Seminars in AI & Robotics (2021/2022)
Paper 2: Autonomous Robotics: Manipulation and Motion Planning Problems

Leandro Maglianella[1]

[1]La Sapienza, University of Rome

Abstract

This paper is an in-depth journey regarding some of the main topics in Autonomous Robotics: Manipulation and Motion Planning. After an initial introduction on the socio-economic aspects of robotics and human-robot interactions, the robotic tasks of Motion and Manipulation in question are defined and all the main scientific improvements resulting from the past decades of research are reconstructed. In particular, aSyMov will be discussed extensively: it is a Robot Task Planner specifically designed to address intricate robot planning problems where geometric constraints are relevant and influence the symbolic plan. During the analysis not only information from the point of view of the robots is included, but the human side in cooperation with them will also be considered.

1. Introduction

During the recent decades, the presence of robots in our modern societies has been rapidly increasing more and more [1]. In Figure 1 it is shown the rising of the amount of operating industrial robots worldwide: in 2020 they were more than 3 million and today the numbers keep on growing accordingly. As a further reference in addition to the figure, consider that industrial robots' popularity began approximately in 1970: in 1973 there were only 3 thousand active robots, in 1983 66 thousand, in 1993 575 thousand and in 2003 800 thousand [1].

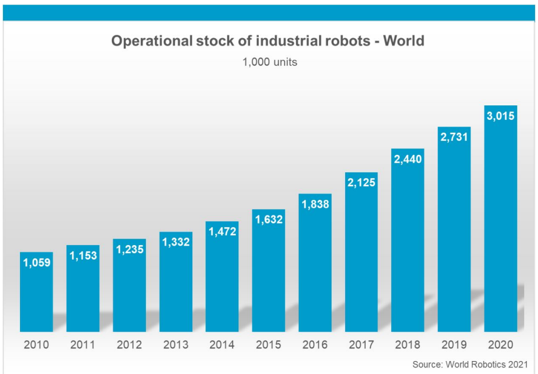

Figure 1: Statistical graphic representing the recent annual growth of active industrial robots globally employed.

Originally the need for efficiency and speed in industries found in robots a perfect candidate for performing simple and repetitive tasks. However, researches now aims at developing new methods to delegate to robots increasingly complex and critical tasks, even in private environments, that require some high-level reasoning skills with great precision and accuracy. For instance, we can consider the widespread use of robots in Amazon's fulfilment centres that manage the cataloguing and movement of the infinite amount of objects that the company sells and ships all over the world; or we could consider even more critical and important domains: many robots in fact find their application in hospitals and centres for the care of the elderly [2], still others are employed as teachers and support for children [3]. As we can imagine, in these kinds of tasks robots must be specifically designed so to be able to have the best possible interaction with humans: the efficiency of the task is not the most relevant aspect anymore; as a matter of fact it is replaced by new important ones related to social sciences as psychology, communications, relational and cognitive science.

In the continuation of this paper, we will retrace most of the work done by the main researches in the Autonomous Robotics field of study. In particular, I will only focus on Manipulation and Motion planning problems: we will study all the relevant definitions and initially consider a simple illustrative example. Later, we will move to some more modern and harder classes of problem and their aspects will be outlined. We will focus extensively on aSyMov, a Robot Task Planner specifically designed to address intricate robot planning problems where geometric constraints are relevant and influence the symbolic plan. Finally, we will give a glance at some extremely recent researches and future works about human-robot cooperation and human-aware planning in realistic settings.

maglianella.1792507@studenti.uniroma1.it (L. Maglianella)
www.linkedin.com/in/leandro-maglianella (L. Maglianella)
© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

2. Manipulation Planning Definitions

Let us begin our study giving the basic notions for Manipulation tasks [4].

- A ***Manipulation Problem*** can be defined as a path planning task for robots manipulating *movable objects* among *static obstacles*.
- In particular, the ***Input*** to this kind of problem should be a *movable object* (or a set of them) and a *goal position* for it (or a set of goal positions for each one of the considered movable objects).

Given these first two definitions, it is clear that the robot, as a start, should be able to tell the difference between other possible robots present in the environment, movable objects and obstacles. More specifically, movable objects are those objects with which the robot can freely interact and move in the considered scenario and, on the contrary, obstacles are those that the robot must avoid. In most of the following studied cases the obstacles are supposed to be static, meaning that not only the robot should not touch them, but also that they cannot move on their own. When the robot transfers a movable object, it must only drop it in a *stable position*: also movable objects cannot move on their own.

Talking about positions, it is also obvious that a robot should be able to auto-localise itself in the world and to interpret the input and goal positions in its own coordinate system and map and, finally, it should be able to detect and identify objects in its surroundings.

- A ***Goal*** must not be a *full description* of the state of the world in the desired final state. It should only be a partial description of what is needed.
- A ***Solution*** is an autonomously derived manipulation path to reach the *Goal* in such a way that satisfies the physical constraints of the *Problem*.

A solution usually consists in the decomposition of the task in a sequence of smaller collision-free motions: the goal state is reached in an incremental fashion by solving the necessary preconditions first. Therefore, the achieved solution usually involve a list of *Transit Path* and *Transfer Path*.

- In a ***Transit Path*** the robot moves alone from its current position to any other position where it can perform a grasping action on a movable object.
- After having grasped an object, in a ***Transfer Path*** the robot moves carrying it to a desired stable position. Here the object is released.

Moreover, it has to be noted that the motion during a Transfer Path should be collision-free for the entire system (robot + object).

Let us now consider a simple example [5] (shown in Figure 2) to apply all the definitions stated until now.

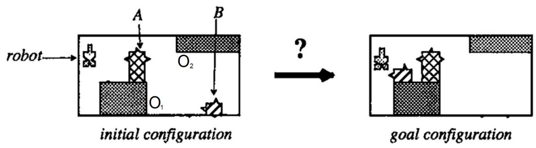

Figure 2: Simple Manipulation Planning example.

In the proposed example there are a robot R, two movable objects A and B and two static obstacles O_1 and O_2. The manipulation task requires R to move B from its initial position configuration p_{B_i} to a goal position p_{B_g}. It should be remarked that the formulated goal gives only a partial description of the scenario: indeed, it does not mention the goal position of the robot p_{R_g} or of the other movable object p_{A_g}. A correct solution for this task would consist in a alternation of collision-free transit and transfer paths: in particular, the robot has to consider the geometrical constraints of the environment; for instance it cannot directly move B to p_{B_g} because A is blocking the passage.

3. Harder modern problems

Only using *"geometric" methods*, although being it a limitation, we can still successfully solve a lot of possible problems. In this section we will only consider two research papers focused on solving some of them that I personally found the most interesting and, then, in the following section we will switch to aSyMov, a more robust, elegant and general solution approach.

3.1. Two-hand Regrasping

This first paper [6] studies the particular case where a humanoid robot has to manipulate to a final position an object with a complex shape in a congested environment. Indeed an immediate solution is not feasible because of an obstacle and of physical limitations of the robotic arms, therefore the robot must move the object between its two hands to reach the goal.

Essentially, the robot is equipped with a planner which includes the rapidly-exploring random tree (*RRT*) search algorithm for motion and a generic grasp planner for multi-fingered hands. In particular, *RRT* is a sampling-based probabilistically complete method which works by sampling the input space and forward integrating the system generating dynamically feasible motion primitives towards the goal. This planner is used to compute a path in which, at some point, a configuration is found where an exchange *double grasp*[1] is feasible so that it can switch grasp between its two hands. In particular, the robot is able to solve the task without knowing in advance that it is required to perform a regrasp and, consequently, does not know where it should perform it: it is only given an input containing the movable object's initial and final desired positions, it is the planner's job to find out if a regrasp is necessary and how to perform it. Figure 3 shows the double grasp for a couple of successful solutions.

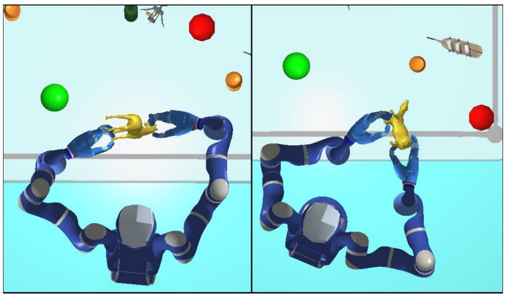

Figure 3: Double grasp configuration of two solutions.

3.2. Pivoting Based Manipulation

This second paper [7] analyses and reports the efficiency of pivoting motions for manipulating big and heavy objects with humanoid robots.

A solution path is obtained by finding a continuous path to the desired final goal which has an ε-free space surrounding it: thanks to this extra free space the robot can slightly incline the object and perform small rotation motions with it around a supporting contact corner, change contact point and then rotate it a bit again around the different corner and repeat this action until the destination is approached. This kind of motion is defined as a *pivoting* manipulation of the object and allows its transfer without demanding the robot to lift it, but using the contact point of the object with the ground as support, resulting in a solution with more dexterity, stability and adaptability. Figure 4 describes one step of the pivoting sequence. The entire robot-object structure is modelled as subjected to two kinematic constraints: the system must not slide (i.e. the classical non-holonomic constraint) and a path is admissible only if its curvature is always lower than some certain defined angle threshold. In this way, the resilience of the system is ensured to be maximised.

[1] With *"double grasp"* it is meant a grasp configuration of one hand which, at the same time, allows the other hand to perform a grasp too. Clearly, this same concept can also be used in situations where the robot has to hand an object to a human.

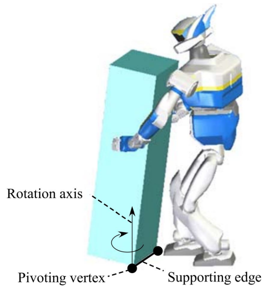

Figure 4: Representation of one step of a pivoting sequence.

As before, the motion path is computed using *RRT* and needs to be approximated in a pivoting sequence usable by the robot: it is done as shown in Figure 5. A arc of circle of angle α and radius R is produced through the following formulae:

$$\alpha = \arctan\left(\frac{2l\sin\beta}{R - l + 2l\cos\beta}\right) \quad \text{and} \quad \gamma = \beta - \alpha$$

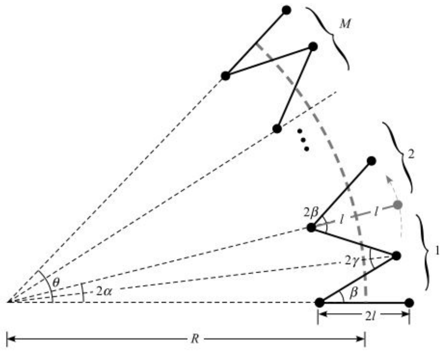

Figure 5: Approximation of an arc of circle into a pivoting.

As a reference and for completeness, I am also citing the other studied papers ([8], [9], [10] and [11]). They were not mentioned in this section but greatly helped in providing a good knowledge base to understand the later works: interested readers are invited to check them out.

4. aSyMov

Let us now consider ***aSyMov*** [12]. Until now, all the experiments have been performed using methods that only consider the geometry of the surrounding environment: all the geometric constraints were correctly considered but, nevertheless, these approaches proved to be feasible only in simple scenarios while they become too demanding in complex ones. Therefore, researchers had to find out a lighter and more "symbolic" way to represent and respect geometric constraints.

The aSyMov planner is designed as a hybrid planner including both a symbolic planner (*Metric-FF*) and a geometric planner (*Move3d*). The geometric planner implements a *Probabilistic Roadmap* method, similarly to the previous *RRT*, and is used to produce three different roadmaps for *transit*, *transfer* and *placement* compliant to our previous definitions. The novelty here lies in a new symbolic reasoning component, developed using *PDDL 2.1* to specify symbolic representations and configurations of the world. In particular, the authors implemented a basic set of predicates (such as *"composed", "belongs-to", "has-purpose", "connection", "on"*) and actions (*"goto", "grasp", "release", "switch-motion"*) to solve manipulation tasks.

The roadmaps' nodes are labelled using *symbolic positions* so that they can be used by the symbolic planner: this is the special manner with which it is possible to create an interplay between the symbolic level and the geometric level. Mainly, the symbolic planner is used to output an high-level plan to the geometric planner that will then try to solve the problem with it. Anyway, the geometric planner has also to possibility, if the symbolic plan turns out impossible, to consult back the symbolic planner for a new plan. Figure 6 shows more in detail the enforced *"Extend State"* algorithm.

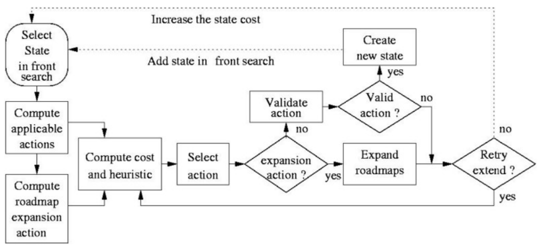

Figure 6: *Extend state* algorithm.

Symbolic positions represents sub-spaces of one roadmap that satisfy some given properties. A symbolic position p follows the following format:

$$P\_\{robot-name\}\_\{roadmap-name\}\_\{property\}$$

For instance, the symbolic position *P_R_TI_TA_O* represents the nodes of the transit roadmap of robot R that can be composed with some nodes of the placement roadmap of the object O to form a node in the transfer roadmap of $R + O$: thus this symbolic position simply corresponds to the positions where R can grasp O.

With all the theory being explained, let us now see an illustrative example on which aSyMov was applied: the Geometric Hanoi Tower Problem with 3 disks (*GHTP-3*). This is a geometrical transposition in 3D of the classic Hanoi Tower Problem: three disks have to be moved from the first to the third stack with the constraint that a disk cannot be placed on a smaller one. Figure 7 shows the problem symbolically.

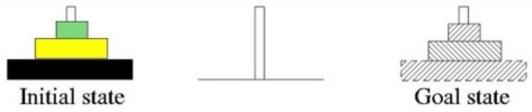

Figure 7: Hanoi Tower Problem's symbolic representation.

GHTP-3 was solved using non-holonomic fork-lift robots and it was tested in different experimental instances (without and with obstacles of different sizes, with one or multiple robots cooperating) achieving fantastic results[2]. Figure 8 displays some of the various experimental instances used.

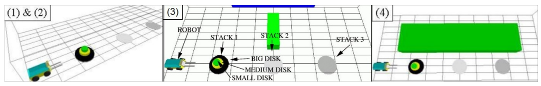

Figure 8: Experimental instances tested for GHTP-3.

5. Human-Robot Cooperation

In this final small section I will briefly introduce Manipulation topics being faced in the present times and considered for the future.

Robots are coming out of industries and evolving into entities that are expected to be interacted with by humans: thus, methods are being built to allow constructive co-operations between them. Robots can be used to carry out some task assigned by humans, for instance giving them some requested object, but can also be used to help in cooperative tasks where a human can demand a robot's action but the robot can ask for the human's help too: hence, robots have to be able to reason on human's feasible actions and how they can help it based on the environment context [13]. Moreover, the collaboration has to be performed by robots while trying to ensure the visibility and accessibility of the used objects [14].

Approaches are also studied to ensure that robots try to maximise safety: the manipulation should always be *human aware* [15]. This means, for example, that robots should attempt to stay as far as possible from humans while moving and to only approach them from the front. Moreover, they should also manage to recognise the *"category"* of the users interacting with them to act for their best comfort: for instance, if the users are elderly, the robot will have to try to be even more cautious and to get closer to them [16].

[2] Videos of the results can be found at this website: https://homepages.laas.fr/rachid/drupal/node/35#deux

References

[1] I. F. of Robotics, Ifr presents world robotics 2021 reports, 2021. URL: https://ifr.org/ifr-press-releases/news/robot-sales-rise-again.

[2] N. Savage, Robots rise to meet the challenge of caring for old people, 2022. URL: https://www.nature.com/articles/d41586-022-00072-z.

[3] Z.-J. You, C.-Y. Shen, C.-W. Chang, B.-J. Liu, G.-D. Chen, A robot as a teaching assistant in an english class, in: Sixth IEEE International Conference on Advanced Learning Technologies (ICALT'06), 2006, pp. 87–91. doi:10.1109/ICALT.2006.1652373.

[4] R. Alami, T. Simeon, J.-P. Laumond, A geometrical approach to planning manipulation tasks. The case of discrete placements and grasps, in: H. Miura (Ed.), The fifth international symposium on Robotics research , MIT Press, 1990, pp. 453–463. URL: https://hal.archives-ouvertes.fr/hal-01309950.

[5] R. Alami, J.-P. Laumond, T. Simeon, Two manipulation planning algorithms, in: K. Goldberg, D. Halperin, J.-C. Latombe, R. Wilson (Eds.), WAFR Proceedings of the workshop on Algorithmic foundations of robotics , A. K. Peters, Ltd. Natick, MA, USA, 1994, pp. 109–125. URL: https://hal.archives-ouvertes.fr/hal-01310030.

[6] J.-P. Saut, M. Gharbi, J. Cortés, D. Sidobre, T. Siméon, Planning pick-and-place tasks with two-hand regrasping, in: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2010, pp. 4528–4533. doi:10.1109/IROS.2010.5649021.

[7] E. Yoshida, M. Poirier, J.-P. Laumond, O. Kanoun, F. Lamiraux, R. Alami, K. Yokoi, Pivoting based manipulation by a humanoid robot, Autonomous Robots 28 (2010) 77–88. doi:10.1007/s10514-009-9143-x.

[8] J. Cortés, T. Simeon, J.-P. Laumond, A Random Loop Generator for Planning the Motions of Closed Kinematic Chains using PRM Methods, in: 2002 IEEE International Conference on Robotics and Automation (ICRA 2002), IEEE, Washington, United States, 2002, pp. 2141–2146. URL: https://hal.laas.fr/hal-01988698.

[9] T. Simeon, J.-P. Laumond, J. Cortés, A. Sahbani, Manipulation Planning with Probabilistic Roadmaps, The International Journal of Robotics Research 23 (2004) 729–746. URL: https://hal.laas.fr/hal-01987879.

[10] J.-P. Saut, A. Sahbani, S. El-Khoury, V. Perdereau, Dexterous manipulation planning using probabilistic roadmaps in continuous grasp subspaces, in: 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 2907–2912. doi:10.1109/IROS.2007.4399090.

[11] E. Yoshida, M. Poirier, J.-P. Laumond, O. Kanoun, F. Lamiraux, R. Alami, K. Yokoi, Regrasp planning for pivoting manipulation by a humanoid robot, in: 2009 IEEE International Conference on Robotics and Automation, 2009, pp. 2467–2472. doi:10.1109/ROBOT.2009.5152540.

[12] S. Cambon, F. Gravot, R. Alami, A Robot Task Planner that Merges Symbolic and Geometric Reasoning, in: 16th European Conference on Artificial Intelligence, Valencia, Spain, 2004. URL: https://hal.laas.fr/hal-01972672.

[13] M. Gharbi, R. Lallement, R. Alami, Combining symbolic and geometric planning to synthesize human-aware plans: toward more efficient combined search, in: 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015, pp. 6360–6365. doi:10.1109/IROS.2015.7354286.

[14] L. de Silva, M. Gharbi, A. K. Pandey, R. Alami, A new approach to combined symbolic-geometric backtracking in the context of human-robot interaction, in: 2014 IEEE International Conference on Robotics and Automation (ICRA), 2014, pp. 3757–3763. doi:10.1109/ICRA.2014.6907403.

[15] E. A. Sisbot, L. F. Marin-Urias, R. Alami, T. Simeon, A human aware mobile robot motion planner, IEEE Transactions on Robotics 23 (2007) 874–883. doi:10.1109/TRO.2007.904911.

[16] J. Mainprice, M. Gharbi, T. Siméon, R. Alami, Sharing effort in planning human-robot handover tasks, in: 2012 IEEE RO-MAN: The 21st IEEE International Symposium on Robot and Human Interactive Communication, 2012, pp. 764–770. doi:10.1109/ROMAN.2012.6343844.