Computer Aided Motion: Move3D within MOLOG

T. Siméon, J-P. Laumond, C. Van Geem and J. Cortes

LAAS-CNRS
7, avenue du Colonel-Roche
31077 Toulouse Cedex - France
{nic,jpl}@laas.fr

Abstract

This paper reports on our current effort for applying probabilistic path planning techniques to logistics and operation in huge industrial installations (e.g., power plants). We show how the specific domain constraints impose a dedicated software architecture to take advantage of the generality of probabilistic approaches. In addition, such an architecture should be compatible with existing CAD systems making critical the interface issues. We conclude on three study cases currently under development within the European project MOLOG.

1 Introduction

Today CAD systems are widely used in manufacturing and more generally to help the operation of complex systems and complex tasks. They are supported by powerful dedicated software including 3D visualization possibilities, geometric tools and friendly interfaces.

In the framework of logistics of industrial installations, CAD systems provide tools for the manipulation and storage of plant layout and design information. Accurate, full color 3D computer models are today available. They can be accessed easily by designers, engineers and project managers. In highly complex projects like operation, maintenance, building, CAD systems consistently enable dramatic savings in cost and time compared with traditional 2D design methods. These savings can halve the design time and save 15% of the total project cost - worth many millions of dollars on a large project.

MOLOG is a long term research action of the European Esprit program running from 1999 to 2002[1]. The objective of the project is to extend the range of CAD systems applications to help the operator in decision making via the integration of new techniques of geometric reasoning, mainly in motion planning and handling task planning.

Several motion planners are now available from the research community[24, 13]. Motion planning applications are emerging today in various domains such as computer graphics, drugs design, medicine... In the context of logistics the challenge is to face highly complex environment models including tens of thousands of objects. Nevertheless the handling devices have few degrees of freedom (usually four). This balance between high geometric complexity of the models and rather simple kinematics of the handling devices defines the range of the problems that should be addressed within MOLOG. A generic scenario addressed by the project is as follow: an operator should decide whether and how a given body (e.g., tools, parts, loads...) can be moved from a place to another one by using given transportation machines (e.g., manipulator robots, gantries, mobile platforms...). If necessary, the operator chooses and evaluates by himself the more adapted device for the task.

This paper overviews the architecture of the software platform Move3D which supports the current research developments conducted at LAAS within MOLOG. After having analyzed the requirements imposed by the application fields (Section 2), the global architecture of Move3D is presented (Section 3). This architecture is induced by the needs of a generic approach to path planning. It offers a framework for rapidly prototyping and evaluating new algorithms, mainly in the context of the probabilistic approaches [4]. Section 4 presents the set of planning algorithms which are currently implemented. Three examples of real size problems are then commented (Section 6).

Therefore the paper does not introduce new algorithms, nor new analysis. It should be viewed as an experience feedback in developing motion planning technology within the context of a well focused application field.

[1] Partners are two academic institutions (LAAS-CNRS and Utrecht University), an end-user (EDF) and a provider of CAD systems (Cadcentre).

2 Critical issues in logistics

This section analyzes the constraints imposed by the application field. Some of them deal with technical issues: the problem to be answered is less to provide a new global and self-contained solution to the planning problems than to offer the operator a set of generic tools which have the potential to be integrated within existing CAD systems and to process geometric models which are not necessarily well formated with respect to path planning algorithms. Other constraints deals with more theoretical issues: the proposed solutions should be generic and should work for a large class of mechanical systems.

Do not propose a solution from scratch: the data translation problem and the API Over the past years the functionality and use of CAD systems has rapidly increased since the mid-80's where first instances of complex systems appeared for managing complex models (see for instance the N4 model of a nuclear power plant of Figure 1, and the model of the Boeing 777[16]). These models contain a huge amount of information and are not necessarily dedicated to motion representation.

With respect to motion planning, the problem is first to extract only the pertinent information, i.e. the geometric part of the model. Then the geometric data should be formated to be processed by geometric operations, e.g. collision-checking. The function should be filled by a dedicated module, the so called `Translator`.

On the other hand, a communication protocol should be defined between the motion planners and the CAD system hosting them. This is the role of the application programmable interface (API). In addition to the description of the static environment usually present in CAD systems, the API provides the means to describe mechanical systems (e.g., handling devices), moving bodies (e.g., freights), a path planning problem (e.g., pick and place configurations). See [9] for a detailed description of the API developed within MOLOG.

Face geometric complexity: the data filtering problem Complex industrial installations, like power plants, oil refineries, offshore platforms and ships, use a large extent of space to implement a process with a large amount of machinery, piping, tanks and other equipment (see Figure 1). A motion planning task is often localized on a small part of the environment. The role of the `Filter` is to pre-process the geometric data basis to extract the only bodies that may interact when searching a collision-free path. The filter either uses inputs given by the operator (e.g., to

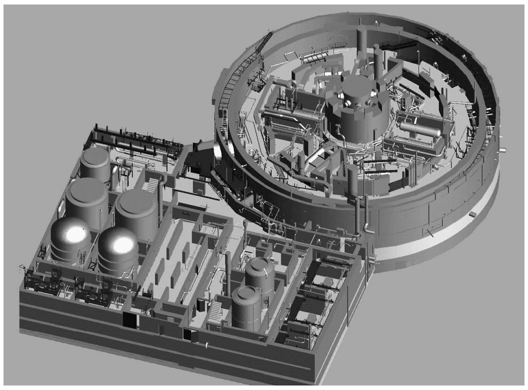

Figure 1: Auxiliaries building (level 5) and reactor building (level 6) of a N4 nuclear power plant (EDF, Engineering and Construction Division).

constrain the path to lye in a user-defined area) or automatically computes the workspace spanned by the bodies to be moved. Consider for instance a problem of path planning for a mobile platform as illustrated in Figure 4: in that case, the filter automatically removes all the parts of the environment that cannot interact with the cart (outside the slice defined by two horizontal planes). For the example shown in Figure 5, the filter retains most of the obstacles for potential collisions with the freight, but determines that no obstacle can collide with the first vertical body of the crane and very few with the second horizontal one. It remains that the filtered model may still include a huge number of geometric primitives. It is well known that the most time consuming operation in motion planning are interference detection and collision-checking along a given path.

The algorithms [11] developed within Move3D for the `Interference-detector` combine techniques proposed in [25] while allowing to process non convex polyhedra together with other volumic primitives (eg. spheres, tubes, torus...) as in [6]. Computation is performed in two stages: selection of possibly colliding pairs based onto simple bounding volumes that approximate the geometry of the objects, and a precise interference detection limited to the pairs selected at the first stage. A hierarchical structure based onto OBB-trees is constructed on top of the convex primitives (or facets), instead of the triangular decomposition required by [12]. This allows to reduce the size of the data structures when facing with large CAD models where most of the objects (eg. piping) are modeled by a collection of simple primitives. Also, CAD systems often group into a same object several components sharing the same semantic but localized at different places within the scene (eg. one single object may represent the piping part of the installation).

Such components are therefore grouped according to their workspace occupancy to improve the efficacy of the hierarchical models. The `Collision-checker` [11] developed for determining whether a given path is collision-free or not, is currently performed by multiple calls to the interference detection algorithms, using a dichotomic sampling of the path and a non uniform step computed from the distance to the obstacles.

Mechanical systems: generic solutions There is no free-flying body in a power plant. Most of the moving bodies are submitted to kinematic constraints: the problem of the transportation of a freight by a cart inherits from the nonholonomic constraints of the cart, the motion of a crane or a rolling bridge should follow a strict protocol imposing for instance to move one degree of freedom at once. Some tasks may require the collaboration between several handling devices, inducing closed kinematic chains. The motion planners algorithms should account for such constraints in a *generic* manner, i.e. without requiring to devise specific motion planners for specific devices. The recent probabilistic approaches allow to address a such generality level.

The probabilistic roadmap algorithms first introduced in [17, 28, 18] and now investigated by numerous researchers [3, 4, 7, 8, 14, 15, 19, 20, 30] answer this generality criterion. A roadmap is a graph that tends to capture the connectivity of the collision-free configuration space. The nodes are collision-free configurations while the edges indicate the existence of an admissible collision-free path between two configurations. Roadmaps are computed in a probabilistic manner by selecting the nodes of the graph randomly. Implementing such approaches only requires two basic geometric procedures: a steering method to compute admissible paths for a given mechanical system and a collision-checker which is used both to select the nodes of the roadmap and to check whether an admissible path is collision-free or not.

In most of the applications, collision-checking is problem independent while the steering methods are device dependant. The following methods illustrated by Figure 2 are today integrated within Move3D:

- `Linear` computes a straight line segment between two configurations: this method works for any holonomic system like a manipulator arm.
- `Nonholonomic` computes smooth paths for carts [29] or articulated mobile robots [23] as well.
- `Manhattan` accounts for the constraints to move one degree of freedom at once.
- `Closed-chain` accounts for particular types of closed kinematic chains.

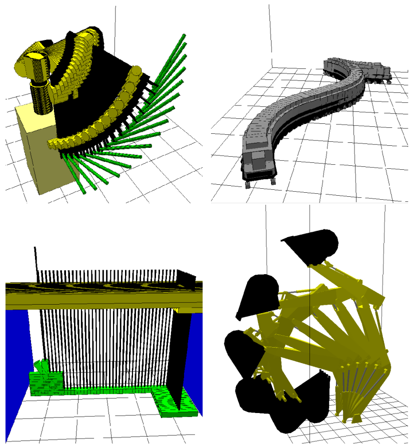

Figure 2: Examples of elementary steering methods

Other methods can be easily integrated into this library. They can also be combined to design more complex steering methods for mechanical systems subject to different motion constraints (e.g. for a mobile manipulator, the mobile platform and the manipulator are respectively controlled by the `Nonholonomic` and the `Linear` methods).

Some mechanical systems also require to consider passive joints that can not be directly controlled, but have to follow the motion of other leading joints. In presence of such motion constraints, the steering method only acts onto the leading joints, while the passive ones are computed by dedicated "follow" functions that express the coupling relations. This mechanism allows to handle systems involving simple closed kinematic chains like 4 dof linkages (e.g. *RRPR* or *4R*) that can be solved efficiently by analytical methods [26]. The right example of Figure 2 shows an hydraulic excavator that contains three *RRPR* linkages required to model the hydraulic systems, and one *4R* linkage for the mechanism controlling the motion of the bucket.

The same mechanism is also used to account for *attachments constraints* that appear when a freight has to be handled by a device. An example of such constraint is illustrated by Figure 5 where the freight has to slide on the ground before its reaches its vertical position.

Maintain the operator in the loop: the interactivity problem and the UI The User Interface allows interactions between the CAD system and an operator for specifying the maintenance operation and using the facilities offered by the motion planning tools. It also provides some feedback with the computed solutions. The use of the planning tools should be as easy as possible since the operator does not know anything about the planning algorithms. See [10] for a detailed description of the UI developed within MOLOG.

3 Move3D architecture

Figure 3 shows the global architecture of the motion planning software. This architecture is derived from the requirements addressed in the previous section. Move3D is composed of the following set of modules:

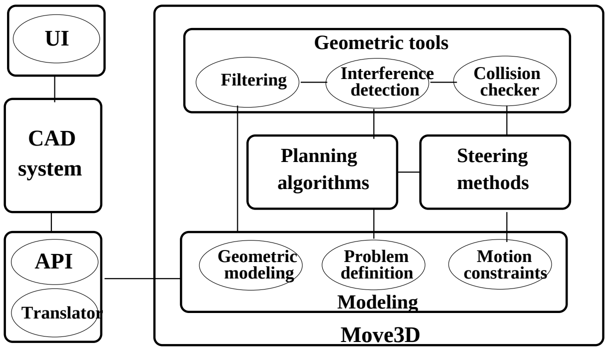

Figure 3: Architecture of Move3d

Mechanical systems and environment modeling Within MOLOG, the input of this module is provided through the API used to connect Move3D to the external CAD system and to transform the geometric data of the CAD system into the geometric data structures of Move3D

Geometric tools for filtering the geometric database and initializing the interference detection algorithms only with the part relevant to the specified planning problem.

Library of steering methods allowing to compute local paths admissible with respect to the own motion constraints of the mechanical systems involved by the planning problem.

Planner module which contains several algorithms based onto randomized techniques (see next section) for computing collision-free paths.

Finally, Move3D has its own Developer Interface module (not displayed onto the Figure) allowing to call the algorithms and to visualize the solutions independently of any external CAD system.

4 Motion planners

This module currently integrates four of the randomized planning techniques proposed in the literature. Three of them share the underlying concept of the Probabilistic Roadmap Methods that first construct a roadmap connecting collision-free configurations picked at random, and then use this roadmap to quickly answer multiple queries.

`Basic-PRM` is based onto the basic PRM scheme [18]. The search space is uniformly sampled at random. All collision-free samples are added to the roadmap and checked for connections with all connected components. The planner offers the choice between several strategies for selecting the promising nodes inside the components, and allows the tuning of several parameters (adjacency neighborhood, size of the roadmap...).

`Visib-PRM` computes visibility roadmaps [30] that are bipartite graphs defined with two types of nodes: the guards and connectors. Collision-free samples are kept as a new guard node when they cannot be connected to the current roadmap, or as a new connector if they improve the connectivity of the roadmap. An interest of this algorithm is the small size of the computed roadmaps. Another one is the possibility to control the quality of the roadmap in term of coverage.

`Gaussian-PRM` [8] uses a non-uniform sampling strategy in order to create a higher density of nodes near the boundary of the free-space. The gaussian sampler generates pairs of configurations separated by a random distance. It only retains a collision-free configuration of the pair when the other one lies in the collision space. This gaussian sampling strategy can be used when computing basic or visibility roadmaps.

The last planner is based onto the `RRT-Connect` algorithm [22] that was designed to answer single-query problems without requiring the preprocessing of the roadmap. The algorithm develops simultaneously two random trees rooted at the initial and goal configurations, and that explore the space while advancing toward each other.

5 Case studies

The environments of Figures 4,5,6 represent canonical examples we are working on within MOLOG.

The first scene corresponds to the model of a steam generator in a nuclear power plant. The problem is to check the feasibility of moving a non-holonomic cart underneath a steam generator. The path displayed onto the figure was computed by the `Visib-PRM` planner using the Reed&Shepp [29] steering method to generate feasible local paths. The model contains around 40.000 facets and the construction of the roadmap required a few minutes. After that, queries (including some path smoothing) can be processed in one second.

The industrial installation of the second example is a stabilizer (subset of a plant in chemical industry) modeled by 300.000 facets. The rotating crane (around 1000 facets) has to place a tank inside the metallic structure, starting from an initial position

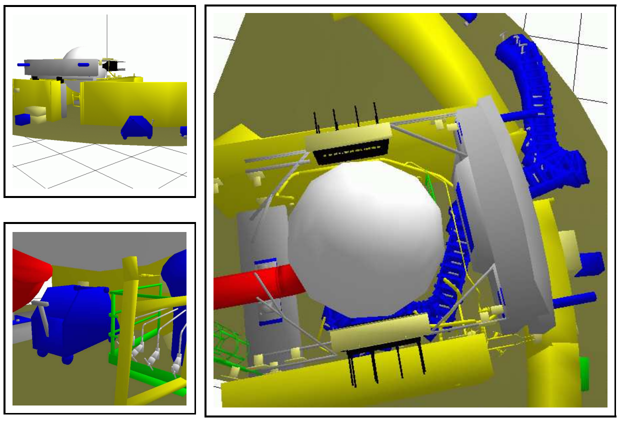

Figure 4: Nonholonomic cart in a steam generator of a nuclear plant (model provided by EDF)

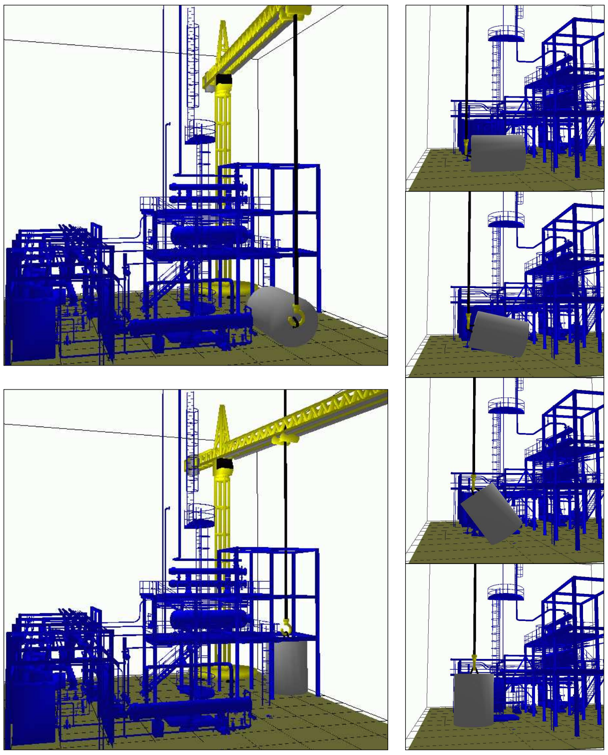

Figure 5: Stabilizer environment with a rotating crane (model provided by Cadcentre)

where the tank is horizontally placed onto the ground. Here the steering method produces `Manhattan` paths for the crane, while accounting for the constrained motion of the tank which slides onto the ground before it reaches a vertical position.

The third example shows a partial view of the engine room of a nuclear plant. Here, the purpose of the maintenance operation is to repair the water turbopump composed of three elements which have to be moved to a lower floor with a traveling crane. The path displayed onto Figure 6 corresponds to the motion of the traveling crane computed to grasp the first element of the pump. For this scene (same geometric complexity than the stabilizer), the construction of the roadmap took less than ten minutes allowing queries to be processed in a couple of seconds.

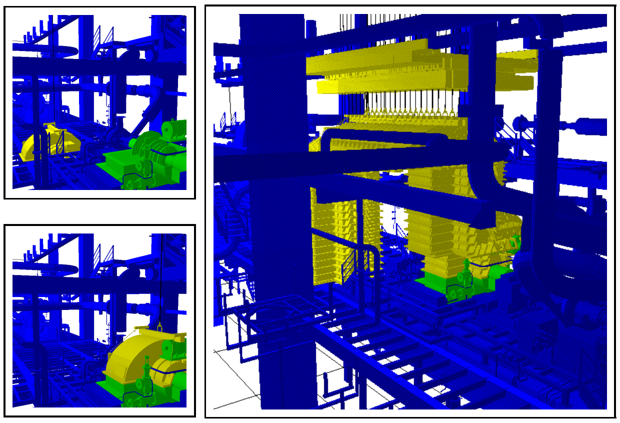

Figure 6: Travelling crane in the engine room of a nuclear plant (model provided by EDF)

6 Conclusion

The aim of the work reported in this paper is to develop a general planning software for providing CAD systems with motion planning facilities. The challenge for application fields such as logistics is to face real size problems involving a large class of mechanical systems. This is the direction of the research developments conducted at LAAS within the MOLOG project. The examples shown in the paper illustrate the kind of problems that can be solved today by the algorithms integrated within Move3D. It remains that additional work still needs to be done for improving the efficacy of the planning techniques. Another challenging issue is to develop more sophisticated handling planning algorithms that also support the choice and the use of several handling devices for carrying out objects within an industrial installation. This problem referred to as the manipulation planning problem [2] remains a practical challenge because of its additional complexity, although several promising results [21, 1, 27] have been recently obtained using probabilistic techniques.

Acknowledgments: The practical results have been obtained from the motion planning software Move3D developed in collaboration with C. Nissoux. We also thank EDF and Cadcentre for providing models of industrial scenes.

References

[1] J.M. Ahuactzin, K.K. Gupta and E. Mazer. Manipulation planning for redundant robots: a practical approach. *Practical Motion Planning in Robotics*, K.K. Gupta and A.P. Del Pobil (Eds), J. Wiley, 1998.

[2] R. Alami, J.P. Laumond and T. Siméon. A geometrical approach to planning manipulation tasks. In *Robotic Research: the fifth international symposium*, Tokyo (Japan), 1989.

[3] N. Amato, O Bayazit, L. Dale, C. Jones and D. Vallejo. OBPRM: an obstacle-based PRM for 3D workspaces. In *Robotics: The Algorithmic Perspective (WAFR98)*, P. Agarwal and all (Eds), AK Peters 1998.

[4] J. Barraquand, L. Kavraki, J.C. Latombe, T.Y. Li, R. Motvani and P. Raghavan. A random sampling sheme for path planning. In *Robotics Research*, G. Giralt and G. Hirzinger (Eds), Springer, 1996.

[5] J. Barraquand and JC. Latombe. Robot motion planning: a distributed representation approach. In *International Journal of Robotics Research*, 10 (6), 1991.

[6] G. van den Bergen. A fast and robust GJK implementation for collision detection of convex objects. In *Journal of Graphics Tools*, 4 (2), 1999.

[7] P. Bessiere, J. Ahuactzin, T, El-Ghazali and E. Mazer. The "Ariane's clew" algorithm: Global planning with local methods. In *IEEE International Conference on Robots and Systems*, 1993.

[8] V. Boor, M. Overmars, A. van der Stappen. The Gaussian Sampling Strategy for Probabilistic Roadmap Planners. In *IEEE International Conference on Robotics and Automation*, 1999.

[9] R. Davies, M. Overmars, V. Boor, A. Kamphuis, C. Van Geem, and T. Siméon. API: Requirements and architecture for motion planning interaction with PDMS. MOLOG Report A.1.1, 1999.

[10] W. Du, A. Drouin, L. Masquelier. User Interface Definition. MOLOG Report A.2.1, 1999.

[11] C. van Geem, T. Siméon. A Practical Collision Detector for Path Planning in Factory Models. *In preparation.*

[12] S. Gootschalk, M.C. Lin and D. Manocha. OBB-Tree: A hierarchical structure for rapid interference detection. In *ACM Siggraph'96*, 1996.

[13] D. Halperin, L. Kavraki and J.C. Latombe. Robotics. In *Handbook of Dicrete and Computational Geometry*, J.E. Goodman and J. O'Rourke (Eds), CRC Press, 1997.

[14] D. Hsu, JC. Latombe, R. Motwani. Path planning in expansive configuration spaces. In *IEEE International Conference on Robotics and Automation*, Albuquerque (USA), 1997.

[15] D. Hsu, L. Kavraki, JC. Latombe, R. Motwani and S. Sorkin. On finding narrow passages with probabilistic roadmap planners. In *Robotics: The Algorithmic Perspective (WAFR98)*, P. Agarwal et al (Eds), AK Peters, 1998.

[16] A, K. Jones. CAD geometry algorithms in a large industrial entreprise. In *Algorithms for Robotic Motion and Manipulation (WAFR96)*, J.P. Laumond and M. Overmars (Eds), AK Peters, 1997.

[17] L. Kavraki and JC Latombe. Randomized preprocessing of configuration space for fast path planning. In *IEEE International Conference on Robotics and Automation*, San Diego (USA), 1994.

[18] L. Kavraki, P. Švestka, J.-C. Latombe and M.H. Overmars. Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces, In *IEEE Transactions on Robotics and Automation*, 12 (4), 1996.

[19] L. Kavraki and JC. Latombe. Probabilistic roadmaps for robot path planning. In *Practical Motion Planning in Robotics*, K. Gupta and A. del Pobil (Eds), Wiley Press, 1998.

[20] L. Kavraki, M. Kolountzakis and J.-C. Latombe. Analysis of Probabilistic Roadmaps for Path Planning, In *IEEE Transactions on Robotics and Automation*, 14 (1), 1998.

[21] Y. Koga, K. Kondo, J. Kuffner and J.C. Latombe. Planning motion with intentions. In *ACM SIGGRAPH*, Orlando (USA), 1994.

[22] J. Kuffner and S.M. La Valle. RRT-Connect: An efficient Approach to Single-Query Path Planning. In *IEEE International Conference on Robotics and Automation*, San Francisco (USA), 2000.

[23] F. Lamiraux and JP. Laumond. Flatness and small-time controllability of multi-body mobile robots: application to motion planning. In *IEEE Transaction of Automatic Control*, 45 (10), 2000.

[24] J.C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

[25] M. Lin, D. Manocha, J. Cohen and S. Gottschalk. Collision detection: Algorithms and applications. In *Algorithms for Robotic Motion and Manipulation (WAFR96)*, JP. Laumond and M. Overmars (Eds), AK Peters 1997.

[26] J.M. McCarthy. Geometric Design of Linkages (preliminary version) Springer Mathematics, 2000.

[27] Ch. Nielsen, L. Kavraki. A two-level fuzzy PRM for manipulation planning. In *IEEE Int. Conference on Intelligent Robots and systems*, Japan, 2000.

[28] M. Overmars and P. Svestka. A Probabilistic learning approach to motion planning. In *Algorithmic Foundations of Robotics (WAFR94)*, K. Goldberg et al (Eds), AK Peters, 1995.

[29] J. A. Reeds and R. A. Shepp. Optimal paths for a car that goes both forward and backwards. *Pacific Journal of Mathematics*, 145 (2), pp. 367–393, 1990.

[30] T. Siméon, J.P. Laumond and C. Nissoux Visibility based probabilistic roadmaps for motion planning. In *Advanced Robotics*, 14 (6), 2000. A short version appeared in IEEE IROS 1999, Korea.