2011 IEEE International Conference on Robotics and Automation
Shanghai International Conference Center
May 9-13, 2011, Shanghai, China

Multi-Robot System for Artistic Pattern Formation

Javier Alonso-Mora, Andreas Breitenmoser, Martin Rufli, Roland Siegwart and Paul Beardsley

Abstract— **This paper describes work on multi-robot pattern formation. Arbitrary target patterns are represented with an optimal robot deployment, using a method that is independent of the number of robots. Furthermore, the trajectories are visually appealing in the sense of being smooth, oscillation free, and showing fast convergence. A distributed controller guarantees collision free trajectories while taking into account the kinematics of differentially driven robots. Experimental results are provided for a representative set of patterns, for a swarm of up to ten physical robots, and for fifty virtual robots in simulation.**

I. INTRODUCTION

In the last decade robotic pattern formation and formation control has experienced a rise of attention along with the advances in multi-robot systems. The applications of pattern formation are broad, and include multi-robot navigation tasks for exploration, escorting and rescue missions, alignment of aerial vehicles, or cooperative control of mobile sensor networks to maintain surveillance or coverage.

The pattern formation task involves the assignment of robots to goal positions that define the final pattern, but also the control of the robots to actually establish the formation. Typical work on pattern formation in robotics like [1] focuses on the positioning of the robots in a specified pattern and measures against the accuracy of the patterns achieved, as the main goal is the final robot formation, which often is the basis to accomplish another task.

In contrast, in this work focus is put on the pattern formation as such. The aim is to generate both visually convincing final formations by optimizing the robots' goal positions, as well as simple and smooth robot motions at the transitions of patterns. Although visual appeal is not directly encoded in our method, the different algorithms of this work were selected with the clear aim for visual appeal.

A recent survey [2] describes the challenges of robotic pattern formation in detail. Many of the identified challenges, such as pattern transformation, collision avoidance, scalability of patterns or formation of multiple patterns are covered by our work and will be presented in the following sections.

The presented method first generates uniform sets of goal positions from input pattern templates by Voronoi partitioning. Thus an accurate representation of the final patterns is provided. Then the robots are driven towards the goal positions in an iterative process, by first performing a multi-robot goal assignment based on the Hungarian algorithm [3] to coordinate the robots, which leads to short paths and fast convergence of the robots to the goal. Finally, the robots smoothly avoid collisions using the distributed collision avoidance method presented in [4]. This two last steps are repeated until convergence.

We demonstrate our approach in experiments with up to ten differentially-driven robots and in simulations with fifty robots. The presented multi-robot system is capable of creating multi-pattern formations as well as transitions between arbitrary patterns, involving filling and non-filling patterns likewise. Further, reconfiguration of patterns is achieved in real time without need for costly preprocessing and robots recover from failures and adapt to changes in patterns reliably.

Similar to our goal assignment approach, the Hungarian algorithm was implemented for shape-reconfiguration in [6] among others. In [7] a scalable and distributed algorithm for shape transformation based on median consensus was presented and showed a performance close to that of the optimal assignment by the Hungarian algorithm.

Methods such as [8] represent an efficient way to control the overall shape of a robot team by global parameters, where the positions of the individual robots in the formation are not explicitly controlled. Patterns can also be created by applying a potential or force field. A method based on generalized social potential fields is presented in [9] that directs the robots to the goal positions on collision-free but relatively jagged paths which might get trapped in local minima.

The work in [10] presents a solution to the coverage problem by partitioning the area with a Voronoi diagram. The shape of the Voronoi regions and thus the positions of the robots are controlled through a density function over the environment, which results in a limited set of simple uniform robot patterns.

Other decentralized approaches, such as the gradual pattern formation presented in [11], investigate pattern formation under the local information provided by the robots' sensors. Due to the lack of global knowledge, optimal choreographic motions cannot be achieved.

Furthermore, related work to pattern formation extends beyond the field of robotics to animation and computer graphics, where the motion coordination of large groups of agents is studied. The concept of spectral transformations is introduced in [12] to sequentially achieve smooth trajectories and collision avoidance. The approach produces visually appealing motions and the interpolation paths are controllable, but collision avoidance is not taken into account at the planning stage. Moreover, the method seems to require substantial manual tuning.

J. Alonso-Mora, A. Breitenmoser, M. Rufli and R. Siegwart are with the Autonomous Systems Lab, ETH Zurich, 8092 Zurich, Switzerland {jalonso,andrbrei,ruflim,rsiegwart}@ethz.ch

J. Alonso-Mora, P. Beardsley are with Disney Research Zurich, 8092 Zurich, Switzerland {jalonso,pab}@disneyresearch.com

978-1-61284-385-8/11/$26.00 ©2011 IEEE

The remainder of the paper is structured as follows. Section II states the problem and gives an overview of the presented approach to pattern formation. Section III describes the goal generation for representation of the final pattern, whereas Section IV presents the control. In Section V simulations and experiments are discussed. Finally, Section VI concludes and gives an outlook on our future work.

II. Problem definition and overview

Given m non-intersecting patterns $\{P_1, \ldots, P_m\}$ and n robots $\{R_1, \ldots, R_n\}$ of radius $\{r_1, \ldots, r_n\}$, two complementary problems arise: first, to obtain the best possible distribution of the n robots over the m patterns so that in their final positions the patterns are optimally represented; second, to guarantee visually appealing motions of the robots and fast convergence, both for holonomic and non-holonomic robots.

The solution here presented consists of two main steps, each one solving one of the aforementioned problems.

First, a set of n goal positions is computed. The goal positions represent the patterns and are independent of the initial positions of the robots. This initial computation is equivalent to a coverage method and can be computed offline if the pattern is known a priori. The result of this step is a set of $\sum_{l=1}^{m} n_l = n$ goal positions, where n_l goal positions represent pattern P_l.

Second, robots are driven towards the set of goals by a real-time controller. This iterative controller is subdivided into three parts: first the robots are optimally and uniquely assigned to goal positions, second each robot computes a preferred velocity towards its goal independently of the other robots, third each robot chooses a collision free velocity close to its preferred velocity, taking the current positions and velocities of its neighbors into account, and moves according to this computed velocity. Visual appeal in the trajectories is obtained thanks to the optimal goal assignment and reciprocal collision avoidance which avoids oscillations and produces smooth trajectories.

III. Goal generation

Given m patterns, the objective is to find the positions of n samples which optimally represent them. The samples are spread over the patterns according to their relative area. The ratios $a(P_1) : a(P_2) : \cdots : a(P_m)$ are kept close to $n_1 : n_2 : \cdots : n_m$, with $\sum_{l=1}^{m} n_l = n$ and $a(P_l)$ the area of pattern P_l. Therefore, the problem is reformulated as optimizing the positioning of a set of n_l particles representative for each pattern P_l. This is equivalent to a coverage problem, as stated in [10], where a cost function given by

$$\sum_{j=1}^{n_l} \int_{W_j^l} ||\mathbf{q} - \mathbf{q}_j^l|| \rho_l(\mathbf{q}) d\mathbf{q} \tag{1}$$

is to be minimized. The set $\{W_j^l\}_{j \in [1,n_l]}$ is a partition of the convex region $\hat{P}_l$ that entirely encloses P_l, $\mathbf{q}_j^l$ are the positions of the particles, i.e. the generators of the partition and $\rho_l(\mathbf{q})$ a mass density function which takes high values in P_l and decreases towards zero outside. Note that an enclosing convex region $\hat{P}_l$ is needed for computation of the partition, but the pattern itself P_l can be of arbitrary shape. The Lloyd algorithm presented in [10] guarantees convergence of the final particle configuration to a local minimum.

Fig. 1 demonstrates the process. The algorithm is initialized to n_l particles uniformly sampled within the pattern P_l. In each step the optimal partition given by the current samples is computed. This is the Voronoi tessellation,

$$V_j^l = \{\mathbf{q} \in \hat{P}_l \mid ||\mathbf{q} - \mathbf{q}_j^l|| \leq ||\mathbf{q} - \mathbf{q}_s^l||, \forall s \in [1, n_l], s \neq j\}. \tag{2}$$

The position of each particle $\mathbf{q}_j^l$ is updated to its optimal placement, the centroid of its cell,

$$C_{V_j^l} = \frac{\int_{V_j^l} \mathbf{q} \rho_l(\mathbf{q}) d\mathbf{q}}{\int_{V_j^l} \rho_l(\mathbf{q}) d\mathbf{q}}. \tag{3}$$

Due to the proximity between initial and final positions of the samples the iterative optimization quickly converges to a local minimum of the coverage function (1), which in general is a good representation of the pattern. Implementation of the Lloyd algorithm can be either distributed or centralized. For each pattern, the centralized optimization presents $\mathcal{O}(n_l \log n_l)$ overall time complexity of one step of the computation and a good distribution can be obtained in ten to twenty iterations if the start positions are sampled from a uniform distribution [13].

Let us call G_l the set of n_l particles that optimally represents pattern P_l and has been obtained through the presented Voronoi coverage optimization. Finally, the set of goals G is formed by the union $G = \bigcup_{l=1}^{m} G_l$. Fig. 1 shows the initial and final position of the particles for representing a triangle.

IV. Control

In order to achieve fast convergence and smooth trajectories, a simple real-time control is proposed. In each iteration, each agent R_i is uniquely assigned to a goal position $\mathbf{g}_j \in G$, computes a preferred velocity independently of the other agents and modifies it to avoid collisions with its neighbors.

A. Goal assignment

In each iteration the first step is to find a bijection between the n agents and the n goal positions which minimizes a cost

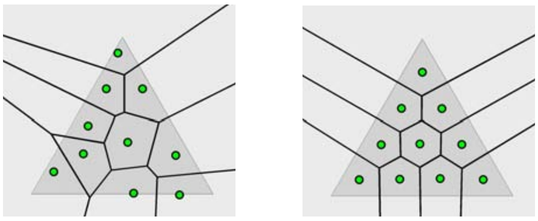

Fig. 1. Left: Initial samples for representing a triangle and Voronoi tessellation. Right: Final position of the samples after convergence and their Voronoi tessellation.

function, here given by the sum of squared distances to the goal

$$\sigma^k_{GA} = \operatorname*{argmin}_{\sigma^k} \sum_{i=1}^{n} ||\mathbf{g}_{\sigma^k(i)} - \mathbf{p}^k_i||^2, \tag{4}$$

where k is the iteration index, σ^k the assignment function, $\mathbf{p}^k_i$ the position of robot R_i at iteration k and $\mathbf{g}_{\sigma^k(i)}$ its assigned goal.

The centralized Hungarian algorithm for optimally solving the goal assignment problem with cost $O(n^4)$ was given by Kuhn and Munkres [3] and later improved by Edmonds and Karp to cost $O(n^3)$ [14]. In [16] a distributed task assignment method with market based coordination is presented, which finds a sub-optimal solution in a maximum of $n(n+1)/2$ iterations for each robot. In our present simulations and experiments a centralized version of the Hungarian algorithm is used, thus the optimal assignment is found. This computation becomes prohibitive for groups of robots larger than the ones used in the experiments and simulations. In that case, a sub-optimal goal assignment should be computed. Moreover, due to the low number of reassignments observed throughout iterations, this layer could run at a lower rate in a parallelized approach.

B. Preferred velocity

Each robot R_i selects a preferred velocity $\mathbf{v}^k_{pref_i}$ towards its assigned goal without taking into account the other robots. This velocity is given by a simple proportional controller

$$\mathbf{v}^k_{pref_i} = V_p \min\left(1, \frac{||\mathbf{g}_{\sigma^k_{GA}(i)} - \mathbf{p}^k_i||}{K_a}\right) \frac{\mathbf{g}_{\sigma^k_{GA}(i)} - \mathbf{p}^k_i}{||\mathbf{g}_{\sigma^k_{GA}(i)} - \mathbf{p}^k_i||}, \tag{5}$$

where the constant $V_p > 0$ is the preferred speed of the robot and is chosen slightly lower than the maximum speed. This is in order to successfully avoid collisions by speeding-up if needed. The constant $K_a > 0$ is the distance to the goal from which the preferred velocity is reduced linearly. In order to guarantee convergence without oscillations K_a must verify $K_a \geq V_p \Delta t$, where Δt is the time step of the controller. Note that the non-holonomicity of the robots is taken into account in the local collision avoidance.

C. Local collision avoidance

For each robot, given a preferred velocity $\mathbf{v}^k_{pref_i}$ and the current velocities and positions of its neighbors, a collision free velocity $\mathbf{v}^k_{cf_i}$ and appropriate controls are computed. In order to avoid collisions while guaranteeing visually appealing motions local reciprocal collision avoidance is implemented. The method is based on Velocity Obstacles [17] in velocity space and exploits the fact that all controlled robots in the environment react following the same scheme. For holonomic robots refer to [18]. The kinematic constraints of non-holonomic robots are taken into account in NH-ORCA [4], which is the chosen method here shortly described.

Assume a differential-drive robot with constraints in its linear and angular velocities given by $|v| \leq v_{max} - |\omega|\frac{l_w}{2}$ and $|\omega| \leq \omega_{max}$,where l_w is the inter-wheel distance and v_{max} and ω_{max} the maximum linear and angular speeds. Further consider a set of basic trajectories described by two segments, the first segment at v and ω constant and the second one starting at a fixed time $T_o \geq \Delta t$, at constant v and $\omega = 0$.

Given a holonomic robot with constant velocity $\mathbf{v}_H$, its trajectory can be tracked by a non-holonomic robot moving along a trajectory subject to the aforementioned constraints in speed and path, within a certain tracking error. The set S_{AHV_i} now defines the velocities that can be tracked by robot R_i within an error below a fixed upper limit $\epsilon_{max} \geq 0$, and P_{AHV_i} is its polygonal approximation. In [4] the closed form of S_{AHV_i} was presented.

The set of collision-free velocities $ORCA^{\tau}_i$ for robot R_i with horizon τ is given by

$$ORCA^{\tau}_i = P_{AHV_i} \cap \bigcap_{j=1, j\neq i}^{n} ORCA^{\tau}_{i|j}\,, \tag{6}$$

where $ORCA^{\tau}_{i|j}$ is the set of collision-free velocities for horizon τ, and a holonomic robot at position p^k_i and radius $r_i + \epsilon_{max}$ with respect to a holonomic robot at position p^k_j, radius $r_j + \epsilon_{max}$ and velocity $\mathbf{v}^{k-1}_{cf_j}$. As shown in [18], $ORCA^{\tau}_{i|j}$ is given by the half plane in velocity space computed geometrically from the generated Velocity Obstacle. Finally, $\mathbf{v}^k_{cf_i}$ is selected as

$$\mathbf{v}^k_{cf_i} = \underset{\mathbf{v}\in ORCA^{\tau}_i}{argmin} \, ||\mathbf{v} - \mathbf{v}^k_{pref_i}||. \tag{7}$$

An example with five robots is presented in Fig. 2.

Then feasible controls (v^k_i, ω^k_i) are chosen for robot R_i, which minimize the tracking error of $\mathbf{v}^k_{cf_i}$ while following the aforementioned basic trajectories. The mapping between $\mathbf{v}_{cf}$ and (v, ω) was given in closed form in [4]. Moreover, ϵ_{max} is dynamically decreased to guarantee $r_i + r_j + 2\epsilon_{max} \leq ||\mathbf{p}^k_i - \mathbf{p}^k_j||$. In addition, for crowded scenarios the time horizon can be optimized in each step by solving a 3D optimization [18], algorithm that runs in $\mathcal{O}(n_i)$ expected time for each robot R_i, where n_i is the number of neighboring robots.

The result is an algorithm that guarantees collision-free trajectories for non-holonomic robots even in highly crowded environments. The collision avoidance layer can be fully par-

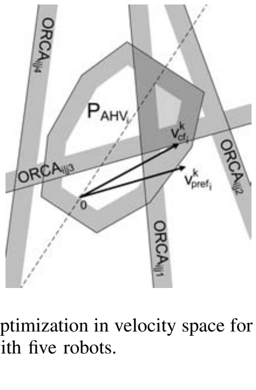

Fig. 2. NH-ORCA optimization in velocity space for a differentially driven robot in a scenario with five robots.

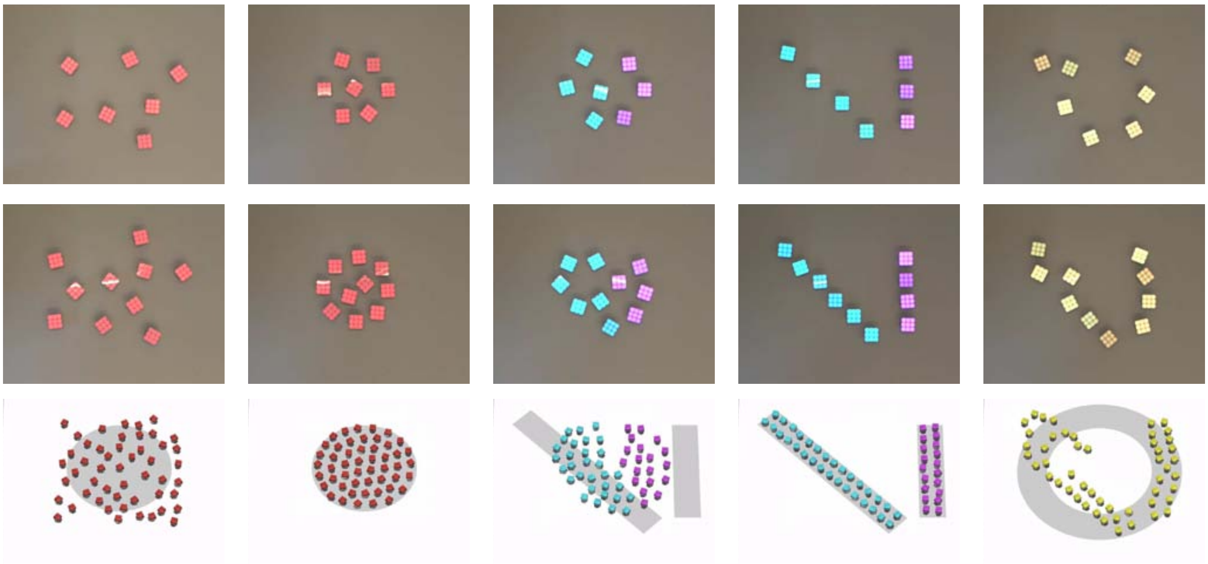

Fig. 3. Multi-robot pattern formation. Top row, left to right: seven robots starting from random positions transition to form sequentially a circle, two lines and a ring, which follows in Fig. 4; intermediate and final frames are displayed. Middle row: Identical transformation with ten robots. Bottom row: Identical transformations with fifty simulated robots. The target patterns are displayed in the background at the bottom row.

allelized and information exchange is limited to knowledge of size, position and velocity of the robots' neighbors.

V. Simulation and experiments

The pattern formation and control methods presented in this paper are tested both in experiments with real robots and in simulation. For the experiments a representative set of patterns is chosen, which is composed of polygonal convex and non-convex patterns, line patterns, filled patterns and patterns with hole, and formations with multiple patterns at a time. The chosen patterns are a circle, two lines, a ring, a triangle and three stars.

These patterns are visible as the background shapes in the bottom rows of Figs. 3 and 4. We show the representation of these patterns using four, seven and ten physical robots, and fifty simulated robots. We also show that trajectories present fast convergence, are smooth and exempt of undesired oscillations, thus they are visually pleasing.

A. Setup for pattern formation

The arena consists of a white 1.5 m x 1.5 m flat area on which the robots operate. An overhead camera tracks them through unique infrared LED codes. Communication to a centralized computing unit is realized via generic radio receivers at 10 Hz. In the following experiments we employ a maximum number of ten modified e-pucks [5], which are small disk-shaped differentially-driven robots. Furthermore, each robot is equipped with an array of 3 x 3 RGB color LEDs to portray a desired pattern. In Table I the relevant parameters of the modified e-puck are given, together with the NH-ORCA collision avoidance constants. Note that ϵ_{max} is decreased when robots are extremely close and horizon τ might be decreased to guarantee feasibility of the optimization, as described in Section IV. The simulations are carried out identically to the physical experiments, except that the tracking part is omitted and the vehicle kinematics are simulated by adding a similar amount of actuation noise as found in the real system. Note that noise is almost negligible in this setup.

B. Experimental results

Three runs of experiments are presented where the number of robots is increased; four, seven and ten robots are used. In all runs the robots are placed randomly on the arena and sequentially move from one pattern formation to the other. The intermediate and final frames for seven and ten robots are displayed starting in Fig. 3 and continuing in Fig. 4. In the top row of Fig. 3 and from left to right, seven robots start from random positions to form sequentially a circle, two lines and a ring, which follows in Fig. 4. In the middle

TABLE I
Parameters of an e-puck robot and NH-ORCA constants

Symbol	Value	Units	Description
l_w	0.0525	m	Distance between wheels
d_w	0.041	m	Diameter of the wheels
d_A	0.09	m	Maximum diameter of the robot
v_{max}	0.13	m/s	Maximum linear speed
ω_{max}	4.96	rad/s	Maximum angular speed
ϵ_{max}	0.01	m	Maximum tracking error
τ	2	s	Time to collision horizon
T_o	0.35	s	Time to achieve orientation
V_p	0.12	m/s	Preferred speed
K_a	0.1	m	Linear speed reduction distance

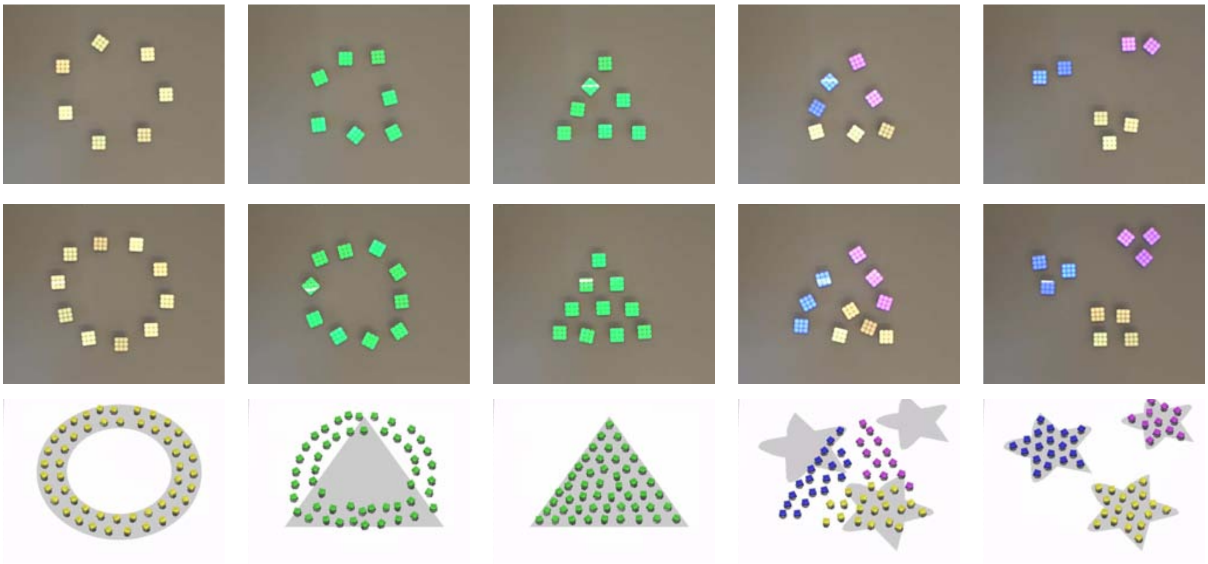

Fig. 4. Multi-robot pattern formation (cont'd from Fig. 3). Top row, left to right: seven robots after achieving a ring pattern transition to form sequentially a triangle and three stars; intermediate and final frames are displayed. Middle row: Identical transformation with ten robots. Bottom row: Identical transformations with fifty simulated robots. The target patterns are displayed in the background at the bottom row.

row identical transformations with ten robots are displayed. The target patterns are displayed in the background at the bottom row. The sequences continue in Fig. 4, where after achieving a ring pattern, the robots first form a triangle and then three stars. Each sequence, from the first to the last pattern, has a duration of approximately 20 seconds and shows that the robots display the pattern optimally in their final positions. Even for low numbers of robots the method is able to represent a target pattern closely, although the visual effect is clearly affected by the low resolution, as can be seen from Fig. 5 where only four robots represent the patterns.

In Fig. 6 the trajectories for several pattern transformations are depicted, where each line represents the path of a robot, with the light blue stars the start positions and the black circles the end positions. The presented transformations are ordered from left to right: ten robots from two lines to ring, fifty simulated robots from two lines to ring and from ring to triangle. Smooth and oscillation-free trajectories are obtained. Collisions are fully avoided while keeping elegant motions. A video that shows the conducted experiments in full length and real speed accompanies the paper.

C. Simulation results

In simulation the number of robots is increased to fifty and the same sequence of patterns described in Section V-B is simulated. The intermediate and final frames are displayed starting in the bottom row of Fig. 3 and continuing in the bottom row of Fig. 4. The target patterns are displayed in the background of the images. As expected, increasing the number of robots improves the quality of the representation. This sequence is included in the accompanying video in full length and at double speed.

D. Recovery from perturbations

Thanks to the goal assignment performed in each iteration, the system adapts to major perturbations in the position of the robots by redistributing them to reform the pattern quickly. In Fig. 7 the recovery from a perturbation (top-right) is shown. Note that initial (top-left) and final (bottom-right) distributions are identical, although the robots have been redistributed. This experiment is presented in full length and real speed in the accompanying video.

VI. Conclusion and future work

In this work a multi-robot control method for pattern formation is presented and tested in experiments and simulation. First, optimal final formations are achieved independently of the number of robots. Nevertheless, the minimum number of robots needed for obtaining a visually pleasant representation depends on the given pattern and remains difficult to specify in general. To obtain the final positions the Voronoi coverage method is applied before the robots start moving. This computation can be distributed. Second, visually appealing trajectories (oscillation-free, smooth and of similar length for all robots) are obtained with fast convergence to the final formation from any start conditions. To achieve this, a real-time controller is implemented. The goal assignment step is implemented in a centralized scheme, but existing distributed algorithms could be utilized instead; the local collision avoidance is inherently distributed. Fast computation of the goal positions and fast convergence to the final pattern was achieved in all of our experiments, where no failure was detected. Furthermore, motions are guaranteed to be collision-free for the case of multiple differentially-driven robots.

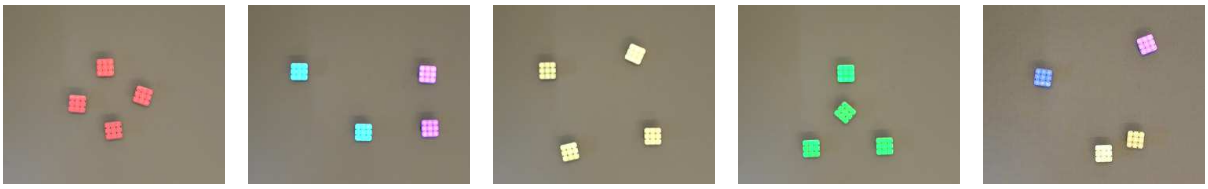

Fig. 5. Five patterns represented with only four robots. Left to right: circle, two lines, ring, triangle, three stars. Even with very low resolution patterns are noticeable

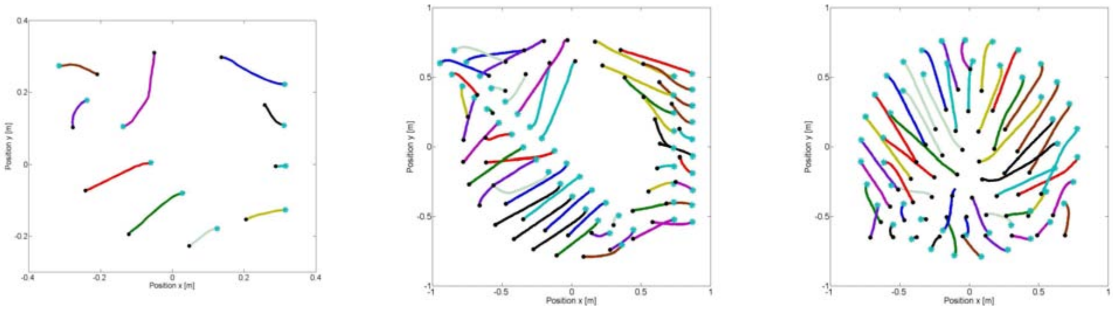

Fig. 6. Left: Trajectories for ten robots from two lines to ring. Middle: Trajectories for fifty robots (simulation) from two lines to ring. Right: Trajectories for fifty robots (simulation) from ring to triangle. Start and final positions are represented with light blue stars and black circles respectively.

The system in this paper provides a platform for several future investigations, including the handling of static and dynamic obstacles, as well as dynamic patterns.

References

[1] G. Antonelli, F. Arrichiello and S. Chiaverini, ”The entrapment/escorting mission: An experimental study using a multirobot system”, in *IEEE Robotics and Automation Magazine*, vol. 15, no. 1, pp. 22-29, 2008.

[2] B. Varghese and G. McKee, ”Towards a Unifying Framework for Pattern Transformation in Swarm Systems”, in *Proc. of the AIP Conference*, vol. 1107, no. 1, pp. 65–70, 2009.

[3] H. W. Kuhn, ”The hungarian method for the assignment problem”, in *Naval Research Logistics*, vol. 2, no. 1, pp. 83–97, 1955.

[4] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, R. Siegwart, ”Optimal Reciprocal Collision Avoidance for Multiple Non-Holonomic Robots”, in *Proc. Int. Symp. on Distributed Autonomous Robotics Systems*, 2010.

[5] F. Mondada, M. Bonani, X. Raemy, J. Pugh,C. Cianci, A. Klaptocz, S. Magnenat, J.-C. Zufferey, D. Floreano and A. Martinoli, ”The e-puck, a Robot Designed for Education in Engineering”, in*Conf. Aut. Rob. Syst. Compet.*, pp. 59–65, 2009.

[6] S. Yun, D.A. Hjelle, E. Schweikardt, H. Lipson and D. Rus, ”Planning the Reconfiguration of Grounded Truss Structures with Truss Climbing Robots that Carry Truss Elements”, in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2009.

[7] R. Ravichandran, G. Gordon and S. C. Goldstein. ”A scalable distributed algorithm for shape transformation in multi-robot systems”, in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, 2007.

[8] C. Belta and V. Kumar, ”Abstraction and control for groups of robots”, in *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 865-875, 2004.

[9] R. Gayle, W. Moss, M. C. Lin and D Manocha, ”Multi-robot coordination using generalized social potential fields”, in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2009.

[10] F. Bullo, J. Cortés and S. Martínez, ”Distributed Control of Robotic Networks”, *Princeton University Press*, 2009.

[11] Y. Ikemoto, Y. Hasegawa, T. Fukuda and K. Matsuda, ”Gradual spatial pattern formation of homogeneous robot group”, in *Inf. Sci. Inf. Comput. Sci.*, vol. 171, no. 4, pp. 431-445, 2005.

[12] S. Takahashi, K. Yoshida, T. Kwon, K. Hoon Lee, J. Lee and S. Y. Shin, ”Spectral-based group formation control”, in *Computer Graphics Forum*, 2009.

[13] O. Deussen, S. Hiller, C. van Overveld and T. Strothotte, ”Floating Points: A Method for Computing Stipple Drawings”, in *Computer Graphics Forum*, vol. 19, pp. 40–51, 2000.

[14] J. Edmonds and R. M. Karp, ”Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems”, in *Journal of the ACM*, vol. 19, no. 2, pp. 248–264, 1972.

[15] L. Liu and D. A. Shell, ”Assessing Optimal Assignment under Uncertainty: An Interval-based Algorithm”, in *Proc. of Robotics: Science and Systems*, 2010.

[16] N. Michael, M. M. Zavlanos, V. Kumar and G. J. Pappas, ”Distributed multi-robot task assignment and formation control”, in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 128–133, 2008.

[17] P. Fiorini and Z. Shiller, ”Motion planning in dynamic environments using Velocity Obstacles”, in *Int. Journal of Robotics Research*, vol. 17, no. 7, pp. 760-772, 1998.

[18] J. van den Berg, S. J. Guy, M. Lin and D. Manocha, ”Reciprocal n-body Collision Avoidance”, in *Int. Symp. on Robotics Research*, 2009.

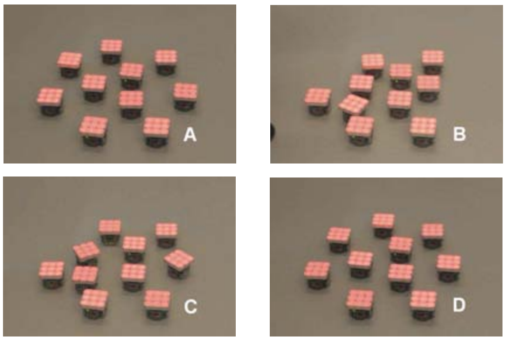

Fig. 7. Perturbation recovery with ten robots displaying a circle. Between frames A and B the positions of some robots are manually modified.