First-Order Synergies for Motion Planning of Anthropomorphic Dual-Arm Robots ⋆

Néstor García * **Raúl Suárez** * **Jan Rosell** *

* *Institute of Industrial and Control Engineering (IOC), Universitat Politècnica de Catalunya (UPC), Barcelona, Spain (e-mails: {nestor.garcia.hidalgo, raul.suarez, jan.rosell}@upc.edu).*

Abstract: This paper addresses the problem of designing a planning algorithm for anthropomorphic dual-arm robotic systems to find paths that mimics the movements of real human beings by using first-order synergies (correlations between joint velocities). The key idea of the proposal is to convert captured human movements into a vector field of velocities, defined in the configuration space of the robot, and use it to guide the search of a solution path. The motion planning is solved using the proposed algorithm, called FOS-BKPIECE, that is a bidirectional version of the KPIECE planner working with an improved version of the extension procedure of the VF-RRT planner. The obtained robot movements follow the directions of the defined vector field and hence allow the robot to solve the task in a human-like fashion. The paper presents a description of the proposed approach as well as results from conceptual and application examples, the latter using a real anthropomorphic dual-arm robotic system. A thorough comparison with other previous planning algorithms shows that the proposed approach obtains better results.

Keywords: Robotics, Path Planning, Dual-Arm Robots, Synergies, Human-Like Motions.

1. INTRODUCTION

Motion planning is nowadays a quite researched issue in robotics, even more since the robots became a vital part of many application fields (e.g. the electronic and medical industries, or the computational biology and computer animation). The importance of this problem is manifested when the motion planning of mechanical hands or anthropomorphic dual-arm systems is attempted, i.e. systems involving a high number of degrees of freedom (DOF). In addition to this, sometimes not only a valid path is required but also the one that optimizes some path quality metric (e.g. minimizing the path length or the execution time). This is a typical problem in the humanoid robotics, where the motion planning should not only focus on the efficient search of a valid solution, but also on the search of robot movements that mimic the movements of the human beings. Pursuing this goal, the human-robot collaboration is facilitated because the humans can adjust their motions to avoid possible injuries or enhance the collaboration since they are familiar with the robot motions (Fukuda et al., 2001).

The motion planning of complex systems have been addressed with different planning algorithms, being the sampling-based planners the most commonly used (Elbanhawi and Simic, 2014). Among them, the Probabilistic Roadmap planners, PRM (Kavraki et al., 1996), or the Rapidly-exploring Random Trees, RRT (Kuffner and LaValle, 2000), are the most outstanding. However, these algorithms are non-optimal. To find an optimal solution, some variants like the PRM* and RRT* algorithms have been proposed (Karaman and Frazzoli, 2011).

In order to obtain human-like movements, the right coordination between the robot joints is crucial and therefore the real movements of a human being are commonly used as a reference (Argall et al., 2009). Relevant pioneering works dealt with the grasping problem analyzing the correlations of the finger joints when the human hand was grasping objects (Santello et al., 2002). These correlations were called *hand postural synergies* and mapped into a mechanical hand (Ciocarlie and Allen, 2009). The synergies existing in the human hand were also used for other objectives such as the analysis and design of robotic hands in order to mimic human grasps (Ficuciello et al., 2014), the design of specific hand control systems (Wimböck et al., 2011), or the identification of the hand pose using low-cost gloves (Bianchi et al., 2013). Nevertheless, there exist other approaches that instead of studying the hand synergies while grasping an object, they compute them from hand movements when the human tries to cover the whole hand configuration space in an unconstrained way (Sun et al., 2010). These synergies can be used then to simplify the motion-planning problem by reducing the dimension of the search space as well as to mimic human postures (Rosell et al., 2011). More recently, a compliant model, called *soft synergies*, was also introduced and used in the selection of grasping forces, in their control, and in the control of the motion of the grasped object (Gabiccini et al., 2011). In addition, the synergies were used in a dual-arm anthropomorphic system while performing manipulation tasks (Suárez et al., 2015).

All these works dealt with synergies involving correlations between joint positions. Nevertheless, it seems natural to complement the information embedded in these traditional

⋆ This work was partially supported by the Spanish Government through the projects DPI2013-40882-P, DPI2014-57757-R and DPI2016-80077-R. N. García is supported by the Generalitat de Catalunya through the grant FI-DGR 2016.

Fig. 1. Human operator performing a task with both hands while wearing the measurement equipment.

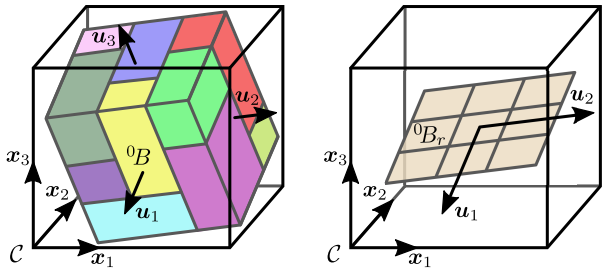


Fig. 2. The box containing the zero-order synergies defines the relevant region of $\mathcal{C}$, called ^0B. In turn ^0B is divided into synergy cells so that the first-order synergies in each cell are different (left). A grid of tree cells in ^0B_r, i.e. the subspace spanned by the first r zero-order synergies ($\boldsymbol{u}_1$ and $\boldsymbol{u}_2$ in the figure), is used to estimate the coverage of the whole $\mathcal{C}$ (right).

synergies with new synergies computed from samples captured in the velocity space of the system, generalizing thereby the concept of postural synergies. These synergies obtained in the space of the first derivative of the configuration trajectories were called *first-order synergies* (García et al., 2015). In that work, the position synergies were used, on the one hand, to detect the relevant region of the configuration space, i.e. the area where the synergies have been computed. On the other hand they were used to classify the first-order synergies by dividing this box into several *synergy cells* where the first-order synergies are significantly different. Moreover, studies made by Grinyagin et al. (2005) and Vinjamuri et al. (2007) expressed the angular velocities of finger joints as linear combinations of a small number of *kinematic synergies*, which were also angular velocities of finger joints. The kinematic synergies were also used in tracking arm movements (d'Avella et al., 2006). However, the velocity synergies have not been investigated in dual-arm movements neither used in the motion planning of bimanual tasks. Therefore, this opens a completely new field of research that is addressed in this work.

After this introduction, Section 2 presents the problem statement and gives an overview of the proposed approach, Section 3 details the proposal, the approach is validated in Section 4 and finally Section 5 presents the conclusions and future work.

2. PROBLEM STATEMENT AND APPROACH OVERVIEW

The goal of this work is to solve the motion planning of an anthropomorphic dual-arm robot trying to mimic the movements that a human does to solve a given task. To this end, a sampling-based planning algorithm is designed and the movements of human operators are used to guide the motion planning. The main features of the proposed approach are the following:

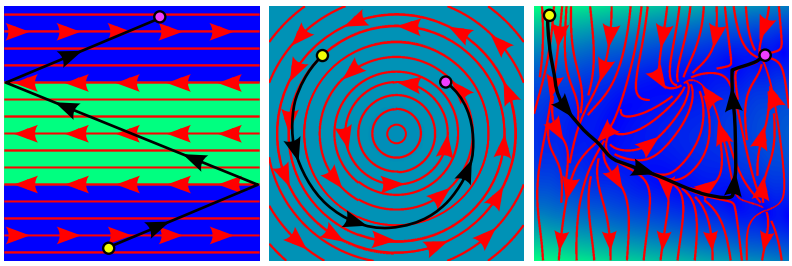
(1) Human movements are captured, and then mapped to the anthropomorphic dual-arm robotic system, in order to obtain the synergies that exist in the dual-arm movements when humans solve a task.
(2) The computed synergies are used to generate a vector field over $\mathcal{C}$, the configuration space of the robot. This vector field guides the motion planning by assigning a desired velocity to each configuration in $\mathcal{C}$.
(3) The synergies are also used to select the subspace ^0B_r, a lower-dimensional subspace in $\mathcal{C}$, that contains a predefined high portion of the sample variance of the captured movements. During the motion planning, the projection of the tree samples into ^0B_r gives an idea of the coverage of $\mathcal{C}$.
(4) A bidirectional sampling-based planning algorithm, called FOS-BKPIECE, is designed to bias the tree growth towards the directions of the vector field. Hence, human-like movements are obtained. The proposed planner is based on the planners VF-RRT (Ko et al., 2014) and KPIECE (Şucan and Kavraki, 2010).

3. PLANNING PROCEDURE

3.1 Generating vector fields from synergies

This subsection presents a procedure to capture synergies from human operators and model them as a vector field of desired velocities. As stated in the previous section, the movements of human beings are used here to plan human-like motions for anthropomorphic dual-arm robots. First, with magnetic trackers and sensorized gloves, the position and orientation of the human wrists are captured during the execution of a given task (see Fig. 1). The wrists of the dual-arm robot are placed at the captured poses, by solving the inverse kinematics, in order to map the human movements to the robotic system (Suárez et al., 2015). Then, the synergies are computed in the robot configuration space. The concept of first-order synergies (correlations between DOF velocities) was introduced by García et al. (2015), and in turn the couplings of DOF positions were called zero-order synergies. The Principal Component Analysis (PCA) of the captured configuration samples in the joint space, returns a new basis of $\mathcal{C}$ with the axes ordered according to the sample dispersion along them. Each axis represents a zero-order synergy and the movement along it, equivalent to a single DOF, implies the coordinated movement of several (or all) the actual DOF of the system. The first-order synergies are obtained similarly, but in this case using velocity samples computed from the measured positions.

The zero-order synergies are used here to detect the relevant region of $\mathcal{C}$, called box ^0B, where the captured motions take place (see Fig. 2). Note that the directions of human movements depend on the region of $\mathcal{C}$ where



a) b) c)

Fig. 3. Benchmarking of the modified VF-RRT. A mobile robot travels from the start to the goal position trough the unit square, where no obstacles are present and different vector fields $\boldsymbol{f}$ are defined (a-c). The directions of $\boldsymbol{f}$ are depicted by red stream lines and the magnitude of $\boldsymbol{f}$ is denoted by the background color (changing from blue to green as $\|\boldsymbol{f}\|$ grows). The black lines show the paths with minimum upstream criterion.

they take place. Taking this into account, ${}^0B$ is divided into *synergy cells*, where the first-order synergies returned by the PCA of the velocity of the samples contained in a cell are significantly different to the ones of the other cells (García et al., 2015). Then, the first-order synergies are used to generate a vector field that assigns a desired joint velocity for each configuration $\boldsymbol{q} \in \mathcal{C}$. For a robot with n DOF the desired velocity of $\boldsymbol{q}$ is randomly picked from the subspace spanned by the first $p \leq n$ first-order synergies of the synergy cell where $\boldsymbol{q}$ lies. If $\boldsymbol{q}$ lies outside ${}^0B$, the closest synergy cell is used. Finally, the zero-order synergies are also used to define a subspace ${}^0B_r \subseteq {}^0B$ that is spanned by the first $r \leq n$ zero-order synergies (see Fig. 2). The projection of the sample trees into ${}^0B_r$ helps to detect the unexplored areas of $\mathcal{C}$.

3.2 Planning on vector fields

This subsection proposes a method to grow sample trees along the directions of a vector field as a parameterless variant of the planner VF-RRT (Ko et al., 2014).

The motion-planning problems where there is a preferred direction of movement for each configuration are well framed as motion planning on vector fields. To solve this kind of problems, an RRT-based planner, called VF-RRT, was proposed. This planning algorithm adjusts the randomly sampled nodes towards the vector-field direction as follows. Let $\hat{\boldsymbol{v}}_{\text{rand}}$ be the advance direction resulting from the RRT sampling and $\hat{\boldsymbol{v}}_{\text{field}}$ be the vector-field direction. Then, both directions are combined to obtain $\hat{\boldsymbol{v}}_{\text{new}}$, the actual advance direction:

$$\hat{\boldsymbol{v}}_{\text{new}} = \omega_{\text{rand}}\, \hat{\boldsymbol{v}}_{\text{rand}} + \omega_{\text{field}}\, \hat{\boldsymbol{v}}_{\text{field}} \tag{1}$$

with $\omega_{\text{rand}} \geq 0$ and $\omega_{\text{field}} \in \mathbb{R}$ ensuring that $\|\hat{\boldsymbol{v}}_{\text{new}}\| = 1$ and that $\|\hat{\boldsymbol{v}}_{\text{rand}} - \hat{\boldsymbol{v}}_{\text{field}}\| \geq \|\hat{\boldsymbol{v}}_{\text{new}} - \hat{\boldsymbol{v}}_{\text{field}}\|$. The weights ω_{rand} and ω_{field} are controlled by a parameter $\lambda > 0$ such that $\hat{\boldsymbol{v}}_{\text{new}} \to \hat{\boldsymbol{v}}_{\text{field}}$ if $\lambda \to \infty$ and $\hat{\boldsymbol{v}}_{\text{new}} \to \hat{\boldsymbol{v}}_{\text{rand}}$ if $\lambda \to 0$. Nevertheless, λ is not a fixed parameter, it is adaptively adjusted according to the progress of the motion planning (i.e. λ decreases if difficulties are found in growing along the vector-field directions and vice versa).

In the work of Ko et al. (2014), λ is initialized to a predefined λ_{init} and is updated every k iterations as follows. First, let $\boldsymbol{q}$ be a configuration candidate to be added to the sample tree and a rectilinear segment in $\mathcal{C}$ be called *motion*. Then, $\boldsymbol{q}$ is considered as being *efficient* if:

a) m, the motion reaching $\boldsymbol{q}$, is collision-free, and
b) δ, the distance between $\boldsymbol{q}$ and its closest configuration in the tree, is greater than a predefined $\delta_{\text{ineff}} \in [0, \epsilon]$, where ϵ is the standard RRT step-size parameter.

Let $E_{\text{ineff}} \in [0, 1]$ be the rate of inefficient nodes found in the last k iterations, and E^*_{ineff} be a reference value for E_{ineff}. Then, λ is updated as $\lambda' = \lambda\,(1 + E^*_{\text{ineff}} - E_{\text{ineff}})$. Thereby, λ grows if $E_{\text{ineff}} < E^*_{\text{ineff}}$ and vice versa. Note that λ_{init}, k, δ_{ineff} and E_{ineff} are user-defined and that some suitable values may be difficult to find. In addition, the parameter configuration is highly problem-dependant.

Table 1. Average results of the motion planning of the benchmarking problems.

$\boldsymbol{f}$	VF-RRT version	Success rate (%)	Planning time (s)	Number of iterations	Upstream criterion	Solution length (m)
a)	Modified	100	0.2665	1266	0.421	1.753
	Original	90.1	0.6222	3135	0.529	1.679
b)	Modified	100	0.1863	672	0.146	1.870
	Original	96.5	0.4629	2540	0.469	1.614
c)	Modified	100	0.1641	541	0.675	1.495
	Original	93.1	0.4000	1956	1.140	1.403

For these reasons, a new method to update λ is proposed here. First, λ is initialized to $\lambda_{\max}$, i.e. a high enough value so that the tree follows the vector field at the first iteration. As opposed to the original procedure, λ is updated at each iteration. Hence a smooth continuous growth of the tree is ensured. The new value λ' is computed as

$$\lambda' = \begin{cases} \lambda\, e^{-1} & \text{if } m \text{ is in collision} \\ \lambda\, e^{1 - 2\left(1 - \frac{\delta}{\epsilon}\right)^{0.3}} & \text{otherwise} \end{cases} \tag{2}$$

Finally, λ is clamped to the range $[\lambda_{\min}, \lambda_{\max}]$ to prevent it from underflowing or growing too much unnecessarily. $\lambda_{\min}$ and $\lambda_{\max}$ are set to 10^{-3} and 10^5, respectively, since no significant changes are observed in $\hat{\boldsymbol{v}}_{\text{new}}$ when λ varies from 0 to $\lambda_{\min}$ or from $\lambda_{\max}$ to ∞. Note that in this way, the user does not need to define any parameter and λ still decreases if it is difficult to grow the tree following the vector field (i.e. m implies collision or $\delta \to 0$).

To test the performance of the proposed modifications of the VF-RRT planner, a set of 2D benchmarking problems have been set up. They consist of a mobile robot navigating in an obstacle-free square of side length 1 m, where three different vector fields have been established (see Fig. 3). The upstream criterion $\mathcal{U}$, proposed by Ko et al. (2014), is used as a quality metric in the comparison of the solution paths obtained with the original and the proposed VF-RRT. $\mathcal{U}$ measures the extent to which a path $\mathcal{P}$ goes against a vector field $\boldsymbol{f}(\boldsymbol{q})$ and is computed as:

$$\mathcal{U} = \int_{\mathcal{P}} \|\boldsymbol{f}(\boldsymbol{q})\| - \frac{\boldsymbol{f}(\boldsymbol{q}) \cdot \dot{\boldsymbol{q}}}{\|\dot{\boldsymbol{q}}\|}\, \mathrm{d}\boldsymbol{q} \tag{3}$$

Thereby a path with lower $\mathcal{U}$ is less deviated with respect to the directions of the vector field and, therefore, is considered to be better. For instance in case the vector field is computed with human movements, the lower $\mathcal{U}$ a path obtains, the better the human movements are mimicked. A maximum planning time of 5 seconds is considered in the experimentation. If a path is not obtained within

Algorithm 1: FOS-BKPIECE

Input : Query configurations $\boldsymbol{q}_{\text{start}}$, $\boldsymbol{q}_{\text{goal}} \in \mathcal{C}$
Output : Valid path $\mathcal{P}$ from $\boldsymbol{q}_{\text{start}}$ to $\boldsymbol{q}_{\text{goal}}$

1: $\mathcal{G}_{\text{A}}$.AddMotion($\boldsymbol{q}_{\text{start}}, \boldsymbol{q}_{\text{start}}, \|\boldsymbol{q}_{\text{goal}} - \boldsymbol{q}_{\text{start}}\|$)
2: $\mathcal{G}_{\text{B}}$.AddMotion($\boldsymbol{q}_{\text{goal}}, \boldsymbol{q}_{\text{goal}}, \|\boldsymbol{q}_{\text{start}} - \boldsymbol{q}_{\text{goal}}\|$)
3: **for** $i \leftarrow 1$ **to** N **do**
4: $\quad \boldsymbol{q}_{\text{init}} \leftarrow$ RandConf($\mathcal{G}_{\text{A}}$.SelectCell( ))
5: $\quad \boldsymbol{q}_{\text{bias}} \leftarrow$ RandConf($\mathcal{G}_{\text{B}}$.TopExteriorCell( ))
6: $\quad \boldsymbol{q}_{\text{new}} \leftarrow$ NewConf($\mathcal{G}_{\text{A}}, \boldsymbol{q}_{\text{init}}, \boldsymbol{q}_{\text{bias}}$)
7: $\quad$ **if** ValidMotion($\boldsymbol{q}_{\text{init}}, \boldsymbol{q}_{\text{new}}$) **then**
8: $\quad\quad \mathcal{G}_{\text{A}}$.AddMotion($\boldsymbol{q}_{\text{init}}, \boldsymbol{q}_{\text{new}}, \|\boldsymbol{q}_{\text{bias}} - \boldsymbol{q}_{\text{new}}\|$)
9: $\quad\quad \boldsymbol{q}_{\text{bridge}} \leftarrow$ RandConf($\mathcal{G}_{\text{B}}$.CellContaining($\boldsymbol{q}_{\text{new}}$))
10: $\quad\quad$ **if** $\boldsymbol{q}_{\text{bridge}} \neq \emptyset$ **and** ValidMotion($\boldsymbol{q}_{\text{bridge}}, \boldsymbol{q}_{\text{new}}$) **then**
11: $\quad\quad\quad \mathcal{G}_{\text{B}}$.AddMotion($\boldsymbol{q}_{\text{bridge}}, \boldsymbol{q}_{\text{new}}, 0$)
12: $\quad\quad\quad$ **return** Path($\mathcal{G}_{\text{A}}, \mathcal{G}_{\text{B}}$)
13: $\quad$ Swap($\mathcal{G}_{\text{A}}, \mathcal{G}_{\text{B}}$)
14: **return** $\emptyset$

Algorithm 2: NewConf

Input : Grid $\mathcal{G}$ and configurations $\boldsymbol{q}_{\text{init}} \in \mathcal{G}$, $\boldsymbol{q}_{\text{bias}} \in \mathcal{C}$
Output : Configuration $\boldsymbol{q}_{\text{new}}$

1: **if** $\|\boldsymbol{q}_{\text{bias}} - \boldsymbol{q}_{\text{init}}\| \leq \epsilon$ **then return** $\boldsymbol{q}_{\text{bias}}$
2: $\hat{\boldsymbol{v}}_{\text{field}} \leftarrow$ FOS($\boldsymbol{q}_{\text{init}}$)
3: **if** RootedAtGoal($\mathcal{G}$) **then** $\hat{\boldsymbol{v}}_{\text{field}} \leftarrow -\hat{\boldsymbol{v}}_{\text{field}}$
4: **if** Rand01( ) $< P_{\text{bias}}$ **then** $\hat{\boldsymbol{v}}_{\text{rand}} \leftarrow$ UnitVector($\boldsymbol{q}_{\text{bias}} - \boldsymbol{q}_{\text{init}}$)
5: **else** $\hat{\boldsymbol{v}}_{\text{rand}} \leftarrow$ RandDir($\boldsymbol{q}_{\text{init}} - \mathcal{G}$.Parent($\boldsymbol{q}_{\text{init}}$))
6: $\hat{\boldsymbol{v}}_{\text{new}} \leftarrow$ NewDir($\hat{\boldsymbol{v}}_{\text{rand}}, \hat{\boldsymbol{v}}_{\text{field}}$)
7: **return** $\boldsymbol{q}_{\text{init}} + \epsilon \hat{\boldsymbol{v}}_{\text{new}}$

this time, the execution is marked as a failure. Table 1 shows the resulting average values of the success rate, the planning time, the number of iterations, the path upstream criterion $\mathcal{U}$ and the solution length. Note that for each vector field, 100 executions have been run for the modified VF-RRT and that 10000 different parameter configurations were used for the original planner. In order to consider all the possible instances of the original VF-RRT, the value of its parameters have been uniformly chosen at random from the corresponding intervals and a high enough maximum value for k has been considered. The approach proposed in this work has been implemented within The Kautham Project (Rosell et al., 2014), a motion planning and simulation environment developed at the Institute of Industrial and Control Engineering (IOC-UPC) for teaching and research, and was run in a 3.40-GHz Intel i7-3770, 4-GB RAM PC. From the simulation results shown in Table 1 it can be appreciated that the modified VF-RRT outperforms the original planner in all aspects: it obtains better solution paths (i.e. with a lower $\mathcal{U}$) and in less time. In fact, the original VF-RRT is not able to find a solution within the time restriction for some executions.

3.3 The FOS-BKPIECE planning algorithm

The proposed planning algorithm, called FOS-BKPIECE, is introduced in this subsection. The basic structure of the algorithm is similar to the bidirectional implementation of the KPIECE planner provided by the *Open Motion Planning Library* (Şucan et al., 2012). The KPIECE planner guides the exploration of $\mathcal{C}$ using a projection of the tree samples into a discretized space: the sample tree is mostly extended from samples lying in the boundary of this discretization. Note that although the KPIECE planner applies a bias towards the unexplored areas of $\mathcal{C}$, it does not use any nearest-neighbor structure to select

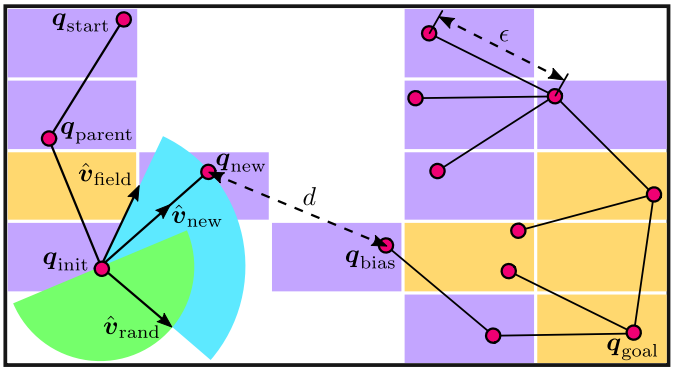


Fig. 4. Hypothetical representation of the planning procedure: Two sample trees, rooted at the configurations $\boldsymbol{q}_{\text{start}}$ and $\boldsymbol{q}_{\text{goal}}$, explore $\mathcal{C}$ pursuing their connection. To estimate the coverage of $\mathcal{C}$, the trees are projected to the subspace ${}^0B_r \subseteq \mathcal{C}$, where a cell-based discretization is established. The cells of ${}^0B_r$ containing tree nodes are classified, based on their number of neighbors, as interior or exterior (filled in orange and purple, respectively). At each iteration, a tree is expanded from a configuration $\boldsymbol{q}_{\text{init}}$ to a new configuration $\boldsymbol{q}_{\text{new}}$. The advance direction $\hat{\boldsymbol{v}}_{\text{new}}$ lies in the blue region and is a combination of a random direction $\hat{\boldsymbol{v}}_{\text{rand}}$, lying in the green region, and the direction $\hat{\boldsymbol{v}}_{\text{field}}$ of the first-order synergies.

the node that must be extended, as any RRT-variant does (e.g. VF-RRT). This produces a considerable reduction of computational load and settles the problems that may appear if the euclidean distance is not a good metric for the configuration space (Palmieri and Arras, 2015).

Additionally, the proposed FOS-BKPIECE planner uses the introduced modifications of the VF-RRT planner to guide the tree towards the synergy directions. Hence, solution paths that mimic the human movements are obtained. The planner is described in Algorithm 1 and has the following main features:

- Two trees, rooted at the start configuration $\boldsymbol{q}_{\text{start}}$ and at the goal configuration $\boldsymbol{q}_{\text{goal}}$ (Lines 1-2), are steered towards each other while exploring the configuration space $\mathcal{C}$. The function AddMotion($\boldsymbol{q}_{\text{init}}$, $\boldsymbol{q}_{\text{new}}$, d) inserts the motion from $\boldsymbol{q}_{\text{init}}$ to $\boldsymbol{q}_{\text{new}}$ in the tree, so that $\boldsymbol{q}_{\text{init}}$ becomes the parent node of $\boldsymbol{q}_{\text{new}}$. d is a measure of the closeness of $\boldsymbol{q}_{\text{new}}$ to the other tree and it is used to figure out how easy it may be to connect the trees trough $\boldsymbol{q}_{\text{new}}$ (see Fig. 4). Hence d can be an estimation of the actual minimum distance between $\boldsymbol{q}_{\text{new}}$ and any configuration in the other tree.
- The trees are projected into ${}^0B_r$ to guess the explored areas of $\mathcal{C}$, or at least their projection into ${}^0B_r$ (see Fig. 2). ${}^0B_r$ has been divided into *tree cells* of a predefined cell size. The cells where each tree lie, referred as *grids* $\mathcal{G}_{\text{A}}$ and $\mathcal{G}_{\text{B}}$, are classified as *interior cells* if all their neighbor cells contain tree nodes; or as *exterior cells* otherwise (see Fig. 4). The tree cells must not be confused with the synergy cells in which ${}^0B$ is divided, and that contain first-order synergies (see Subsection 3.1). Moreover, the cells in $\mathcal{G}_{\text{A}}$ and $\mathcal{G}_{\text{B}}$ are sorted by an heuristic-based score. Preference is given to the tree cells that: a) are exterior; b) have fewer neighbors; c) have been more recently populated;

d) contain fewer tree nodes; and e) their nodes are close to the other tree and have been fewer times used to grow the tree (Şucan and Kavraki, 2010).

- At each iteration, the tree in $\mathcal{G}_\mathrm{A}$ grows from a configuration $\boldsymbol{q}_\mathrm{init}$, randomly selected from the tree nodes in the top-scored cell of $\mathcal{G}_\mathrm{A}$ (Line 4). $\boldsymbol{q}_\mathrm{init}$ is extended to a new configuration $\boldsymbol{q}_\mathrm{new}$ (Line 6), by an increment step ϵ, steered towards the first-order synergies and $\boldsymbol{q}_\mathrm{bias}$ (Line 5), a configuration randomly selected from the top-scored tree cell on the boundary of $\mathcal{G}_\mathrm{B}$, the other grid (see Fig. 4). If the motion from $\boldsymbol{q}_\mathrm{init}$ to $\boldsymbol{q}_\mathrm{new}$ is collision-free (Line 7) it is added to $\mathcal{G}_\mathrm{A}$, using the distance between $\boldsymbol{q}_\mathrm{bias}$ and $\boldsymbol{q}_\mathrm{new}$ as an overestimation of the measure d (Line 8).
- The connection of the trees is attempted trough a motion between $\boldsymbol{q}_\mathrm{new}$ and $\boldsymbol{q}_\mathrm{bridge}$, a configuration randomly selected from the nodes in the tree cell of $\mathcal{G}_\mathrm{B}$ to which $\boldsymbol{q}_\mathrm{new}$ would be projected (Line 9). If the motion is valid, it is added to $\mathcal{G}_\mathrm{B}$ and the solution path from $\boldsymbol{q}_\mathrm{start}$ to $\boldsymbol{q}_\mathrm{goal}$ is returned (Lines 10-12). Otherwise, the tree roles are exchanged (Line 13).

The proposed planner uses the function NewConf, outlined in Algorithm 2, to grow a tree from a given $\boldsymbol{q}_\mathrm{init}$ to some $\boldsymbol{q}_\mathrm{new}$. For that, the next procedure is followed:

- If $\boldsymbol{q}_\mathrm{init}$ is in the neighborhood of $\boldsymbol{q}_\mathrm{bias}$, i.e. the trees are closer than a distance ϵ, no synergy bias is applied and $\boldsymbol{q}_\mathrm{new}$ is $\boldsymbol{q}_\mathrm{bias}$ itself (Line 1). Otherwise, the advance direction $\hat{\boldsymbol{v}}_\mathrm{new}$ is steered by the direction $\hat{\boldsymbol{v}}_\mathrm{field}$ of the first-order synergies and a random direction $\hat{\boldsymbol{v}}_\mathrm{rand}$.
- $\hat{\boldsymbol{v}}_\mathrm{field}$ is the velocity of the vector field associated with $\boldsymbol{q}_\mathrm{init}$ (Line 2) and it is computed as described in Subsection 3.1. When the tree rooted at $\boldsymbol{q}_\mathrm{goal}$ is extended, $\hat{\boldsymbol{v}}_\mathrm{field}$ must be reversed (Line 3).
- $\hat{\boldsymbol{v}}_\mathrm{rand}$ points towards $\boldsymbol{q}_\mathrm{bias}$ with a probability P_bias (Line 4). Otherwise, $\hat{\boldsymbol{v}}_\mathrm{rand}$ is some random unit vector that satisfies $\hat{\boldsymbol{v}}_\mathrm{rand} \cdot (\boldsymbol{q}_\mathrm{init} - \boldsymbol{q}_\mathrm{parent}) \geq 0$ (see Fig. 4), where $\boldsymbol{q}_\mathrm{parent}$ is the parent node of $\boldsymbol{q}_\mathrm{init}$ (Line 5).
- $\hat{\boldsymbol{v}}_\mathrm{new}$ is a combination of $\hat{\boldsymbol{v}}_\mathrm{rand}$ and $\hat{\boldsymbol{v}}_\mathrm{field}$ (Line 6), as shown in Fig. 4 and described in Subsection 3.2. Finally, $\boldsymbol{q}_\mathrm{new}$ is the configuration at distance ϵ from $\boldsymbol{q}_\mathrm{init}$ in the direction of $\hat{\boldsymbol{v}}_\mathrm{new}$ (Line 7).

Good results are obtained setting P_bias to 0.05, being ϵ problem-dependant. The dimension r of ${}^0B_r$ is chosen so that the variance related to the first r zero-order synergies surpasses the 95% of the total variance. Similar procedure is followed with the number p of first-order synergies used to compute $\hat{\boldsymbol{v}}_\mathrm{field}$. As suggested by Şucan and Kavraki (2010), the size of the tree cells is chosen so that 20 cells fit along each axis of ${}^0B_r$.

4. VALIDATION OF THE APPROACH

4.1 Conceptual example

For illustrative purposes, a simple example has been set up consisting of a mobile robot travelling in an obstacle-free square of side length 1 m. The start configuration $\boldsymbol{q}_\mathrm{start}$ is at the top-left corner and the goal configuration $\boldsymbol{q}_\mathrm{goal}$ is at the top-right corner. Four regions with different first-order synergies have been artificially defined (see Fig. 5-a): the first-order synergies point downwards in the left region, rightwards in the middle-bottom region, and upwards in the right region. However the synergies do not establish a clear direction in the middle-top region. Note that in this example $n = r = p = 2$ and ${}^0B_r$ is $\mathcal{C}$ itself. The problem has been solved with the FOS-BKPIECE but also with a non-bidirectional version of the proposed planner, called FOS-KPIECE; the original VF-RRT; and the RRT*. The parameters of the VF-RRT have been empirically set and the RRT* has been modified to minimise $\mathcal{U}$, see Eq. (3). Some of the obtained solution paths are shown in Fig. 5. It can be appreciated how the sample trees grow with a greater pace in the sense defined by the first-order synergies of each region, thus encountering a good quality solution. Note that the sample trees of the FOS-BKPIECE and of the VF-RRT remain close to the solution path. However with the RRT* the sample trees spread completely over $\mathcal{C}$, with the consequent waste of time. Table 2 shows the average results after 100 executions of the mentioned algorithms, with the planning time limited to 5 s. All the planners obtain similar upstream criterion values,

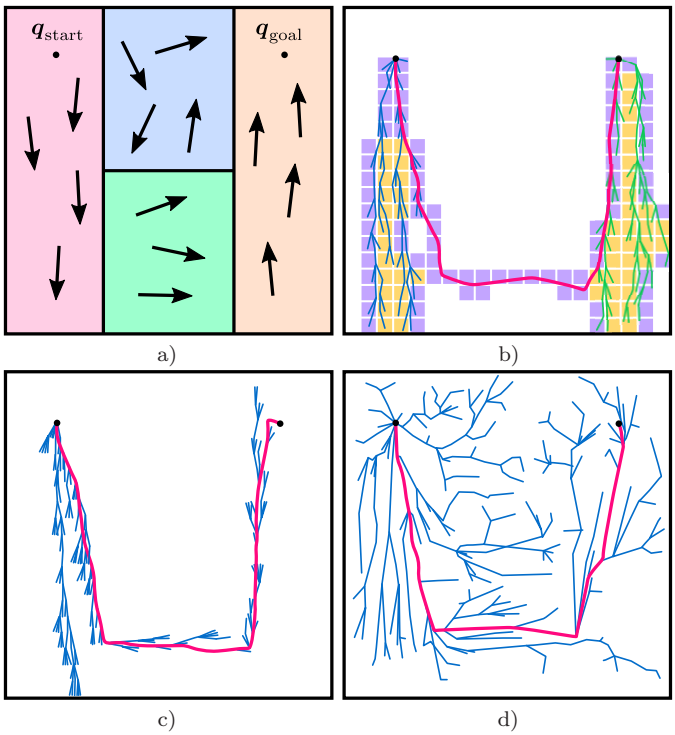


Fig. 5. Conceptual example: A mobile robot must go from $\boldsymbol{q}_\mathrm{start}$ to $\boldsymbol{q}_\mathrm{goal}$, preferably along the directions of the first-order synergies (a). Four regions has been artificially imposed over $\mathcal{C}$, each one with different synergy directions (denoted by arrows). Trees and solutions paths obtained with the FOS-BKPIECE (b), VF-RRT (c) and RRT*(d) planners are shown. The paths are depicted by red lines. Interior and exterior tree cells are filled in orange and purple, respectively.

Table 2. Average results of the motion planning of the conceptual example.

Planner	Success rate (%)	Planning time (s)	Number of iterations	Upstream criterion	Solution length (m)
FOS-BKPIECE	100	0.125	452	0.513	3.831
FOS-KPIECE	100	0.298	775	0.494	3.880
VF-RRT	100	0.406	812	0.491	3.914
RRT*	100	5	992	0.505	3.517

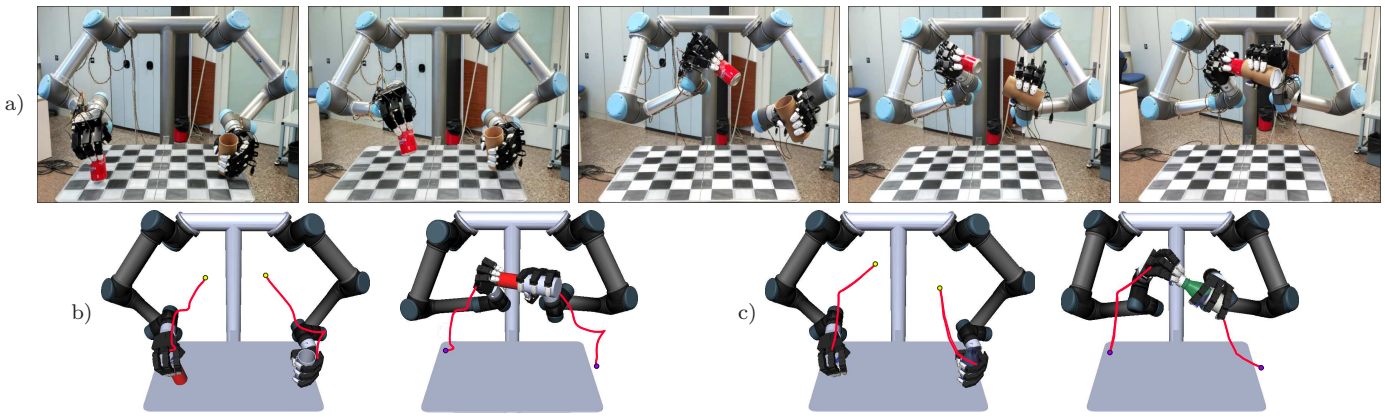


Fig. 6. Solution paths obtained with the planner FOS-BKPIECE: snapshots of the path execution with the real dual-arm robot for the *assembly* task following $\boldsymbol{f}_{\mathrm{F}}$ (a); and start and goal configurations in the simulation environment, showing the translational planned path, for the *assembly* task following $\boldsymbol{f}_{\mathrm{A}}$ (b) and the *bottle* task following $\boldsymbol{f}_{\mathrm{F}}$ (c).

Table 3. Average results of the motion planning of the application problems (1-3) using the planners FOS-BKPIECE (a), FOS-KPIECE (b), VF-RRT (c) and RRT* (d).

Problem	1)				2)				3)			
Planner	a)	b)	c)	d)	a)	b)	c)	d)	a)	b)	c)	d)
Success rate (%)	100	100	100	100	100	100	100	100	100	100	100	100
Planning time (s)	0.489	0.665	3.688	100	0.367	0.886	2.970	100	0.406	0.829	2.158	100
Number of iterations	154	115	290	14036	88	216	232	14423	94	224	206	13737
Upstream criterion	3.234	3.197	3.174	3.042	5.047	4.954	4.746	4.642	4.912	4.735	4.707	4.690
Solution length (rad)	3.862	4.135	4.090	3.818	4.279	4.404	4.368	4.509	4.244	4.378	4.105	4.795
Valid motion rate (%)	91.69	93.25	72.09	55.29	90.64	92.30	74.45	54.77	85.60	94.33	79.71	54.55

being $\mathcal{U}$ a little bit higher with the FOS-BKPIECE due to its greedy attempt to connect the trees; but it is the fastest planner thanks to its bidirectionality (3 and 40 times faster than the VF-RRT and the RRT* planners, respectively). However, the non-bidirectional proposed algorithm FOS-KPIECE is still faster than the others.

4.2 Application example

The planning of the movements of an anthropomorphic dual-arm robotic system is used for a real example of the proposed planning procedure. The used dual-arm robot is composed of two UR5 industrial robotic arms from Universal Robots, assembled emulating the human arm configuration. Each arm has 6 DOF and is equipped with a 16-DOF Allegro Hand from SimLab (see Fig. 6). Hence, for the planning of the arm movements the configuration space $\mathcal{C}$ has dimension $n=12$. For this example, the motion planning of the following tasks is considered:

- *Assembly*: A human-demonstrated task that consists in grasping, from a table in front of the body, a soda can with one hand and a cylindrical box with the other and moving both objects to a pre-assembly pose that allows the insertion of the can into the box.
- *Bottle*: A non-demonstrated task that consists in grasping, from a table in front of the body, a bottle with one hand and the bottle cap with the other and then tapping the bottle.

The movements of three human operators have been captured while performing 10 times the *assembly* task. From the captured data, the zero- and first-order synergies are computed and ${}^0B$ is found, i.e. the region of $\mathcal{C}$ containing the zero-order synergies. Based on synergy differences, ${}^0B$ is split into 21 synergy cells (as those predefined in the conceptual example). Then using the computed synergies, the vector field $\boldsymbol{f}_{\mathrm{A}}$ is obtained. In addition, pursuing the general application of the proposed approach, the same procedure is applied to the movements of the human operators while moving both arms in an unconstrained random way, trying to cover the workspace in front of the body. With this, the natural movements of the human operators are collected even though the whole workspace may have not been completely covered. Using these captured movements, called *free-movements*, ${}^0B$ is split into 64 synergy cells and the vector field $\boldsymbol{f}_{\mathrm{F}}$ is generated.

The motion planning has been solved for the next cases:

(1) *Assembly* task following the directions of $\boldsymbol{f}_{\mathrm{A}}$.
(2) *Assembly* task following the directions of $\boldsymbol{f}_{\mathrm{F}}$.
(3) *Bottle* task following the directions of $\boldsymbol{f}_{\mathrm{F}}$.

Notice that, except in the first case, the motion planning has been solved following the directions of movement of another task. Thereby, it is tested the utility of $\boldsymbol{f}_{\mathrm{F}}$ as a general-purpose vector field in case a task-specific vector field is not available (i.e. when the task has not been demonstrated). As in the conceptual example, the planners that have been used are the proposed FOS-BKPIECE and FOS-KPIECE, a fine-tuned version of the original VF-RRT, and an RRT* modified to minimise the upstream criterion $\mathcal{U}$. For the first two cases, representative solution paths obtained with the FOS-BKPIECE planner are shown in Fig. 6, both in the simulation environment and with the real dual-arm robot. For the three studied cases, Table 3 shows the average results of the motion planning after 100 executions and with a time limit of 100 s. Note that the solution length is measured in $\mathcal{C}$, as the sum of

joint movements, and that additionally the valid motion rate is collected, i.e. the proportion of iterations in which no collisions occur and the tree actually grows. It can be appreciated that the paths that best follow the human movements, i.e. the ones with the lowest $\mathcal{U}$, are obtained with the RRT* but at the expense of a prohibitive planning time. The fastest planner is the proposed FOS-BKPIECE, without significantly impairing the upstream criterion. The planner FOS-KPIECE is still faster than the VF-RRT and RRT*, and obtains similar $\mathcal{U}$ values. For the assembly task, lower $\mathcal{U}$ values are obtained with the task-specific vector field than with $\boldsymbol{f}_{\mathrm{F}}$. However, the paths of both tasks solved with $\boldsymbol{f}_{\mathrm{F}}$ mantain the human apperance (see Fig. 6).

5. CONCLUSIONS AND FUTURE WORK

This paper has proposed a planning algorithm for anthropomorphic dual-arm robots, called FOS-BKPIECE and that founds a path with movements similar to the ones of a human being. For this purpose, the zero- and first-order synergies of the robot (i.e. couplings of DOF positions and of velocities, respectively) have been computed from real human movements. Then, the configuration space $\mathcal{C}$ is split into synergy cells, using zero-order synergies, and each cell is paired with a set of first-order synergies. Thereby, a vector field of velocities is generated (i.e. an assignment of a velocity to each configuration in $\mathcal{C}$ based on the human movements). Paths following this vector field can be obtained with the planner VF-RRT. However, this implies several parameters that need to be finely tuned for each problem. For that reason, a parameterless VF-RRT, that exposes better results than the original one, is introduced in this paper. The proposed FOS-BKPIECE is based on this modified VF-RRT and also on the KPIECE. To illustrate the presented ideas, the presented approach has been compared with the original VF-RRT and the RRT* planners in conceptual and application examples. The obtained results show that the proposed procedure obtains paths that follow better the human movements.

Future work is focused on the application of the proposal to a system composed of several collaborative robots. The extension of the proposed planning method to the kinodynamic motion planning is another interesting topic.

REFERENCES

Argall, B., Chernova, S., Veloso, M., and Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Auton. Systems*, 57(5), 469–483.

Bianchi, M., Salaris, P., and Bicchi, A. (2013). Synergy-based hand pose sensing: Reconstruction enhancement. *Int. J. Robotics Research*, 32(4), 396–406.

Ciocarlie, M. and Allen, P. (2009). Hand posture subspaces for dexterous robotic grasping. *Int. J. Robotics Research*, 28(7), 851–867.

d'Avella, A., Portone, A., Fernandez, L., and Lacquaniti, F. (2006). Control of fast-reaching movements by muscle synergy combinations. *J. Neuroscience*, 96(30), 7791–7810.

Elbanhawi, M. and Simic, M. (2014). Sampling-based robot motion planning: A review. *IEEE Access*, 2, 56–77.

Ficuciello, F., Palli, G., Melchiorri, C., and Siciliano, B. (2014). Postural synergies of the UB Hand IV for human-like grasping. *Robotics and Auton. Systems*, 62(4), 515–527.

Fukuda, T., Michelini, R., Potkonjak, V., Tzafestas, S., Valavanis, K., and Vukobratovic, M. (2001). How far away is "artificial man". *IEEE Robotics and Automation Mag.*, 8(1), 66–73.

Gabiccini, M., Bicchi, A., Prattichizzo, D., and Malvezzi, M. (2011). On the role of hand synergies in the optimal choice of grasping forces. *Auton. Robots*, 31, 235–252.

García, N., Rosell, J., and Suárez, R. (2015). Motion planning using first-order synergies. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2058–2063.

Grinyagin, I., Biryukova, E., and Maier, M. (2005). Kinematic and dynamic synergies of human precision-grip movements. *J. Neurophysiology*, 94(4), 2284–2294.

Karaman, S. and Frazzoli, E. (2011). Sampling-based algorithms for optimal motion planning. *Int. J. Robotics Research*, 30(7), 846–894.

Kavraki, L., Svestka, P., Latombe, J.C., and Overmars, M. (1996). Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robotics and Automation*, 12(4), 566–580.

Ko, I., Kim, B., and Park, F. (2014). Randomized path planning on vector fields. *Int. J. Robotics Research*, 33(13), 1664–1682.

Kuffner, J. and LaValle, S. (2000). RRT-Connect: An efficient approach to single-query path planning. In *Proc. IEEE Int. Conf. Robotics and Automation*, 995–1001.

Palmieri, L. and Arras, K. (2015). Distance metric learning for RRT-based motion planning with constant-time inference. In *Proc. IEEE Int. Conf. Robotics and Automation*, 637–643.

Rosell, J., Pérez, A., Aliakbar, A., Muhayyuddin, Palomo, L., and García, N. (2014). The Kautham project: A teaching and research tool for robot motion planning. In *Proc. IEEE Int. Conf. Emerging Technologies and Factory Automation*.

Rosell, J., Suárez, R., Rosales, C., and Pérez, A. (2011). Autonomous motion planning of a hand-arm robotic system based on captured human-like hand postures. *Auton. Robots*, 31(1), 87–102.

Santello, M., Flanders, M., and Soechting, J. (2002). Patterns of hand motion during grasping and the influence of sensory guidance. *J. Neuroscience*, 22(4), 1426–1435.

Suárez, R., Rosell, J., and García, N. (2015). Using synergies in dual-arm manipulation tasks. In *Proc. IEEE Int. Conf. Robotics and Automation*, 5655–5661.

Şucan, I.A., Moll, M., and Kavraki, L.E. (2012). The Open Motion Planning Library. *IEEE Robotics and Automation Mag.*, 19(4), 72–82.

Şucan, I.A. and Kavraki, L.E. (2010). Kinodynamic motion planning by interior-exterior cell exploration. In *Algorithmic Found. of Robotics VIII*, 449–464. Springer.

Sun, S., Rosales, C., and Suárez, R. (2010). Study of coordinated motions of the human hand for robotic applications. In *Proc. IEEE Int. Conf. Information and Automation*, 776–781.

Vinjamuri, R., Mao, Z.H., Sclabassi, R., and Sun, M. (2007). Time-varying synergies in velocity profiles of finger joints of the hand during reach and grasp. In *Proc. IEEE Annual Int. Conf. Engineering in Medicine and Biology Society*, 4846–4849.

Wimböck, T., Jan, B., and Hirzinger, G. (2011). Synergy-level impedance control for a multifingered hand. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 973–979.