Finding Formations for the Non-prehensile Object Transportation with Differentially-Driven Mobile Robots

Henrik Ebel(✉), Daniel Niklas Fahse, Mario Rosenfelder, and Peter Eberhard

Institute of Engineering and Computational Mechanics, University of Stuttgart, Pfaffenwaldring 9, 70569 Stuttgart, Germany
henrik.ebel@itm.uni-stuttgart.de

Abstract. This paper proposes a formation-finding procedure for the non-prehensile transportation of arbitrarily-shaped polygonal objects with differentially-driven mobile robots. The proposed procedure is conceptually novel by taking a multibody system dynamics perspective, explicitly taking into account the robots' non-holonomic constraints. Being based on proper first principles such as Jourdain's principle, if the modeling assumptions are met, the approach nominally guarantees to only produce formations that are actually useful to manipulate the object in a given dynamic situation. As a byproduct, the scheme can directly provide a set of robot propulsion forces fitting the desired object motion. Simulation results confirm the chosen formations' favorable qualities.

1 Introduction and Transportation Approach

Cooperative object manipulation is a popular model problem in distributed robotics research [11] since it is a theoretically very challenging task while intuitively demonstrating the benefits of cooperation. Moreover, schemes devised in laboratories may be developed further to be applicable in logistics applications or even in disaster relief scenarios, e.g., to clear paths of debris. Over the years, researchers have studied many variants of cooperative transportation and a wide variety of solution approaches. This paper focuses on non-prehensile transportation, in which the robots can push but not pull the transported object, without gripping it in any way, making it possible to use simple, cost effective robot designs without complicated gripping mechanisms.

For these reasons, non-prehensile, pushing-based transportation has received continued and ample interest in research. However, it seems that most proposed methods are restricted to very simple object shapes, often enough to rectangular boxes [6–8,12,14]. Indeed, in the rather encompassing overview given in [11], not

This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Grant 433183605 and through Germany's Excellence Strategy (Project PN4-4 Theoretical Guarantees for Predictive Control in Adaptive Multi-Agent Scenarios) under Grant EXC 2075-390740016.

A. Kecskeméthy and V. Parenti-Castelli (Eds.): ROMANSY 2022, CISM 606, pp. 163–170, 2022.
https://doi.org/10.1007/978-3-031-06409-8_17

a single pushing-based, non-prehensile scheme appears that can deal with object shapes as general as this contribution. Interestingly, in the very recent work [1], the authors also venture to cooperatively push objects with robots of the same type as in this study. However, they neglect the employed robots' non-holonomic constraints when evaluating the usefulness of a formation for object manipulation and leave their inclusion to future work. In contrast, this contribution is built around the explicit and proper consideration of the non-holonomic constraints of differentially-driven mobile robots.

In our previous work, upon which this contribution builds, we developed a comprehensive distributed transportation scheme which lets omnidirectional mobile robots organize self-reliantly to transport arbitrarily shaped rigid objects, without any centralized decision making instance [2,4,5]. However, compared to omnidirectional robots, differentially-driven ones are more prevalent in commercial applications due to their simpler, more cost effective designs using, e.g., fewer drives. For this reason, this paper aims to enable the extension of our previous formation-building approach to differentially-driven mobile robots.

Among other hardware-validated, pushing-based schemes, a key differentiator of our transportation scheme is its unique versatility since it self-reliantly adapts to arbitrary object shapes, paths to follow, and numbers of robots, automatically reorganizing whenever necessary and without needing to retune or relearn controllers or behavior policies, see for instance the digital supplementary material from [4], which gives a glimpse of the versatility in video form. To achieve this, the transportation task is decomposed into two main subtasks. Firstly, formations around the edges of the object are devised so that they are useful to translate and rotate the object as currently necessary. Then, distributed formation control is utilized to translate and rotate the formation and thereby the object as desired. Both are dealt with by schemes based on distributed optimization. Concretely, distributed model predictive control is employed for formation control [3], and also the requirements for a useful formation are cast into an optimization problem that is solved with a custom distributed augmented Lagrangian particle swarm optimization algorithm [2,4].

For distributed control, we have already described how to accurately control a formation of differentially-driven mobile robots in [9]. This leaves as novel contribution here synthesizing formations as the remaining, major subtask to be evolved in the light of the non-holonomic constraint of differentially-driven robots. Therefore, in the interest of a concise paper, the subsequent disquisitions focus solely on formation synthesis.

To that end, in Sect. 2, the precise problem setup is introduced. Section 3 forms the core of the paper, i.e., characterizing and finding formations useful for transportation. The resulting optimization problem can be solved with the same distributed optimization algorithm used in our previous work [4]. Finally, in Sect. 4, simulations show that solutions of the formation synthesis problem indeed correspond to formations for which the robots can manipulate the object as necessary.

2 Assumptions

The differentially-driven robots used for transportation shall have a circular footprint of radius r_{g}. It is assumed that the motion of the ith robot can be described with sufficient accuracy by the kinematics $\dot{x}_i = \cos(\theta_i)v_i$, $\dot{y}_i = \sin(\theta_i)v_i$, $\dot{\theta}_i = \omega_i$, where ${}_i\boldsymbol{x}^{\mathsf{T}} := \begin{bmatrix} x_i \; y_i \; \theta_i \end{bmatrix}$ is the robot's pose, ${}_i\boldsymbol{r}^{\mathsf{T}} := \begin{bmatrix} x_i \; y_i \end{bmatrix}$ is its position in the inertial frame of reference, θ_i is the robot's orientation measured as an angle relative to a fixed reference, and v_i and ω_i are its translational and angular velocities. Moreover, the transported object shall be rigid, not subject to any kinematic constraint, and its shape shall be describable with sufficient accuracy by polygons.

It is assumed that $N \geq 2$ robots cooperate. Furthermore, the robots shall only rely on forces orthogonal to the object surface for transportation since tangential friction forces are usually hard to predict and thus may not reliably contribute to the transportation. It is assumed that robots and object move in the plane. Subsequently, without special notation, all vectors denoting spatial quantities shall be given in the coordinates of and relative to the inertial frame of reference $\mathcal{K}_{\mathrm{I}}$. However, the notation ${}^{\alpha}\boldsymbol{z}$ subsequently denotes the quantity $\boldsymbol{z}$ expressed in the coordinates of the coordinate frame $\mathcal{K}_{\alpha}$. Similarly, ${}^{\check{\alpha}}\boldsymbol{z}$ denotes the quantity not only expressed in the coordinates of $\mathcal{K}_{\alpha}$ but also relative to its origin, which may be moving. If not giving rise to ambiguities, time dependencies are largely omitted notation-wise in the interest of a compact representation of formulae. Matrices are written as bold uppercase symbols, vectors as bold lowercase symbols, and scalar quantities in regular font.

3 Formation Synthesis

The system consisting of the transported object and N robots is understood as a planar multibody system with the bodies being subject to non-holonomic as well as holonomic constraints. This includes unilateral constraints since the robots do not grip the object. The configuration of the multibody system can be fully described by means of the vector $\boldsymbol{z}^{\mathsf{T}} := \begin{bmatrix} {}_{\mathrm{O}}\boldsymbol{x}^{\mathsf{T}} \; {}_{1}\boldsymbol{x}^{\mathsf{T}} \cdots {}_{\mathrm{N}}\boldsymbol{x}^{\mathsf{T}} \end{bmatrix}$ with ${}_{\mathrm{O}}\boldsymbol{x} \in \mathbb{R}^3$ describing the object pose. The goal is to formulate an optimization problem whose feasible points correspond to formations that are useful to accelerate the object into the direction currently required. Hence, the primary task is to formulate a set of constraints that encapsulates the requirements for a fitting formation. In other words, given a desired motion of the object and a candidate formation, a procedure needs to be found that allows to decide whether the candidate formation can make the object move as desired.

As a first observation, when pushing the object, the robots' centers need to have a distance of robot radius r_{g} from the object edge(s) that they are in contact with. Thus, the robot positions useful for transportation lie on the edges of an object dilated by the robot radius. These edges are parametrized by the function ${}^{\check{S}}\boldsymbol{E} : \mathcal{I}_{\mathrm{e}} \to \mathbb{R}^2$ for an interval $\mathcal{I}_{\mathrm{e}}$. The edges of the dilated object are described in the object-fixed coordinate frame $\mathcal{K}_{\mathrm{S}}$ whose origin is

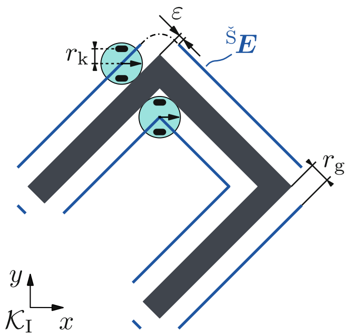

Fig. 1. Non-convex object with two differentially-driven robots.

fixed to the object's center of mass. Pushing the object at pointed corners is not advantageous since the contact normal directions change quickly at these corners. Hence, in the edge parameterization, areas around pointed corners are clipped by the safety distance ε, see Fig. 1. Formation synthesis can then be understood as finding, along the parameterized edges, N one-dimensional positions $w_i \in \mathcal{I}_\mathrm{e}$, $i \in \{1, \ldots, N\}$, $\boldsymbol{w}^\mathsf{T} := \begin{bmatrix} w_1 \cdots w_N \end{bmatrix}$ that are useful to manipulate the object. Subsequently, it is derived what the latter actually means in a mathematical manner.

Since the object shape can be non-convex, a formation may have $n_\mathrm{c}(\boldsymbol{w}) \geq N$ contact points with the object. The geometric situation of a formation around an object is fully defined by the vector $\boldsymbol{w}$, although we will mostly not explicitly write geometric relationships in terms of $\boldsymbol{w}$ for conciseness of presentation. Importantly, for a given configuration $\boldsymbol{w}$ and a given desired motion of the object, we prescribe specific orientations θ_i of the robots that are chosen so that the robots' centers and the object together perform one common motion, i.e., so that the motion of the robots' centers and the object can be described as a pure rotation about the same instant center of rotation while respecting the robots' kinematic constraints. Thus, also the robot orientations can be expressed as a function of $\boldsymbol{w}$ and the desired object motion. For the jth contact point of robot i, the outward-pointing contact normal shall subsequently be denoted by ${}^{S}_{i}\boldsymbol{n}_j$, the robot's position in the formation by ${}^{\check{S}}_{i}\boldsymbol{r} = {}^{\check{S}}\boldsymbol{E}(w_i)$, its orientation by θ_i, and the position of the corresponding contact point on the object by ${}^{\check{S}}\boldsymbol{c}_j$. Postulating that robot i shall not lose contact with the object leads to the contact constraint $g_{\mathrm{c},i,j} = {}^{S}_{i}\boldsymbol{n}_j^\mathsf{T}\left({}^{\check{S}}_{i}\boldsymbol{r} - {}^{\check{S}}\boldsymbol{c}_j\right) - r_\mathrm{g} = 0$. One needs to take care that enforcing this constraint does not require pulling forces, which cannot be exerted by the robots. We will exclude pulling forces by other means later on, otherwise one may instead enforce $g_{\mathrm{c},i,j} \geq 0$. Apart from constraints arising from robot-object contacts, the robots themselves are constrained in a non-holonomic fashion in the form $g_{\mathrm{r},i} = \begin{bmatrix} -\sin(\theta_i) \ \cos(\theta_i) \ 0 \end{bmatrix} {}_{i}\dot{\boldsymbol{x}} = 0$. We will later prescribe a desired acceleration of the object. Hence, to move to acceleration-level, we take the total time derivative of the non-holonomic constraint and can then write it in the form $\boldsymbol{G}_{\mathrm{r},i}(\boldsymbol{z})\,\ddot{\boldsymbol{z}} + \bar{\bar{\gamma}}_{\mathrm{r},i}(\boldsymbol{z}, \dot{\boldsymbol{z}}) = 0$, where $\bar{\bar{\gamma}}_{\mathrm{r},i} = \frac{\mathrm{d}\boldsymbol{G}_{\mathrm{r},i}(\boldsymbol{z})}{\mathrm{d}t}\dot{\boldsymbol{z}}$ and $\boldsymbol{G}_{\mathrm{r},i} \in \mathbb{R}^{1\times 3(N+1)}$. These constraints are stacked into one equation system for later usage, yielding $\boldsymbol{G}_{\mathrm{r}}(\boldsymbol{z})\,\ddot{\boldsymbol{z}} + \bar{\bar{\boldsymbol{\gamma}}}_{\mathrm{r}}(\boldsymbol{z}, \dot{\boldsymbol{z}}) = \boldsymbol{0}$. In the same way, but differentiating twice, also the contact constraints can be brought to an analogous form. Then, stacking all constraints, for each robot and each of its contact points, leads to the large constraint equation system

$$\boldsymbol{G}(\boldsymbol{z})\,\ddot{\boldsymbol{z}} + \bar{\bar{\boldsymbol{\gamma}}}(\boldsymbol{z}, \dot{\boldsymbol{z}}) = \boldsymbol{0} \tag{1}$$

with $\boldsymbol{G} \in \mathbb{R}^{(N+n_c)\times 3(N+1)}$, $\bar{\bar{\boldsymbol{\gamma}}} \in \mathbb{R}^{(N+n_c)}$. Isolating each robot and the object and neglecting friction beyond the necessary friction forces between the robots' wheels and the ground, the Newton and Euler equations for the whole multibody system can be written in the form $\boldsymbol{M}\ddot{\boldsymbol{z}} = \boldsymbol{f}^{\mathrm{a}} + \boldsymbol{f}^{\mathrm{r}}$ with the mass matrix $\boldsymbol{M}$ containing the masses and moments of inertia of each robot and the object, and $\boldsymbol{f}^{\mathrm{a}}$ and $\boldsymbol{f}^{\mathrm{r}}$ containing the applied and the reaction forces and moments, respectively. It is possible to directly include a model for friction forces acting on the object into the Newton and Euler equations. Indeed, in the simulation scenarios of this paper's application, the simulated objects will be subject to friction. However, usually, it is unrealistic to assume that the robots have precise knowledge of the object's friction properties. Hence, for the purposes of this study, we assume that the robots are not aware of the friction properties of the object and, thus, do not consider them in finding formations. Later, the simulation results, which consider friction, will show that this does not inhibit the robots' capability to find useful formations.

As an implication of Jourdain's principle, as shown, e.g., by Woernle [13, Sect. 5.3.4], there exists a vector of Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^{N+n_{\mathrm{c}}}$ such that the reactions can be expressed in the form $\boldsymbol{f}^{\mathrm{r}} = \boldsymbol{G}^{\mathsf{T}}\boldsymbol{\lambda}$, yielding

$$\boldsymbol{f}^{\mathrm{a}} + \boldsymbol{G}^{\mathsf{T}}\boldsymbol{\lambda} = \boldsymbol{M}\ddot{\boldsymbol{z}} \tag{2}$$

when inserted into the Newton and Euler equations. The vector $\boldsymbol{\lambda}_{\mathrm{c}} \in \mathbb{R}^{n_{\mathrm{c}}}$ shall contain those entries of $\boldsymbol{\lambda}$ that correspond to the contact constraints. Through the definition of the contact constraints, positive entries of $\boldsymbol{\lambda}_{\mathrm{c}}$ correspond to the robots pushing the object, whereas negative ones correspond to pulling. Thus, later on, we will enforce $\boldsymbol{\lambda}_{\mathrm{c}} \geq \boldsymbol{0}$. With what has been found by now, if desired accelerations $\ddot{\boldsymbol{z}}_{\mathrm{d}}^{\mathsf{T}} = \left[{}_{\mathrm{o}}\ddot{\boldsymbol{x}}_{\mathrm{d}}^{\mathsf{T}}\ {}_{1}\ddot{\boldsymbol{x}}_{\mathrm{d}}^{\mathsf{T}} \cdots {}_{\mathrm{N}}\ddot{\boldsymbol{x}}_{\mathrm{d}}^{\mathsf{T}}\right]$ for the object and each robot were given, it would be possible to calculate applied forces and fitting Lagrange multipliers that lead to the desired acceleration. Yet, in practice, only the desired motion (and acceleration ${}_{\mathrm{o}}\ddot{\boldsymbol{x}}_{\mathrm{d}}$) of the object will be given. However, the accelerations of robots and object need to jointly satisfy the constraint equation system (1), which, for a given desired object acceleration, can be solved to obtain fitting robot accelerations. There may be infinitely many solutions for the latter, in which case we take the minimum Euclidean norm solution, obtained by using the Moore-Penrose inverse of the submatrix of $\boldsymbol{G}$ consisting of the columns corresponding to robot accelerations. If no solution exists, the candidate formation can be discarded. Finally, given the desired acceleration $\ddot{\boldsymbol{z}}_{\mathrm{d}}$ obtained that way, we solve for the applied forces and Lagrange multipliers by solving the quadratic program

$$\begin{aligned}
\underset{\boldsymbol{f}^{\mathrm{a}},\, \boldsymbol{\lambda}}{\text{minimize}} \quad & \boldsymbol{f}^{\mathrm{a}\mathsf{T}} \boldsymbol{f}^{\mathrm{a}} && \text{(3a)}\\
\text{subject to} \quad & \boldsymbol{f}^{\mathrm{a}} + \boldsymbol{G}^{\mathsf{T}} \boldsymbol{\lambda} = \boldsymbol{M} \ddot{\boldsymbol{z}}_{\mathrm{d}}, && \text{(3b)}\\
& f_1^{\mathrm{a}} = f_2^{\mathrm{a}} = f_3^{\mathrm{a}} = 0, && \text{(3c)}\\
& \boldsymbol{G}_{\mathrm{r}} \boldsymbol{f}^{\mathrm{a}} = \boldsymbol{0}, && \text{(3d)}\\
& \boldsymbol{\lambda}_{\mathrm{c}} \geq \boldsymbol{0}. && \text{(3e)}
\end{aligned}$$

Therein, constraint (3b) stems from (2), constraint (3c) accounts for the fact that only the robots, not the object, are directly actuated, constraint (3d) ensures that the robots, by means of their motors, can only apply forces into their free movement direction, and constraint (3e) ensures that the robots only exert pushing forces. If this optimization problem has a feasible solution, the candidate formation is useful to manipulate the object.

The above disquisitions describe a procedure to check whether a candidate formation is useful, but not yet a procedure to obtain a useful formation. Hence, we let an augmented Lagrangian particle swarm optimization algorithm search for feasible formations [10]. In the optimization problem solved to do so, the above procedure is used in the constraints to ascertain that the formations considered are kinetically and kinematically useful for manipulation. Further constraints ensure that the picked robot positions have enough distance between them to prevent collisions, that the reaction forces corresponding to nonholonomic constraints stay below a bound preventing the robots from slipping sideways, and that the requested applied forces do not exceed the robot motors' capabilities. The cost function prefers formations needing smaller applied forces to achieve the desired motion while, following ideas from [4], trying to also prefer formations which are expected to less frequently require reorganizations if the transportation direction changes.

4 Simulation Results

The simulations in this section are designed to study mainly the formation finding procedure. In the simulations, reorganizations happen instantly. That way, the performance observed does not depend on the time needed by the robots to drive around to reorganize but mostly on the formations picked. The robots reorganize if the current formation does not meet the requirements of a formation useful in the current situation anymore, i.e., if it violates the constraints previously formulated. The desired object path is followed with a distributed model predictive formation controller. The custom simulation environment employed has already proven itself in our previous studies with omnidirectional robots, where the transportation scheme carried over its favorable properties from simulations to hardware experiments [2]. Figure 2 exemplarily depicts three results, with the reference path for the object's center of mass dashed in red.

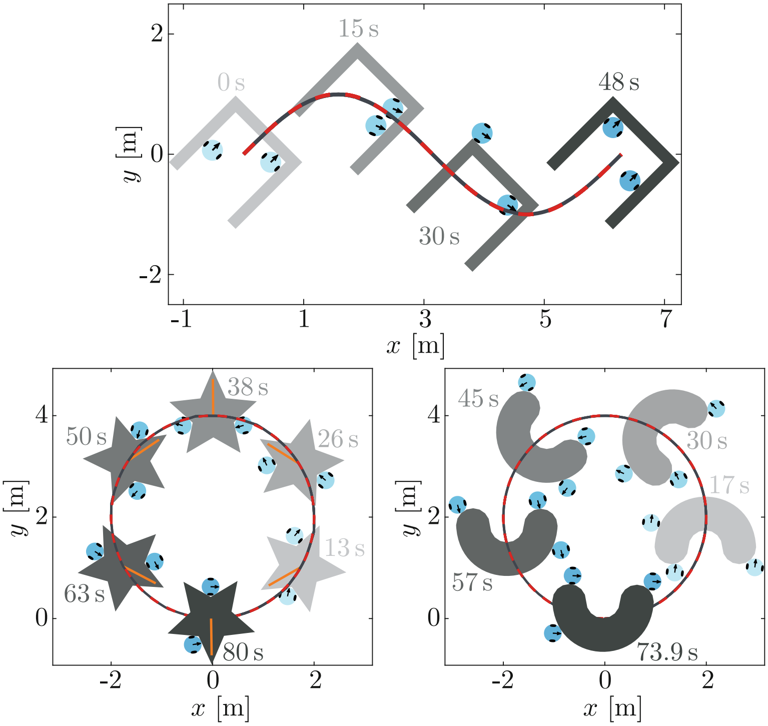

Fig. 2. Three simulations results employing differentially-driven robots.

The U-shaped object shall not be rotated. For the C-shaped object, the orientation shall be constant in the lower half of the circle, whereas the object shall be rotated by 180° in the upper half. The star-shaped object shall be rotated to complete one full rotation. All rotations shall happen at a constant rate. From top to bottom, left to right, the robots reorganize 6, 2, and 10 times before reaching the goal. In all studied scenarios, the scheme is always able to find useful formations, resulting in an accurate tracking performance.

5 Outlook

Due to the promising results of the formation finding scheme, we are currently preparing hardware experiments to test all facets of the transportation scheme in practical transportation tasks. This includes in particular effects caused by realistic friction in hardware experiments since friction is notoriously hard to model in a priori simulations. We hope to obtain a similarly favorable performance as we previously achieved for omnidirectional mobile robots. Furthermore, the approach described in this paper should extend well to more general use cases, e.g., to non-rigid objects.

References

1. Bertoncelli, F., Ruggiero, F., Sabattini, L.: Characterization of grasp configurations for multi-robot object pushing. In: Proceedings of the 2021 International Symposium on Multi-Robot and Multi-Agent Systems (MRS), Cambridge, pp. 38–46 (2021)
2. Ebel, H.: Distributed control and organization of communicating mobile robots: design, simulation, and experimentation. Dissertation, Schriften aus dem Institut für Technische und Numerische Mechanik der Universität Stuttgart, vol. 69. Shaker Verlag, Düren (2021). https://doi.org/10.2370/9783844081725
3. Ebel, H., Eberhard, P.: A comparative look at two formation control approaches based on optimization and algebraic graph theory. Robot. Autonom. Syst. **136**, 103686 (2021)
4. Ebel, H., Eberhard, P.: Non-prehensile cooperative object transportation with omnidirectional mobile robots: organization, control, simulation, and experimentation. In: Proceedings of the 2021 International Symposium on Multi-robot and Multi-agent Systems, Cambridge, pp. 1–10 (2021)
5. Ebel, H., Eberhard, P.: Cooperative transportation: realizing the promises of robotic networks using a tailored software/hardware architecture. at - Automatisierungstechnik. Accepted for publication (2022)
6. Kube, C.R., Zhang, H.: The use of perceptual cues in multi-robot box-pushing. In: Proceedings of the 1996 IEEE International Conference on Robotics and Automation, Minneapolis, pp. 2085–2090 (1996)
7. Matarić, M.J., Nilsson, M., Simsarian, K.T.: Cooperative multi-robot box-pushing. In: Proceedings of the 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems, Human Robot Interaction and Cooperative Robots, Pittsburgh, pp. 556–561 (1995)
8. Rahimi, M., Gibb, S., Shen, Y., La, H.M.: A comparison of various approaches to reinforcement learning algorithms for multi-robot box pushing. In: Proceedings of the International Conference on Engineering Research and Applications, Thái Nguyên, pp. 16–30 (2019)
9. Rosenfelder, M., Ebel, H., Eberhard, P.: Cooperative distributed model predictive formation control of non-holonomic robotic agents. In: Proceedings of the 2021 International Symposium on Multi-Robot and Multi-Agent Systems, Cambridge, pp. 11–19 (2021)
10. Sedlaczek, K., Eberhard, P.: Using augmented Lagrangian particle swarm optimization for constrained problems in engineering. Struct. Multidiscip. Optimiz. **32**(4), 277–286 (2006)
11. Tuci, E., Alkilabi, M.H.M., Akanyeti, O.: Cooperative object transport in multi-robot systems: a review of the state-of-the-art. Front. Robot. AI **5** (2018)
12. Wang, Y., de Silva, C.W.: Multi-robot box-pushing: single-agent Q-learning vs. team Q-learning. In: Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, pp. 3694–3699 (2006)
13. Woernle, C.: Mehrkörpersysteme. Springer Vieweg, Berlin (2016). https://doi.org/10.1007/978-3-662-46687-2
14. Yamada, S., Saito, J.: Adaptive action selection without explicit communication for multirobot box-pushing. IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) **31**(3), 398–404 (2001)