

Motion Planning for Human-Robot Interaction in Manipulation Tasks *

Claudia Esteves, Gustavo Arechavaleta and Jean-Paul Laumond
LAAS-CNRS, 7 avenue du Colonel Roche, 31077 Toulouse, France

Abstract—This paper presents a motion planner to automatically compute animations for virtual (human, humanoid or robot) mannequins cooperating to move bulky objects in cluttered environments. The main challenge is to deal with 3D collision avoidance while preserving the believability of the agents behaviors. To accomplish the coordinated task, a geometric and kinematic decoupling of the system is proposed. This decomposition enables us to plan a collision-free path for a reduced system, then to animate locomotion and grasping behaviors in parallel, and finally to clean up the animation from residual collisions. These three steps are automatically applied making use of different techniques such as probabilistic path planning, locomotion controllers, inverse kinematics and path planning for closed-kinematic mechanisms.

I. INTRODUCTION

Over the last few decades many different attempts for animating the human figure have been performed [9], [17], each of them attaining a certain degree of autonomy, interactivity and user-controllability.

In this work we cope with developing an automated motion strategy for the cooperation of two or more virtual mannequins that transport an object in a 3-dimensional cluttered environment. The motivation is mainly supported by applications in PLM (Product Lifecycle Management), e.g. maintenance and operation in industrial facilities [2]. The mannequins are considered to be either human figures with walking capabilities or virtual mobile robots.

In this context several behaviors should be combined with a single animation sequence: agents should walk or roll while manipulating a bulky object coordinately with other mannequins. Here, the main challenge is to deal with 3D collision avoidance while preserving the believability of the agents behaviors.

From an algorithmic perspective our approach is a centralized one. This work is an extension of the approach developed in [1] where a single agent manipulating an object was considered. The global task is modeled with a single system that gathers all the degrees of freedom of the agents and the object. This system is automatically built by computing a so-called “*reachable cooperative space*”. Then, three consecutive steps are performed automatically:

- 1) Plan a collision-free trajectory for a reduced model of the system.

- 2) Animate locomotion and manipulation behaviors in parallel.
- 3) Tune the generated motions to avoid residual collisions.

These steps are applied by making use of a probabilistic motion planner to compute the collision-free paths; motion controllers adapted for each kind of mannequin of both, locomotion and grasping behaviors and path planning algorithms for closed kinematic chains to deal with coordinated behaviors. Section II makes a brief overview of the different techniques used in this work. In Section III our system is defined and the underlying principles of our approach are stated. Section IV details the three steps performed in order to generate an animation. Experimental results are shown and discussed in Section V. Finally, conclusions and future work are given in Section VI.

II. RELATED WORK AND CONTRIBUTION

In order to build our global planner we have chosen among several techniques the ones that best provide the desired characteristics on the resulting animation, i.e. believable and automated motions, precision in manipulation and combined behaviors.

Eye-convincing human-like motions can be obtained by using motion-capture based techniques. These kind of techniques have proven to be suitable for real-time applications. However, given the limited number of captured examples in a motion library, the generated animation risks of being repetitive. This problem is partially solved by generating new motions from different sequences of existing ones using interpolation procedures as in [21], [26], [27]. Because of the cyclic nature of human walking, a motion-capture based locomotion controller is particularly well adapted to provide the required believability to the animation.

To attain precise control on the manipulation behavior, we borrow techniques from biomechanics-based approaches where forward and inverse kinematics (IK) algorithms are applied to specify the motions of each of the joints of the articulated figure [3]. Although these techniques are suitable for controlling the motions when a specific target is given, generating believable human-like motions is problematic. In our work, as the locomotion controller will provide the believability to the motions, we solve the grasping task by using an IK algorithm adapted for the arms of the virtual mannequins.

Regarding automated motions, several motion planners for virtual humans can be found in the literature. Within

*C. Esteves and G. Arechavaleta benefit from SFERE-CONACyT grants. This work is partially funded by the European Community Projects FP5 IST 2001-39250 Movie and FP6 IST 002020 Cogniron. The authors would like to thank Julien Pettré and Juan Cortés for their valuable help. Video sequences related to this work can be found at <http://www.laas.fr/RIA/RIA-research-motion-character.html>.

these planners, locomotion and manipulation have been treated separately. Two-dimensional locomotion planners have been proposed in [6], [14]. In our work, the 3-dimensionality of the environment is taken into account. In [22] a simple motion planner that combines behaviors (walking, crawling and side-walking) in a sequential manner is presented. However, these works do not address manipulation issues. In the approaches developed in [13], [28], manipulation is tackled by first planning the motion of the object to be carried and then closing the loop for grasping. However, simultaneous locomotion and manipulation tasks are not considered.

The problem of combining behaviors of articulated human figures using motion captured data has been frequently tackled by applying behavior based controllers (e.g. [4], [5], [18]). Here, motion captured data is labeled as containing one particular behavior or characteristic (run, walk, scratch head, ...). A new "walking-scratching head" sequence can be generated by interpolating configurations of the two original captured data. In our work, behaviors are combined by using the geometric and kinematic decoupling approach described in Section III which involves the decomposition of the mannequin degrees of freedom.

Our contribution is thus, to address all these approaches in a single scheme and at the same time to deal with 3D collision avoidance.

III. MODELING THE SYSTEM

A. Human and mobile robot mannequins

Our system is composed of two or more virtual mannequins and a movable object lying in a 3-dimensional cluttered environment. Each of the virtual mannequins is classically represented as a hierarchy of rigid links connected by joints. In this work, we consider two kinds of mannequins: human figures and mobile robot manipulators.

The skeleton of our virtual human mannequins is composed of 20 rigid bodies articulated by 18 joints with 53 degrees of freedom (DOFs). These joints and bodies are arranged in five kinematic chains that converge in the mannequins root located on its pelvis (see Figure 1(a)).

Analogously, our virtual robot manipulator mannequins are composed of 7 rigid bodies linked by 7 joints with 9 DOFs. These joints form one kinematic chain attached to the base of the robot (Figure 1(b)).

On top of the geometric model, kinematic constraints are imposed. For instance, we might want to consider that virtual human mannequins are allowed to walk only forwards or mobile manipulators to be differential drive robots. Kinematic constraints can also be imposed on the object depending on the application, e.g. keeping a tray with wine glasses horizontal. These constraints will be treated within the motion planning strategy described in Section IV.

B. Behavior-based kinematic model

The main underlying principle of our work is a geometric decoupling of the system. This means that the system DOFs

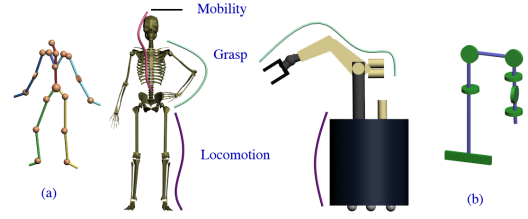


Fig. 1. Our system degrees of freedom are decomposed in three groups: locomotion, grasp and mobility.

are decomposed in groups according to the main task they are sensed to perform. The system contains thus the three groups of DOFs illustrated in Figure 1: locomotion, grasp and mobility,

Locomotion DOFs are the ones involved mainly in the steering of the mannequin in the environment. For the virtual human mannequin these are the DOFs located in its legs and pelvis. For robot mannequins, locomotion DOFs are the ones in its base.

In a similar way, *Grasp* DOFs are in charge of the tasks that involve manipulation, i.e. the arms of the mannequins. In this work, a human mannequin arm is a redundant 7-dof kinematic chain and a 6-dof non-redundant manipulator is considered to be the robot arm.

Mobility DOFs are those that are involved neither in locomotion nor in manipulation but that allow a complementary posture control. For the human mannequin this DOFs lie in the head and spine. In our current robot mannequin there are no mobility DOFs.

The advantage of such a geometric decoupling approach is that a reduced model of the system is obtained for each of the different steps of the planner. In this way, the control and description of the current task is simplified.

C. Reachable cooperative space

To attain cooperation between the mannequins, a description of the space where the object can be manipulated is needed. For a single virtual mannequin, the space in which the object has to lie in order to remain reachable is defined by its arms inherent inverse kinematics. Figure 2 shows an approximation of such a space for each of the mannequins.

In this work, as the mannequins are supposed to carry the same object, we consider the *reachable cooperative space* as the intersection of all the individual spaces. In order to achieve this intersection a non-rigid link between the mannequins is considered. Note that in the case of a large object as in Figure 2 individual reachable spaces are not intersecting, nevertheless both mannequins are still holding the object. This is achieved by considering the object as the end effector of the arms kinematic chain for each of the mannequins. The individual space is therefore enlarged and the cooperative workspace obtained.

In our approach, individual reachable spaces are automatically approximated with the *spherical shells* technique

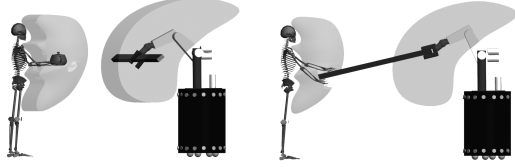


Fig. 2. The individual reachable spaces (left) are enlarged with the object size and give rise to the cooperative space (right).

borrowed from [7]. This technique consists on computing the intersection of the volume between basic shapes (concentric spheres and cones) and parameterize it with the geometry of the closed kinematic chain.

IV. ALGORITHM

Our approach relies mainly on three stages: planning, animating and tuning. Several techniques are suitable for solving each stage. We have adopted some of them in a hierarchical manner to finally achieve collision-free planning motions for the whole system. These stages are described in the following paragraphs.

A. Path Planning

In the planning phase, a reduced geometric model of the system is employed. This simplified model is defined by three different elements: two boxes bounding the locomotion DOFs of each mannequin and a box bounding the object DOFs (see Figure 3). This means that a 12-DOF system is considered at this level. Six of them are the 3-dimensional position and orientation of the object. The other six are the planar position and orientation of each mannequin box. 3 DOFs will be added to the reduced model for each additional mannequin present in the scene.

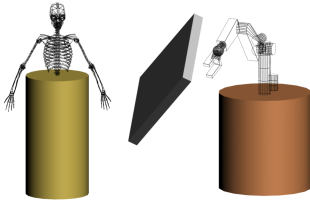


Fig. 3. A 12-DOF reduced geometric model of the system is employed in the planning phase.

Given the user-defined initial and final configurations of the system in the 3D environment, the first step is thus to plan a trajectory for the simplified model described above. For this, a probabilistic roadmap method is applied [12]. This approach is performed in two steps: a learning and a query phase. The principle of the learning phase is to generate random valid configurations for the system within the allowed range of the articulation limits of each DOF. During the sampling, a local planner tries to connect pairs of random configurations to incrementally construct a roadmap (graph) that captures the topology of

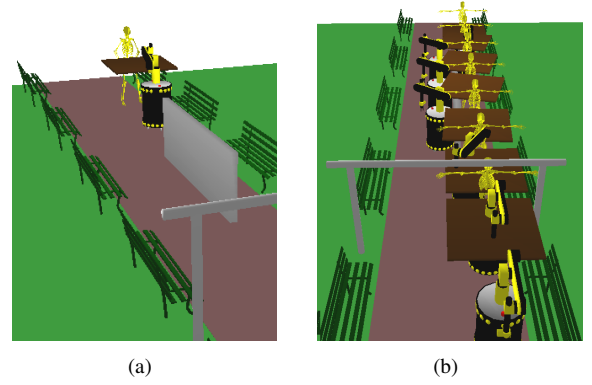


Fig. 4. (a) Initial configuration on the 3D cluttered environment. (b) Some configurations obtained after the planning step.

the configuration space. Having the pre-computed roadmap, the query phase performs a graph search to find feasible paths between the initial and final configurations.

The sampling strategy we use is described in [24]. This strategy captures well the topology of the configuration space in a compact roadmap. Local paths are computed by applying the adequate steering method. The selection of the steering method depends on the kinematic structure of each mobile entity that is part of the system. Based on this consideration, Bezier curves of third degree are computed for the human mannequin. For the object to be manipulated, as well as for holonomic mobile robots, a straight line segment is used. For the case of non-holonomic (differential constraints) mobile robots, a dedicated local planner can be used [15].

After the roadmap is constructed, a connecting path between the initial and final configurations is searched. If the path is found, then it is transformed into a trajectory (i.e. a time parametrized path) with user-defined velocity and acceleration constraints. It is important to note that at this level only locomotion and object DOFs are ensured to be free of collisions.

Figure 4(a) shows a configuration of the system in the 3D environment. Here, the system is composed by a virtual human mannequin, a holonomic mobile manipulator and an object. The barrier at the middle of the walk path forces the robot to take the left side while the human mannequin can still walk on the right side. In Figure 4(b) some configurations for the 12-DOF system on the computed trajectory are illustrated. In this case, the human mannequin follows its bounding cylinder while the object and the robot follow their own steering method according to the trajectory. Note that the object remains within the reachable space between the human mannequin and the robot.

B. Behavior control

At this stage, the trajectory for the 12-DOF simplified model is already planned. The next step is to synthesize the motions for the complete system involving the locomotion, mobility and grasp DOFs. This is done by applying two

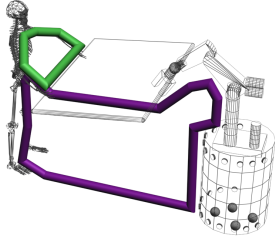


Fig. 5. In the animation phase two closed kinematic chains are formed between the mannequins and the object.

different techniques: a locomotion controller to animate locomotion DOFs (pelvis and legs) and mobility DOFs (spine and head) and an inverse kinematics algorithm adapted for each kinematic chain labeled as grasp DOFs (the human mannequin and the robot arms).

The locomotion controller we have adopted [19] is based on motion capture blending techniques. The result of this controller is a walking sequence for the digital human mannequin.

In order to synthesize the coordinated manipulation motions between the virtual mannequins, the IK solution for each arm is computed in order to reach the values imposed by the object configurations in the first stage. For anthropomorphic arms the inverse kinematics algorithm proposed in [25] is used. After applying the steps mentioned above, a closed-chain mechanism is formed. As it is shown in Figure 5, two closed-loops exist. One is formed by the human mannequin and the object (body-arms-object) and the other by both mannequins (body-object-robot-floor).

Having in mind that the mobility, as well as the grasp DOFs of each mannequin could be in collision, a post-processing stage is performed. In such a tuning phase, closure constraints are considered while the believability of the motions is preserved.



Fig. 6. Some configurations of the obtained manipulation and locomotion behaviors.

Figure 6 illustrates the synthesized motions in the animation stage. Here, some frames (configurations) of the generated animation are shown. Note that the mannequins head is still in collision.

C. Tuning

The purpose of this last stage is to solve the possible residual collision along the animated sequence. This is at-

tained by a local deformation of either the mobility (spine-head) or grasp (arms-object) kinematic chains until a valid random collision-free configuration is reached. Thereafter, a warping method is computed (see [20]) to preserve the smoothness of the animation. Finally, the configurations that have been generated are optimized to obtain a minimal deformation.

Considering the case of collisions involving the grasp DOFs, a local planner addressing on closed kinematic chains should be used. A few path planning methods for closed kinematic chains have been proposed in the literature [8], [10], [16]. However, in order to deal with multi-loop closure constraints we use the variant of RLG algorithm for multiple robots proposed in [7]. For this, we consider the object DOFs as the active part of the closed kinematic chain and the grasp DOFs as the passive part. The goal is to make the active chain reachable by all passive chains simultaneously. This is done by performing a guided-random sampling of the active chain within the intersection space formed between the reachable space of each passive chain. The configurations of passive chains are found using inverse kinematics.



Fig. 7. Sequence of modified motions in the tuning step to avoid collisions with the head of the mannequin.

Figure 7 illustrates the tuning stage by a sequence where the human mannequin avoids collision with his head.

D. Failure Recovery

After these three stages, if there are no collision-free configurations found, the path generated in the first planning stage is invalidated. We remove from the roadmap the edge corresponding to the local path where the tuning step failed. Then a new global search is performed in order to find a new path: it consists in extending the initial roadmap.

V. EXPERIMENTAL RESULTS

Our algorithm has been implemented within the motion planning platform Move3D [23] as an extension of the architecture proposed in [19]. In the following paragraphs examples in different scenarios and different mannequins are presented.

A. In the Factory

In this example a virtual human and a robot mannequin have to transport a slab in a typical industrial environment with plenty of complex obstacles (pipes, drums, beams, ventilation units, etc.). Figure 8 shows some of the configurations of the resulting animation illustrating the trajectory followed by the system. As the system approaches to the

final configuration, it is seen how the slab is turned as a result of the planning step in order to avoid the pipes.

In the animation stage the locomotion behavior is animated and inverse kinematics applied in order to grab the object in its new configuration. At the beginning of the trajectory the human mannequin head collides with the balcony. This collision is automatically solved in the tuning step by bending the spine and head of the mannequin. Once the obstacle is left behind, the digital mannequin smoothly regains its posture to continue the cooperative task until the final configuration is reached.

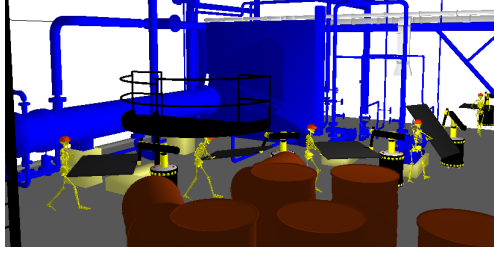


Fig. 8. The agents deal with several obstacles while cooperatively transporting a slab.

Figure 9 shows a set of frames extracted from the final animation. Here, we can see that the motion of the system remains plausible after applying the three steps in the algorithm. Note that the trajectory planned for each of the agents in the scene remains collision-free and their position close enough to keep holding the slab.

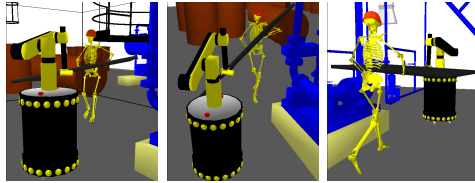


Fig. 9. Some configuration extracted from the final animation.

B. Transporting the Piano

In the second example we treat a version of the Piano Mover's problem with two mobile manipulators and a virtual human mannequin. Here, we want to transport the grand piano in a living-room environment. The living-room is small and contains large objects (tables, a desk ...), the walkways are therefore narrow and collisions are likely to occur.

A collision-free path for the 15-DOF reduced system was found between the desk and the piano chair. Figure 10 illustrates some of the synthesized motions for the mobile manipulators and the human mannequin. The virtual robots open the way in order to avoid colliding with the piano chair and then regaining their posture.

The cooperation among the agents is ensured along the animation. The 3-dimensionality of the resulting animation is mainly seen in the piano configuration. Figure 10 shows

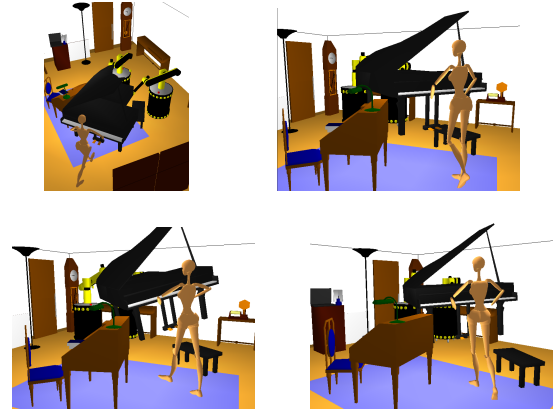


Fig. 10. Two mobile manipulators and a virtual human mannequin cooperating to transport a piano. The piano should be raised in order to avoid collision with the desk.

sequenced frames of the planned animation in order to avoid collision between the piano and the desk. In this example no residual collisions were found.

C. Computational time

The planner has been tested on a workstation Sun-Blade-100 with a 500MHz UltraSparc-IIe processor and 512 MB RAM. In Table I the number of polygons in the different environments as well as the polygons in the system are presented. In the industrial environment we have identified the polygons that participate in the collision test (i.e. the ones no higher than the human mannequin head). In the columns and in the living-room environment, all polygons are considered for collision test purposes.

TABLE I
MODEL COMPLEXITY (NUMBER OF POLYGONS).

	Environment	System
Factory		
- Complete	159,698	18,347
- Col.Test	92,787	
Living-Room		
- Complete	19,077	16,210

The required time to compute the examples presented above are given in Table II (averaged over 100 runs). Here, the results of the planning step are expressed considering a pre-computed graph of the environment product of the learning phase. The time taken to build this roadmap was 31.4 sec. and 3.2 sec. for the factory and the living-room respectively. The time it takes to compute such graph varies with the complexity of the environment and the fact that it is a probabilistic approach. The tables present only the results of the queries. Two different animations were generated for each trajectory, the second doubling the number of frames.

As it was expected, the time it takes the planning step in the industrial environment is significantly higher given its size and complexity. Note that when the number of frames

TABLE II
COMPUTATIONAL TIME (IN SECONDS).

	Factory	Living-Room
No. Frames	268 530	78 151
Stages		
I. Planner		
- Path	6.5 – 6.5	3.3 – 3.3
- Trajectory	2.1 – 4.5	0.6 – 1.3
II. Animation		
- Locomotion	0.8 – 1.6	0.2 – 0.5
- Manipulation	0.4 – 0.8	0.1 – 0.3
III. Residual Col.	5.7 – 11.4	0.0 – 0.0

in the animation vary the time it takes to compute the path does not change because at this stage time is not yet taken into account.

In the animation step it is clearly seen that computational time increases proportionally with the number of frames. It is also seen that the animation step is the fastest step of the algorithm.

The tuning step relies heavily on the complexity of the environment but also on the number of frames with collision. Note that in the living-room example the time it takes to clean the animation is zero because residual collisions were not found.

VI. CONCLUSION

We presented an approach to plan and synthesize collision-free motions for virtual mannequins handling a bulky object in a 3D environment. To accomplish this coordinated task, a geometric and kinematic decoupling of the system is proposed. This decomposition enables us to plan a collision-free path for a reduced system, then to animate the locomotion and grasp behaviors in parallel and finally to clean up the animation from residual collisions. These three steps are automatically applied consecutively making use of different techniques such as motion planning algorithms, locomotion controllers, inverse kinematics and path planning for closed-kinematic mechanisms.

At this stage manipulation planning is not considered. This problem has been previously tackled in computer animation mainly by [13]. However, more complicated instances of this problem involving several mannequins and movable objects can be developed. This is one of our goals in the near future.

Work should also be done in order to obtain a larger set of motions for our virtual mannequins. We intend to achieve this by incorporating simulation-based approaches (e.g., [11]) in our global planning framework.

REFERENCES

- [1] G. Arechavaleta, C. Esteves, and J.-P. Laumond, "Planning fine motions for a digital factotum," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004.
- [2] N. Badler, C. Erignac, and Y. Liu, "Virtual humans for validating maintenance procedures," *Commun. ACM*, vol. 45, no. 7, pp. 56–63, 2002.
- [3] N. I. Badler, C. Phillips, and B. Webber, *Simulating Humans: Computer Graphics Animation and Control*. Oxford University Press, Inc., 1993.
- [4] B. Blumberg and T. Galyean, "Multi-level direction of autonomous creatures for real-time virtual environments," *Computer Graphics*, vol. 29, no. Annual Conference Series, pp. 47–54, 1995.
- [5] M. Brand and A. Hertzmann, "Style machines," in *Proc. of SIGGRAPH'00*. ACM Press/Addison-Wesley Publishing Co., 2000, pp. 183–192.
- [6] M. Choi, J. Lee, and S. Shin, "Planning biped locomotion using motion capture data and probabilistic roadmaps," *ACM Transactions on Graphics*, vol. 22, no. 2, pp. 182–203, 2003.
- [7] J. Cortés and T. Siméon, "Probabilistic motion planning for parallel mechanisms," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2003.
- [8] J. Cortés, T. Siméon, and J.-P. Laumond, "A random loop generator for planning the motions of closed kinematic chains using prm methods," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2002.
- [9] R. Earnshaw, N. Magnenat-Thalmann, D. Terzopoulos, and D. Thalmann, "Guest editors' introduction: Computer animation for virtual humans," *IEEE Computer Graphics Applications*, vol. 18, no. 5, pp. 20–23, 1998.
- [10] L. Han and N. Amato, "A kinematics-based probabilistic roadmap method for closed chain systems," in *Proc. of International Workshop on Algorithmic Foundations of Robotics WAFR*, 2000.
- [11] F. Kanehiro, H. Hirukawa, and S. Kajita, "Openhrp: Open architecture humanoid robotics," *International Journal of Robotics Research*, vol. 23, no. 2, pp. 171–186, 2004.
- [12] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.
- [13] Y. Koga, K. Kondo, J. Kuffner, and J.-C. Latombe, "Planning motions with intentions," in *Proc. of SIGGRAPH*. ACM Press, 1994, pp. 395–408.
- [14] J. Kuffner, "Goal-directed navigation for animated characters using real-time path planning and control," *Lecture Notes in Computer Science (CAPTECH)*, vol. 1537, pp. 171–186, 1998.
- [15] J.-P. Laumond, Ed., *Robot Motion Planning and Control*. Springer-Verlag, 1993.
- [16] S. M. LaValle, J. Yakey, and L. E. Kavraki, "A probabilistic roadmap approach for systems with closed kinematic chains," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 1999.
- [17] R. Parent, *Computer Animation: Algorithms and Techniques*. The Ohio State University: Morgan-Kaufmann Publishers, 2001.
- [18] K. Perlin, "Real time responsive animation with personality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 1, no. 1, pp. 5–15, 1995.
- [19] J. Pettré, J.-P. Laumond, and T. Simeón, "A 2-stages locomotion planner for digital actors," in *Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA)*, San Diego, California, 2003, pp. 258–264.
- [20] —, "3D collision avoidance for digital actors locomotion," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, Nevada, October 2003, pp. 400–405.
- [21] C. Rose, M. Cohen, and B. Bodenheimer, "Verbs and adverbs: Multidimensional motion interpolation," *IEEE Computer Graphics and Applications*, vol. 18, no. 5, pp. 32–41, 1998.
- [22] Z. Shiller, K. Yamane, and Y. Nakamura, "Planning motion patterns of human figures using a multi-layered grid and the dynamics filter," in *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, 2001.
- [23] T. Siméon, J.-P. Laumond, and F. Lamiraux, "Move3d: A generic platform for motion planning," in *Proc. 4th International Symposium on Assembly and Task Planning (ISATP)*, 2001.
- [24] T. Siméon, J.-P. Laumond, and C. Nissoux, "Visibility based probabilistic roadmaps for motion planning," *Advanced Robotics*, vol. 14, no. 6, 2000.
- [25] D. Tolani, A. Goswami, and N. I. Badler, "Real-time inverse kinematics techniques for anthropomorphic limbs," *Graphical Models*, vol. 62, pp. 353–388, 2000.
- [26] M. Unuma, K. Anjyo, and R. Takeuchi, "Fourier principles for emotion-based human figure animation," in *Proc. of SIGGRAPH*. ACM Press, 1995, pp. 91–96.
- [27] A. Witkin and Z. Popovic, "Motion warping," in *Proc. of SIGGRAPH'95*, 1995, pp. 105–108.
- [28] K. Yamane, J. Kuffner, and J. K. Hodgins, "Synthesizing animations of human manipulation tasks," in *Proc. of SIGGRAPH*, 2004.