Proceedings of the 2006 IEEE International Conference on Robotics and Automation
Orlando, Florida - May 2006

Towards Provable Navigation and Control of Nonholonomically Constrained Convex-Bodied Systems

David C. Conner, Howie Choset, and Alfred A. Rizzi
Robotics Institute, Carnegie Mellon University
Pittsburgh, Pennsylvania 15213
Email: {dcconner+,choset+,arizzi+}@ri.cmu.edu

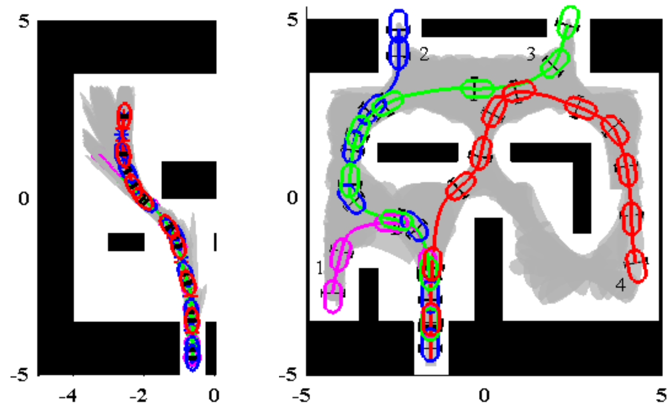
***Abstract*— In this paper, we develop a generic class of control policies that respect nonholonomic constraints and are provably safe with respect to obstacles for a convex-bodied mobile robot. We instantiate this class of policies over local regions of configuration space, and compose the resulting local policies to address the global navigation and control problem for a wheeled mobile robot navigating amongst obstacles. Simulation and experimental results are given.**

I. Introduction

We explore a general solution to the problem of simultaneously planning and controlling the motion of a convex-bodied vehicle in a cluttered planar environment. Our approach relies on a generic class of local control policies that we instantiate over local regions of the robot's free configuration space (termed cells). The automated sequential application of the local policies effectively solves the global navigation and control problem simultaneously. The result is a global strategy, which, given the current configuration (position and orientation) of the robot determines which local policy to apply, and calculates the control inputs based on that policy. The determination of which policy to apply follows from a partial order of the cells that is determined off line, and each local policy specifies a configuration dependent velocity reference defined over its region of applicability. We guarantee that the resulting essentially global vector field respects the system constraints, and that the policies are composed in such a way that the resulting system trajectories are safe with respect to environmental obstacles. Throughout this paper, we will refer to the combination of a configuration space cell and its associated reference vector field as a *policy*. We explore these ideas both through a formal outline of their technical correctness, and experimentally in simulation and on a laboratory mobile robot system. Figure 1 presents some representative paths taken from two different rounds of experiments discussed in Sections IV and V.

II. Related Work

Our approach is based on the technique of sequential composition [1]. This technique enables the construction of switched control policies that have guaranteed behavior, and provable convergence properties. In its original form, sequential composition used positively invariant state regulating feedback control policies whose domains of attraction partition the state space into cells. Given two control policies, Φ_2 is said to *prepare* Φ_1 if the goal set of Φ_2 is contained in the domain of Φ_1.



a) Initial experiments using ellipsoidal cells.

b) Recent experiments using piecewise superquadric cells.

Fig. 1. Paths induced by our hybrid control policy for both simulation and actual runs on a laboratory robot. The light gray regions show the projection of the policy deployment's domain into the workspace. The results shown for the robot experiments are based on dead-reckoned data.

The overall control policy induced by sequential composition is fundamentally a hybrid control policy [2], [3]. The stability of the underlying control policies guarantees the stability of the overall policy [1]. Disturbances are robustly handled provided their magnitude and rate of occurrence is relatively small compared to the convergence of the individual policies.

Sequential composition-like techniques have been applied to mobile robots. Both overlapping polytopes [4] and balls [5] in configuration space have been used to specify domains for holonomic systems. These methods have been extended to second order systems with complicated bounds on acceleration and velocity [4].

Sequential composition has also been used to control wheeled mobile robots using visual servoing [6], [7]. In both cases, the control policies were designed based on careful analysis of the system, its constraints, and the problem at hand.

Others have focused on automated deployment of generic policies [8], [9]. These approaches focused on idealized (point) fully-actuated systems in simulation. The former defined potential functions over local polytopes, and composed them to generate global control policies. The later used affine control polices defined over polytopes, and was extended to an idealized robot with unicycle kinematics using feedback linearization.

III. Definition of Local Policies

For this paper, we assume a kinematic model of the system where the velocity is given by $\dot{q} = A(q)\,u$ for some configuration $q \in \mathcal{Q}$, bounded input $u \in \mathcal{U} \subset \mathbb{R}^m$, and connection $A(q)$, which maps inputs to configuration velocities; that is $A(q) : \mathcal{U} \to T_q\mathcal{Q}$. Given the current configuration of the robot, q, our policy calculates a control input, u, that drives the robot to the local policy's goal set without leaving the cell first.

We expand the conventional definition of *prepares* from a relation between two policies, to a relation between a policy and a set of policies. Now, we say a selected policy prepares a set of policies if the goal set of the selected policy is contained in the union of the domains of the policies in the set. We also allow policies that flow through a designated goal set; for example, driving out a specified doorway.

We will describe the development of our generic policy class in three steps. First, we define the constraints imposed by our input bounds and nonholonomic constraints on velocity. Next we describe our technique for respecting the constraints imposed by the workspace obstacles. Finally, we describe our technique for generating a continuous vector field that respects the system constraints and drives the system to the designated goal set.

1) Limitations Due to Velocity Constraints: In order for the cell to be valid, the cell must be *conditionally positive invariant* under the influence of the control for any initial condition in the cell [6]. Furthermore, the system must reach the designated goal set in finite time. To relate these restrictions to our system constraints, assume there exists an outward pointing unit normal, n, almost everywhere on the cell boundary. Let $\omega(q) = (n(q))^T A(q)$; $\omega(q)$ defines two half-space constraints in the input space. Let, $\omega^+(q)$ ($\omega^-(q)$) denote the strictly positive (negative) half-spaces defined by $\omega(q)$. For a flow-through policy, which has its goal set on the boundary of the cell, we must exit the cell via the goal set; that is $n \cdot \dot{q} > 0$. Hence, we have the necessary condition that $\omega^+(q) \bigcap \mathcal{U} \neq \emptyset$. To satisfy the conditional positive invariance property, we must be able to generate a robot velocity that moves the system into the cell from all other points on the boundary; that is $n \cdot \dot{q} < 0$. Thus, we have the necessary condition $\omega^-(q) \bigcap \mathcal{U} \neq \emptyset$. This relationship is shown in Figure 2. To verify the validity of a given cell, we must know $n(q)$, $A(q)$, $\mathcal{U}$, and the designated goal set.

We must also show that our policy can actually bring any initial state within the cell to the goal in finite time. We cannot give a proof due to space limitations, but one intuitive approach is to grow the cell as a wavefront expansion from the goal set, as shown in Figure 3. We require that $\omega^-(q) \bigcap \mathcal{U} \neq \emptyset$ be satisfied for all wavefronts behind the current wavefront,

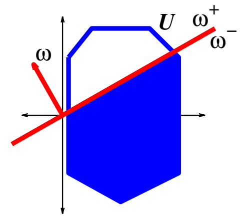


Fig. 2. The input space is constrained to satisfy the configuration velocity constraint.

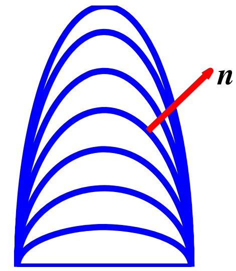


Fig. 3. Growing a cell as a family of level sets.

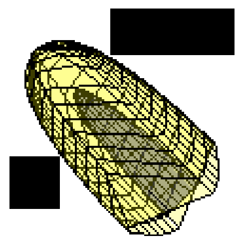


Fig. 4. Projection of a padded cell into the workspace. The cell is the dark inner surface projection; the lighter region is the projection of the padded cell.

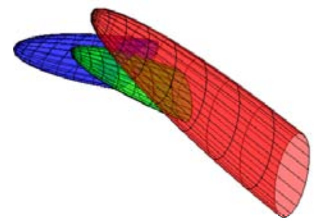


Fig. 5. Deployment of several ellipsoidal cells in the free configuration space.

and that the wavefront stop expanding prior to $\omega^-(q) \bigcap \mathcal{U} = \emptyset$. Given a family of such wavefronts, there is always a control action that drives the system to a configuration corresponding to a wavefront prior to the critical wavefront; thus bringing the system closer to the goal.

2) Limitations Due to Workspace Obstacles: Another necessary condition on our cells is that they exist in the free configuration space of the robot. This, coupled with conditional invariance, guarantees that the system will not collide with an obstacle while the corresponding policy is being executed. In order to avoid constructing the free configuration space, and testing the cell to see if it intersects the configuration space boundary, our approach is to "pad" the cell. This "padding", for each configuration on the cell boundary, represents the extent of the robot body beyond the cell boundary. We project this padded cell into the x-y plane corresponding to the robot workspace, as shown in Figure 4. If the projection does not intersect a workspace obstacle, then the cell is contained in the free configuration space and any trajectory remaining in the cell is guaranteed to be collision free.

3) Definition of the Vector Field: We generate a reference vector field and solve our control problem at the same time by considering a constrained convex optimization. We assume that we have defined our cell such that there is a family of level sets expanding from the goal until they reach the cell boundary, as described before. For any configuration in the cell, we determine the corresponding level set, and calculate the normal vector as shown in Figure 3. We calculate $\omega(q) =$

$(n(q))^T A(q)$, which we then use as an input space constraint in our optimization. We desire a continuous vector field, so we define our cost function to include a simple quadratic term, and solve for the optimal action $u^* \in \mathcal{U}$ such that $\omega \cdot u^* < 0$. For any valid cell, a solution will exist. By definition, the induced vector field , $A(q)\, u^*$, is admissible for a valid cell in that it obeys the nonholonomic constraints, is non-zero everywhere, and its flow brings the system to the goal set.

IV. Initial Results Using Ellipsoidal Cells

To test our approach, we began by experimenting with half-ellipsoidal cells. These cells have a natural expression both parametrically and implicitly as the level set of a function. Figure 5 shows a limited deployment of three ellipsoidal cells.

The major drawback of ellipsoidal cells is that they do not represent a natural shape with respect to the trajectories induced by the system constraints [10]. This means that only a limited region can be bounded by a given cell. As a consequence, it takes a large number of cells to execute a turn. In these initial experiments we used 129 policies to execute the curve shown in Figure 1-a. In spite of this limitation, we successfully demonstrated the utility of the approach in both simulation and on an actual robot. Figure 1-a shows some of the results of the simulations and actual robot experiments on a combined plot.

We simulated a differential-drive robot with bounded car-like steering, using a variety of initial conditions, along with some disturbances. The first disturbance added random noise with uniform distribution, zero-mean, and amplitude equal to the amplitude of our input space to our control input calculation, u^*. The second disturbance added a 10% bias to the control input. The next simulation combined both noise and bias; this is shown in Figure 1-a for two initial conditions. In all cases, in spite of the high noise level, the control system drove the system to the goal. The resulting workspace trajectories were nearly indistinguishable.

We implemented our control strategy on a Nomad Scout 2 mobile robot using the same deployment generated for our simulation. Two of our experimental runs are plotted in Figure 1-a with similar initial conditions to our plotted simulations. The resulting workspace trajectories are nearly indistinguishable from the simulation, with the caveat that we are displaying dead reckoned results and not ground truth.

V. Conclusion

The results of our simulations and experiments validate the basic approach. All of plots in Figure 1, both simulations and robot experiments, appear to be following a specific path, when in fact the induced paths are integral curves of the switched policies. A desired path is never explicitly calculated.

The results also show (as expected) the limitations of using ellipsoids as the basic cell shape. Our recent work has expanded our cells to include cells defined by two superquadric-like surfaces; we report on this most recent work in [11]. A portion of our deployment with 184 new policies is shown in Figure 6; some results from our latest experiments using a different robot are shown in Figure 1-b. As before, these plots show configuration estimates based on dead-reckoning. We have plans to implement a vision-based localization system to determine performance under real world conditions.

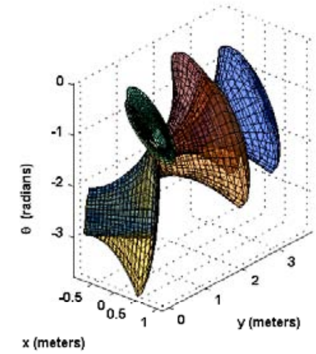


Fig. 6. Deployment of several superquadric based cells in the free configuration space.

We plan to extend the results to more complicated configuration spaces and connections, such as those for the 4-variable Ackermann steered car and a diff-drive towing a trailer. Finally, we plan to extend the work to develop policies for systems with 2nd order control dynamics and actuator limits.

References

[1] R. R. Burridge, A. A. Rizzi, and D. E. Koditschek, "Sequential composition of dynamically dexterous robot behaviors," *International Journal of Robotics Research*, vol. 18, no. 6, pp. 534–555, 1999.

[2] T. A. Henzinger, "The theory of hybrid automata," in *Proceedings of the 11th Annual Symposium on Logic in Computer Science (LICS)*. IEEE Computer Society Press, 1996, pp. 278–292.

[3] M. S. Branicky, "Studies in hybrid systems: Modeling, analysis, and control," Ph.D. dissertation, MIT, Dept. of Elec. Eng. And Computer Sci., June 1995.

[4] A. Quaid and A. A. Rizzi, "Robust and efficient motion planning for a planar robot using hybrid control," in *IEEE International Conference on Robotics and Automation*, vol. 4, April 2000, pp. 4021 – 4026.

[5] L. Yang and S. M. Lavalle, "The sampling-based neighborhood graph: An approach to computing and executing feedback motion strategies," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 419–432, June 2004.

[6] G. Kantor and A. A. Rizzi, "Feedback control of underactuated systems via sequential composition: Visually guided control of a unicycle," in *Proceedings of 11th International Symposium of Robotics Research*, October 2003.

[7] S. Patel, S.-H. Jung, J. P. Ostrowski, R. Rao, and C. J. Taylor, "Sensor based door navigation for a nonholonomic vehicle," in *IEEE International Conference on Robotics and Automation*, Washington,DC, May 2002, pp. 3081–3086.

[8] D. C. Conner, A. A. Rizzi, and H. Choset, "Composition of Local Potential Functions for Global Robot Control and Navigation," in *IEEE/RSJ Int'l. Conf. on Intelligent Robots and Systems*, Las Vegas, NV, October 2003, pp. 3546 – 3551.

[9] C. Belta, V. Iser, and G. J. Pappas, "Discrete abstractions for robot planning and control in polygonal environments," *IEEE Transactions on Robotics*, vol. 21, no. 5, pp. 864–874, October 2005.

[10] M. Vendittelli, J. Laumond, and C. Nissoux, "Obstacle distance for car-like robots," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 4, pp. 678–691, 1999.

[11] D. C. Conner, H. Choset, and A. A. Rizzi, "Provably correct navigation and control for non-holonomic convex-bodied systems operating in cluttered environments," in *submitted to Robotics:Science and Systems*, 2006.