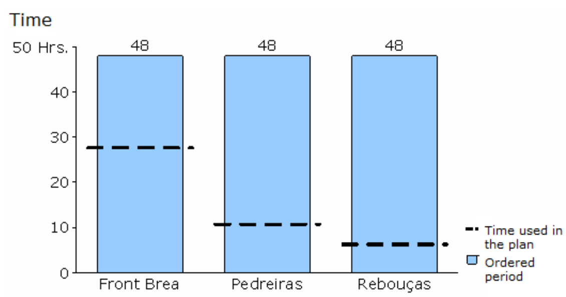

Figure 17: Time used vs. Ordered period of tanker operation in case 2

Even with some restriction in the model concerning continuous time, the approach showed satisfactory results in two challenging planning problems. The domain discussed in this paper gathers features that usually challenges most of recent planning techniques, such as time, large amounts of numeric resources, quality metrics and optimization. Moreover, these are some of features commonly found in real planning problems. As discussed in our previous work on this domain (Sette et al. 2008), few planners can handle domain models that combined all these features, but these applications give a clear road map for planning algorithm development.

Conclusion

In this work, a real complex planning problem, such as the planning/scheduling of the daily activities of a crude oil distribution plant, was presented following a AI planning approach. We described the design process used for building a domain model along with the KE tool itSIMPLE that supports mainly the initial phases of the design and also the plan analysis activity. As an extension of the work done in (Sette et al. 2008), in this paper we have investigated the domain model for the oil distribution activities in the Sao Sebastiao port (Sao Paulo) that considers time constraints and quality-metrics that guide the plan efficiency and quality.

In order to validate the model in real scenarios, two case study were describe and tested using the SGPlan. The first one considers a semi-realistic scenario and the second brings a realist case. Both case study used real data for a more accurate investigation. The planner was chosen based on the domain model requirements and also based on its capability of dealing with the domain features (durative actions, numeric variables, and metrics). The metrics considered in this problem focus on the minimization of different parameters such as allocation of different oil types decreasing the oil quality, interface losses in pipelines, and time spent for the activities.

Experimental results showed that in both cases SGPlan was able to provide satisfactory solutions for managing the activities of docking tankers, distributing and allocating crude oil to be sent to a refinery throughout a pipeline system. Moreover, the solution also showed to be efficient according to non-considered cost such as cost of docking period and overtime docking. In fact, experience brought by this model have motivated the improvement of itSIMPLE towards time-based models. The resulting model will be made available in order to share our results on this domain.

References

Dahal, K.; Galloway, S.; Burt, G.; McDonald, J.; and Hopkins, I. 2003. Port system simulation facility with an optimization capability. *International Journal of Computational Intelligence and Applications* 3(4):395–410.

Edelkamp, S.; Jabbar, S.; and Nazih, M. 2006. Large-scale optimal pddl3 planning with mips-xxl. In *In 5th International Planning Competition Booklet (IPC-2006*.

Fox, M., and Long, D. 2003. Pddl2.1: An extension of pddl for expressing temporal planning domains. *Journal of Artificial Intelligence Research (JAIR)* 20:61–124.

Hsu, C. W.; Wah, B. W.; Huang, R.; and Chen, Y. X. 2006. New features in sgplan for handling soft constraints and goal preferences in pddl3.0. In *Proc. Fifth International Planning Competition, International Conf. on Automated Planning and Scheduling*.

Li, J.; Wenkai, L.; Karimi, I. A.; and Srinivasan, R. 2005. Robust and efficient algorithm for optimizing crude oil operations. In *American Institute of Chemical Engineers Annual Meeting*.

Mas, R., and Pinto, J. M. 2003. A mixed-integer optimization strategy for oil supply in distribution complexes. *Optimization and Engineering* 4(1-2):23–64.

OMG. 2003. *UML 2.0 OCL Specification m Version 2.0*.

OMG. 2005. *OMG Unified Modeling Language Specification, m Version 2.0*.

Sette, F. M.; Vaquero, T. S.; Park, S. W.; and Silva, J. R. 2008. Are automated planners up to solve real problems? In *Proceedings of the 17th World Congress The International Federation of Automatic Control (IFAC'08), Seoul, Korea*, 15817–15824.

Vaquero, T. S.; Romero, V.; Tonidandel, F.; and Silva, J. R. 2007. itSIMPLE2.0: An integrated tool for designing planning environments. In *Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS 2007). Providence, Rhode Island, USA.*