Vision-based Robot Formations with Bézier Trajectories

Siou Y CHIEM
École Centrale de Nantes
Nantes, France

Enric CERVERA
Robotic Intelligence Lab
Jaume-I University of Castelló, Spain

Abstract. In this paper, we present a simple method for the deployment of multiple autonomous robots, with leader-follower roles. Each follower robot estimates the position and orientation of its leader with a fast color-tracking vision system, and builds a Bézier curve that describes the trajectory between its current position, and the position of the leader robot. The generated trajectories are accurate enough to extend the approach to more follower robots, while keeping a line formation. By the introduction of virtual points, formations of different shapes can be generated. We validate our method both in simulation and with physical robots.

1. Introduction

This paper presents a simple method for generating smooth trajectories for mobile robots in formation. Our goal is to drive a formation of different shapes, as seen in Fig. 1, where one of the robots is the *leader* (marked with a black triangle) and the others are the *followers*. We assume that the leader robot has a precise positioning system, e.g. a laser rangefinder, and the follower robots are endowed with vision systems, capable of determining the position and orientation of other robots (the leader, or other followers).

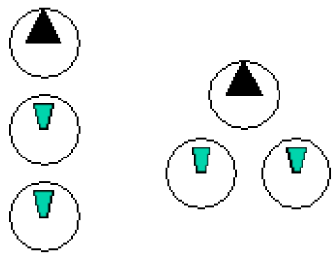

Figure 1: linear and triangular robot formations.

The interest in robot formations lies in the ability to deploy a number of robots, potentially heterogeneous. Research in this area has grown significantly in recent years. Das et al. [4] present a paradigm for switching between simple decentralized controllers that allows for changes in formation, and they use of information from a single type of sensor, an omnidirectional camera, for all the controllers. On the other hand, Michaud et al. [9] use a pan camera head, sonar readings and wireless communication; their robots are able to move in formation, avoid obstacles and switch formations, but also to initialize and determine by themselves their positions in the formation.

Local information is preferred in large formations, due to observability constraints. Jadbabaie, et al. [8], develops a nearest neighbor rule for *n* autonomous agents in the plane with the same speed but with different headings.Local sensing and control is proposed by Fredslund and Mataric [7], but in their team of homogeneous robots, each one is equipped with sonar, laser, camera, and a radio link for communicating with the others. Desai et al. [5] use methods of feedback linearization for controlling formations that uses only local sensor-based information, in a leader-follower motion.

Akella and Hutchinson [1] address the task of coordinating the motions of multiple robots when their trajectories (defined by both the path and velocity along the path) are specified. Other more complex mathematical approaches include potential functions, as developed by Balch and Hybinette [2], graph theoretic methods, proposed by Desai [6], or trajectory generation for minimizing energy functions, presented by Belta and Kumar [3].

The rest of the paper is structured as follows: Section 2 proposes a simple method for a following behavior, onto which formations are built, based on Bézier curves [10]. The method is validated through simulations in Section 3, and extended to manage other formations than the line. In Section 4, experiments with real robots are presented. Finally, Section 5 draws some conclusions and sketches the future work.

2. Bézier trajectory between leader and follower robots

Determination of the position and orientation of the leader robot can be achieved by estimating its distance and relative orientation with regard to the follower robot. As seen in Fig. 2, only the position (x,y) and the orientation θ needs to be estimated. In the proposed setup, observer cameras are mounted on pan-tilt platforms, and the pan angle ψ can be read from the platform. Thus, the position of the observed robot is obtained from its estimated distance, and the pan angle. By using pan-tilt platforms, the orientation of the observer robot needs not to be the same as that of its camera. Thus, the camera is free to follow the observed robot, and keep its image well centered. As will be seen in the following, trajectories do not follow straight lines between robots, so the orientation of the robot may be very different from that of its camera.

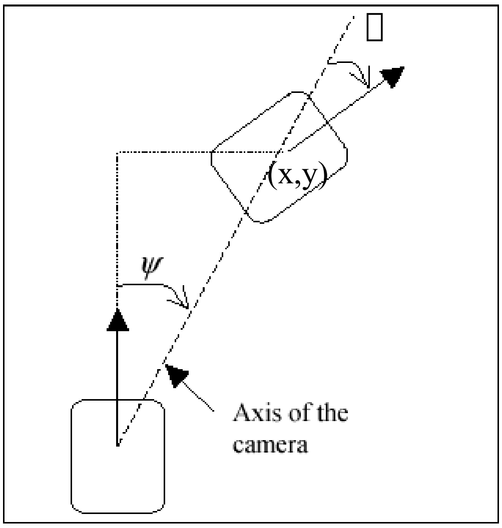

Figure 2: Position and orientation of the leader in the follower's frame of reference.

The key idea is to define a Bézier cubic curve [10] between the leader and follower robots (Fig. 3). One such curve is defined by four points: the *endpoints* P_0 and P_3 are given by the positions of the leader and follower robots (codenamed *Sneezy* and *Happy* in our experiments); the *control points* P_1 and P_2 are chosen along the lines defined by the orientations of the robots.

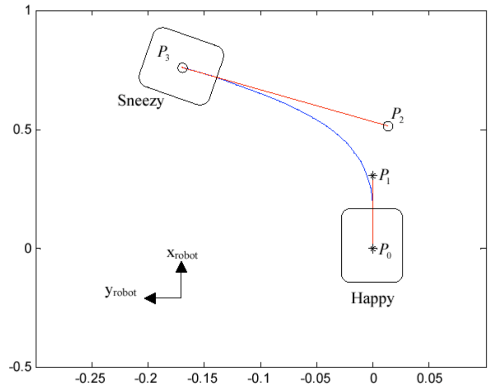

Figure 3: definition of the four points of the Bézier cubic curve.

The cubic Bézier curve is given by:

$$P(t) = at^3 + bt^2 + ct + P_0, \forall t \in [0,1] \qquad (1)$$

where the vectors a, b, c are defined as follows:

$$\begin{aligned} c &= 3(P_1 - P_0) \\ b &= 3(P_2 - P_0) - c \\ a &= P_3 - P_0 - c - b \end{aligned} \qquad (2)$$

and the points are defined in the follower robot's reference frame:

$$P_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad P_1 = \begin{bmatrix} d \\ 0 \end{bmatrix} \quad P_2 = \begin{bmatrix} x - d\cos\theta \\ y - d\cos\theta \end{bmatrix} \quad P_3 = \begin{bmatrix} x \\ y \end{bmatrix} \qquad (3)$$

The values of x, y and θ are obtained from the vision system; the value of d, though, can be chosen arbitrarily. The well-established theory of Bézier curves states two very interesting properties regarding the endpoints:

- the curve passes through the endpoints themselves, and
- the curve is tangent to the vectors $P_1 - P_0$ and $P_3 - P_2$ at the endpoints.

The control points can be arbitrarily chosen along the lines, but very different trajectories are obtained as depicted in Fig. 4.

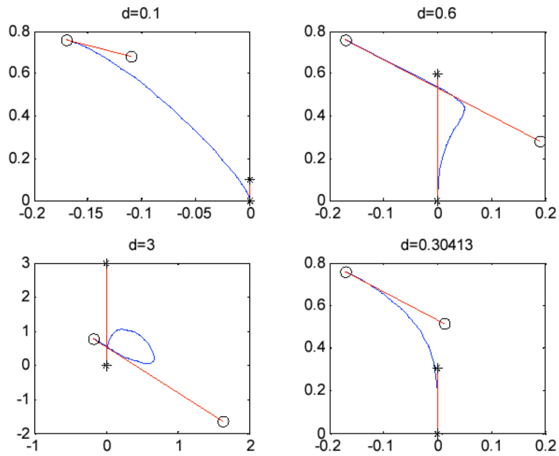

Figure 4: Bézier cubic curves with different values of the scale factor *d*.

In order to obtain invariance to scale, control points are spaced up to a value proportional to the distance between robots, and the same value d is used for both pairs of endpoint and control point. In the rest of the paper, the value of the control point distance is:

$$d = \frac{2}{3}\sqrt{2}\left(\sqrt{2} - 1\right)\left\|\mathbf{P}_3 - \mathbf{P}_0\right\| = \frac{2}{3}\sqrt{2}\left(\sqrt{2} - 1\right)\sqrt{x^2 + y^2} \qquad (4)$$

such that, if the endpoints belong to the circular arc of a quadrant, then the midpoint of the Bezier curve (t=1/2) would lie on the midpoint of that circular arc.

The leader robot is assumed to follow an arbitrary trajectory, with a constant, known linear velocity; in order for another robot to follow the leader, we will assume that it moves with the same linear velocity v than that of the leader. Thus, only the angular velocity of the follower ω needs to be computed at each time step. Such angular velocity must correspond to the *curvature* of the Bézier trajectory at P_0:

$$\omega = \frac{v}{R} = v\kappa \tag{5}$$

where R and κ are the radius and curvature of the trajectory, respectively. The curvature of any parametric curve is:

$$\kappa = \frac{x'y'' - y'x''}{\left(x'^2 + y'^2\right)^{3/2}} \tag{6}$$

Since there is only need to compute the curvature at t=0, then it happens that:

$$\begin{aligned} x' &= v \\ y' &= 0 \end{aligned} \tag{7}$$

so that the curvature at this point is:

$$\kappa = \frac{y''}{x'^2} = \frac{2(y - d\sin\theta)}{3d^2} \tag{8}$$

where d is obtained from (4), and the desired angular velocity is computed (5). The velocity is updated at each video frame, when the new position of the leader is estimated, the Bézier points are defined, and the velocity is computed. In our experiments, the whole process is performed 25 times per second.

3. Simulation results

3.1 Line formations

Simulation results confirm the validity of the presented approach. In the first experiment, two robots are used. Starting from position (0,0), the leader robot (*Sneezy*) moves along a pre-programmed path, consisting of a straight line, an arc to the left, and another straight line. The follower robot (*Happy*) starts 1 meter away from the leader, with the same orientation. As seen in Fig. 5, the trajectory of the follower robot coincides very accurately with that of the leader robot.

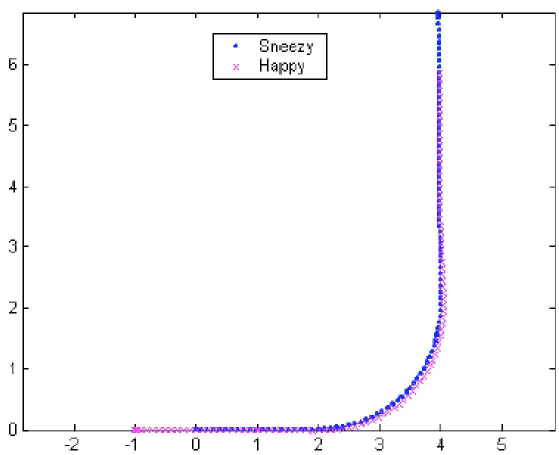

Figure 5: simulation with one *leader* and one *follower* robots.

Line formations with many robots can be easily built: each robot should follow its preceding one. Though trajectories are not exactly coincident, simulated experiments confirm the validity of the approach. See e.g. Fig. 6 for a 3-robot line formation.

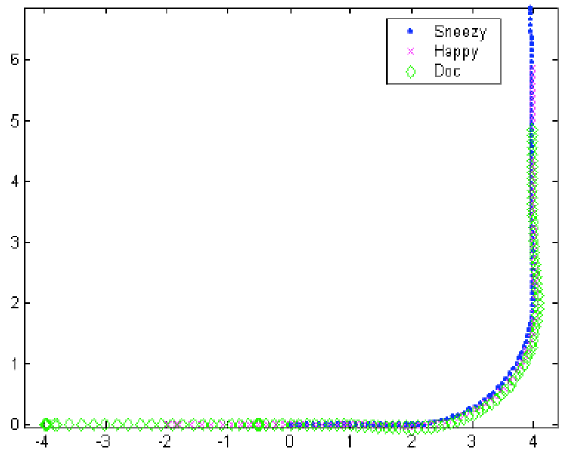

Figure 6: simulation of a 3-robot line formation.

3.2 Extension to other formations

Formations of arbitrary shapes can be obtained by the introduction of *virtual points*. These points do not correspond to the exact position of a leader robot, but with an added displacement. For example, Fig. 7 depicts a triangular formation where the reference for the follower robots (Happy and Doc) is not the leader robot but a point located at a given distance at its left and right respectively. Since the position and orientation of the robot is known, the estimation of the virtual points is trivial.

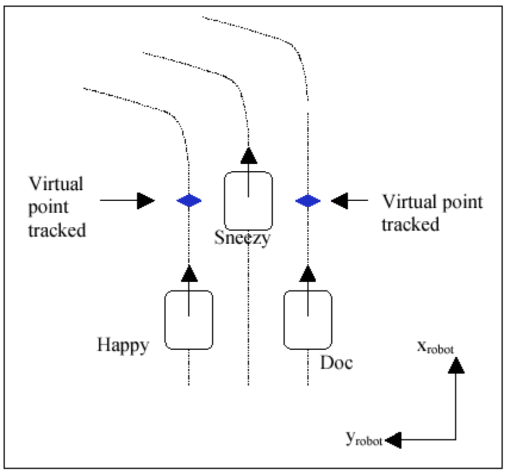

Figure 7: triangular formation with virtual points.

Simulations confirm again the validity of the approach. As depicted in Fig. 8, the follower robots start in a line formation, then move to a triangle, and keep this formation with respect to the leader. It must be noted that the linear velocity of the follower robots is not constant, due to the different radius of their respective trajectories. A simple gain factor has been added, proportional to the distance to the leader robot.

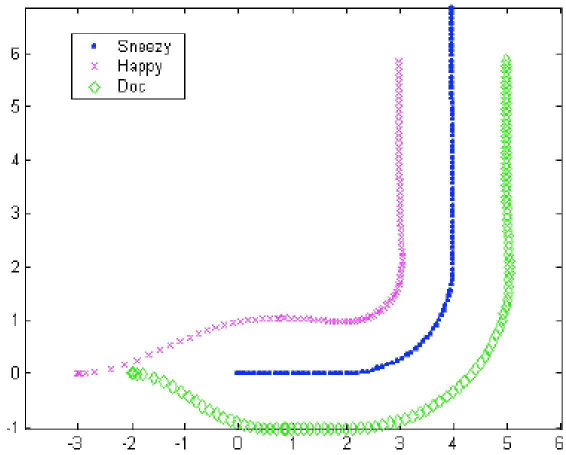

Figure 8: simulation of a 3-robot triangular formation.

4. Experimental results

Pose estimation is based on the tracking of color regions attached to the robot. Fig. 9 depicts a color pattern consisting of a central purple rectangle, and two lateral green rectangles. The central rectangle provides an estimate to the distance based on its perceived and real height. The difference between the perceived heights of the lateral rectangles provides an estimate of the orientation of the pattern with respect to the observing robot.

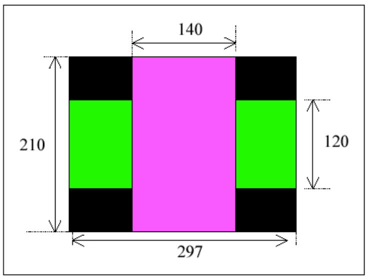

Figure 9: color pattern used for pose estimation (millimeters).

Experiments with real robots in line formations are presented. As stated before, the position of the leader robot is estimated at video rate (25 fps). The points of the Bézier trajectory are then estimated, with the final result being the angular velocity of the follower. The most computer-intensive step is the tracking of the leader.

In the first one, as shown in Fig. 10, the leader robot carries a laser range finder for precise positioning. The follower robot uses a pan/tilt camera and a color-tracking vision system to estimate the position and orientation of the leader robot at video rate.

In the experiment, both robots describe a trajectory with a 90-degree turn to the left. As a straightforward extension, a third robot is added to the line formation, which follows the second one with exactly the same algorithm. As seen in Fig. 11, the trajectories described by both follower robots start and end up in the desired line formation.

5. Conclusion

The presented approach can be extended to more robots, by the appropriate definition of the leader and follower roles for each robot, and the virtual points that must be followed. The vision-based pose estimation is precise enough so that the followers do not need accurate positioning systems. Only the leader robot carries such a system, in order to command the whole formation.

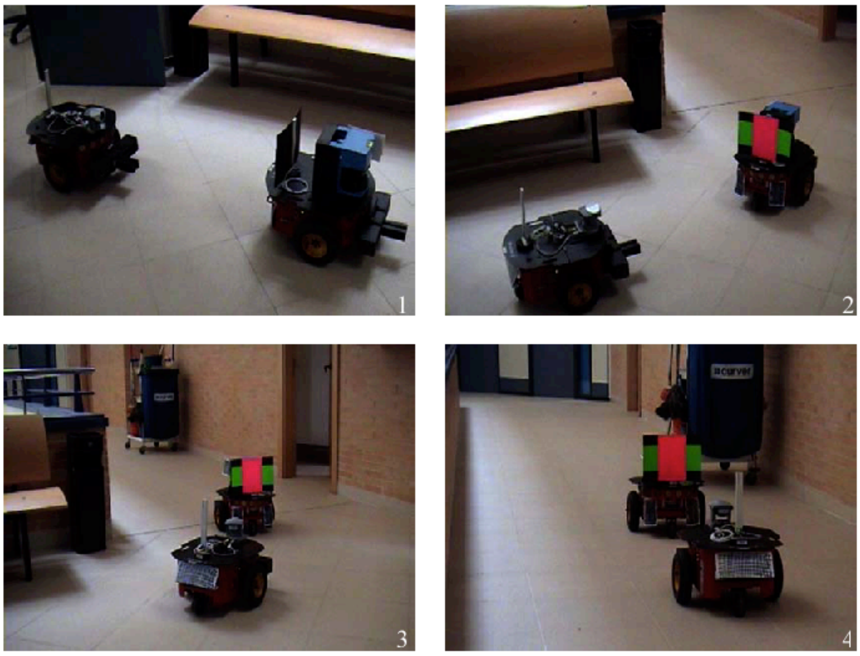

Figure 10: line formation with two mobile robots.

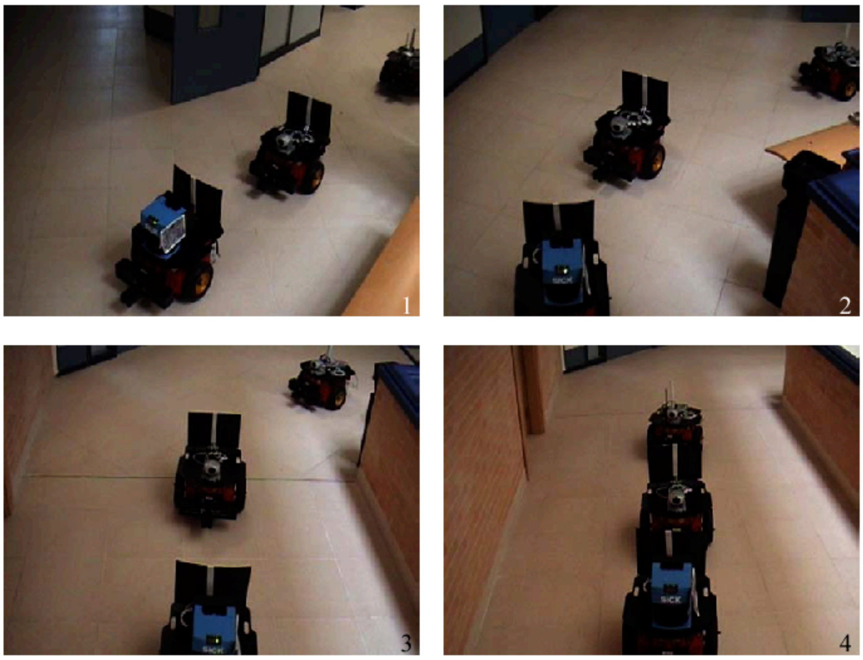

Figure 11: line formation with three mobile robots.

An interesting extension would consist of commanding robot without cameras, from a robot that carries the vision system (Fig. 12). This robot should estimate the position of the leader and the *dummy* robots, and build the appropriate Bézier trajectories. Then the velocity of each robot can be computed and sent to it via radio modem or similar.

The main problem is the need to track different color patterns, which may interfere or cause occlusions, but the advantage is the low-cost of the dummy robots.

Acknowledgements

This work has been partially funded by the Generalitat Valenciana (CTIDIA/2002/195), the Spanish Ministry of Science and Technology (DPI2001-3801, HF2001-0112, FIT-020100-

2003-592), and the Fundació Bancaixa (P1-1B2001-28). The authors gratefully acknowledge this support.

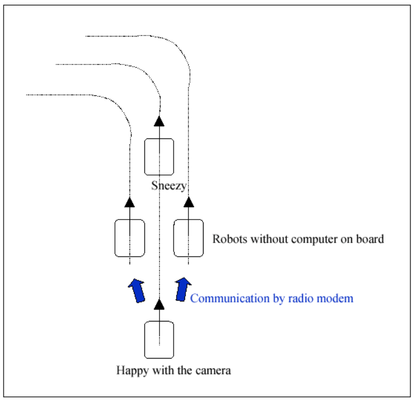

Figure 12: diamond formation with four mobile robots, and only one camera.

Bibliography

[1] Akella, S.; Hutchinson, S.; “Coordinating the motions of multiple robots with specified trajectories”, Proc. of the IEEE International Conference on Robotics and Automation, 2002, vol. 1, pp. 624 – 631.

[2] Balch, T.; Hybinette, M.; “Social potentials for scalable multi-robot formations”, Proc. IEEE International Conference on Robotics and Automation, 2000, vol. 1, pp. 73 – 80.

[3] Belta, C., V. Kumar, V., “Motion generation for formations of robots: a geometric approach”, Proc. of the IEEE International Conference on Robotics and Automation, 2001, vol. 2, pp. 1245 – 1250.

[4] Das, A.K.; Fierro, R.; Kumar, V.; Ostrowski, J.P.; Spletzer, J.; Taylor, C.J. “A vision-based formation control framework”, IEEE Transactions on Robotics and Automation, Volume: 18 Issue: 5, Oct. 2002 , pp. 813 – 825.

[5] Desai, J. P., Ostrowski, J., and Kumar, V., "Controlling formations of multiple mobile robots," in Proc. of the IEEE International Conference on Robotics and Automation, 1998, pp. 2864 – 2869.

[6] Desai, J.P., "Modeling Multiple Teams of Mobile Robots: A Graph Theoretic Approach", Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001, vol. 1, pp. 381 – 386.

[7] Fredslund, J.; Mataric, M.J.; “Robot formations using only local sensing and control”. Proc. of the IEEE International Symposium on Computational Intelligence in Robotics and Automation, 2001, pp. 308 – 313.

[8] Jadbabaie, A.; Lin, J.; Morse, A.S.; “Coordination of groups of mobile autonomous agents using nearest neighbor rules”, Proc. of the 41st IEEE Conference on Decision and Control, 2002, vol. 3, pp. 2953-2958.

[9] Michaud, F.; Letourneau, D.; Guilbert, M.; Valin, J.-M.; “Dynamic robot formations using directional visual perception”, Proc. of the IEEE/RSJ International Conference on Intelligent Robots and System, 2002, vol. 3, pp. 2740 – 2745.

[10] Prautzsch, H.; Boehm, W.; Paluszny, M; “Bezier and B-Spline Techniques”, Springer-Verlag, 2002.