A Task Restricted Hierarchical Control Scheme Facilitating Small Logistics

Xanthi S. Papageorgiou*, Dimitris Kavroulakis, Dimitris Ntalaperas, Thanassis Bouras
Robotics & Cognitive Systems Unit, UBITECH, Limassol, Cyprus
xpapageogiou@ubitech.eu

***Abstract*—In the context of logistics for small scale enterprises, we have proposed a framework for a robotic indoor transport system, specially design for coping with semi-structured, flexible environments, typically encountered in small residences, warehouses, or medium-sized industrial facilities where there is minimal potential for infrastructural and/or procedural augments to the process, to facilitate the robotic automation. Such environments pose a significant problem to automated solutions, since the environment is flexible and partially unknown, cluttered and human-center. For such problems, hierarchical abstractions is an almost natural way to decompose the problem, but this line of attack is not without problems. Specifically, for complicated tasks, it is not in general possible to guarantee that the low level controller will be able to facilitate the task. In this approach we discuss the idea of using a hierarchical scheme to disengage the low level robot motion with the task reasoning and task planning and to facilitate the execution of the low level, while restricting the task level to a prescribed lexicon and grammar of tasks, that can be executed reasonably well, by the low level controllers. The grammar is used in conjunction with the task specification of each specific plant, to produce a family of tasks that are both easily executable and provide a large coverage in the overall task space.**

***Index Terms*—Motion Tasks Representation, Knowledge Base, Multi Robot Task and Motion Planning, Smart Robotics and Automation, Hierarchical Control**

I. Introduction

Many companies/organizations accommodated in small buildings involve a vast amount of material transport through hallways, elevators, basements, and customer/patient units. Logistic transporting robots are used in big hospitals, malls, and industrial areas, but there is no cost-effective autonomous logistic robotic system adapted to small residences, warehouses, or medium-sized industrial facilities. In healthcare institutions, several employees spend many man-hours transporting food, clothes, waste, and other materials. Although automation solutions exist for large scale structured environments, they do not exist for small buildings or semi-structured environments.

In this paper, we present our work on a framework for a robotic indoor transport system towards that can be utilized for wholesalers for pharmaceuticals, small residences, hotels, supermarkets/warehouses, and other small semi-structured buildings, using a hierarchical control scheme with a restricted task level planner. The core of our idea is to restrict the tasks handled by the tasks planner to a subset, easily solvable by the low level control system.

We present a concrete example, based on a hospital setting, where a smart intuitive, flexible, **Task Management System (TMS)** is used with novel sensor technology and optimized TMS software to provide the most cost-effective and fastest to install solution and lower the transportation costs by $50\% - 80\%$.

The tasks primitives defined on the tasks' system are used in conjunction with the with intra-logistics software of the plant to automate the whole value chain from the back end to the customer/patient point. In addition, the proposed system shall increase the service quality by providing traceability during delivery and controlled environmental conditions, which are especially useful for hospitals, hotels, and the food industry.

II. Related Work

The problem of planning for a robot that operates in environments containing a large number of objects, taking actions to move through the world and change the state of the objects, is known as Task and Motion Planning (TMP). TMP problems contain elements of discrete task planning, discrete–continuous mathematical programming, and continuous motion planning and thus cannot be effectively addressed by any of these fields directly. In [1], a class of TMP problems and survey algorithms for solving them, characterizing the solution methods in terms of their strategies for solving the continuous space sub problems and their techniques for integrating the discrete and continuous components of the search are reported.

Task planning refers to the ability of a given specification in a high-level language for the robot to translate this specification to the lower-level controllers to accomplish a specific task. The problem of task planning for robotic systems has been successfully addressed in the context of formal methods [2]. The high-level specification is given by high-level languages such as the Linear Temporal Logic (LTL), Metric Interval Temporal Logic (MITL) and Signal Temporal Logic (STL). A reconfigurable motion planning and control in obstacle cluttered environments under timed temporal tasks are presented by [3], [4]. In [5], a human-in-the-loop task and motion planning strategy for mobile robots with mixed-initiative control is presented. A learning-based dynamic task assignment framework for a class of human-robot interactive systems in that humans and robots work closely to accomplish a mission with LTL constraints proposed in [6]. One of the essential requirements of autonomous motion planning and robot navigation is accurate localization. Uncertainty is prevalent in most problem domains, and a fundamental requirement is inferring the robot state (or other variables of interest) precisely. Moreover, in large-scale problem domains, employing multiple robots offer enhanced performance capabilities over a single robot performing the same tasks. In addition, complex real-world scenarios present the need for planning at different levels to accomplish a given set of tasks. High-level (task) planning helps break down a given set of tasks into a sequence of sub-tasks. Finding the appropriate robot motions to execute these sub-tasks requires determining the proper set of low-level control actions. Hence, planning should be performed in the task-motion or the discrete-continuous space [7]–[10].

A dynamic and efficient task management algorithm to enable a multi-robot system to effectively handle diverse and unexpected situations consists of task scheduling and allocation procedures. The task scheduling procedure efficiently schedules the waiting tasks of the multi-robots in the system to minimize the total time consumed by all the tasks. The task allocation procedure obtains the working information from the task scheduling procedure to assign jobs to each robot. In [11], an algorithm is proposed and compared with the tree-based task scheduling method. Also, designing and building Intelligent robots involves integrating various functionalities such as navigation, various recognitions, reasoning, planning, and others. Furthermore, such functional components are often distributed over several processors, even in a single robotic system. Manifold functionalities and inherent complexity of robotic systems require a well-organized, uniform control of the functional components to make the formidable integration of the functionalities manageable. In [12], an agent-based task management architecture for the control of intelligent robots to simplify the integration task is presented. The task manager works as an integration middleware and provides a consistent and unified control view for the functional components, which may be distributed over a network. Also, in [13], a robot task design and management system has been developed for tool manipulation tasks done by a sensor-based robot. This design and management system can be used for all processes, from teaching to task execution. A task description can be generated quickly and intuitively using a graphical user interface.

In the proposed approach, we developed an algorithmic module for augmenting task-replanning based on completed recorded tasks to utilize the smart transportation system. The proposed system will develop and implement an algorithmic solution to facilitate task replanning. Equivalent tasks will be defined, examined, and simulated whether productivity is enhanced. Furthermore, this algorithmic solution will be open to human interventions since human expertise is crucial to facilitate this step.

III. The well known blocking problem

A key issue in any hierarchical dichotomy, is, in essence, that the dichotomy is almost always incomplete. It is, in general, next to impossible to actually divide a general, open system into discrete hierarchy levels, "solving" the levels independently, and, as it is well known, a very large literature body has been developed to cope this issue, i.e. to establish conditions where hierarchical solutions work. In certain settings (i.e. discrete event systems) a lot of work, during the last years, has led to theoretical solutions that allow provable behavior of certain hierarchical control systems.

But, a general solution to arbitrary hierarchical motion systems, especially if one considers multi-robot setups, seems difficult. This is especially prominent in problems where the complexity of the problems on the higher level becomes large, with this complexity rapidly diffusing to the lower levels. For example, given a group of mobile robots, one can almost intuitively divide the problem into a higher level, **Task Planning Level**: *where & when the robots should be* and to a lower level the **Task Execution Level**: *how the robots should move to avoid collisions and to optimize the trajectory*.This intuitive dichotomy is capable to cope with a large number of tasks.

This dichotomy is not immune to problems. The task execution level can lead to problems in the task planning level, and vice versa, the task planning level to problems in the task execution, especially in complex, dynamic environments. The motion control of a multi-robotic system is complex, and this complex behavior will complicate the task controller.

IV. Proposed Approach

Our approach to this problem is effectively to constrain the task level, in a way to increase the overall capability of the system to handle the tasks. The system handles task belonging to a subset of the task space, specially selected to ensure a) large coverage of the task space, b) easy-to-provable implementation of the tasks on the task execution level. In essence, we propose to constraint the task space using a lexicon and a grammar on the task space, tailored to tasks related to logistics issues. Let $\mathcal{T}$ be the set of tasks that can be handled by the high level task controller. We denote as

$$\mathcal{T}_s \subseteq \mathcal{T}$$

the subset of tasks that can be handled provably, in bounded time from the low level controller, and as

$$\mathcal{T}_l \subseteq T_s$$

the subset of tasks derived from a lexicon of elementary tasks τ_i and a grammar on these tasks, defined as a set of functions

$$G = \{g_j | g_j : \mathcal{T}_l \times \mathcal{T}_l \to \mathcal{T}_l\}$$

Given this notation, our approach is to map the the human-level transportation tasks of the specific plant, to tasks belonging to T_l.

In the example we discuss, we use an appropriate algorithmic module that assists the human engineer utilizing a task specification language and an environment agnostic approach, permitting a fast and straightforward task codification.

Then, the motion task is allocated to the different robots of the system and using a task planner -to synchronize the tasks- are given to the individual robots, and then executed using a motion control scheme. This approach results in a conceptual architecture versatile enough to handle many

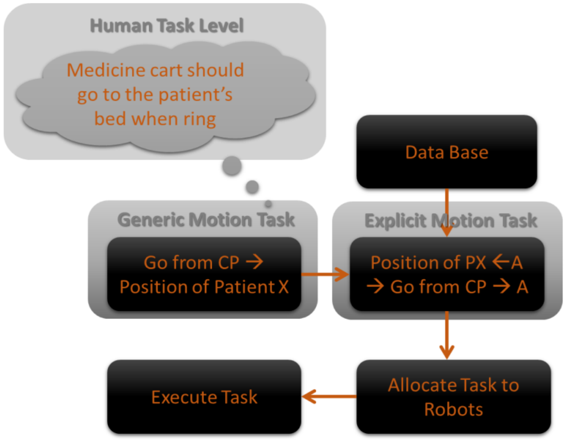

Fig. 1. Conceptual hospital task codification scheme.

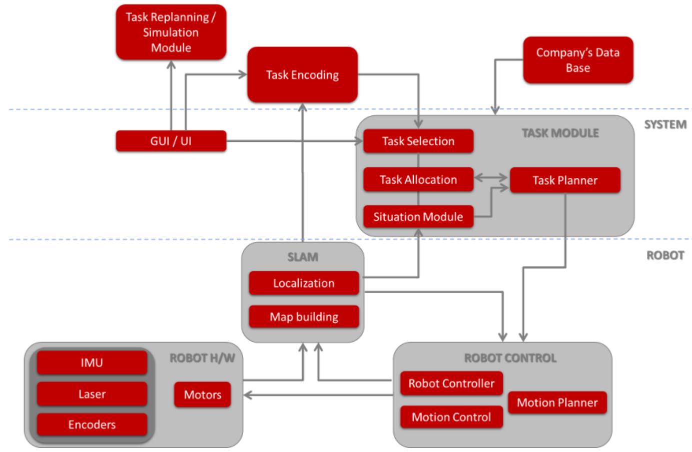

Fig. 2. Conceptual architecture of the proposed system.

different human level tasks in different environments and results in an inherently robust solution, where the tasks are executed even in the presence of robot failures. Moreover, this layered approach, and the introduction of generic tasks, permits lower engineering effort in codifying the problem.

V. A Hosptital case

We discuss, as an example, how this scheme is used in a hospital case, depicted in Fig. 1. This use case scenario contains a task, on a human level, which is to send a medicinal trolley to the patient's bed when needed. The task is codified as a generic motion task: Go from **Current Position (CP)** → Position of **Patient X (PX)**. During execution, the generic task produces an explicit motion task: Go from Current Position → PX using Hospital DB of the number of patients in need, as well as his/her position (A). Finally, the task is allocated to the robots and executed. We can conceptualize the general architecture of our system as depicted in Fig. 2.

A. System Components

1) Robot Algorithmic Components: The **Hardware Abstraction Layer (HAL)** that reads the data from the available sensors (IMU, wheel encoders and Laser Scanner) and controls the robot actuators. This part is greatly facilitated by using ROS compatible components.

The **Robot Motion Module (RMM)**, to which the planning and execution of the robot motion is assigned to, as well as the overall control of the robotic platform.

The **SLAM** module, for each robot of the team. The SLAM module is responsible for estimating the position of the robotic system and the shape of the environment, including possible dynamic entities on the environment, which can be other robots, humans and/or other moving machinery.

B. System Level Algorithmic Components

The **Task Module (TM)**, which is the main module of the system. The task module comprises first from:

- a **Situation Awareness Module (SAM)**, integrates information from all the SLAM of all the robots and from other potential sensors (i.e., global vision, possible human detection, etc.);
- the **Task Selection Module (TSM)**, which using the GUI information and the information from the system DB transcribes a generic task into a concrete task to be executed;
- the **Task allocation/task Planner Module (TPM)**, which allocate the tasks to all available robots and create appropriate motion sequence task to be executed by the robots. These modules react dynamically and can reallocate resources should this become necessary.

Moreover, the system level algorithm includes two modules running only for configuring the system:

The **Task Codification Module (TCM)**, transcribes the human level tasks into generic motion tasks

The important **Task Replanning Module (TRM)**, is used to assist in an organizational replanning of the overall process to maximize productivity.

C. System Operation

The interaction between the modules can be described as follows: The HAL module sends the sensor data to the SLAM module, which is also connected to the RMM since efficient slam requires the expected input to the system as well. The HAL module outputs a) the positional information of the robot to the RMM to control the motion of the robot using a closed-loop approach, as well as b) positional information and object information to the SAM module to construct a consistent view of the overall workspace. The HAM module is controlled by the RMM, both for logical commands and mainly for motion commands towards the robot actuators.

The RMM is interfaced with the SLAM module for positional information, while the TM gives the tasks allocated to the specific robot. The TM is interfaced with the GUI/UI for general control of the system, and, using a suitable bridge interface, with the company DB. Moreover, TM uses the output of the TCM, since TCM produces generic motion tasks which are specified to exploit motion tasks by the TSM of

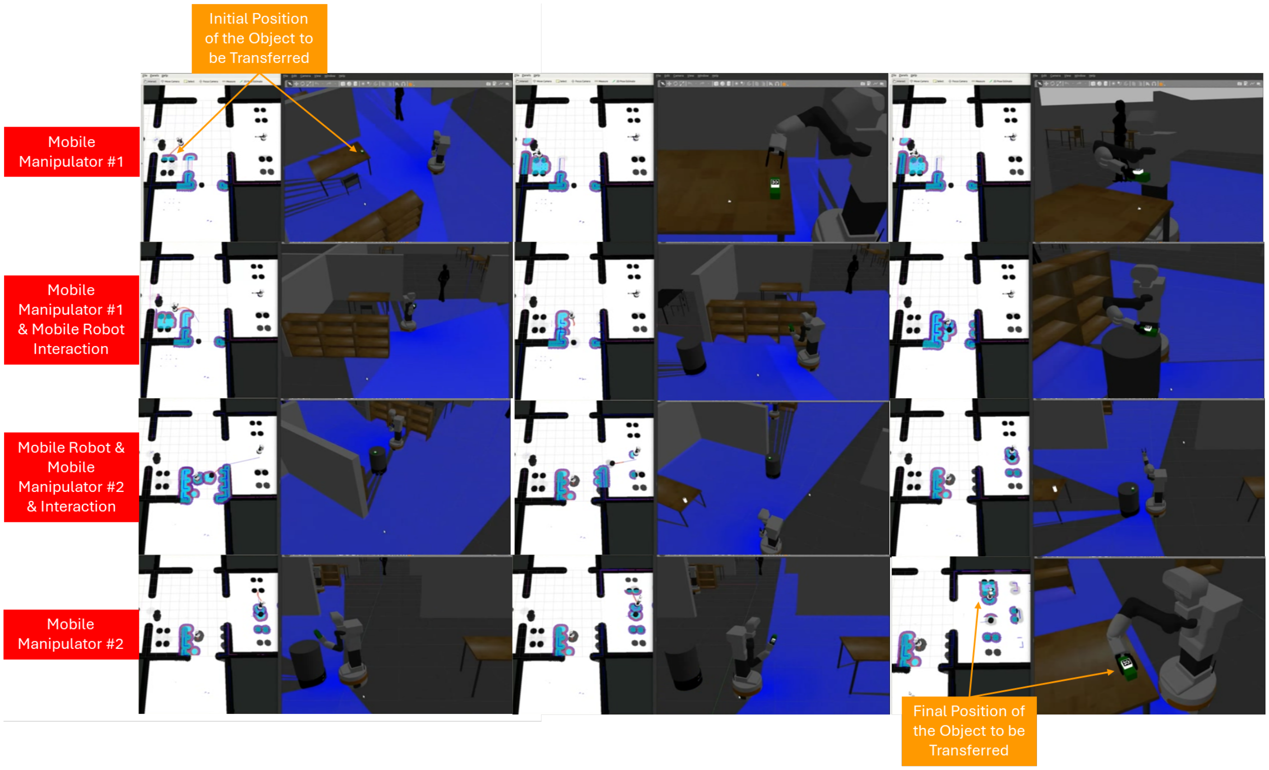

Fig. 3. Simulated results execution sequence.

TCM. TM is also interfaced with the SAM to assess the overall status of the system (location of the robots, environment etc.).

The TCM is a crucial part of the architecture, as it transcribes human specification to computer generic motion tasks. TCM is interfaced with GUI/UI and the SLAM, RMM and TM modules. When TCM is executed, the environment is scanned, and the robot runs with a specialized mode to chart the environment. The output of TCM is a codified set of generic motion tasks. Finally, the TRM is a generic "super module" interfaced with the whole system. It uses the available data of the system and already completed motion tasks, and its purpose is to assist a human operator in redesigning the overall system so that productivity is maximized.

D. Simulated Results

The proposed robotic system was simulated by used existing types of autonomous mobile robots, and manipulators (one Robotnik RB-1 Base robot and two PAL TIAGo robots). The sequence of the execution is depicted in Fig. 3. In an office environment containing a number of obstacles, one mobile manipulator grasp an object and transfer it to a mobile robot. The mobile robot drove the object to another location close to the second mobile manipulator. Then the robot manipulator pick up the object and following delivered it to the final working station.

VI. Conclusions

In this paper, we propose a framework for a robotic indoor transport system specially suited for wholesalers for drug-stores, small residences, hotels, supermarkets/warehouses, and other small semi-structured buildings and will automate the transport process and free workforce for tasks that entail higher added value. The system is based on a two level hierarchy, with the task planning and the task execution level. To facilitate their correct interaction, we propose a solution where the tasks allocated to the system are not arbitrary, but belong in a subset that is easily executed by the planner. We examine in particular how our proposed system is implemented with intuitive, flexible, Task Management System (TMS).

Acknowledgment

This work is a part of "Smart Autonomous RObotic deliveries for small loGIstics" – AROGI, ENTERPRISES/0223/Sub-Call1/0117 project, funded by the European Union Recovery and Resilience Facility of the NextGenerationEU instrument, through the Research and Innovation Foundation. Also, this work is a part of ARISE: "Advanced AI and RobotIcS for autonomous task pErformance" project, that has received funding from the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101135959.

References

[1] C. Garrett, R. Chitnis, R. Holladay, B. Kim, T. Silver, L. Kaelbling, and T. Lozano-Pérez, "Integrated task and motion planning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 4, 05 2021.

[2] C. Baier and J.-P. Katoen, *Principles of Model Checking*, 01 2008, vol. 26202649.

[3] C. K. Verginis, C. Vrohidis, C. P. Bechlioulis, K. J. Kyriakopoulos, and D. V. Dimarogonas, "Reconfigurable motion planning and control in obstacle cluttered environments under timed temporal tasks," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 951–957.

[4] S. Ahlberg and D. V. Dimarogonas, "Human-in-the-loop control synthesis for multi-agent systems under hard and soft metric interval temporal logic specifications," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 788–793.

[5] M. Guo, S. Andersson, and D. V. Dimarogonas, "Human-in-the-loop mixed-initiative control under temporal tasks," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 6395–6400.

[6] B. Wu, B. Hu, and H. Lin, "A learning based optimal human robot collaboration with linear temporal logic constraints," *CoRR*, vol. abs/1706.00007, 2017. [Online]. Available: http://arxiv.org/abs/1706.00007

[7] F. Lagriffoul, N. T. Dantam, C. Garrett, A. Akbari, S. Srivastava, and L. E. Kavraki, "Platform-independent benchmarks for task and motion planning," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3765–3772, 2018.

[8] M. M.G and A. Salgoankar, "A survey of robotic motion planning in dynamic environments," *Robotics and Autonomous Systems*, vol. 100, 12 2017.

[9] G. Rajendran, U. V, and B. O'Brien, "Unified robot task and motion planning with extended planner using ros simulator," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 9, pp. 7468–7481, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157821001695

[10] J. Alonso-Mora, J. DeCastro, V. Raman, D. Rus, and H. Kress-Gazit, "Reactive mission and motion planning with deadlock resolution avoiding dynamic obstacles," *Autonomous Robots*, vol. 42, 04 2018.

[11] J. Park, J. Kim, and D. Kim, "Task management algorithm for multi-robot system in intelligent space," *International Journal of Control and Automation*, vol. 10, pp. 173–186, Jan. 2017.

[12] J. Lee and B. Kwak, "A task management architecture for control of intelligent robots," 08 2006, pp. 59–70.

[13] K. Kanayama, M. Mizukawa, S. Iwaki, S. Matsuo, T. Okada, and Y. Nakamura, "A robot task design and management system for industrial applications," in *1997 8th International Conference on Advanced Robotics. Proceedings. ICAR'97*, 1997, pp. 687–692.