

Application

Eric Medvet*, Stefano Seriani, Alberto Bartoli, and Paolo Gallina

Evolutionary Optimization of Sliding Contact Positions in Powered Floor Systems for Mobile Robots

Abstract: Mobile robotics is a rapidly expanding technology due to its potential for increased safety and lower costs. In many applications, power is supplied to the robot through sliding contacts and a powered floor. Deciding the positions of the contacts on the robot is a difficult task: for any position/orientation of the robot, at least one contact has to touch a positive strip and at least one a negative strip. In this work, we tackle the problem using Differential Evolution (DE). We formally define problem-specific constraints and objectives and then describe how to use DE for evolving contact positions that satisfy those constraints and maximize those objectives. We validate experimentally our proposal by applying it to three real robots and by studying the impact of the main problem parameters on the effectiveness of the evolved designs for the sliding contacts.

Keywords: Multi-objective optimization, Automatic design, Differential Evolution

PACS: ...

1 Introduction and related work

Mobile robots are playing a role of increasing importance in current society. They are and will be aiding humans in performing tasks which may be dangerous, fatiguing, or boring due to repetitiveness. Examples range from persons transportation and goods delivering to surveillance.

One significant challenge in the field of mobile robots is how to provide power to the robots. The most common solution is to use batteries, which have, however, the apparent limitation that have to be recharged, causing a stop in the robot operations. A viable alternative consists in delivering power to the robot by electrifying the environment where it moves: this solution builds on the long-established experience on supplying powers to mobile machines, the most notable examples being in transportation systems, e.g., trains [12] and trams [9]. Another large family of solu-

tions is based on wireless power delivery, often realized by means of resonating coils [16, 18]. Finally, delivering power directly via conductors is often achieved through umbilical cables [11, 10].

In this paper, we focus on powered floors: the surface on which the robot moves is covered with conductive strips, interleaved by narrow non-conductive strips, alternatively connected to positive and negative poles of a power source; the robots receive the power from the strips using a number of sliding contacts positioned on the bottom of their bodies. This solution fits well the scenario of lab experimentation with Swarm Robotics [1] or Evolutionary Robotics [8]. In these settings, small prototypical robots have to run for long time, without obstacles to their movements (as, e.g., wires for power supply), in a physically constrained environment which has to be easily observable and accessible to researcher, for experiment “debugging” purposes. The latter condition may be not met if, e.g., robots get the power from the floor and the ceiling. Indeed, powered floors have already been used in similar settings [17] and also inspired the design of robotic platforms tailored purposely to experimentation [6].

Designing a working powered floor system (i.e., robots and floor) requires to decide the width of the strips and the positions of the sliding contacts on the robot. While the first task is not subjected to many constraints, for the positions of the contacts one has to take in to account the shape of the robot and the presence of moving parts, parts which should not be covered (e.g., sensors), or parts which are too far from or too close to the floor. Moreover, the fundamental requirement is that the contacts should be positioned in a way that guarantees that at least one contact is on a positive strip and one on a negative strip for any rotation and position of the robot. A fixed contacts design which aims at satisfying this condition, consisting of four contacts, has been already proposed in the 50s by Claude Shannon [14, p. 678], but it might be unsuitable for some robots due to the constraints described above.

We here propose to use a form of evolutionary optimization, called Differential Evolution (DE) [15], for finding automatically the positions of the sliding contacts, given a

*Corresponding author: Eric Medvet, Stefano Seriani, Alberto Bartoli, Paolo Gallina, Department of Engineering and Architecture, University of Trieste, Trieste, Italy

description of the region of the robot which is suitable for hosting them, which guarantee that the robot can receive power from the floor in any rotation and position. DE is an Evolutionary Algorithm (EA) for continuous optimization which can be used for solving single- or multi-objective problems.

We formally define the condition for the robot to be always powered, given the positions of its contacts. We introduce, based on the formal definition, two representations for the solution to be optimized (i.e., the positions of the contacts), one capable of exploiting the symmetries in the shape of the robot, and four quantitative objectives suitable for driving the search with DE.

We experimentally assess the ability of our proposed solution to find the contact positions for three real robotic platforms (Thymio II, mBot, and Elisa-3) and investigate how the solution representation and the objective impact on the effectiveness and efficiency of the optimization. Then, we explore the design space of the powered floor system problem: we focus on the two most relevant problem parameters, the width of the conductive strips and the maximum number of sliding contacts. The experimental evaluation suggests that our proposed solution based on DE is capable of designing automatically the salient part of a powered floor system, i.e., the positions of the contacts, in many different settings. The experiments also show that the quantitative objectives that we adopted for driving the search are indeed effective and that exploiting the symmetry, using an appropriate solution representation, better solutions can be obtained faster.

The present work is based in part on a preliminary study by the same authors [7]. We here extend the cited paper along many lines: (i) we propose a new solution representation that can exploit existing symmetries in the robot shape and validate on three real robotic platforms (Thymio II, mBot, and Elisa-3); (ii) we describe a new quantitative objective that measures to which degree the contact positions result in a balanced power delivery between positive and negative poles; (iii) we conduct a more detailed discussion of the experimental results, including a deeper comparison of the different design choices; and (iv) we present a prototype implementation of a powered floor system based on one of the commercial robots used in the method experimentation.

2 Powered floor system

We consider a scenario in which a mobile robotic platform (a *robot*) moves on a *powered floor*, that is, on a floor composed

of strips of conductive material interleaved by strips of non-conductive, insulating material (see Figure 9b). The conductive strips are alternatively connected to the positive and negative poles of a constant power source. The robot may move, by translating and rotating, on the powered floor without any constraint.

In order to get powered, the robot is equipped with an *array of sliding contacts*, whose positions are fixed with respect to the geometry of the robot. Each of the contacts of the robot is always in physical contact with the floor. The robot is in a *powered condition* if there is at least one contact on a positive conductive strip and at least one contact on a negative conductive strip. We call the contacts which enable this condition the *powered contacts*. There are no constraints on which contact is on which polarity: a simple rectifier circuit can be used to obtain a constant power source out of the powered contacts—we briefly describe an example of a rectifier circuit in Section 4.

In this work we are concerned with the problem of *optimizing the contact positions* so as to ensure that the robot is always in the powered condition irrespective of its position on the powered floor. This optimization problem must be solved subject to any constraints introduced by the geometry of the robot. Accordingly, we define the problem as follows. Given (a) a maximum number n of sliding contacts and (b) a description of a *feasible region* for the position of the sliding contacts with respect to the geometry of the robot, find a *design of the array* of up to n sliding contacts which (a) are in the feasible region and (b) allow the robot to be in a powered condition for any possible position, i.e., any possible combination of translation and rotation. We call a *powered array* an array of contacts which satisfies the two previous conditions, i.e., an array which guarantees that the robot is always in the powered condition.

We make several assumptions for ease of modelling and without loss of generality. The powered floor consists of infinite strips of infinite length, all of them parallel with the y -axis. All the conductive strips have the same width (denoted w) and all the non-conductive strips have the same width (denoted v). The first strip is a conductive strip and is connected to the positive pole—this strip thus starts at $x = 0$ and ends at $x = w$. Finally, we assume that each of the contacts of the robot is point-like (i.e., its size is zero along every axis).

We denote by x the position of the a reference point of the robot, by ω the rotation of the robot, and by $A = \{(r_1, \phi_1), \dots, (r_n, \phi_n)\}$ the array, i.e., the list of the n positions (r_i, ϕ_i) , expressed in polar coordinates, of each i -th contact with respect to the reference point of the robot. Figure 1 shows a schematic representation of the considered scenario and the corresponding notation.

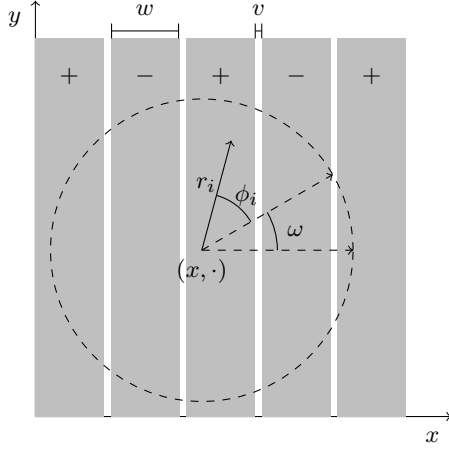


Fig. 1: A schematic representation of the considered scenario. The conductive strips are colored in gray, whereas the non-conductive strips are not colored; the robot reference point is in (x, \cdot) (the y coordinate is irrelevant) and its rotation is ω ; there is a single sliding contact (r_i, ϕ_i) .

We assume that the description of the feasible region for the contacts is provided as a function $p : [0, +\infty[\times [-\pi, \pi[\rightarrow \{0, 1\}$. A position (r, ϕ) , expressed in polar coordinates, is in the feasible region if and only if $p(r, \phi) = 1$.

A sliding contact (r_i, ϕ_i) is over a positive strip if and only if:

$$\exists k \in \mathbb{N} : 2k(w+v) \leq x + r_i \cos(\phi_i + \omega) \leq (2k+1)(w+v) - v \quad (1)$$

The same condition can also be expressed in terms of remainder of the division, which allows to remove k :

$$(x + r_i \cos(\phi_i + \omega)) \bmod (2w + 2v) \leq w \quad (2)$$

We define a function f_c^+ which is 1 if the condition is met and 0 otherwise:

$$f_c^+(r, \phi, x, \omega) = \begin{cases} 1 & \text{if } (x + r_i \cos(\phi_i + \omega)) \bmod (2w + 2v) \leq w \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Similarly, we define the function f_c^- for negative strips:

$$f_c^-(r, \phi, x, \omega) = \begin{cases} 1 & \text{if } w + v \leq (x + r_i \cos(\phi_i + \omega)) \bmod (2w + 2v) \leq 2w + v \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The powered condition of a robot in position x and rotation ω is hence met when both of the following hold:

$$f^+(A, x, \omega) = \sum_{i=1}^{i=n} f_c^+(r_i, \phi_i, x, \omega) p(r_i, \phi_i) \geq 1 \quad (5)$$

$$f^-(A, x, \omega) = \sum_{i=1}^{i=n} f_c^-(r_i, \phi_i, x, \omega) p(r_i, \phi_i) \geq 1 \quad (6)$$

where the factor $p(r_i, \phi_i)$ means that only contacts which are in the feasible region can be taken into account. The powered condition can also be written as:

$$f^\pm(A, x, \omega) = \min \left(f^+(A, x, \omega), f^-(A, x, \omega) \right) \geq 1 \quad (7)$$

where f^\pm is the number of powered contacts with the robot in position x and rotation ω . Finally, the condition that the robot is in a powered condition in any position x and rotation ω , i.e., the condition for A being a powered array, can be written as:

$$f(A) = \min_{\substack{x \in [0, 2(w+v)[\\ \omega \in [0, 2\pi[}} f^\pm(A, x, \omega) \geq 1 \quad (8)$$

The problem of finding a powered array may be hence solved by finding an A for which $f(A) \geq 1$, which can be done by optimizing A with the goal of maximizing $f(A)$, whose upper bound is $\lfloor \frac{n}{2} \rfloor$. Note that the larger $f(A)$, the better the positions of the contacts of the array, since the larger the number of powered contacts in the worst position and rotation of the robots.

2.1 Secondary objectives

It can be seen that, due to the presence of min and mod operators and of the binary co-domain of the function p , f may be extremely non-smooth. Figure 2 shows how f^\pm varies with the position x (top) and rotation ω (bottom) of a robot with 5 sliding contacts equally spaced on a circle of radius $r_0 = 20$ mm centered in the reference point—i.e., $(r_i, \phi_i) = (r_0, (i-1)\frac{2\pi}{5})$ with $i \in \{1, \dots, 5\}$ —which moves on a powered floor with $w = 9$ mm and $v = 1$ mm: for this specific contacts array A , $f(A)$ is 0, which means that there are positions and rotations for which the powered condition is not met, as visible in Figure 2 for, e.g., $x = 11$ cm or $\omega = 2.5$ rad.

The non-smoothness of f may negatively affect the effectiveness and efficiency of the optimization for a powered array. In order to address this problem, we consider a function \hat{f} which is the average value of f^\pm , instead of the min value, as in Equation (8):

$$\hat{f}(A) = \frac{1}{4\pi(w+v)} \iint_{\substack{x \in [0, 2(w+v)[\\ \omega \in [0, 2\pi[}} f^\pm(A, x, \omega) dx d\omega \quad (9)$$

Instead of measuring the number of powered contacts in the worst condition as f , \hat{f} measures the average number of powered contacts across all the possible conditions. It is worth to note, however, that while $f \geq 1$ implies the the robot is always (i.e., for any x and ω) in a powered

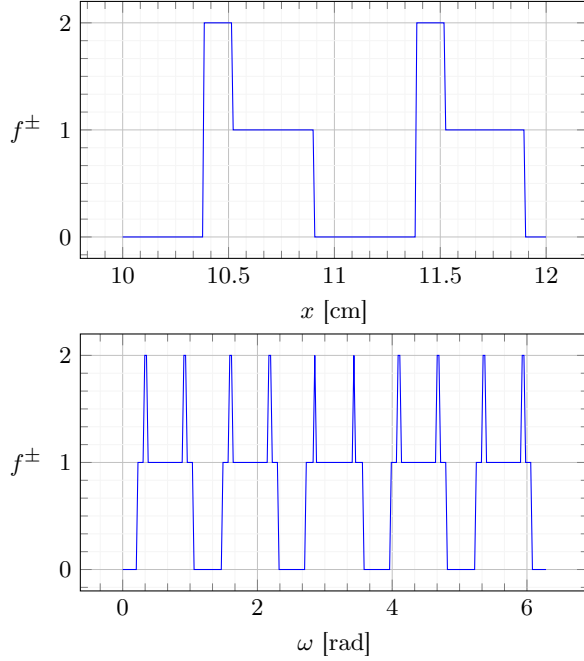


Fig. 2: Values of the function $f^\pm(A, x, \omega)$ for a simple array of 5 contacts $(r_0, (i-1)\frac{2\pi}{5})$ (equally spaced on a circle of radius $r_0 = 20$ mm centered in the robot reference point) for different positions (x , top) and rotations (ω , bottom) of the robot, with $w = 9$ mm and $v = 1$ mm.

condition, $\hat{f} \geq 1$ does not. That is, $\hat{f}(A) \geq 1$ does not guarantee that an array A is a powered array. In the next sections, we will show how we used f and \hat{f} together to drive the search for a powered array.

2.1.1 Distance among contacts

The function f does not explicitly take into account the relative positions of the sliding contacts. In particular, it does not penalize designs of the contacts array where the sliding contacts are too close to each other: those design might be harder to be realized in practice. To take into account this aspect, we introduce a third objective function d which measures the average distance of the contacts to their closest contact:

$$d(A) = \frac{1}{n} \sum_{i \in V} \min_{j \in P, j \neq i} \sqrt{r_i^2 + r_j^2 - 2r_i r_j \cos(\phi_i - \phi_j)} \quad (10)$$

where $P = \{1 \leq i \leq n : p(r_i, \phi_i) = 1\}$ is the set of the indexes of the contacts which are in the feasible region—i.e., as for f and \hat{f} , sliding contacts which are not in the feasible region are not taken into account by d .

2.1.2 Contact array balance

The function f^\pm of Equation (7) allows to express the powered condition for an array A in position x and rotation ω as $f^\pm(A, x, \omega) \geq 1$. The condition is met if at least one contact is connected to the positive strip, i.e., if $f^+(A, x, \omega) \geq 1$, and at least one contact touches the negative strip, i.e., if $f^-(A, x, \omega) \geq 1$. The difference in the number of contacts connected with the two polarities is therefore not taken into account by f^\pm , and hence by f and \hat{f} . An array minimizing that difference corresponds to a more balanced contact configuration, a feature being in general desirable; if we consider the total number of contacts fixed, a balanced configuration would mean more redundancy. For example, if 10 is the number of total contacts, a 6 positive/4 negative distribution is more redundant than an unbalanced 9/1 distribution, meaning that 3 negative poles could be removed, or fail to actually touch the floor for mechanical reasons, without power transmission failure.

In order to pursue this goal, we introduce another objective function which measures the average *balance* of the array. We first consider the balance of an array A in position x and rotation ω :

$$b'(A, x, \omega) = - \left| f^+(A, x, \omega) - f^-(A, x, \omega) \right| \quad (11)$$

where $f^+(A, x, \omega)$ and $f^-(A, x, \omega)$ measure, respectively, the number of contacts connecting with positive and negative strips (see Equations (5) and (6)). The minus sign outside the absolute value of the difference is inserted to make b' consistent with the semantics it should convey: great values (at most 0) mean good balance, i.e., small difference in the number of contacts, lower values mean poor balance, i.e., large difference. Then, we define the average balance by considering all possible positions and rotations of the robot (similarly to f' in Equation (9)):

$$b(A) = \frac{1}{4\pi(w+v)} \iint_{\substack{x \in [0, 2(w+v)[\\ \omega \in [0, 2\pi[}} b'(A, x, \omega) dx d\omega \quad (12)$$

2.2 Evolutionary optimization

We resort to Differential Evolution (DE) [15] for solving the problem of the design of contacts array. DE is an EA which can be used for optimization in continuous (real-valued) search space both for single- and multi-objective problems and hence fits the scenario of this study. There exist many variants of DE which are commonly identified with the DE/a/b/c naming scheme [3], where a represents the base vector to be perturbed, b represents the number of difference

vectors considered for perturbation, and c stands for the type of crossover being used, if any. In this work, we used the DE/rand/1, i.e., a variant without the crossover operator which selects the individual to be mutated randomly.

In brief, DE/rand/1 evolves a population of n_{pop} vectors $\mathbf{x} \in \mathbb{R}^{n_s}$ as shown in Algorithm 1. The population is first initialized (lines 1–8) by randomly setting vectors elements. Then, the following steps are repeated until a termination criterion is met: (i) three different individuals (i.e., vectors) $\mathbf{x}, \mathbf{y}, \mathbf{z}$ are randomly selected in the population (with uniform probability); (ii) a new solution \mathbf{x}' is built in which each element x'_j is set either to x_j (with a probability $1 - c_r$) or to a linear combination $x_j + d_w(y_j - z_j)$ of the corresponding elements of the selected individuals (with a probability c_r); (iii) if \mathbf{x}' is fitter than \mathbf{x} , then \mathbf{x}' replaces \mathbf{x} in the population, otherwise it is discarded. A usual termination criterion consists in having performed n_{ev} iterations, which correspond to n_{ev} fitness evaluations.

```

1  $P \leftarrow \emptyset$ 
2 foreach  $i \in \{1, \dots, n_{\text{pop}}\}$  do
3    $\mathbf{x} \leftarrow \mathbf{0}$ 
4   foreach  $j \in \{1, \dots, n_s\}$  do
5      $x_j \leftarrow U(0, 1)$ 
6   end
7    $P \leftarrow P \cup \{\mathbf{x}\}$ 
8 end
9 foreach  $i \in \{1, \dots, n_{\text{ev}}\}$  do
10   $\mathbf{x}, \mathbf{y}, \mathbf{z} \leftarrow \text{PickRandomly}(P, 3)$ 
11   $\mathbf{x}' \leftarrow \mathbf{x}$ 
12  foreach  $j \in \{1, \dots, n_s\}$  do
13    if  $U(0, 1) < c_r$  then
14       $x'_j \leftarrow x_j + d_w(y_j - z_j)$ 
15    end
16  end
17  if  $\mathbf{x}' \succ \mathbf{x}$  then
18     $P \leftarrow P \setminus \{\mathbf{x}\}$ 
19     $P \leftarrow P \cup \{\mathbf{x}'\}$ 
20  end
21 end

```

Algorithm 1: DE/rand/1 as used in this work.

In a multi-objective problem, the comparison between two individuals \mathbf{x}, \mathbf{x}' for determining the fittest one can be done in several ways. In this work, we use lexicographical ordering. Let $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^m$ be the m -dimensional fitness vector of the solution \mathbf{x} and let assume, without loss of generality, that we want to maximize fitness objectives. We say that \mathbf{x}' is fitter than \mathbf{x} , denoted by $\mathbf{x}' \succ \mathbf{x}$, if and only

if $\exists k \leq m : (\forall i < k : f_i(\mathbf{x}') = f_i(\mathbf{x})) \wedge (f_k(\mathbf{x}') > f_k(\mathbf{x}))$. That is, if \mathbf{x} and \mathbf{x}' are equivalent according to the first objective, then the second objective is compared, and so on.

2.2.1 Solution representation

We considered two different solution representations, i.e., ways of encoding the positions of the contact array A in a numerical vector suitable to be optimized using DE.

In the first representation, we encode A by means of a function $e : (\mathbb{R}^+ \times [0, 2\pi])^n \rightarrow \mathbb{R}^{2n}$ as follows:

$$\mathbf{x} = e(A) = \left(\frac{r_1}{r^*}, \frac{\phi_1}{2\pi}, \dots, \frac{r_n}{r^*}, \frac{\phi_n}{2\pi} \right) \quad (13)$$

where n is the maximum number of sliding contacts and r^* is the maximum value of r for which $p(r, \phi) = 1$, i.e., the distance of the farthest point of the feasible region from the reference point. This way, the initialization procedure of the population, which sets the elements of \mathbf{x} by sampling $U(0, 1)$ (see Algorithm 1) is appropriate for the domains of r_i, ϕ_i .

We also consider a variant of the representation in which a form of symmetry in the positions of the contacts is imposed. Imposing symmetry might be beneficial in two ways. First, it might result in arrays with improved feasibility, since the symmetry of the array can be set to be consistent with the symmetry of the robot. Second, it might result in a more efficient optimization (i.e., faster), since a search space of lower dimension is used for the same maximum number of contacts with respect to the case of free (i.e., without imposed symmetry) contact positions.

In principle, any kind of symmetry may be imposed by a proper solution representation. In this study we deal with three robots which exhibit three different symmetries (see Section 3): central, along one axis, along two orthogonal axes. For the sake of clarity and to allow comparisons among the three cases, we actually considered the symmetry along one axis (namely the y -axis), which is common among the three considered robots. In this case the solution is encoded using $e_{\text{sym}} : (\mathbb{R}^+ \times [0, 2\pi])^n \rightarrow \mathbb{R}^{2n}$ as follows:

$$\mathbf{x} = e_{\text{sym}}(A) = \left(\frac{r_1}{r^*}, \frac{\phi_1}{2\pi}, \dots, \frac{r_{\frac{n}{2}}}{r^*}, \frac{\phi_{\frac{n}{2}}}{2\pi}, \dots, \frac{r_1}{r^*}, \frac{-\phi_1}{2\pi}, \dots, \frac{r_{\frac{n}{2}}}{r^*}, \frac{-\phi_{\frac{n}{2}}}{2\pi} \right) \quad (14)$$

where the symmetry is reflected by the fact that for every contact r_i, ϕ_i there is a contact in $r_i, -\phi_i$.

2.2.2 Fitness

We explored different options for driving the search, i.e., different f based on f , \hat{f} , d and b functions (see Section 2.1).

Concerning f , \hat{f} , and b , the actual global minimum or average value for $x \in [0, 2(w+v)[$ and $\omega \in [0, 2\pi[$ (see Equations (8), (9) and (12)) cannot, in general, be obtained. Instead, a numerical approximation can be computed. We compute the values of the functions in the $n_{\text{points}}^2(x, \omega)$ pairs resulting from evenly sampling the two corresponding domains and take the minimum (for f) or average (for \hat{f} and b) of those values.

The four different options we considered are:

$$\begin{aligned} \mathbf{f}_M(\mathbf{x}) &= (f(\mathbf{x}')) \\ \mathbf{f}_{MD}(\mathbf{x}) &= (f(\mathbf{x}'), d(\mathbf{x}')) \\ \mathbf{f}_{MAD}(\mathbf{x}) &= (f(\mathbf{x}'), \hat{f}(\mathbf{x}'), d(\mathbf{x}')) \\ \mathbf{f}_{MABD}(\mathbf{x}) &= (f(\mathbf{x}'), \hat{f}(\mathbf{x}'), b(\mathbf{x}'), d(\mathbf{x}')) \end{aligned}$$

where $\mathbf{x}' = \mathbf{e}^{-1}(\mathbf{x})$ or $\mathbf{x}' = \mathbf{e}_{\text{sym}}^{-1}(\mathbf{x})$ depending on which representation is used and \mathbf{e}^{-1} is the inverse of the function \mathbf{e} . Using \mathbf{f}_M corresponds to driving the search with only the function f , i.e., a single-objective optimization, whereas the other cases correspond to a multi-objective optimization.

3 Experiments and results

We aimed at gaining insights about our proposed solution from the point of view of the evolutionary optimization—which representations and objectives lead to a more effective and efficient optimization—and of the considered application—how do the main problem parameters impact on the optimized arrays.

To this end, we conducted several experiments applying the proposed method to three different robots: Thymio II¹, mBot², and Elisa-3³. The three robots, shown in Figure 3, have similar size and were designed to ease learning and experimenting, in particular for children. They are also suitable (and have been actually used [4, 5, 13]) for experimenting with Evolutionary Robotics [8].

The feasible regions for the three robots are determined by the respective shapes and positions of the wheels and other parts which are designed to be in contact with the floor.



Fig. 3: Thymio II (left), mBot (center), and Elisa-3 (right) robots.

The feasible region for the Thymio II is defined as:

$$p(r, \phi) = \begin{cases} 1 & \text{if } -50 \text{ mm} \leq r \sin \phi \leq 0 \wedge |r \cos \phi| \leq 75 \text{ mm} \\ 1 & \text{if } 0 \leq r \sin \phi \leq 30 \text{ mm} \wedge |r \cos \phi| \leq 110 \text{ mm} \\ 0 & \text{otherwise} \end{cases} \quad (15)$$

The feasible region for the mBot is defined as:

$$p(r, \phi) = \begin{cases} 1 & \text{if } |r \cos \phi| \leq 45 \text{ mm} \wedge |r \sin \phi| \leq 45 \text{ mm} \\ 0 & \text{otherwise} \end{cases} \quad (16)$$

Finally, the feasible region for the Elisa-3 is defined as:

$$p(r, \phi) = \begin{cases} 1 & \text{if } 25 \text{ mm} \leq r \leq 30 \text{ mm} \\ 0 & \text{otherwise} \end{cases} \quad (17)$$

A graphical representation of the feasible regions is shown in Figure 4.

Concerning the parameters of the EA (population size n_{pop} , crossover rate c_r , and differential weight d_w), it should be noted that setting appropriate values when using DE is not trivial [2]. Based on some preliminary experiments and previous knowledge, we set $n_{\text{pop}} = 100$, $c_r = 0.8$, and $d_w = 0.5$. We verified that small variations of those parameters do not substantially alter the qualitative findings of the experimental evaluation.

For each experiment, we performed 30 independent runs by varying the initial random seed: we present mean and standard deviation computed across the runs for the salient indexes. We set $n_{\text{ev}} = 10\,000$ and $n_{\text{points}} = 100$. We performed the experiments on a machine equipped with an Intel(R) Core(TM) i5-3470 CPU at 3.20 GHz with 8 GB of RAM: one run took, on average, ≈ 80 s to be performed.

¹ <https://www.thymio.org/en:thymio>

² <https://www.makeblock.com/steam-kits/mbot>

³ <http://www.gctronic.com/doc/index.php/Elisa-3>

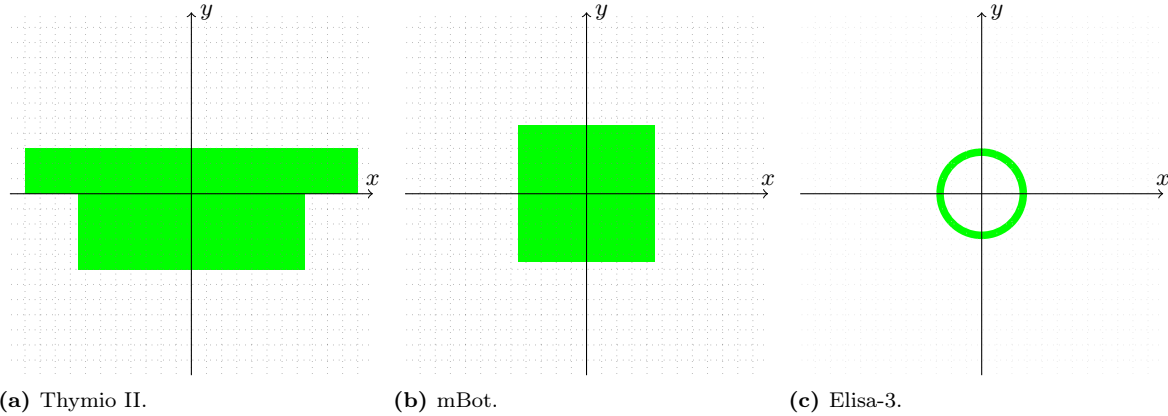


Fig. 4: Graphical representation of the feasible regions for the three robots, using Cartesian coordinates x, y , rather than polar coordinates r, ϕ : regions whose points are such that $p(r, \phi) = 1$, with $r = \sqrt{x^2 + y^2}$ and $\phi = \tan^{-1} \frac{y}{x}$, are plotted in green. The scale is the same for the three plots.

Tab. 1: Final value (mean and standard deviation) of f for the three robots and the four search variants with $w = 25$ mm, $v = 3$ mm, and $n = 10$.

Robot	MABD	MAD	MD	M
Thymio II	2 \pm 0	2 \pm 0	0.2 \pm 0.44	0.8 \pm 0.38
mBot	2 \pm 0	2.0 \pm 0.19	2 \pm 0	2 \pm 0
Elisa-3	2 \pm 0	2 \pm 0	1.5 \pm 0.51	1.8 \pm 0.44

3.1 Results and discussion

3.1.1 How to drive the search

Table 1 shows the value of f obtained at the end of the evolutionary search for the three robots with the four fitness variants with $w = 25$ mm, $v = 3$ mm, and $n = 10$.

It can be seen from the figures in Table 1 that there are differences among the robots and among the fitness variants. With the simplest fitness variant M, our proposed solution is able to obtain, on average, a final f of 1.76 for Elisa-3, 2 for mBot, and 0.83 for Thymio II. We recall that an array of contacts is a powered array (i.e., with that array the robot is in a powered condition in any position and rotation) when $f \geq 1$ (see Equation (8)): hence, our proposed solution with the M variant is able to design a powered array for two of the three robots. For the Thymio II robot, the M variant is not able to design a powered array, neither is the MD variant. With the MAD and MABD variants, instead, our solution always designs a powered array for each robot in almost all the runs (see the standard deviation values in Table 1)—alls for Elisa-3 and mBot. We hypothesize that the reason for which the search struggles for the Thymio

II robot is because its feasible region has a more complex shape than those of the other robots.

By analyzing the raw results, we verified that MAD and MABD outperform M and MD because, when the evolution stagnates on a value of f , improvements in the value of \hat{f} allow to gradually improve the array design. In other words, thanks to its better smoothness, \hat{f} can drive the search when f cannot. The d function—which, we recall, measures the average distance among contacts in the feasible region (see Equation (10))—does not seem to be able to drive the search better than f : the MD variant, in facts, obtains worse results than the M variant. This difference between \hat{f} and d is not surprising, however. On one hand, we introduced \hat{f} with the precise goal of mitigating the poor smoothness of f , hence as a helper for driving the search together with f . On the other hand, d represents a different goal, i.e., designing a contact array which, by avoiding contacts which are too close, is easier to be realized. From a different point of view, f and \hat{f} are not competitive objectives, whereas f and d are: in the extreme case, d pushes the search towards removing contacts (i.e., moving them away from the feasible region), because this increases the average distance; a counter-effect of removing a contact is that f may become lower. Note, however, that since we use lexicographical ordering of objectives for designs comparison (see Section 2.2), a change in a design which negatively affects f will never be kept during the search, regardless of the fact that it improves d .

Table 1 also shows that the difference between MAD and MABD, in terms of the value of f at the end of the optimization process, is negligible. By examining the results, we found that the search does not improve in efficiency (faster reach of the final value), nor in effectiveness (greater final

value): see also Section 3.1.2. Interestingly, in particular, the value of b is not significantly better when using MABD instead of MAD. We interpret this fact as a sort of redundancy of b and \hat{f} from the point of view of their ability to drive the search.

Figure 5 shows the average values of f during the search for the three robots (plots) and the four variants (line colors).

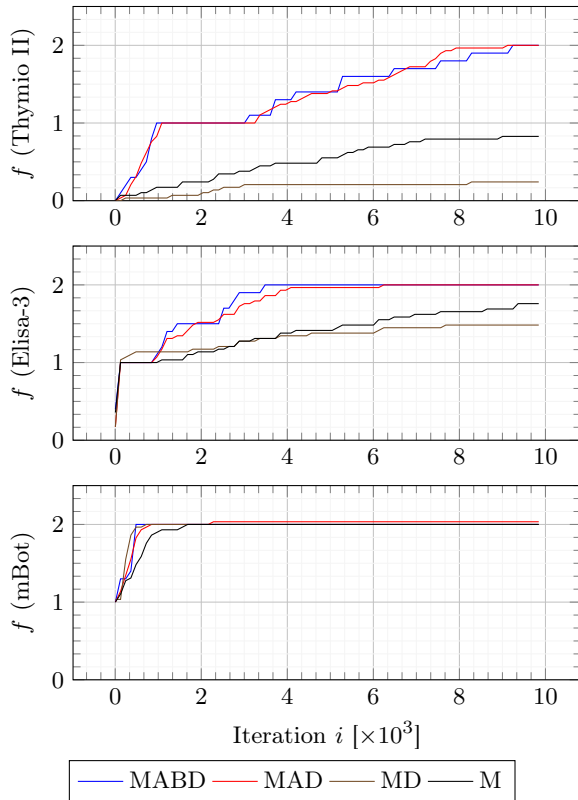


Fig. 5: Value of f during the evolution for the three robots (plots) with the four fitness variants (line color) with $w = 25$ mm, $v = 3$ mm, and $n = 10$.

The figure makes apparent the fact that, for two on three robots (Elisa-3 and Thymio II), the MAD and MABD variants are more effective and more efficient than M and MD; that is, the final value of f is greater and it is reached in fewer iterations. For the mBot the differences among M, MD, and MAD variants are negligible: we motivate this finding with the fact that the feasible region for mBot is easier than for Elisa-3 and Thymio II. In particular, it is more regular than the one of the latter and much larger than the one of the former (see Figure 4).

3.1.2 Secondary objectives

In order to better understand how the optimization leads to the final solution, we analyzed in detail the values of the four indexes (f , \hat{f} , d , and b) for the two best fitness variants: MAD and MABD. Figure 6 shows the results, i.e., the mean value across the 30 runs during the evolution.

It can be seen that the relations among the indexes are well represented by the respective lines. In particular, f and \hat{f} show similar trends, but \hat{f} is smoother and assume greater values: this is consistent with the fact that f is the minimum (across robot position and rotation) number of contacts touching the stripes, whereas \hat{f} is based the average.

Concerning d , it can be seen that it has a different trend than f and \hat{f} : indeed, there is a trade-off between the objective of maximizing the distance among contacts and the one of obtaining a powered array. This trade-off, however, is apparent only after the first stage of the evolution, i.e., when the solution is becoming promising.

Finally, Figure 6 shows that b exhibits a trend similar to f and \hat{f} , confirming the observation that it is not suitable for improving the optimization. Interestingly, however, b highlights the differences among the three robots: obtaining a balanced contact is easier with mBot than with Thymio-II.

3.1.3 Position of contacts

Figure 7 shows, for each robot, the outcomes of the application of our method obtained with $w = 25$ mm, $v = 3$ mm, and $n = 10$. For space constraints, we include here only four on 30 arrays that we randomly selected: we visually inspected the other arrays and they do not alter the following considerations.

As expected, since they are the result of an optimization process that employs some degree of randomness, no significant patterns emerge in contact positions. Interestingly, the figure confirms the fact that the Thymio-II robot is the one for which the method struggles more: the position of the contacts in the shown arrays appear to not exploit the full feasible region.

3.1.4 Symmetry

We here compare the two solution representations, the latter one exploiting the symmetry along the y -axis of the feasible region of the three considered robots. Table 2 shows the final value (mean and standard deviation across the 30 runs) of the four indexes with the two representations e and e_{sym} .

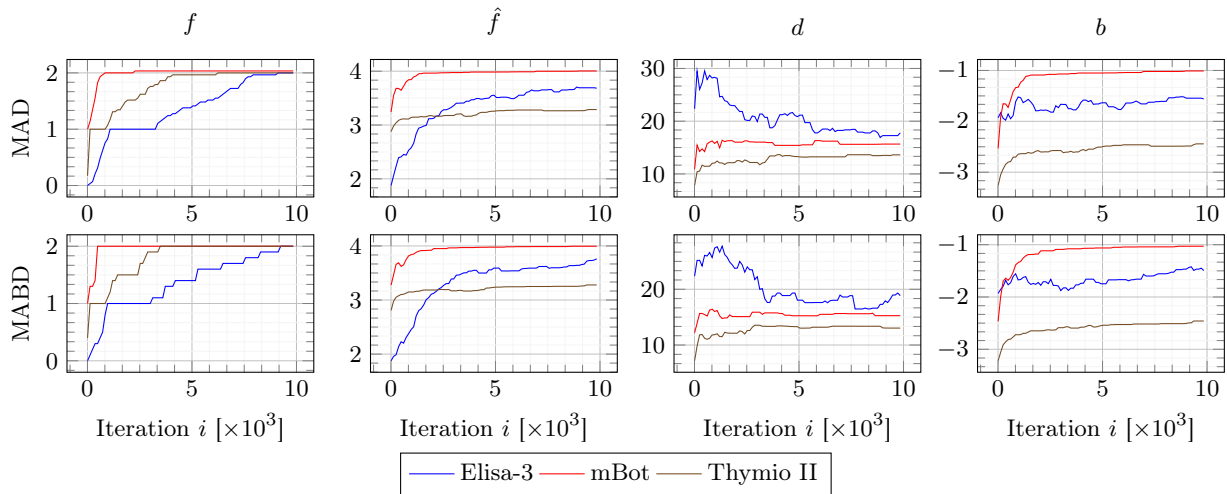


Fig. 6: Value of the salient indexes (columns of plots) during the evolution for the three robots (line color) with the two best fitness variants (rows of plots) with $w = 25$ mm, $v = 3$ mm, and $n = 10$.

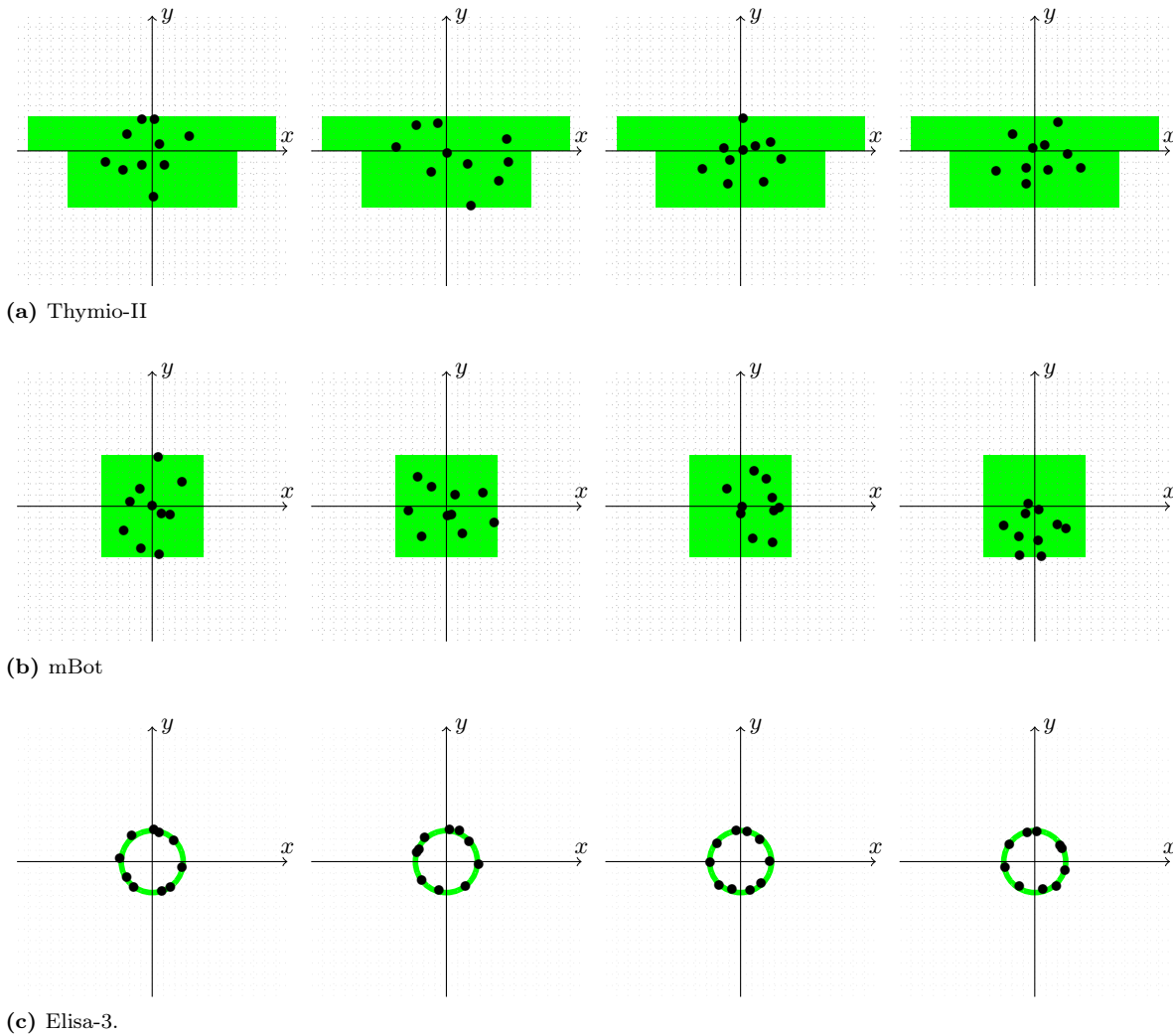


Fig. 7: Graphical representation of four solutions for each robot obtained with the MAD fitness variant and $w = 25$ mm, $v = 3$ mm, $n = 10$: each dot corresponds to a contact of the array.

Tab. 2: Final value (mean and standard deviation) of the four indexes using the two solution representations with $w = 25$ mm, $v = 3$ mm, and $n = 10$.

Rep.	Robot	f	\hat{f}	d	b
e_{sym}	Elisa-3	2 ±0	3.5±0.04	12.1±1.3	-2 ±0.1
	mBot	2.2±0.41	4.1±0.07	13.3±2.8	-0.8±0.1
	Thymio II	2.1±0.26	4.1±0.06	13.1±2.4	-0.8±0.1
e	Elisa-3	2 ±0	3.3±0.06	13.6±0.7	-2.4±0.1
	mBot	2 ±0	4 ±0.04	15.7±2	-1 ±0.1
	Thymio II	2 ±0	3.7±0.17	17.7±1.9	-1.5±0.2

3.1.5 Impact of main parameters

We considered only the best performing variant of our proposed solution (MAD) and performed a set of experiments by varying the value for $w \in \{15, 20, 25, 30, 35\}$ mm and for $n \in \{4, 5, 6, 8, 10, 12, 15\}$, i.e., the width of the conductive strips and the maximum number of contacts in the array, respectively. Figure 8 presents the average final values of f, \hat{f}, d for each robot (color line) and each one of the two parameters (w and n , row of plots)—as for the previous experimental campaign, we performed 30 runs for each combination of parameters.

The dependency between the three objectives and n , visible in Figure 8, looks as expected. The greater the (maximum) number of contacts n , the greater the values of f and \hat{f} and the smaller the value of d . Concerning f, \hat{f} , we observe that increasing n results, obviously, in more contacts in the array, and hence an opportunity for more powered contacts. Concerning the average distance among contacts d , the finding is sound: if more contacts are spread on the same feasible region, they are in general closer. Interestingly, the two leftmost plots in the top row of plots of Figure 8 also constitute an evidence of the better smoothness of \hat{f} with respect to f : the lines for the former are essentially straight lines, whereas in the lines for f sort of steps can be spotted in the plot.

From the bottom row of plots of Figure 8 it can be seen that the relation between w and the objectives is more complex than the one between n and the objectives. For example, the final \hat{f} for the Elisa-3 robot has a minimum for $w = 20$ mm, i.e., greater \hat{f} values are obtained for both $w = 15$ mm and $w = 25$ mm. More in general, it can be seen from these three plots that the relation between the objective and the parameter w varies across objectives and across robots. From a very high level point of view, these plots suggest that the larger the conductive strips, the better: we motivate this finding by observing that increasing w while keeping v (width of non-conductive strips) constant

corresponds, basically, to increasing the ratio of the floor area which is actually powered. Concerning the average contacts distance d , Figure 8 suggests that the wider the conductive strips, the larger the distance among contacts.

4 Prototype of a power system

We present in the following a preliminary implementation of the contact array applied to the Elisa-3 robot. The purpose is to briefly discuss the other issues of the design of the powered floor systems different than the one of determining the position of the contacts.

The prototype is illustrated in Figure 9. Specifically, Figure 9a shows the arrangement of the strips, which are interdigitated according to a negative, isolated, positive, isolated pattern.

In order to operate, the contact array system must be act as a rectifier, between the contacts and the power supply unit of the robot. This is necessary since the contacts receive indifferently a positive or negative voltage, but the circuitry on board the robot require a stable polarized source. In fact, the PCB visible on top of the robot in Figure 9b is a 8-pole rectifier circuit. The diagram for this can be seen in Figure 9c.

For the contact elements we used, in this preliminary embodiment, M1.5 screws with a round head. The chassis for the contact array was fabricated using an Ultimaker 2+ 3D printer; the material used was the polymer PLA. Thanks to this approach, we were able to provide a certain elasticity to the contact elements, which helped to increase adherence to the powered floor, hence improving conductivity and the overall robustness of the system.

5 Concluding remarks

In this paper, we focused on powered floor systems for delivering power to mobile robots. We considered the problem of automatically finding the positions of the sliding contacts which guarantee that the robot actually receives power from the floor in any position and rotation, the positions being constrained in a given feasible region defined depending on the robot shape and equipment. We introduced a formal formulation for the problem including four objectives to be maximized and two representations for the solutions, one of which allows to exploit possible symmetries in the robot. We tackled the resulting multi-objective optimization problem with Differential Evolution (DE).

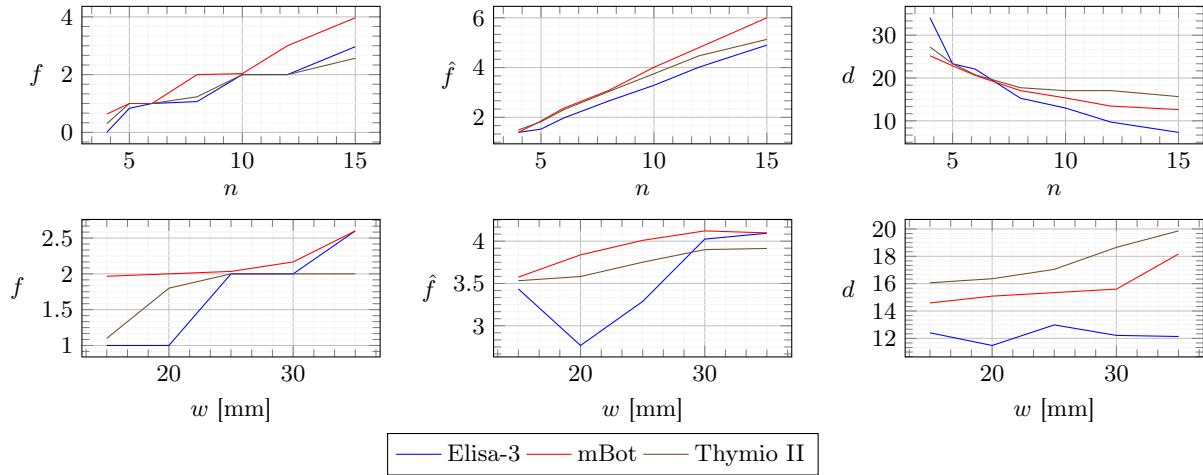


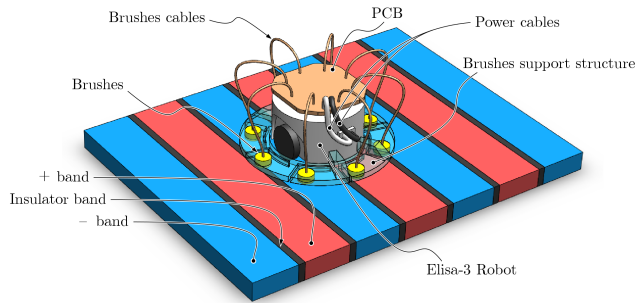
Fig. 8: Final values, averaged across the 30 runs, for the indexes f , \hat{f} , and d for different values for the maximum number n of contacts (top row of plots, with $w = 25$ mm) and different values for the width w of the conductive strips (bottom row of plots, with $n = 10$ and with $v = 3$ mm).

We experimentally verified that our proposed solution was indeed able to design effective arrays of sliding contacts for three real robots (Thymio II, mBot, Elisa-3). We investigated about the ability of the three objectives to drive the search with DE. We also experimentally explored the impact of the two most important problem parameters (the width of the conductive strips and the maximum number of sliding contacts) on the effectiveness of the contact arrays designed automatically with DE. The experimental results suggest that our solution may be useful in practice for assisting the design of powered floor systems.

Finally, we described a preliminary implementation of a powered floor system for an Elisa-3 mobile robot and briefly discussed issues of its design different than the one concerning the positioning of the contacts.

References

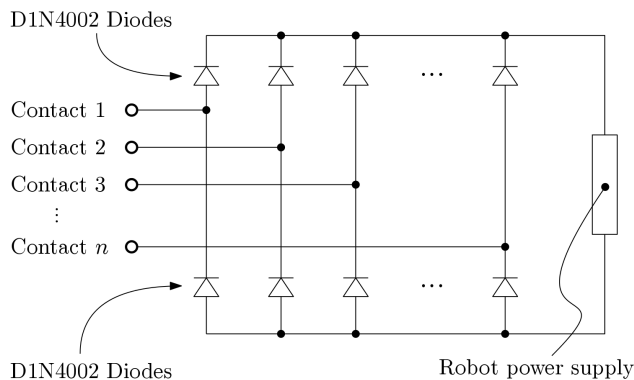
- [1] Manuele Brambilla, Eliseo Ferrante, Mauro Birattari, and Marco Dorigo. Swarm robotics: a review from the swarm engineering perspective. *Swarm Intelligence*, 7(1):1–41, 2013.
- [2] Janez Brest, Sao Greiner, Borko Boskovic, Marjan Mernik, and Viljem Zumer. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE transactions on evolutionary computation*, 10(6):646–657, 2006.
- [3] Swagatam Das and Ponnuthurai Nagarathnam Suganthan. Differential evolution: a survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, 15(1):4–31, 2011.
- [4] Jacqueline Heinerman, Massimiliano Rango, and AE Eiben. Evolution, individual learning, and social learning in a swarm of real robots. In *Computational Intelligence, 2015 IEEE Symposium Series on*, pages 1055–1062. IEEE, 2015.
- [5] Jacqueline Heinerman, Alessandro Zonta, Evert Haasdijk, and Agoston Endre Eiben. On-line evolution of foraging behaviour in a population of real robots. In *European Conference on the Applications of Evolutionary Computation*, pages 198–212. Springer, 2016.
- [6] John Klingner, Anshul Kanakia, Nicholas Farrow, Dustin Reishus, and Nikolaus Correll. A stick-slip omnidirectional powertrain for low-cost swarm robotics: Mechanism, calibration, and control. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 846–851. IEEE, 2014.
- [7] Eric Medvet, Stefano Seriani, Alberto Bartoli, and Paolo Gallina. Design of powered floor systems for mobile robots with differential evolution. In *International Conference on the Applications of Evolutionary Computation (Part of EvoStar)*, pages 19–32. Springer, 2019.
- [8] Stefano Nolfi, Josh Bongard, Phil Husbands, and Dario Floreano. Evolutionary robotics. In *Springer Handbook of Robotics*, pages 2035–2068. Springer, 2016.
- [9] Luigi Pastena. A catenary-free electrification for urban transport: An overview of the tramwave system. *IEEE Electrification Magazine*, 2(3):16–21, 2014.
- [10] S. Seriani, P. Gallina, and A. Wedler. A modular cable robot for inspection and light manipulation on celestial bodies. *Acta Astronautica*, 123:145–153, 2016.
- [11] S. Seriani, P. Gallina, and A. Wedler. *Dynamics of a*



(a) CAD view.



(b) The Elisa-3 robot with the contact array.



(c) Circuit.

Fig. 9: Implementation of a 8-contact array on a Elisa-3 robot. In 9a a 3D model is shown of the system, with the main components; in 9b a photo is shown of the system; in 9c the circuit is shown that drives current from the powered floors, through the contacts and finally to the robot power supply module.

- [12] AWC Shing and PPL Wong. Wear of pantograph collector strips. *Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit*, 222(2):169–176, 2008.
- [13] Fernando Silva, Luís Correia, and Anders Lyhne Christensen. Evolutionary online behaviour learning and adaptation in real robots. *Royal Society open science*, 4(7):160938, 2017.
- [14] Neil JA Sloane and Aaron D Wyner. *Claude Elwood Shannon: Collected Papers*. IEEE press, 1993.
- [15] Rainer Storn and Kenneth Price. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.
- [16] Junhua Wang, Meilin Hu, Changsong Cai, Zhongzheng Lin, Liang Li, and Zhijian Fang. Optimization design of wireless charging system for autonomous robots based on magnetic resonance coupling. *AIP Advances*, 8(5):055004, 2018.
- [17] Richard A Watson, Sevan G Ficici, and Jordan B Pollack. Embodied evolution: Distributing an evolutionary algorithm in a population of robots. *Robotics and Autonomous Systems*, 39(1):1–18, 2002.
- [18] Mingbo Yang, Guodong Yang, En Li, Zize Liang, and Hao Lin. Modeling and analysis of wireless power transmission system for inspection robot. In *Industrial Electronics (ISIE), 2013 IEEE International Symposium on*, pages 1–5. IEEE, 2013.