Motion Optimization of Robots, Application to HRP-2

Sylvain Miossec, Sebastien Lengagne, Abderrahmane Kheddar, Kazuhito Yokoi

Motion Optimization of Robots, Application to HRP-2

Sylvain Miossec
Institut PRISME, Univ. d'Orléans, IUT de Bourges
Email: sylvain.miossec@bourges.univ-orleans.fr

Sébastien Lengagne
LIRMM
Email: sebastien.lengagne@lirmm.fr

Abderrahmane Kheddar (LIRMM/CNRS), Kazuhito Yokoi (AIST)
Email: {abderrahmane.kheddar,kazuhito.yokoi}@aist.go.jp

1 Problem Statement

We presents an implementation of motion optimization for robots. The exact motion optimization problem we solve is the following

$$\min_{q(t),u(t),F(t),t_f} \mathscr{C}(q(t),\dot{q}(t),u(t),t_f) \tag{1a}$$

subject to

$$u = f(q,\dot{q},\ddot{q},F) \tag{1b}$$

$$c_{\text{meq}}(q,\dot{q},\ddot{q},u) = 0 \tag{1c}$$

$$c_{\text{mineq}}(q,\dot{q},\ddot{q},u) \leq 0 \tag{1d}$$

$$c_{\text{mt}}(q(t_d),\dot{q}(t_d),\ddot{q}(t_d),u(t_d)) = 0 \tag{1e}$$

where q is the vector of parameters of the system, u is the control vector and F the vector of forces and torques applied by the environment, (1b) is the dynamic model of the system, (1c) and (1d) are the semi-infinite constraints at every instant of the motion and (1e) constraints at fixed instants.

To solve the functional optimization problem (1) we implemented a direct method that consists in the discretization of joint angles j and forces k as B-splines (2), while joint torques u are obtained from q and F with the inverse dynamic model. Semi-infinite motion constraints (1c) and (1d) are simply discretized on an equally-spaced grid during the motion.

$$\begin{array}{c} q_j(p,t) = \sum_{i=1}^{n_{qj}} B_{q,i}(t) c_{q,ij} \\ F_k(p,t) = \sum_{i=1}^{n_{Fj}} B_{F,i}(t) c_{F,ik} \end{array} \tag{2}$$

where n_j is the number of basis functions $B_i(t)$, c_{ij} are the B-spline coefficients, N_q is the total number of joints, N_F is the total number of force components. The parameters of the motion are then $p = \{c_{q,ij} \mid j \in [1,N_q], i \in [1,n_{qj}]\} \bigcup \{c_{F,ik} \mid k \in [1,N_F], i \in [1,n_{Fj}]\}$..

Many constraints can be added for the definition of the motion characteristics. Gradients are also computed to improve convergence. The resulting optimization problem is solved with sparse program IPOPT [1] enhanced with a BFGS hessian approximation method. Our contributions include: (i) a model of joint friction using *sign* function, which is regularized to obtain a C^2 problem, (ii) an implementation of an automatic scaling method for minimizing gradients disparity, (iii) a regularized distance computation which gives a C^1 problem [2] in order to include collision avoidance with IPOPT which needs derivable functions.

Table 1: Criteria and iteration number for kicking optimization. Reference is with joint friction, dense BFGS, automatic scaling. No friction case is the criteria with friction model for an optimization without friction. L-BFGS is for the limited memory BFGS with an history of 6 iter.

	reference	no friction	L-BFGS	hand scale
Crit.	192.0	349.8	200.6	192.0
Iter.	243	219	330	327

2 Results

We optimized kicking, throwing and lifting trajectories for the 30 dof Humanoid robot HRP-2 [3]. We considered joints limits, actuators limits, no sliding, no take off, and no turn over of the contacts, as well as constraints for the definition of the motion. We considered 12 control points per joint B-spline, and obtained between ≈ 100 and ≈ 400 parameters in total. Optimizations lasted between 1 min and 1h. Table 1 presents those results in more detail. One can notice that friction must be considered for optimal motion generation. BFGS and automatic scaling is improving convergence. BFGS gives a dense hessian while it is approximately half composed with zeros. A sparse BFGS implementation may improve convergence.

Auto-collision avoidance was tested for motions where the solution was giving auto-penetration. Optimizations successfully converged with little overhead due to collision avoidance computations.

References

[1] A. Wachter and L. T. Biegler, "On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[2] A. Escande, S. Miossec, and A. Kheddar, "Continuous gradient proximity distance for humanoids free-collision optimized-postures," in *IEEE RAS/RSJ Conference on Humanoids Robots*, 2007.

[3] K. Kaneko, F. Kanehiro, S. Kajita, H. Hirukawa, T. Kawasaki, K. Hirata, M. Akachi, and T. Isozumi, "Humanoid robot hrp-2," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, 2004.