Robots in Retirement Homes: Applying Off-the-Shelf Planning and Scheduling to a Team of Assistive Robots (Extended Abstract)*

Tony T. Tran†, Tiago Vaquero‡, Goldie Nejat†, and J. Christopher Beck†
† University of Toronto, Toronto, Ontario, Canada
‡ Massachusetts Institute of Technology, Cambridge, Massachusetts, USA
{tran,nejat,jcb}@mie.utoronto.ca, tvaquero@mit.edu

Abstract

We investigate Constraint Programming and Planning Domain Definition Language-based technologies for planning and scheduling multiple robots in a retirement home environment to assist elderly residents. Our robotics problem and investigation into proposed solution approaches provide a real world application of planning and scheduling, while highlighting the different modeling assumptions required to solve such a problem. This information is valuable to the planning and scheduling community as it provides insight into potential application avenues, in particular for robotics problems. Based on empirical results, we conclude that a constraint-based scheduling approach, specifically a decomposition using constraint programming, provides the most promising results for our application.

1 Introduction

The aging of global populations has had, and will continue to have, profound impact on our society [United Nations, 2002] with one particular challenge being the physical, cognitive, and psychological welfare and well-being of the elderly. Without an increase in the number of caregivers, the growth of the older population will lead to a strain on existing personnel to meet these needs. To address the lack of human resources, human-robot interaction (HRI) and robot companionship have been proposed and shown to have positive results on the human psychological state [Banks *et al.*, 2008].

Our long-term project is the deployment of intelligent mobile robots in retirement homes to engage residents in stimulating recreational activities [Booth *et al.*, 2016; Louie *et al.*, 2014a; 2014b; 2015; Li *et al.*, 2016; Mohamed and Nejat, 2016]. We have designed the robot, Tangy, to: 1) navigate using a laser range finder and 3D depth sensors, 2) detect users with 2D cameras, and 3) interact with users through speech, gestures, and a touch screen. While the implementation of the robot behaviors addresses robotics challenges (e.g., sensing, HRI, person and activity recognition), herein we focus on the planning and scheduling of the daily activities of multiple socially assistive robots such as Tangy. These plans and schedules are to be generated prior to the start of a day and executed by the team of robots during the day.

This paper studies the modeling and solving of a particular application. Such case studies serve as valuable feedback for researchers focusing on the theory and algorithms which form the core of planning and scheduling research.

The main contributions of this work are:

- Three modeling approaches for a complex multi-robot HRI problem using two different solving technologies: Planning Domain Definition Language (PDDL) and Constraint Programming (CP);
- One of the first applications of CP to a multi-robot planning and scheduling problem;
- The development of a CP-based decomposition that outperforms the other tested methodologies.

2 Problem Definition

We need to create a daily schedule for a team of robots in a retirement home environment. In this section, the main elements of the proposed problem are defined. The parameters and constraints were obtained from meetings with directors, healthcare professionals, and residents from Toronto area retirement homes [Louie *et al.*, 2014a; 2015].

2.1 The Retirement Home Environment

The retirement home is discretized into a set of locations, L, representing different rooms in the environment with a known distance between any two locations l and m, denoted as d_{lm}.

For a set of users, U, each user, $u_i \in U$, has a personal profile that specifies a schedule, including his/her location and availability throughout the day. The user profile also represents preferences specifying that the user, u, wishes to participate in between att_min_u and att_max_u multi-user HRI activities (Bingo games).

We are also given a set of robots, R. Each robot, $r_i \in R$, starts the day at a charging station and is capable of moving in the retirement home environment at a fixed speed, v_i. The robot facilitates the HRI activities by traveling to the appropriate location and interacting with the user(s) for the required amount of time. While the robot is traveling and performing activities, it consumes battery power at a rate dependent on

*This paper is an extended abstract of an article in the *Journal of Artificial Intelligence Research* [Tran *et al.*, 2017].

the task. The battery level, bl_i of a robot, r_i, must stay between $bl\_min_i \leq bl_i \leq bl\_max_i$. During the day, a robot may need to travel to and recharge at a charging station.

The HRI activities are telepresence sessions and Bingo games. A set of telepresence sessions, S, represent the single-user activities that are required to be scheduled during the day. Each telepresence session, $y \in S$, is associated with a user, u, a duration, dur_y, and a set of valid time intervals when it can occur. A set of Bingo games, G, represents the multi-user activities, optional tasks that add value to the daily schedule for users. A Bingo game, $g \in G$, is characterized by its location, the duration of the game dur_g, and a set of valid time intervals when it can be played. For each Bingo game, g, the number of participants must be between three and ten. The participating users must be available during the time of the Bingo game and each player must be reminded of the game by a robot 15 to 120 minutes before it begins.

2.2 Input and Goal

The inputs to the problem are the sets of locations, L; users, U (with their profiles); charging stations, $K \subset L$; available robots, R (with their initial locations, velocity, and battery level and consumption details); the requested telepresence sessions, S; and possible Bingo games, G. The goal is to create a plan of robot tasks in which all the requested telepresence sessions are scheduled and the requested Bingo games and reminders are scheduled, if possible, given that user attendance preferences have to be satisfied. Furthermore, the participating users must be chosen for each Bingo game as part of the problem solving. At the end of the day, all robots must be at a location with a charging station.

As a multi-objective optimization problem, we want to: 1) perform as many Bingo games as possible, 2) have as many users playing Bingo as possible, 3) provide reminders as close as possible to the game times, and 4) expend as little battery power as possible. The objective function considered is:

$$\min \quad f = 500B + 1000(|U| - P) + \delta + \epsilon. \tag{1}$$

Here, B is the number of Bingo games played and P is the total number of participants across all Bingo games. The values of δ and ϵ represent the sum of time between all reminders and their corresponding Bingo games and the total energy usage of all robots, respectively.

3 Solution Methods

We present three different approaches to solving the robot task scheduling problem: PDDL-based planning, constraint-based scheduling, and a constraint-based scheduling decomposition. In the interest of space, we only present partial models. For the complete models, and five alternative PDDL models that do not perform as well as the one presented here, see our full paper [Tran *et al.*, 2017].

Planning and scheduling differ in the abstractions used to model tasks in each paradigm. In the former, it is sufficient to model tasks by utilizing operators, which dictate how the state of the world may be changed. An operator is instantiated to create a ground action and the planner decides how many times to instantiate each operator and their sequence to reach a goal state. For example, to charge the battery on a robot, the planner has access to a battery recharging operator that can be executed by any robot as many times as necessary.

In contrast, typical scheduling solvers require that every potential task is modeled as an optional task [Laborie, 2003]. This approach allows one to model a task but not necessarily execute it. For example, the maximum number of charging tasks must be known prior to scheduling in order to know how many optional tasks to create. The same must be done for the reminders and Bingo games since it is not known a priori which users are assigned to a Bingo game or whether a game will be played or not. The bounds introduced for the scheduling models provide the schedulers with additional information not available to the planners.

3.1 PDDL-Based Planning

The first methodology uses PDDL [Ghallab *et al.*, 1998], specifically PDDL2.2 with processes from PDDL+. The main classes of the model are: *Location*, *ChargingStation*, *Robot*, *User*, *TelepresenceSession*, *BingoGame*, and *Global*.

A robot can perform seven operations: *move*, *recharge*, *remind*, *do_telepresence*, *play_Bingo*, *interact*, and *skip_Bingo*. Other than *recharge* and *do_telepresense*, all other operators are used to model Bingo games. The *move* operator moves a robot from one location to another to seek users for reminders and to facilitate a Bingo game in the games room, while having a duration and energy consumption based on the distance.

Each Bingo game has the properties *dur*, to represent duration; *not_done* and *done*, to represent whether the game has been performed; and timed-initial literals *must_be_done_during*, to represent the time windows in which the task can be performed. In addition to the properties of the games introduced in the problem description, we have added the fluents *p_num* and *p_cur* to control the number of users reminded by the robots and the number of users playing the game, as well as *delivery_time* to control the time at which each user is reminded about the game.

In the *remind* operator, a robot must be ready to perform the task and the user has to be available at the same location as the robot. Since the users are moving in the environment, we model their location over time and require that the robot and user be in the same location then the reminder occurs. The time of the reminder is recorded in the fluent *delivery_time*, which becomes a condition for the Bingo operators.

Modeling the time between the delivery of a reminder and its associated Bingo game is done by *processes* [Fox and Long, 2006]. A process (called *clock_ticker*) models an exogenous activity that is triggered for as long as a condition holds (in this case the fluent *can_start_clock*), regardless of the action selection process. This mechanism allows us to increment the fluent *current_time* one minute at a time, simulating the passage of time in discrete one-minute intervals.

In order to facilitate a game after the reminders, a robot has to first start the *play_Bingo* action, then it has to concurrently perform the *interact* action with each participant. The *play_Bingo* action can only finish when the robot has performed the *interact* action with all assigned players. The *interact* action is used to ensure that users are participating in the Bingo game for the duration of the game.

In the scenario where a Bingo game is not played, the action *skip_Bingo* is performed to ensure that the Bingo activity is considered *done* for the goal state.

3.2 Global-CP

The second methodology uses CP and is called Global-CP.

Each robot task is represented by an interval variable [Laborie, 2009], a_j, defined by a start time, end time, and size, which refers to the battery power required to perform the task. Other than the telepresence sessions, these tasks are optional; i.e., $presenceOf(a_j)$ can be equal to 1 (performed) or 0 (not performed). Furthermore, each HRI-related task has a number of *clone* tasks, which are required to model the alternative robots that can complete the tasks. For each task j, there are $|R|$ additional tasks indexed by i and denoted by α_{ij}.

The scheduler must decide which robot executes each of the executed tasks via the *clone* tasks. We link these tasks with an *alternative* constraint, $alternative(a_j, \{a_{1j}, \ldots, a_{|R|j}\})$ that ensures that if a task a_j is present in the schedule, then exactly one other task from the set of tasks $\{a_{1j}, \ldots, a_{|R|j}\}$ is also present. Thus, if a task a_j is scheduled, then an appropriate task a_{ij} is also present and this task corresponds with robot i which is assigned to execute the task.

To ensure that robots perform at most one task at a time, a cumulative function, rc_i is used for each robot i. Cumulative functions are piecewise functions over time with discrete value changes made at the start and end times of interval variables [Laborie, 2009]. We use a *pulse* effect for every task a_{ij} on cumulative function rc_i, which increases (decreases) the cumulative function by one at the start (end) of the interval variable. By restricting $rc_i \leq 1$, we ensure that each robot does not perform multiple tasks in parallel.

Similarly, users can only be active in one HRI activity at a time. A cumulative function, uc_u, is used, where all tasks pertaining to a user u must also exhibit a *pulse* effect. Additionally, these tasks must be restricted to only occur during times when the user is available. We make use of calendars available in CP Optimizer to restrict the start times of a task.

One complication of our problem domain is the representation of robot and user movement within the environment. Typically, travel times are represented as minimum separation requirements between two tasks, where the separation must be greater than or equal to the time needed to travel between locations [Pesant *et al.*, 1998]. However, because the robots must interact with users and the users themselves move within the environment, the exact location of and therefore, the travel time before, an interaction depends on the time that the action takes place. Our approach is to create additional tasks for each reminder based on the possible locations for a reminder for a particular user throughout the day. It is then possible to treat travel times as sequence-dependent setup times as is standard in scheduling (e.g., [Tran *et al.*, 2016]) to ensure that the robot is given sufficient time for travel.

To model the interaction of users in Bingo games, we create an additional set of $|U|$ optional interval variables for each Bingo game. This set is used to ensure that a user who is assigned to a Bingo game will participate in the game at the appropriate time (similar to the *interact* operator from the PDDL model). A user who has not been reminded of a Bingo game will not participate in the game. However, if a user was reminded, then the scheduler must ensure that the corresponding user-Bingo game interaction task is performed. This task will have a pulse effect on the relevant user's cumulative function, uc_u, similar to how other interval variables affect the availability of a robot. Additionally, we constrain the start time of executed reminder tasks to be between 15 and 120 minutes before the appropriate Bingo game task by bounding the separation between the two tasks.

3.3 CP-Based Decomposition

Smith *et al.* [2000] noted that one of the weaknesses of scheduling is the inability to adequately handle environments with cascading effects. An action with a cascading effect changes the system state and leads to requirements for one or more other actions. For example, if a user is to play a Bingo game, he or she must be reminded of the game. However, if a user is not participating in a Bingo game, the reminder should not take place. Due to such dependencies, the scheduling model becomes very large since we create alternative interval variables for every possible action. We propose a decomposition of the CP model, Decomposed-CP, that attempts to improve upon Global-CP and better handle cascading effects.

The decomposition is comprised of two stages: a master problem and a sub-problem. The master problem is the Global-CP model, but with a simplified objective function: $\min f = |U| - P$. The master problem solution is feasible for the complete problem, however, it may be of poor quality since most of the objective function is ignored.

The solution of the master problem gives an assignment of users to Bingo games that is used in the sub-problem. Only games that were played in the master problem are available to be played in the sub-problem with the same players, but without fixed start-times. The upper bound for the number of charging tasks per robot is set to the total number of charging tasks across all robots in the master solution. We choose this upper bound to allow for some flexibility in changing how often a recharge occurs, while guaranteeing that a feasible schedule exists. The objective function of the sub-problem is the original objective function (Equation 1). Note that in this decomposition, the optimal assignment of games and the number of recharges might not be optimal for the original problem.

When using the Decomposed-CP model, we set a time limit for the master problem that is half the total time limit. We switch to solving the sub-problem when either the master problem has been solved to optimality or when the time-limit is reached; whichever occurs first.

Table 1: The number of objects in the five scenarios.

Scenario	Users	Robots	Telepresence	Bingo
1	5	2	2	1
2	10	2	4	2
3	15	3	6	3
4	20	3	8	4
5	25	4	10	5

Table 2: Empirical results for the five scenarios tested. Results are not shown when no feasible solutions were found.

Model	Scenario	Runtime (s)		Participants		Objective Value	
		first	last	first	last	first	last
PDDL Planning	1	0.04	786.12	0	3	5,506.13	2,070.75
	2	0.18	0.88	0	0	11,019.38	11,016.66
	3	0.84	9.38	0	0	16,530.31	16,525.73
	4	2.18	23.84	0	0	22,024.92	22,024.92
	5	7.24	89.46	0	0	27,557.30	27,554.54
Global-CP	1	0.08	9.26	0	5	5,623.00	192.00
	2	1.96	33.84	6	9	4,549.00	1,243.00
Decomposed CP	1	0.05	0.23	5	5	486.00	192.00
	2	0.08	67.49	6	10	5,039.00	847.00
	3	0.36	37.87	4	15	12,296.00	1,213.50
	4	0.41	2,567.80	10	20	12,107.00	1,430.50
	5	0.37	3,382.83	4	25	23,494.50	1,929.00

4 Empirical Evaluation

We consider a retirement home environment in which residents undertake several activities in different locations during the day. We assume that each user has seven one-hour non-interruptible activities (e.g., physiotherapy, doctor's appointment, family visit, nap, meals). Other, interruptible, activities (e.g., walk in the garden, read in a common area) allow robot interactions, but in various locations. At least one interruptible activity is assumed for each user. We analyze the proposed models for five full-day scenarios representing the requirements of the retirement home, but with a varying number of users, robots, and HRI activities (see Table 1). User schedules were obtained from our collaborating retirement homes.

We run each model on the five scenarios using a 64-bit Ubuntu Linux machine with 12 GB of memory and a one-hour time limit. The OPTIC planner [Benton *et al.*, 2012] is used to solve the PDDL planning model. The CP models are solved using IBM ILOG CPLEX CP Optimizer 12.6.2. We measure runtime, number of participants in Bingo games, and the objective function as performance metrics.

The performance of each methodology is presented in Table 2. The PDDL solver always finds a feasible solution, but these plans are low quality as they do not contain any Bingo games except for scenario 1. OPTIC struggles to improve upon the initial solution. In contrast, Global-CP is unable to find a feasible solution within one hour for scenarios 3 to 5, but, for the smaller scenarios, the solution quality is better and does improve over time. Thus, the PDDL solver behaves by prioritizing feasibility while mostly ignoring the objective function and the CP solver more aggressively optimizes the objective function, but can do so at the cost of not being able to find a schedule.

Decomposed-CP performs the best as it is able to consistently find solutions of high quality. The decomposition is able to find schedules with the maximum number of possible participants for every scenario and in general does so very quickly. Once a partial schedule is found with decisions made regarding the presence of a Bingo game and its players, the problem becomes significantly easier as many actions with cascading dependencies are eliminated. There is a large reduction in the number of charging tasks (about 98%) and reminder tasks (between 80% and 96%).

5 Conclusion

The properties of the retirement home environment create a complex problem to solve. Based on numerical experiments, we find that CP, in particular a decomposition model using CP, is the most suitable technology to use in our system.

CP is better equipped than PDDL planning for handling optimization, but can struggle to find feasible schedules for larger problems. We found that CP is better at obtaining high quality solutions, but it is interesting to note that the best approach is to only consider a simple objective function. Although the large number of optional tasks used in the CP model is also a culprit for the poor performance, we suspect for similar problems with complex objective functions, it is easier and likely just as effective to decompose a CP model to handle the objective function in stages rather than to remove or reduce the optional activities. We conclude that CP is better suited for our application; however, we do not claim that CP will be better than PDDL-based planning for all robot applications. To state one obvious case, CP is an inappropriate choice if the bound on the number of tasks to perform is either very poor or undefined.

Our full paper [Tran *et al.*, 2017] presents more details of our work including: the full models, along with five alternative PDDL models; the modeling limitations and feature support of available solvers; a number of problem modifications that are also studied to obtain insights on the problem characteristics that are difficult for the solving technologies; and, lastly, a discussion of each of the technologies, the effects of modeling decisions, and a comparison between planning and scheduling.

Acknowledgements

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada, Dr. Robot Inc., and the Technology Evaluation in the Elderly Network (which is supported by the Government of Canada through the Networks of Centres of Excellence).

References

[Banks *et al.*, 2008] Marian R. Banks, Lisa M. Willoughby, and William A. Banks. Animal-assisted therapy and loneliness in nursing homes: use of robotic versus living dogs. *Journal of the American Medical Directors Association*, 9(3):173–177, 2008.

[Benton *et al.*, 2012] J. Benton, A. J. Coles, and A. I. Coles. Temporal planning with preferences and time-dependent continuous costs. In *Proceedings of the Twenty Second International Conference on Automated Planning and Scheduling (ICAPS-12)*, pages 2–10, June 2012.

[Booth *et al.*, 2016] Kyle E. C. Booth, Tony T. Tran, Goldie Nejat, and J. Christopher Beck. Mixed-integer and constraint programming techniques for mobile robot task planning. *IEEE Robotics and Automation Letters*, 1(1):500–507, 2016.

[Fox and Long, 2006] Maria Fox and Derek Long. Modelling mixed discrete-continuous domains for planning. *Journal of Artificial Intelligence Research*, 27:235–297, 2006.

[Ghallab *et al.*, 1998] M. Ghallab, A. Howe, C. Knoblock, D. McDermott, A. Ram, M. Veloso, D. Weld, and D. Wilkins. PDDL—The Planning Domain Definition Language, 1998.

[Laborie, 2003] Philippe Laborie. Algorithms for propagating resource constraints in AI planning and scheduling: existing approaches and new results. *Artificial Intelligence*, 143(2):151–188, 2003.

[Laborie, 2009] Philippe Laborie. IBM ILOG CP Optimizer for detailed scheduling illustrated on three problems. In *Proceedings of the Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, pages 148–162. Springer, 2009.

[Li *et al.*, 2016] Jacob Li, Wing-Yue Geoffrey Louie, Frances Despond, and Goldie Nejat. A user-study with tangy the bingo facilitating robot and long-term care residents. In *Proceedings of the IEEE International Symposium on Robotics and Intelligent Sensors (IRIS)*. IEEE, 2016.

[Louie *et al.*, 2014a] Wing-Yue Geoffrey Louie, Jacob Li, Tiago Vaquero, and Goldie Nejat. A focus group study on the design considerations and impressions of a socially assistive robot for long-term care. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 237–242. IEEE, 2014.

[Louie *et al.*, 2014b] Wing-Yue Geoffrey Louie, Tiago Vaquero, Goldie Nejat, and J. Christopher Beck. An autonomous assistive robot for planning, scheduling and facilitating multi-user activities. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 5292–5298. IEEE, 2014.

[Louie *et al.*, 2015] Wing-Yue Geoffrey Louie, Jacob Li, Tiago Vaquero, and Goldie Nejat. Socially assistive robots for seniors living in residential care homes: User requirements and impressions. In *Human-Robot Interactions: Principles, Technologies and Challenges*, pages 75–108. Nova Science Publishers, Inc, 2015.

[Mohamed and Nejat, 2016] Sharaf C. Mohamed and Goldie Nejat. Autonomous search by a socially assistive robot in a residential care environment for multiple elderly users using group activity preferences. In *Proceedings of the 26th International Conference on Automated Planning and Scheduling (ICAPS) Workshop on Planning and Robotics (PlanRob)*, pages 58–66, 2016.

[Pesant *et al.*, 1998] Gilles Pesant, Michel Gendreau, Jean-Yves Potvin, and Jean-Marc Rousseau. An exact constraint logic programming algorithm for the traveling salesman problem with time windows. *Transportation Science*, 32(1):12–29, 1998.

[Smith *et al.*, 2000] David E. Smith, Jeremy Frank, and Ari K. Jónsson. Bridging the gap between planning and scheduling. *The Knowledge Engineering Review*, 15(1):47–83, 2000.

[Tran *et al.*, 2016] Tony T. Tran, Arthur Araujo, and J. Christopher Beck. Decomposition methods for the parallel machine scheduling problem with setups. *INFORMS Journal on Computing*, 28(1):83–95, 2016.

[Tran *et al.*, 2017] Tony T. Tran, Tiago Vaquero, Goldie Nejat, and J. Christopher Beck. Robots in retirement homes: Applying off-the-shelf planning and scheduling to a team of assistive robots. *Journal of Artificial Intelligence Research*, 2017. To appear.

[United Nations, 2002] United Nations. *World population ageing, 1950-2050*. Number 207 in ST/ESA/SER.A. New York: United Nations. Department of Economic and Social Affairs, 2002.