Contact Planning for Acyclic Motion with Task Constraints and Experiment on HRP-2 Humanoid

Adrien Escande
CEA - LIST
Fontenay-aux-Roses, France

Abderrahmane Kheddar
CNRS-UM2 LIRMM, Montpellier, France
CNRS-AIST JRL, UMI3218/CRT, Tsukuba, Japan

***Abstract*—This video illustrates our work on contact points planning and its recent enhancement by two new functionalities. First, by taking advantage of the possibilities offered by our initial posture generator, we include additional tasks in the planning that are not related to locomotion. Second, we refine the potential function that guides the planner so as to cope with more challenging scenarios. We then test these novelties on difficult problems with success, and experiment the output of one of the planned scenario on a HRP-2 humanoid robot.**

I. Introduction

Cyclic motions, although useful in practice, prove to be limited and there are plenty of situations where acyclic motion with eventually additional links' support are required to achieve complex transportation motions. A simple example could be the humanoid robot grasping a ramp to ease climbing high stairs. Non-gaited motion planning has been more formally addressed in simulation in [1] and [2]. A different approach has been proposed by the authors in [3] and [4] where real experiments have been conducted on the HRP-2 humanoid robot. Acyclic motions having contact supports that could occur on any part of the robot with any part of the environment is on the stage of full realization, but there remain many problems to be solved and future extension to be addressed that have been discussed in [5].

In this video, we show that our approach allows also an interesting extension at nearly no cost to handle additional constraints related to tasks that are not linked to locomotion. A companion paper is submitted as a regular paper in this edition of IROS which thoroughly explains the increment on our previous achievements.

II. Background

A. Overall planner

In [3] we presented a planner with the following principle: planning is made in the space of sets of contacts SC, by building incrementally a tree of such sets. The difference between a father and its son is exactly one contact (more or less). To drive the planning, a potential function f is given. At each iteration, the best leaf of the tree (according to f) is selected and its sons are built by adding or removing a contact. If some of the new leaves are too close to existing nodes or leaves, they are discarded. This mechanism is inspired by the potential-field-based planner Best First Planning (BFP), see [6]. However, we are planning here in SC, which allows a dramatic reduction of the search space compared to the usual configuration space. Yet it does not allow to take into account the geometrical and physical limitations of the robot: two contacts of a set may be too far from each another, a contact may force collisions or instability of the robot, etc. Feasibility of a set must be checked, and this is done with a posture generator (section II-B), see Fig. 1. Upon failure of the posture generator, the set of contacts is discarded.

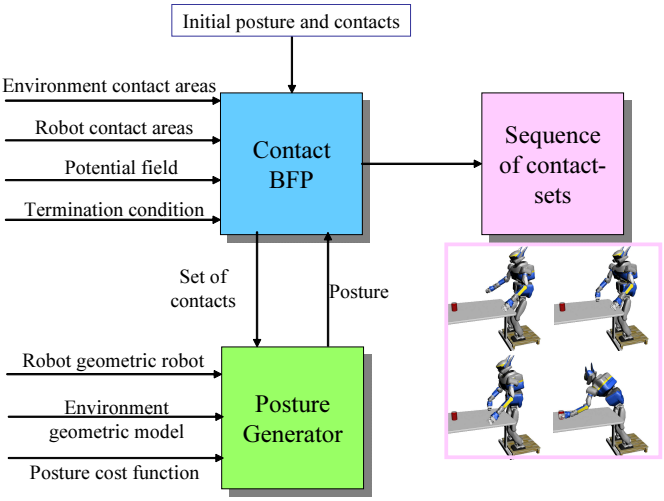

Fig. 1. Overview of the planner and its components.

In [4], we introduced some major concepts regarding the choice of new contacts, the design of the potential function, and a positive interaction between both of these concepts. The choice of a new contact is made directly during the posture generation attempt and is based on the criterion of the posture generator. We thus reduce significantly the number of feasibility checks to perform and especially the number of failures of generation attempts, which are the most time costly computations.

The design of f relies on a rough trajectory T in the configuration space. This trajectory is defined by several key postures between which a linear interpolation is made. It does not necessarily lies completely in the free configuration space, nor does it need to be in the robot stability space. It is just a guide upon which we build a descending valley-like potential whose minimum is at the end of T. This trajectory was first given manually through key configurations. We now developed

a method to generate it automatically, see [7]. Refinement in f have also been made, namely to reduce the search space and converge fastly toward the solution in constraint spaces (see IROS companion paper). In [4] and recently we used f as part of the criteria in posture generation, and take this into account in the BFP-like part of the planner to generate far less nodes.

Planning is thus made in the sets of contacts space, but with a constant link to the configuration space. The inputs of our planner are the data of the environment, the data of the robot, a feasible starting set of contacts and some end conditions. Output is a sequence of sets of contacts along with their associated witness postures.

B. Posture Generator

For a given set of contacts $\{C_i\}$ as input, the posture generator writes and attempts to solve an optimization problem with (non-linear) constraints, whose variables are the n degrees of freedom of the robot $\mathbf{q}$:

$$\min_{\mathbf{q}\in\mathcal{Q}} f(\mathbf{q}) \tag{1}$$

where Q is the set of admissible postures for these contacts:

$$\mathcal{Q} = \begin{cases} q_i^- \leq q_i \leq q_i^+, \quad \forall i \in [|1,n|] & (2)\\ \epsilon_{ij} \leq d(r_i(\mathbf{q}), r_j(\mathbf{q})), \quad \forall (i,j) \in \mathcal{I}_{\text{auto}} & (3)\\ \epsilon_{ik} \leq d(r_i(\mathbf{q}), O_k), \quad \forall (i,k) \in \mathcal{I}_{\text{coll}} & (4)\\ s(\mathbf{q}) \leq 0 & (5)\\ g_i(\mathbf{q}) = 0 \quad \forall\ C_i & (6)\\ h_i(\mathbf{q}) \leq 0 \quad \forall\ C_i & (7) \end{cases}$$

Inequalities (2) are the joint limits; eqs. (3) define constraints for auto-collision avoidance between pairs of robot's bodies (r_i, r_j) given by $\mathcal{I}_{\text{auto}}$. d is the minimum distance between two objects, that need to be positive; eqs. (4) deal also with collision avoidance between r_i and any object O_k of the environment, pairs are defined by $\mathcal{I}_{\text{coll}}$; ϵ_{ij} and ϵ_{ik} are security margins for these constraints; s is a static stability criterion –s can simply be the belonging of the projection of the center of mass to the convex hull of contacting points. An extension of this criterion can be used as proposed by [8]–; g_i and h_i are respectively the equality and inequality constraints describing the i^{th} contact –basically, they force a point and a frame of a link to correspond to a point and a frame of the environment.

The optimization criterion f is optional. It allows the user control the overall look of the obtained posture. For example, the user may want to have human-like postures. In [3], we used the distance to a reference posture, in [4], the potential function f of the planner level is used as optimization criterion.

To solve such an optimization problem we use FSQP which allows the use of any function provided it is at least twice continuous. In particular, it copes with non-linear criteria and constraints.

Let's take as an example the following mission illustrated by the attached video: the robot must carry a glass containing a liquid. If the robot does not have the glass in the gripper, it must first reach it. This part of the planning was solved in [3].

Separating the problem into sub-missions as "go to the glass", "grasp the glass", "carry it", and so on, would be the purpose of an higher-level task planner and is beyond the scope of this work. Here we assume the robot with the glass grasped. In order not to spill a drop of liquid, the robot must keep at all time the glass in a vertical position, and in particular at each witness posture associated to a node of our planner. This task can be described with:

$$\mathcal{T}_{\text{glass}} = \begin{cases} \mathbf{n}(\mathbf{q}).\mathbf{i} = 0 \\ \mathbf{n}(\mathbf{q}).\mathbf{j} = 0 \\ -\mathbf{n}(\mathbf{q}).\mathbf{k} < 0 \end{cases} \tag{8}$$

where $\mathbf{n}(\mathbf{q})$ is the axis of the hand carrying the glass, and $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ is a frame of the world. This task is added to every posture generation. Every posture of the output plan will respect its constraints. Adding a task in the planning decreases the number of degree of freedom of the robot, meaning a witness posture will be found for fewer sets of contacts. It has thus a direct impact on the plan, since it reduces the possibilities of the planner.

Using the framework described in [4], we played the output of the glass scenario on HRP-2 robot. Since some posture were really demanding for the robot, and some step redundant we manually removed some nodes. We were able to play successfully the plan with a glass filled with Japanese tea. The video blends virtual reality simulation showing at the beginning what the contact planner attempts are, all the postures that have been found are displayed (fast). In the second stage the video illustrates the sequence of contacts set that have been chosen, and then how the trajectory is generated between successive contact sets. At the end, the overall motion planning is played in the HRP-2 robot.

See also paper ID980.

REFERENCES

[1] K. Hauser, T. Bretl, and J.-C. Latombe, "Non-gaited humanoid locomotion planning," in *IEEE/RSJ International Conference on Humanoid Robots*, December 5-7 2005, pp. 7–12.

[2] K. Hauser, T. Bretl, K. Harada, and J.-C. Latombe, "Using motion primitives in probabilistic sample-based planning for humanoid robots," in *Workshop on the Algorithmic Foundations of Robotics*, 2006.

[3] A. Escande, A. Kheddar, and S. Miossec, "Planning support contact-points for humanoid robots and experiments on HRP-2," in *IEEE/RSJ International Conference on Robots and Intelligent Systems*, Beijing, China, October 2006, pp. 2974–2979.

[4] A. Escande, A. Kheddar, S. Miossec, and S. Garsault, "Planning support contact-points for acyclic motion and experiments on HRP-2," in *International Symposium of Experimental Robotics*, Athens, Greece, 14-17 July 2008.

[5] A. Kheddar and A. Escande, "Challenges in contact-support planning for acyclic motion of humanoids and androids," in *International Symposium on Robotics*, 2008.

[6] J.-C. Latombe, *Robot motion planning*. Boston-Dordrecht-London: Kluwer Academic Publishers, 1991.

[7] K. Bouyarmane, A. Escande, F. Lamiraux, and A. Kheddar, "Collision-free contacts guide planning," in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, 12-17 May 2009.

[8] T. Bretl and S. Lall, "Testing static equilibrium for legged robots," *IEEE Transactions on Robotics*, vol. 24, pp. 794–807, August 2008.