Proceedings of the 2002 IEEE/RSJ
Intl. Conference on Intelligent Robots and Systems
EPFL, Lausanne, Switzerland • October 2002

Capture of Homotopy Classes with Probabilistic Road map

E. Schmitzberger [1 2], J.L. Bouchet [1], M. Dufaut [2], D. Wolf [2], R. Husson [2]

[1]*Electricité De France, EDF R&D, Paris, France*
[2]*Centre de Recherche en Automatique de Nancy, INPL, Nancy, France*

Abstract

Feasibility tests in virtual reality for nuclear power plant maintenance or dismantling operations are a source of problems for motion planning because finding a way in a cluttered environment is not easy for the bulky loads, mobile devices and robots used in such operations. Standard Probabilistic Roadmap Methods (PRM) have been successfully used to answer such feasibility problems. These methods provide at the most a single solution but do not provide a complete overview of the possible motions which have to be evaluated in a complete engineering task. We focus here on the open question of building Probabilistic Roadmaps which can provide an exhaustive list of all the solutions which can not be distorted from one to another while staying collision free. We call such Roadmaps Homotopy Preserving Probabilistic Roadmap (HPPR). We propose a new algorithm for creating HPPR.

1. Introduction

A motion planning problem consists in searching collision free motion(s) between two given positions for a device evolving in 3D spaces amidst obstacles. We treat problems where the set of all physically admissible positions is a manifold of finite dimension n and can be mapped by an open bounded part of $\Re^n$ called the free configuration space C_{free}. Furthermore we consider here that any continuous path of C_{free} defines a physically admissible motion. The motion planning problem can thus be reformulated as searching for a path in C_{free} between two given configurations. This formulation covers a large class of mechanical systems, called holonomic systems, and includes some cases with closed kinematic chains [SIM^2]. Some extensions to the larger set of small time controllable symmetric systems like objects rolling without slipping have been done [LAU]. In that case, any continuous path of C_{free} just has to be the limit of admissible paths in the topology of uniform convergence. All these topological properties guarantee the existence of path-wise connected neighborhoods of any configurations. Several topological bases can thus be defined on the configuration space. This topological background enables one to build finite coverage and laid the foundations of digital representation and algorithmic treatment of motion planning among obstacles. We sum up this approach in section §2. Then we introduce the use of road maps which are graphs embedded in the configuration space to represent the possible motions. The digital representation of homotopy [MAR,DEY] is then presented in the framework of the probabilistic motion planning and leads to the algorithm HPPR in section §3.

2. Digital representations of connectivity

The configuration space C maps the set of all physically admissible positions but allows penetration between objects. C is a compact connected part of $\Re^n$ and contains the set of (collision) free configuration C_{free}. C has the Euclidean metric of which the distance is noted d_C. We treat the holonomic case where linear paths of C respect the motion constraints.

2.1 Use of Balls

In this part, we show that the use of canonical trajectories simplifies the computational treatment without loss of the topological properties induced by controllability and symmetry. We define a linear **local path** in the configuration space as:

$$\forall (x,y) \in C^2,\ \forall s \in [0,1],\ l_{x,y}(s) = x + s.(y-x)$$

We can check that the symmetry property remains:

$$\forall s \in [0,1]\quad l_{x,y}(s) = l_{y,x}(1-s)$$

Given that the length of the linear local path $L(l_{x,y})$ is the Euclidian distance, we can easily define path-wise connected balls:

$$B_\varepsilon(x) = \{y \,/\, L(l_{x,y}) = d_C(x,y) < \varepsilon\}$$

The family $\mathbf{B}$ of all such balls forms a topological base of C.

We now introduce the set $\overline{C_{free}}$ of configurations which correspond to a collision between the mobile device and the obstacles. Such configurations can be computationally detected. The collision detection function *free* gives the (overestimated) distance to the closest collision configuration.

$$free(x) = \min_{y \in \overline{C_{free}}} (d_C(x,y))$$

We can now present an overview of the topological proprieties which guarantee the completeness of an algorithmic approach of motion planning. Several subsets can be defined with the help of the collision detection function:

$$\overline{C_{free}} = \{x \in C \,/\, free(x) = 0\}$$
$$C_{free} = \{x \in C \,/\, free(x) > 0\}$$
$$C^{\varepsilon}_{free} = \{x \in C \,/\, free(x) \geq \varepsilon\}$$

The sub-family $\mathrm{B}^{C_{free}} \subset \mathrm{B}$ of balls included in the open C_{free} is a topological base of C_{free}. Furthermore the subset C^{ε}_{free}, called the ε-good subset of C_{free}, is by definition compact with reference to the following **induced topological base**:

$$\mathrm{B}_{C^{\varepsilon}_{free}} = \left\{B \cap C^{\varepsilon}_{free} \,/\, B \in \mathrm{B}^{C_{free}}\right\}$$

This property enables us to say that any coverage of C^{ε}_{free} with elements of $\mathrm{B}_{C^{\varepsilon}_{free}}$ admits a **finite** sub-coverage V:

$$C^{\varepsilon}_{free} = \bigcup_{B \in V \subset \mathrm{B}_{C^{\varepsilon}_{free}}} B \quad, Card(V) = N$$

We can obtain a finite representation of C^{ε}_{free} with any fit ε. This finite representation is the fundamental condition for algorithmic treatment.

A direct consequence of the symmetry property is that the non empty intersection of elements of V is an equivalence relation which means that the elements are path-wise connected. This relation leads to a partition of V in connected components. This partition can be represented by means of a graph $G=(V,E)$ whose **vertices** are the elements of V and whose **edges** E represent some non empty intersections of couples of elements of V. To build such a graph, the standard PRM approach first considers that each element of V is a connected sub-component and then merges couple of sub-components while adding a single edge until the whole components are obtained. Thus, $M=N-ncc$ edges leads to a partition of V in ncc connected components and forms a graph $G=(V,E)$ composed of ncc trees [OV,KAV]. PRM provides at the most one solution to a given motion planning problem.

2.2 Use of Visibility domains

We first extend the definition of the function *free* to local path collision detection. $free(l_{x,y})$ takes a null value as soon as one configuration of the local path corresponds to a collision. The procedure for calculating the collision detection for local paths is provided by coverage of the path with balls of the induced topological base $\mathrm{B}_{l_{x,y}}$. We do not detail this procedure which is out of the scope of this article.

The topological base of balls of C_{free} has been used to define the existence of finite representations but it is, most of the time, too fine and not suited for an algorithmic treatment. That is why we introduce the visibility domain of a configuration x [SIM[1]],

$$D(x) = \{y \in C_{free} \,/\, free(l_{x,y}) > 0\}$$

The configuration x is called a guardian. Visibility domains **do not** form a topological base. But if $x \in C^{\varepsilon}_{free}$, the visibility domain $D(x)$ contains, by definition, a ball of $\mathrm{B}^{C_{free}}$. It thus provides a (finite and) more concise way of coverage (Figure 1).

$$C^{\varepsilon}_{free} = \bigcup_{x \in V} D(x) \text{ with } V = \{x_1, x_2, \ldots, x_N\}$$

$$\text{and } \forall x, y \in V, x \neq y, free(l_{x,y}) = 0$$

The set V is called the set of guardians.

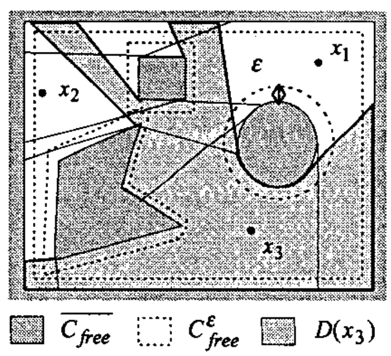

***Figure 1:** Coverage with visibility domains*

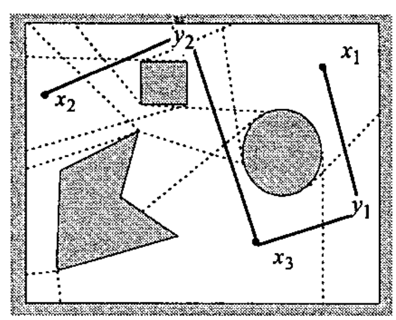

***Figure 2:** One connected component*

In the same manner as in the previous section, the non empty intersection between two visibility domains defines an equivalence relation. The existence of a point y at least belonging to two different visibility domains $D(x_i)$ and $D(x_j)$ is detected as soon as y verifies $free(l_{y,x_i}).free(l_{y,x_j}) > 0$. Standard PRM approach could thus be applied without any restriction. (Figure 2) and produces the graph G=(V,E) with:

$$E = \{(x_{i_1}, y_1, x_{j_1}), (x_{i_2}, y_2, x_{j_2}), ..., (x_{i_M}, y_M, x_{j_M})\}$$

2.3 Probabilistic Approaches

The time needed to cover C^{ε}_{free} depends notably on the dimension of the configuration space. This dependence is exponential [LAT] whatever the choice of the topology. This is obvious for a regular paving of C^{ε}_{free}. Probabilistic techniques of decomposition of the free configuration space minimize the impact of this dependence in the same way as Monte Carlo based integration techniques [LAT]. Furthermore the convergence of the coverage rate can be improved by several heuristics [BOO,AMA,WI]. Probabilistic techniques have been successfully used with several topologies. They proves to be flexible compared to previous motion planning techniques.

3. Digital representation of the homotopy class

3.1 Direct approach

By definition, a direct approach to build a concise homotopy preserving graph would consist in adding to G only new edges which can not be continuously distorted in C_{free} into already existing paths of G. In other words, no **reducible** loop (we mean loops which are continuously distortable in C_{free} into a point) must be created while adding a new edge to G. But such an approach is not suited for an algorithmic treatment. The potential number of loops to be tested increases exponentially with the number of new edges. Furthermore, testing the reducibility of a given loop is a non trivial problem and could lead to local minima if we try to reduce the loop length by distortion.

3.2 Homotopy and Visibility

To overcome these difficulties, we first propose to exclusively consider the reducibility of the elementary loops which are at the border of a local face. We define a linear **local face** for any triplet of points of C as :

$$\forall(x,y,z) \in C^3, \forall(s_1,s_2) \in [0,1]^2,$$
$$h_{x,y,z}(s_1,s_2) = x + s_1.(y-x) + s_2.(z-y)$$

In our case, local faces are controllable configuration subsets. The procedure for calculating the collision detection of local faces is thus provided by covering the face with balls of the induced topological base $B_{h_{x,y,z}}$. We can extend the definition of the function *free*(.) to the collision detection of local faces. The function $free(h_{x,y,z})$ takes a null value as soon as one configuration of the local face corresponds to a collision. In our case, the fact that a local face is collision free, *i.e.* $free(h_{x,y,z})>0$, is a sufficient condition to obtain the reducibility of the corresponding loop. We can remark that the local path defined by two arbitrary chosen vertices of the local face is always included in the visibility domain of the third vertex. This is a direct consequence of the fact that:

$$\forall\sigma \text{ a permutation of } \{x,y,z\},$$
$$free(h_{\sigma(x),\sigma(y),\sigma(z)}) > 0 \Leftrightarrow free(h_{x,y,z}) > 0$$

Collision free local faces could thus be seen as a means of detecting the inclusion of a local path in a visibility domain. The notion of visible sub-graph is presented to introduce the use of elementary reducible loops. Given a graph G=(V,E) and a configuration t, we define a visible sub-graph G_t=(V_t,E_t) from the t-point of view with:

- a sub-list of guardians visible from t,
  $V_t = \{x \in V \,/\, free(l_{t,x}) > 0\}$
- a sub-list of the edges forming elementary reducible loops with t,
  $E_t = \{(x_1, y, x_2) \in E \,/\, free(h_{t,x_1,y}).free(h_{t,x_2,y}) > 0\}$

To complete the analogy started with the definition of visibility, we say that G has a given property from the t-point of view if G_t presents this property.

Let us use a random sampling of C^{ε}_{free}, to test whether a graph G is at least simply connected from any point of view. If the visible part of the graph is not connected, then we add a minimal number of edges to G until its visible part is at least simply connected. The following algorithm describes this procedure.

Algorithm HPPR
$V = \varnothing, E = \varnothing, RL = \varnothing$ (1)
while unsuccessful tries (2)<limit
  Given a random $t \in C^{\varepsilon}_{free}$, build $G_t = (V_t, E_t)$ (3)
  if $V_t = \varnothing$, add t in the G list
  else
    if $G_t = (V_t, E_t)$ is non connected,
      add a minimal number of edges (u_i, t, u_j)
      with $u_i, u_j \in V_t$ in the E and E_t lists to
      make G_t simply connected.
    end
    Extension(t,RL) (1)
  end
end

(1) The notations RL and Extension will be used in the next sections.
(2) We call the try unsuccessful, if the try does not add a new vertex, edge or reducible loop.
(3) The partition of G in connected components has already been treated in section §3.1. and $G_t=(V_t,E_t)$ is built in a single connected component of $G=(V,E)$.

The set $\left\{y \in C^{\varepsilon}_{free} / \exists x_i \neq x_j \in V, y \in D(x_i) \cap D(x_j)\right\}$ is compact and can be finitely covered by balls of the induced topology. It shows that:

The number of edges that one must add to a finite trees-like graph to make it at least simply connected from any point of view is finite in C^{ε}_{free}.

In this way, the graph tends to be connected from everywhere in C^{ε}_{free} with non zero probability (Figure 3). We can then follow any curve (possibly closed) in C^{ε}_{free} while « seeing » at least one path (loop) of G. In fact, several paths of the graph can be associated in that manner because some reducible loops have been created while adding edges. For example the loop $(x_2, y_4, x_3, y_7, x_2)$ of the Figure 3 is reducible. That is why we call the obtained graph a **redundant HPPR.**
We will explain in the following section the way to reduce this redundancy and to obtain an answer to our reducibility problem in a more general case.

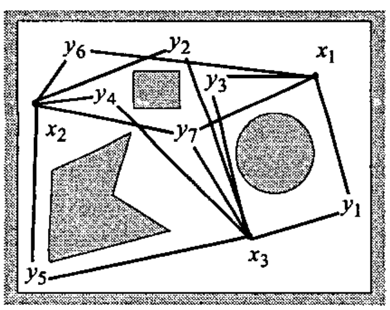

***Figure 3:** A graph after HPPR process (i.e connected from any point of view)*

3.3 Homotopic redundancy

Reducible loops must be eliminate in order to minimize the redundancy in the number of solutions of a motion planning problem. We define and exploit here the inner algebraic structure of reducible loops so as to avoid redundant tests of reducibility. This structure enables us to reduce the time (but not the number) of tests to do to add a new edge. Most of the reducibility tests can be solved by combinatorial operations instead of complex distortion of loops, as it is the case in a direct approach.

A **path** is a list of np points forming a chain of $np-1$ local paths $l_{x_k,x_{k+1}}$. for $k \in \{1..np-1\}$ noted $p=(x_1,x_2,\dots,x_{np})$. This chain can be a **loop** if start and end point merge. We note $P(V)$ the set of all paths using points of V and $P_{x,y}(V)$ the set of all paths of V linking x and y elements of V. Likewise, we call $P_{x,x}(V)$ the set of all possible loops of V with the point x. We can define an operation of composition $\circ$ for path (and loops):

$$(x_1,\dots,x) \circ (x,\dots,x_2) \equiv (x_1,\dots,x,\dots,x_2)$$

We now define two elementary bidirectional transformations for paths. The first one is:

$$(\dots,x,y,x,\dots) \leftrightarrow (\dots,x,\dots)$$

It defines an equivalence relation $\Re_1$ between the two path on $P_{x,x}(V)$. $(P_{x,x}(V)/\Re_1,\circ)$ is a group.
If $free(h_{x,y,z})>0$, we define a second bidirectional elementary continuous distortion for a path by:

(. ., ,. .) .. , , .. .)

We can remark that in our case:

$free \quad _{x,x,y}) \quad 0 \Leftrightarrow free\ l_{x,y})>0$

This relation is thus more general than the first one.

If we can obtain one path p from another path p_2 after a finite number of elementary distortions (Figure 4), we define a relation, noted $p_1 \Re p_2$, between the two paths. This relation means that we have continuously distorted p_1 to p with collision free faces. We call it a digital homotopy relation.

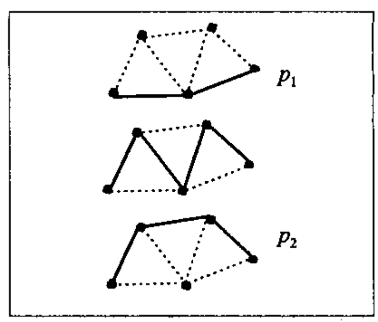

***Figure 4:** Homotopy relation $\Re$ by means of free local faces*

The digital homotopy relation $\Re$ is an equivalence relation on $P(V)$ and is compatible with the path composition. The set of equivalence class of $P_{x,x}(V)$ noted $(P_{x,x}(V)/\Re,\circ)$ is a group. The representatives of the Identity of this group are reducible loops.
It shows that reducible loops form a subgroup of $(P_{x,x}(V)/\Re_1,\circ)$. In our procedure the number of loops is finite. We can build a finite generator RL of the subgroup of reducible loop:

$RL = \{c_1, c_2, \ldots c_{nrl}\}$

$GEN(RL)$ is the theoretical set of all reducible loops generated by compositions of loops of RL. We use the fact that we can test if an arbitrary loop c is in $GEN(RL)$. In fact, this test, called the elementary divisor algorithm, reduces c with element of RL until we obtain a single point or until no further reduction is possible. In that case the residue of c is added to the RL list.

```
Extension(t)
if ∃c ∈ (V_t, E_t) and c ∉ GEN(RL),
        add the residue of c in the reducible
        loop list RL
end
```

We now have a procedure of detecting a broader set of reducible loop as in the precedent section but the question is «Can any reducible loop of G be detected in that manner?»

We can then follow any curve (possibly closed) in C^{ε}_{free} while «seeing» at least one path (loop) of G.

A reducible loop is by definition the border of a continuous surface S of C_{free}. Given an approximation of this surface with a mesh of local faces, we can associate a loop in G to each face. If we refine the approximation of S so that each elementary face tends to be a point, then each associated loop in G tends to be visible from a point and thus are reducible. This proof overview answer to the question and shows that:

If a graph is at least simply connected from any point of view then a reducible loop is either included in a single visibility domain or can be obtained by composition of several reducible loops.

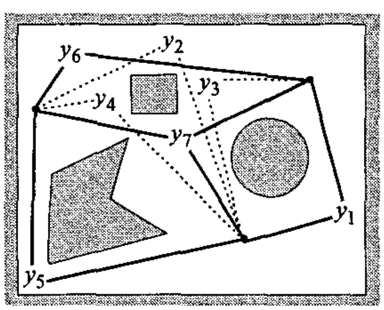

Figure 5: Removal of reducible loops

The given of a complete generator RL enables one to eliminate edges by breaking reducible loop (Figure 5). The result is thus a **non-redundant HPPR.**

4. Results

We have implemented HPPR under Matlab 6.0 in 2D environment as represented in the following figures.

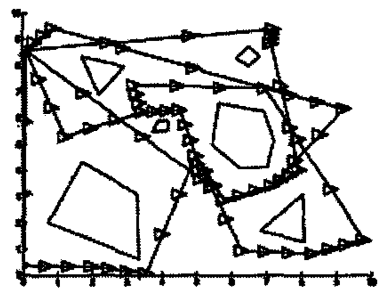

Figure 6: HPPR of a translating mobile body amidst 6 obstacles.

Figure 7: HPPR of an articulated chain (6 degrees of freedom)

We have represented in the following diagram the number of additional loops created for the example of figure 6. This number tends to be constant when the permitted limit of unsuccessful tries grows. These values are the average for 10 iterations of the algorithm with 4 values of the fixed limit (10,50,100,200).

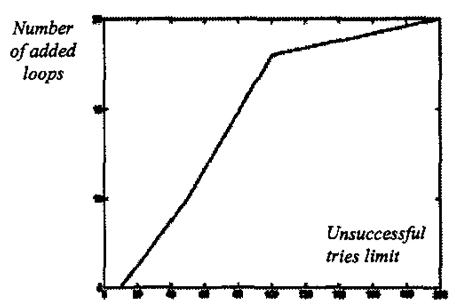

Figure 8: Experimental confirmation of the finite number of additional loops

In each case, all the reducible loops were detected. All non reducible loops were captured for a tries limit greater than 100.

Given $G=(V,E)$ with $N=Card(V)$ and $M=Card(E)$ and a randomly chosen node (or respectively an edge), the algorithm can decide in a computational time linear in N (resp. in M) whether or not a candidate node (resp. an edge) must be added to G. Redundant HPPR procedures do not introduce any additional combinatorial complexity compared to the coverage procedure using visibility domains. The time complexity is more related to the probability of finding a valid new candidate by random sampling. This probability goes down as G grows and increases the number of unsuccessful tries. This problem is widely treated in the literature of probabilistic road map and is an expression of the theoretic results on motion planning time complexity as it was mentioned in the section §2.3.

5. Conclusion

We have proposed in this article a method using the qualities of the probabilistic approach to build HPPR graphs. It enables one to split the motion planning procedures in two phases. The first one lists the homotopy classes of the solutions and the second one determines the most appropriate trajectory of each class. Trajectory optimization techniques must be used, for example to minimize path length or maximize distances to obstacles, but these techniques converge and are efficient if we use homotopic distortion [GRI]. We thus have to list all non homotopic solutions to globally achieve the optimization procedure. Furthermore, in the context of a variational approach of handling planning, [FER,ZEF] we can first consider the motion of the free flying load before introducing the geometric and kinematics constraints due to handling devices. Each homotopy class for a free flying load among obstacles like tubes and machinery of nuclear power plant could leads to very different handling procedures with dedicated devices and must be listed to enable an handling process choice. All these reasons show that the topological information captured by HPPR graph is essential before any further motion planning procedures.

Acknowledgments

This work is supported by the European Esprit Project 2826 MOLOG. *http://www.laas.fr/molog*. We want to thank T. Simeon, F. Lamiraux and A. Sahbani from the LAAS for their support.

References

[AMA] N.M. Amato, O. Burchan Bayazit, L. K. Dale, C. Jones, D. Vallejo, « OBPRM: An Obstacle-Based PRM for 3D Workspaces », In *Robotics: The Algorithmic Perspective (WAFR98)*, P. Agarwal and all (Eds), AK Peters 1998

[BOO] V.Boor, M.H. Overmars, A.F. van der Stappen, « The Gaussian Sampling Strategy for Probabilistic Roadmap Planner », In *IEEE International Conference on Robotics and Automation*, 1999, pp 1018-1023

[DEY] T. K. Dey, H. Edelsbrunner, and S. Guha. «Computational Topology», In *Advances in Discrete and Computational Geometry (Contemporary mathematics 223)*, ed. B. Chazelle, J. E. Goodman, and R. Pollack, American Mathematical Society, 1999, pp 109-143.

[FER] P. Ferbach and J. Barraquand. « A method of Progressive Constraints for Manipulation Planning », In *IEEE Tr. Rob. and Aut.*,1997, 13(4): pp 473-485

[LAT] J-C Latombe, *Robot Motion Planning,*, Ed. Kluwer Academic Publisher, 1991

[GRI] D.Grigoriev, A. Slissenko. « Computing minimum-link path in a homotopy class amidst semi-algebraic obstacles in the plane », In T. Mora and H. Mattson editors, In *Proc. of the Int. Symp. AAECC'12*, pp 114-129. Springer Verlag, 1997. Lect. Notes in Comput. Sci, vol. 1255.

[KAV] Lydia E. Kavraki, Petr Svestka, Jean-Claude Latombe, « Probabilistic roadmaps for path planning in high dimensional configuration spaces », In *IEEE Trans. Robot. Autom.*, 1996, 12(4) : pp 566-580

[LAU] « Guidelines to Non-holonomic Motion Planning » Laumond Ed. www.laas.fr/~jpl

[MAR] T.Martinetz, K Schulten. « Topology representing networks », *Neural Networks* 7 , 1994, pp 507-522

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

[SIM[1]] T. Siméon, J.P Laumond and C. Nissoux. «Visibility-based probabilistic roadmap for motion planning». In *Advanced Robotics Journal* 14(6), 2000

[SIM[2]] T. Siméon, J.Cortès, A. Sahbani and J.P Laumond. « A Manipulation Planner for Pick and Place Operations under Continuous Grasps and Placements », In *IEEE International Conference on Robotics and Automation*, 2002

[WI] S.Wilmarth, N. Amato and P.Stiller. « MAPRM: A Probabilistic Roadmap Planner with Sampling on the Medial Axis of the Free Space ». In *IEEE International Conference on Robotics and Automation*, 1999, pp. 1024-1031

[ZEF] M. Zefran and V. Kumar, « A variational calculus framework for motion planning », In *International Conference on Advanced Robotics*, Monterey, CA, 1997