Path Smoothing Strategy Based on Metaheuristic Algorithms for Probabilistic Foam

Luís Bruno Pereira do Nascimento
Department of Computer and Electrical Engineering
Federal University of Rio Grande do Norte
Natal, RN, Brazil
lbruno@ufrn.edu.br

Vitor Gaboardi dos Santos
Department of Computer and Electrical Engineering
Federal University of Rio Grande do Norte
Natal, RN, Brazil
vitorgaboardi@ufrn.edu.br

Diego da Silva Pereira
Center of Innovation in Computer Technologies
Federal Institute of Rio Grande do Norte
Parnamirim, RN, Brazil
diego.pereira@ifrn.edu.br

Daniel Henrique Silva Fernandes
Department of Computer and Electrical Engineering
Federal University of Rio Grande do Norte
Natal, RN, Brazil
eng.danielhsf@gmail.com

Pablo Javier Alsina
Department of Computer and Electrical Engineering
Federal University of Rio Grande do Norte
Natal, RN, Brazil
pablo@dca.ufrn.br

ABSTRACT

The probabilistic Foam method (PFM) is a sampling-based path planning algorithm that ensures a feasible path bounded by a safe region. This method is ideal for assistive robotics applications, which demands a high level of safety, such as performing a motion by an active exoskeleton. However, PFM generates non-smoothed paths, which results in non-anthropomorphic movements. Thus, this paper presents some optimization strategies based on metaheuristics to smooth the paths generated by PFM. Simulated experiments were performed using the Harmony Search Algorithm, and Genetic Algorithm and they were applied to an exoskeleton to overcome an obstacle. Results show that our proposed approach is capable of smoothing paths for this application, which resulted in more anthropomorphic motions.

Keywords

Assistive Robotics; Active Exoskeleton; Path Smoothing; Genetic Algorithm; Harmony Search Algorithm

1. INTRODUCTION

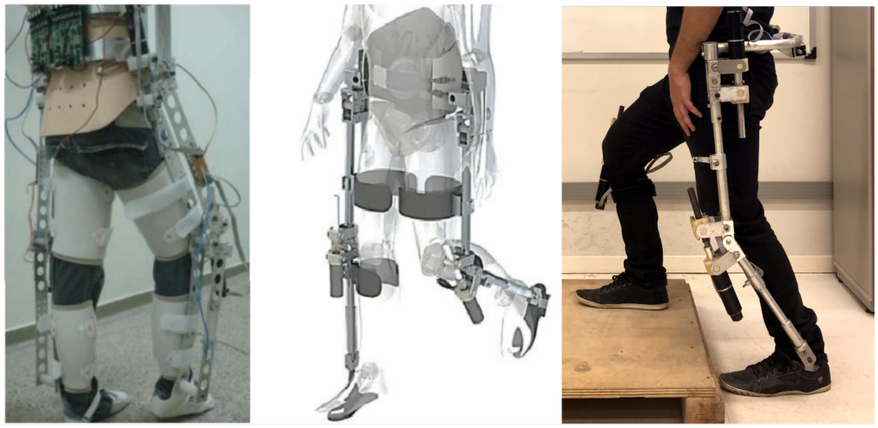
Applications in autonomous robotics are in constant expansion due to the fast technological advances. Moreover, Path planning is one of the most relevant issues related to this field of research, since a planner calculates a set of poses and orientations that a robot must perform to accomplish a specific task, such as moving from an initial to a goal configuration without colliding with obstacles along the path [11].

Among the several path planning methods, the sampling-based planners are the most promising path planning category since they can find feasible paths using random samples from the free configuration space. Sampling-based Path planners such as Probabilistic Roadmaps (PRM) [9] and Rapidly-Exploring Random Tree (RRT) [12] are the most known. The sampling-based path planners usually use few computer resources [13] and can be applied for robots with many degrees of freedom [14, 1].

Among different types of robots, assistive robots are devices that perform actions that benefit people with some kind of disability. For instance, Ortholeg [15] is a project of a lower limb active exoskeleton that helps physically challenged people in the walking experience. The Ortholeg exoskeleton was designed with the concept of transparency, which can be defined as the capability of the device to make the walking experience as natural as possible, both for the user and the people around him [15]. Figure 1 illustrates the exoskeleton used in this study.

In [3], a path planning algorithm called Probabilistic Foam Method (PFM) [20] was applied to the Ortholeg to provide safe movements for the task of overcoming a single obstacle. The planner PFM guarantees a high clearance path from the obstacles for safe maneuverability. A problem with the approach presented in [3] is that paths generated by PFM are non-smooth, which implies that the motion performed by the exoskeleton has a non-anthropomorphic pattern.

Path smoothing problems that involves many constraints can be solved with optimization techniques, as shown in [19]. In this way, some metaheuristics have been used as path smoothing optimization, as can be seen in the works [8] and [2], where Genetic Algorithm (GA) and Particle Swarm



(a) (b) (c)

Figure 1: The Ortholeg project: A lower limb active exoskeleton. (a) Ortholeg project version 1.0. (b) Design of Ortholeg Project version 2.0. (c) Ortholeg Project 2.0 in development.

optimization (PSO) were applied. Among these optimization methods, the Harmony Search Algorithm (HS) [4] is a metaheuristic with some interesting features [22] and better results in comparison with other algorithms for various problems [17, 10].

In [16], the algorithm HS was applied to smooth the paths generated by PFM. However, it was not performed any comparison with other methods to prove its efficiency. Thus, this work presents the application of the methods GA and HS to improve the smoothness of the paths generated by the Probabilistic Foam Method, guaranteeing the safety of the motion. For this study, we considered the motion planning of overcoming an obstacle by the exoskeleton Ortholeg as a study case, and some of the methods were applied to improve the generated paths.

This paper is organized as follows: Section 2 presents and describes the metaheuristic algorithms used in this work (Genetic Algorithm and Harmony Search Algorithm). Section 3 describes PFM, and its application for the orthosis. Section 4 models the path smoothing process as an optimization problem. Section 5 presents some simulated results and, finally, Section 6 presents some conclusions and future works.

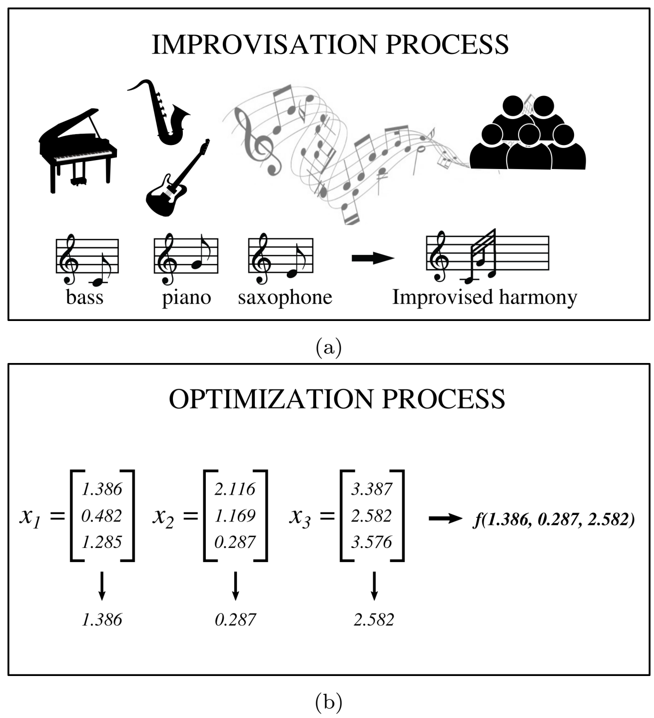


Figure 2: Musical improvisation (a) process versus optimization process (b).

2. THE METAHEURISTIC ALGORITHMS

This section presents some details about two relevant global optimization techniques: Harmony Search Algorithm, a method based on musical concepts and Genetic Algorithm, a method based on process of natural selection.

2.1 Harmony Search Algorithm

Harmony Search Algorithm (HS) is a metaheuristic inspired in musical concepts such as improvisation and memorization [4]. Finding the perfect harmony in music is analogous to find the optimality in an optimization process [22]. Figure 2 illustrates this analogy, where the musical instruments are the decision variables and the notes are the range of each variable. The harmonies represent the candidate solution of the problem and the people reaction is the objective function.

The simple HS algorithm has five steps, as follows:

1. Initialization of the HS parameters;
2. Initialization of the Harmony Memory;
3. Improvisation of a new harmony;
4. Harmony Memory Update
5. Stopping criteria verification

First, it is necessary to initialize the HS parameters: Harmony Memory Size (HMS), Harmony Memory Consideration Rate (HMCR), Pitch Adjustment Rate (PAR), Band-Width (BW) and the Number of Improvisations (NI). The Harmony Memory (HM) is a vector that contains HMS harmonies. The parameter HMCR indicates the probability of selecting an existing harmony from HM during the improvisation process and finally, PAR and BW are parameters related to adjustments in the harmony.

The initialization of the Harmony Memory is by HMS vectors sampled randomly. During the improvisation process, a harmony h is selected from the HM with probability HMCR. The adjustment of the harmony h generates a new harmony h_{new}:

$$h_{new} = h \pm x \times BW, \quad x \in U([0,1]) \tag{1}$$

where x is a random value uniformly distributed on [0, 1], and BW is a parameter between [0,1] that provides the adjustment.

After improvised, the new harmony is evaluated using the objective function. If h_{new} is better evaluated than the worst harmony inside the HM, the h_{new} replaces this worst harmony. Finally, while the stopping criteria is not achieved, new harmonies are improvised.

2.2 Genetic Algorithm

The Genetic Algorithm (GA) is an evolutionary meta-heuristic mainly inspired by Charles Darwin's theory of natural evolution. This technique was first introduced in [7], and popularized in [5], and it can be applied to solve several global optimization problems. This algorithm considers the mechanisms of natural selection where the fittest individuals are selected as parents for a reproduction process in order to produce offsprings.

In this algorithm, a set of possible solutions (individuals) are randomly generated from a search space, forming the Initial Population. All individuals are evaluated by a fitness function (Objective Function) in order to select the best pairs for reproduction. The reproduction process consists basically of two genetic operations: Crossover and Mutation. The Crossover is a genetic operator that combines the information of two parents to generate a new offspring. The Crossover is a genetic operator that combines the information of two parents to generate a new offspring. The second operation is the Mutation, which is responsible for maintaining genetic diversity through some small modifications in the chromosome of an individual. Finally, Elitism is a strategy that guarantees the solution quality by maintaining the best individual of each generation in the next population.

A pseudocode describing the steps of Genetic Algorithm is presented in Algorithm 1.

The genetic algorithm presents some relevant parameters for the convergence process: The Population size *pop_size* is the number of individuals in the population; The Crossover probability *cross_prob* indicates how often the selected parents will be reproduced, and usually is a value close to 1; The Mutation probability *mut_prob* indicates how often the chromosome will mutate and usually is a small value.

Algorithm 1: Basic Genetic Algorithm
Require: *cross_prob*, *mut_prob*, *pop_size*
Ensure: *best_individual*.

```
pop ← generate_initial_population(pop_size);
evaluate_initial_population(pop);
while stop criteria is not met do
    parents ← select_individuals(pop);
    offspring ← cross_individuals(parents, cross_prob);
    offspring ← mut_offspring(offspring, mut_prob);
    pop ← apply_elitism(offspring);
    best_individual ← obtain_best_offspring(pop);
end while
```

3. THE PROBABILISTIC FOAM METHOD

Probabilistic Foam method (PFM) is a robot path planner proposed in [20] with the main feature of generating high clearance paths for safe maneuverability. In this work it was used an implementation of PFM using information from workspace, as in [3]. In this implementation, a basic structure of the method, called bubble, was computed in the configuration space $\mathcal{C}$ using distance information from workspace $\mathcal{W}$ based on the concept of *bubbles of free space* [18].

A bubble b with radius r, centered in a configuration q is defined as a volumetric structure with a hyperspherical shape which is composed by a set of free configurations. The planning of overcoming an obstacle for the orthosis considers the configuration vector $q = [\theta_{\mathrm{h}}, \theta_{\mathrm{k}}, \theta_{\mathrm{q}}]^T$, where θ_{h} is the hip joint, θ_{k} is the knee joint and θ_{q} is the displacement of the hip. In this way, the bubble is defined as a 3-*ball* and its surface is a 2-*sphere* in $\mathcal{C}$, as can be seen in Fig. 3b. The path planning will return a path from the initial configuration q_{init} to the goal configuration q_{goal}, providing the motion shown in Fig. 3a.

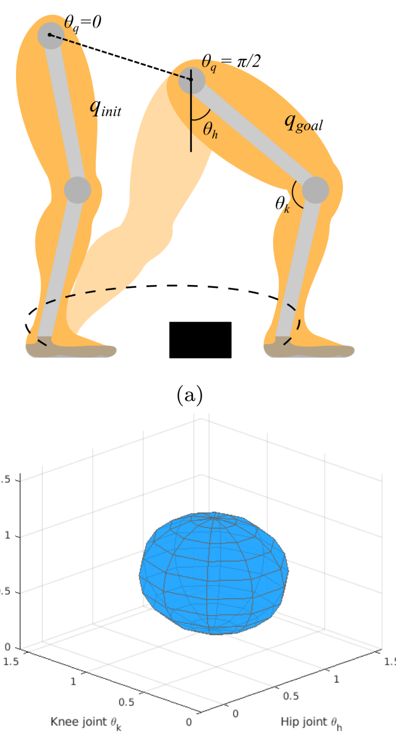


Figure 3: Overcoming an obstacle with PFM. (a) Leg positions in $\mathcal{W}$. (b) Illustration of a bubble in $\mathcal{C}$.

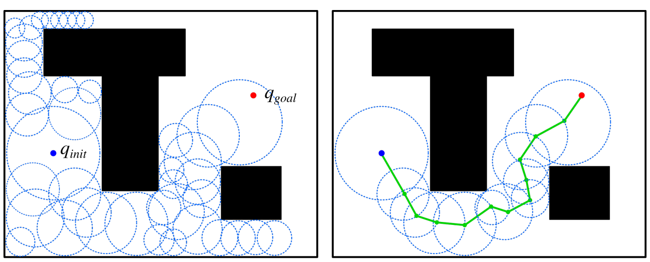
3.1 Foam Propagation

The probabilistic foam F is a structure defined as a set of bubbles, $F = \cup b$, constructed incrementally from the b_{init} (i.e., bubble centered in q_{init}) to b_{goal} (i.e., bubble that encircles q_{goal}) in the free space. This propagation occurs by expanding children bubbles on the surface of parent bubbles from the previous generation. The maximum number N of children bubbles for each parent bubble is defined by:

$$N = K\left(\left\lfloor \frac{r}{r_{\min}} \right\rfloor\right)^{n-1} \tag{2}$$

where n is the dimension of $\mathcal{C}$, r is the radius of the parent bubble and r_{min} is the radius of the smallest allowed bubble. K is a constant value related to the maximum number of children bubbles for the parent bubble with radius r_{min}.

A total of N random configurations are sampled on the surface of the parent bubble. Bubbles sampled in regions covered by other bubbles are removed. Besides, a child bubble can be expanded only if its radius is $r \geqslant r_{min}$. The valid bubbles are stored in a search tree F and added in a queue Q. The choice of a parent bubble in the foam is according to the first-in-first-out (FIFO) rule. The foam propagates until a child bubble encircles q_{goal}. Figure 4a shows the probabilistic foam for a space $n = 2$. When some bubble encloses q_{goal}, the algorithm finishes and a structure called Rosary can be found, as shown in Figure 4b.



(a) **(b)**

Figure 4: (a) Probabilistic foam. (b) Extracted rosary and found path.

The Rosary $\mathcal{R}$ is found by performing a descending search from the bubble b_{goal} to b_{init} through the overlapping region between two adjacent bubbles called *hatch*, and following the parentship relation between bubbles b_{i} with $\{\text{i} \in \mathbb{N}/1 \leqslant \text{i} \leqslant k\}$.

A feasible path can be extracted from the rosary by linking line segments through the center of the bubbles, as can be seen in Fig. 4b (green line). If the path crosses the intersection region between all consecutive bubbles, the path is safe. In this way, path smoothing strategies need to improve the path maintaining these safety constraints.

4. PATH SMOOTHING AS AN OPTIMIZATION PROCESS

In order to model the path smoothing as an optimization problem, we need to deal with two main issues: The decision variable and the objective function. Consider the rosary and path shown in Figure 5.

The blue region in the rosary represents the intersection between two consecutive bubbles b_i and b_{i+1}, which represents the constraints of the problem.

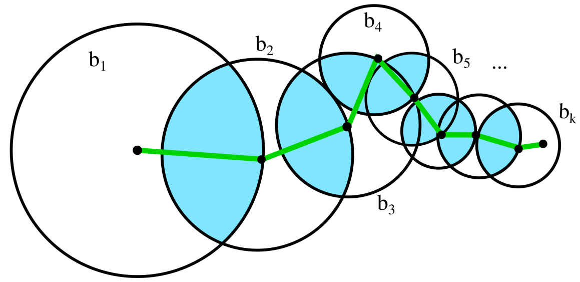


Figure 5: Illustration of a rosary and the extracted path.

4.1 Objective Function

In the optimization process the path must cross all bubbles intersections, and the path σ is formed by the configurations q', as can be seen in Figure 6.

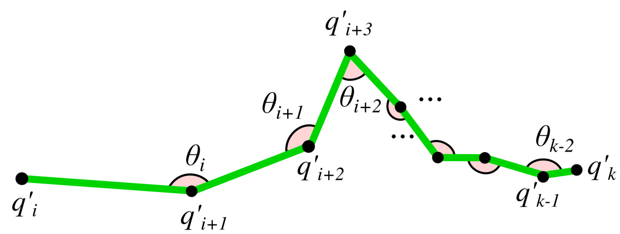


Figure 6: Illustration of the extracted path and the angles between the line segments.

In this case, the decision variable of this optimization process is given by:

$$\sigma = \{q'_1,\ q'_2,\ \ldots,\ q'_{k-2},\ q'_{k-1}\} \tag{3}$$

where k is the number of bubbles in the rosary.

In order to evaluate each path, we used the angles θ_i between line segments, as shown in Figure 5. We consider that the pair of line segments are smoothest when the angle value is $\theta_i = \pi$. Besides, for the special case of the orthosis, we also consider that it is important to decrease the path length σ_{length}. In this way, given the angles between the line segments along the path,

$$\theta_i = \cos^{-1}\frac{(q'_i - q'_{i+1})\cdot(q'_{i+2} - q'_{i+1})}{\left|q'_i - q'_{i+1}\right|\left|q'_{i+2} - q'_{i+1}\right|} \tag{4}$$

the objective function for this problem is:

$$\begin{aligned} &\min\ f(\sigma) = \frac{\sum_{i=1}^{k-2}(\pi - \theta_i)}{k-2} + \sigma_{length} \\ &\text{such that:} \\ &\quad q'_i \in (b_i \cap b_{i+1}) \end{aligned} \tag{5}$$

where $b_i \cap b_{i+1}$ is the *hatch* region between two consecutive bubbles. The minimization of the function $f(\sigma)$ indicates that the angles θ_i are getting closer to π, resulting in smoother paths. Moreover, using the σ_{length} information, the paths also become shorter.

4.2 Strategies used to implement the GA

In this paper, it was considered real-value encoding in the Genetic Algorithm implementation. The selection process was performed using Roulette selection, where it was assigned a higher probability to the best evaluated individuals, i.e, the ones with low values of $f(\sigma)$.

The Heuristic Crossover [21, 6] was used in the reproduction process. In this technique, a new individual σ_f is generated combining two individuals σ_p^1 and σ_p^2. Considering that $f(\sigma_p^1) \leqslant f(\sigma_p^2)$, the individual σ_f is created based on the following equation:

$$\sigma_f = \sigma_p^1 + \beta(\sigma_p^1 - \sigma_p^2) \tag{6}$$

where β is a random uniform value such that $\beta \in [0, 1]$.

In the mutation, the alleles of an individual (the configurations $\{q_i', q_{i+1}', ..., q_{k-1}'\}$ (as shown in (5)) were modified using a subtle adjustment. Considering α a random uniform value between $[a, b]$, i.e., $\alpha \in [a, b]$, q_i' was adjusted by $\alpha q_i'$. Furthermore, after modifying a point, we verify whether it still belongs to the hatch. If not, the point returns to its previous value. Finally, elitism was implemented by replacing the worst rated individual with the best individual of the previous generation.

5. RESULTS

In order to perform the path planning using the Probabilistic Foam method for the scenario where the exoskeleton must overcome a simple obstacle (Figure 3a), we used the parameters $K = 10$ and $r_{min} = 0.07$. The initial configuration was $q_{init} = [0.2\ 0.3\ 0.08]$ rad, and the goal configuration was $q_{goal} = [0.45\ 0.4536\ 1.45]$ rad. The simulation results of path planning with PFM are shown in Figures 7 and 8. The PFM generated a probabilistic foam with a high number of bubbles, as shown in Figure 7.

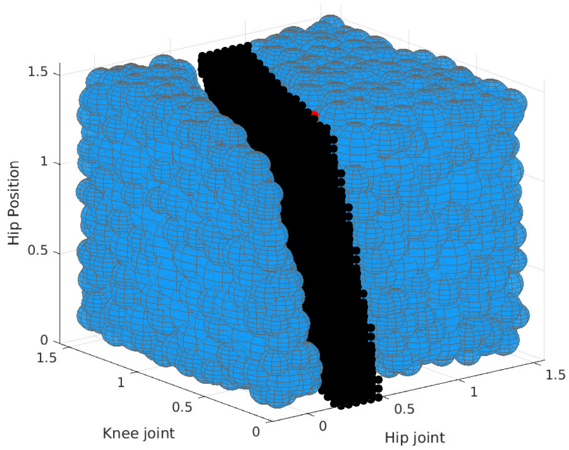
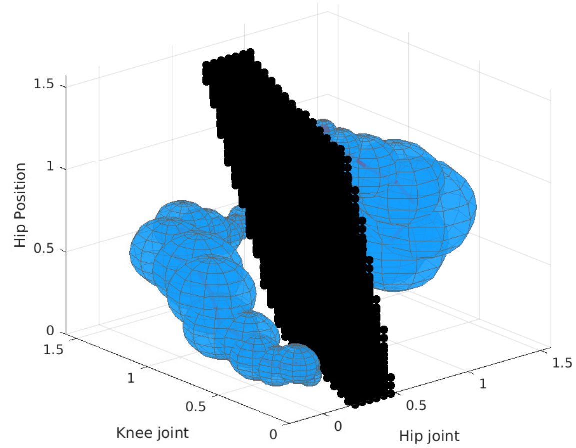


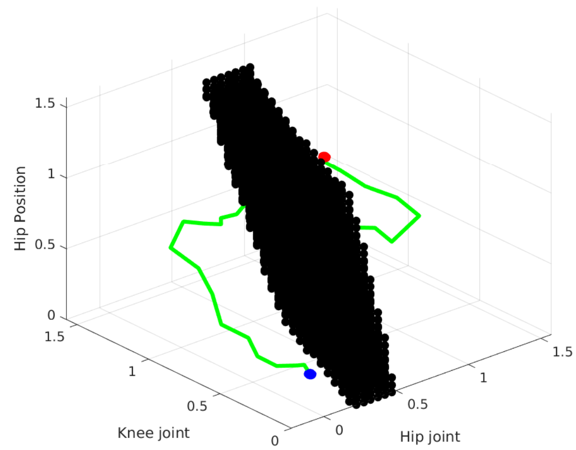
Figure 7: probabilistic foam.

From Figure 7, it was possible to obtain a rosary, as can be seen in Figure 8a. A path could be extracted from this rosary by linking the center of each bubble by line segments, as shown in Figure 8b.

The path provided by the probabilistic Foam method generated a motion that overcomes the obstacle, as can be seen in Figure 9.



(a)



(b)

Figure 8: Path planning for the exoskeleton. (a) The rosary found by the probabilistic foam. (b) The path extracted from rosary.

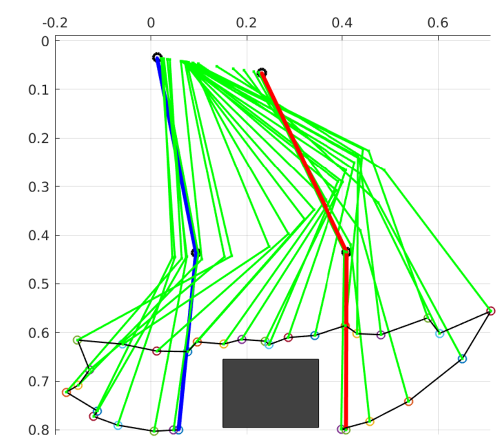


Figure 9: Leg motion obtained by PFM.

Notice that, the movement performed is neither smoothed nor anthropomorphic, which hampers the motion of the exoskeleton user. Therefore, two strategies were used to smooth the original path, the methods HS and GA.

5.1 Results with Harmony Search Algorithm

Using the optimization technique HS, it was possible to improve the path, as can be seen in Figure 10. The HS parameters were $HMCR = 0.99$, $HMS = 50$, $PAR = 0.15$, $BW = 0.01$. These values were obtained empirically. Furthermore, we also considered a stop criteria when achieving a fitness value less than 0.38, i.e, $f(\sigma) < 0.38$.

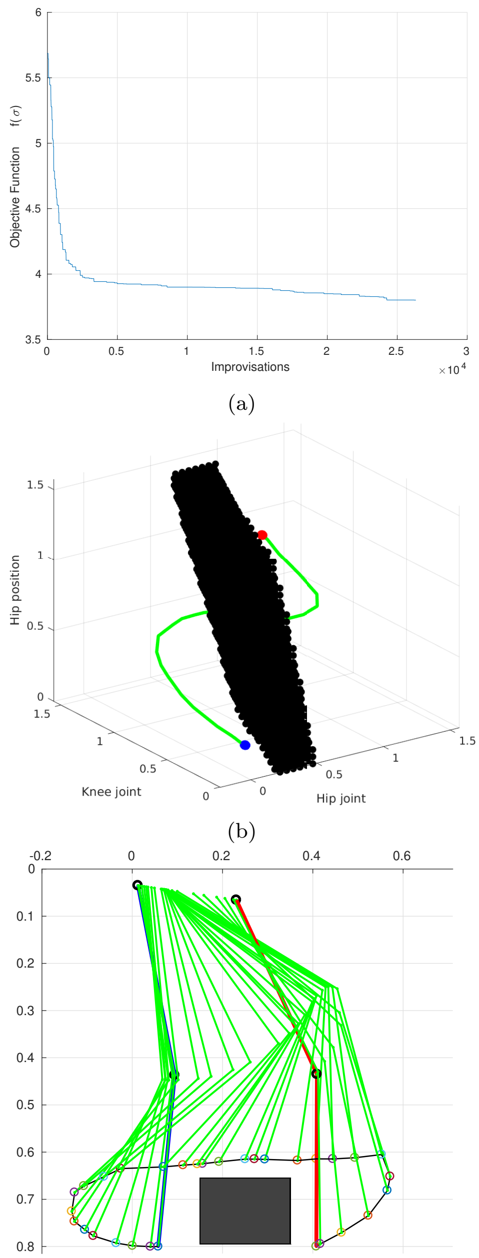


Figure 10: Path smoothing result using HS. (a) Convergence curve for the optimization. (b) Smoothed path by HS. (c) Performing of the improved motion.

The graph shown in Figure 10a indicates the value of the f function related to the best harmony in HM for each improvisation. Figure 10b shows the path related to the best improvised harmony and Figure 10c illustrates the motion provided by this optimized path, which results in a smoother and shorter path when compared to the original one generated by the PFM.

5.2 Results with Genetic Algorithm

The GS parameters used in this paper were defined as follows: crossover probability = 0.9, mutation probability = 0.001, population size = 50 and $\beta = 0.3$. Additionally, we used the same stop criteria which was applied in the HS ($f(\sigma) < 0.38$). The smoothing results obtained with the Genetic Algorithm can be seen in Figure 11.

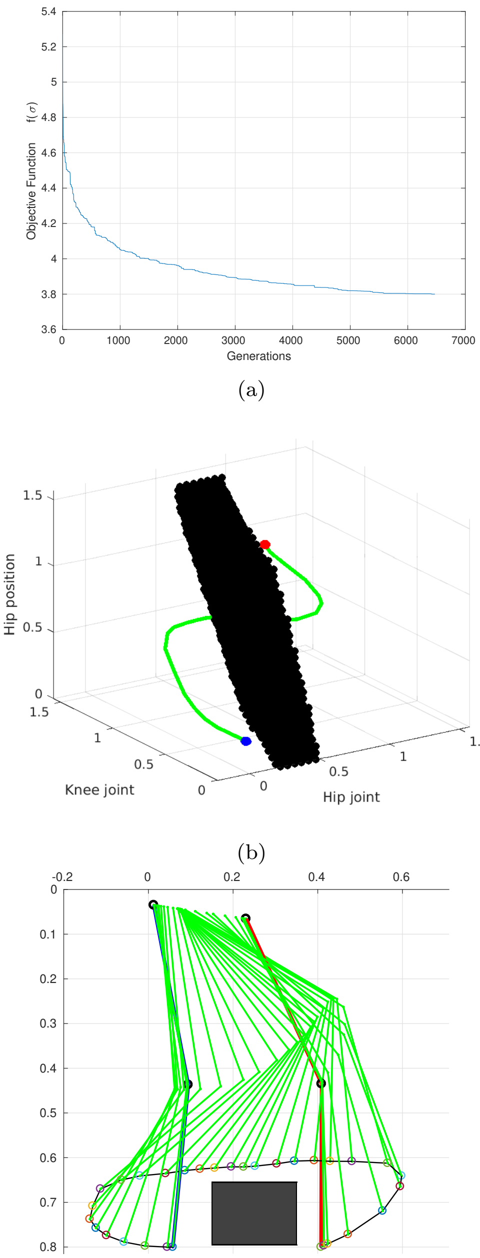


Figure 11: Path smoothing result using GA. (a) Convergence curve for the optimization. (b) Smoothed path by GA. (c) Performing of the improved motion.

Table 1: Comparison for the average values over the best $f(\sigma)$ resulted for 50 runs.

Method	Time (s)				$f(\sigma)$				Iterations			
	Mean	Max	Min	Std	Mean	Max	Min	Std	Mean	Max	Min	Std
HS	17.229	23.432	10.207	4.0113	4.0198	4.1505	3.9115	0.067772	4206.3	6385	3034	962.95
GA	27.112	39.253	16.7	6.3739	3.9307	3.985	3.8576	0.035699	3718.6	5510	2287	935.18

The smooth path obtained in the HS has a length $\sigma_{length} = 3.6155$, while the GA presented a length $\sigma_{length} = 3.6007$. Therefore, even though the paths obtained by HS and GA use the same fitness value in the stop criteria, the path provided by the GA is shorter and also smoother.

5.3 Comparison

To decide which method has the best evaluation, we performed another set of simulations considering the population variability as the new stop criteria. The algorithm must stop when no improvement in the average value of the population fitness is met considering 200 consecutive iterations.

Considering the stochastic nature of this method, the algorithms were subjected to 50 runs in order to analyze the $f(\sigma)$, the processing time (t), and the number of iterations. The simulations with the algorithms were performed on a 1.8 GHz Intel Core i7 processor with 8 GB RAM on Ubuntu 16.04 operating system. The numerical values are shown in Table 1.

From Table 1, we can conclude that the HS provided the best processing time. However, the Genetic Algorithm achieved the best solutions, since $f(\sigma)$ has a lower average value. Besides, the small standard deviation in the $f(\sigma)$ values indicates that even though HS and GA are stochastic processes, the methods tend to provide acceptable convergences for the studied scenario.

In this way, the presented approach provided smoother paths and, consequently, more anthropomorphic motions.

6. CONCLUSIONS

This work presented a strategy based in an optimization by metaheuristics to smooth paths obtained by the path planning method Probabilistic Foam (PFM). The algorithms HS and GS were tested for improve the paths. The methodology presented some satisfactory results, where a path generated by PFM was successfully smoothed. This approach generated improved paths, which provided more anthropomorphic motions for the exoskeleton. Besides, the smoothed path kept bounded by the bubbles from rosary, which means that the path remained safe.

A drawback of this method for the studied scenario is that it was necessary a high number of iterations from both methods to provide acceptable results. However, it was implemented the basic version of the algorithms and perhaps, improved variants of these strategies can deal with this issue. An inherent problem with the Probabilistic Foam Method is that if the rosary has bubbles with irregular size or are placed in a disorganized way, the smoothness process will be more complicated.

In this way, in future works we intend to find strategies to improve the convergence process for harmony search. Besides, a study about the computational effort for this approach is necessary. Finally, we also intend to develop an approach to adjust the bubbles from rosary, improving the smoothness and length of the path.

7. ACKNOWLEDGMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

8. REFERENCES

[1] C.-S. Chen, J.-X. Chang, and S.-K. Chen. Collision avoidance path planning for the 6-dof robotic manipulator. In *Proceedings of the International Conference on Artificial Intelligence and Robotics and the International Conference on Automation, Control and Robotics Engineering*, page 2. ACM, 2016.

[2] X. Chen and Y. Li. Smooth path planning of a mobile robot using stochastic particle swarm optimization. In *2006 International Conference on Mechatronics and Automation*, pages 1722–1727. IEEE, 2006.

[3] L. B. P. do Nascimento, D. da Silva Pereira, A. Sanca Sanca, K. J. Silva Eugenio, D. H. da Silva Fernandes, P. Javier Alsina, M. Valério Araujo, and M. Rabello Silva. Safe path planning based on probabilistic foam for a lower limb active orthosis to overcoming an obstacle. In *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*, pages 413–419, Nov 2018.

[4] Z. W. Geem, J. H. Kim, and G. V. Loganathan. A new heuristic optimization algorithm: harmony search. *simulation*, 76(2):60–68, 2001.

[5] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989.

[6] F. Herrera and M. Lozano. Heuristic crossovers for real-coded genetic algorithms based on fuzzy connectives. In *International Conference on Parallel Problem Solving from Nature*, pages 336–345. Springer, 1996.

[7] J. H. Holland. *Adaptation in Natural and Artificial Systems: an introductory analysis with applications to biology, control, and artificial intelligence*. University of Michigan Press, Ann Arbor, MI, 1975. second edition, 1992.

[8] M.-Y. Ju and C.-W. Cheng. Smooth path planning using genetic algorithms. In *2011 9th World Congress on Intelligent Control and Automation*, pages 1103–1107. IEEE, 2011.

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

[10] Y.-H. Kim, Y. Yoon, and Z. W. Geem. A comparison study of harmony search and genetic algorithm for the max-cut problem. *Swarm and evolutionary*

computation, 44:130–135, 2019.

[11] J.-C. Latombe. *Robot Motion Planning*. Springer US, Boston, MA, 1991.

[12] S. M. Lavalle. Rapidly-exploring random trees: A new tool for path planning. Technical report, Computer Science Dept., Iowa State University, 1998.

[13] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, New York, NY, USA, 2006.

[14] H. Lee, H. Kim, and H. J. Kim. Path planning and control of multiple aerial manipulators for a cooperative transportation. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2386–2391. IEEE, 2015.

[15] N. B. Melo, C. E. Dorea, P. J. Alsina, M. V. Araujo, and M. Gloger. Ortholeg 2.0-design of a transparent active orthosis. *IEEE Latin America Transactions*, 15(10):1869–1874, 2017.

[16] L. Nascimento, V. Santos, D. Pereira, and P. Alsina. A path smoothing strategy based on harmony search algorithm for probabilistic foam. In *Anais da VII Escola Regional de Computação Aplicada à Saúde*, pages 211–216, Porto Alegre, RS, Brasil, 2019. SBC.

[17] H. Piarehzadeh, A. Khanjanzadeh, and R. Pejmanfer. Comparison of harmony search algorithm and particle swarm optimization for distributed generation allocation to improve steady state voltage stability of distribution networks. *Research Journal of Applied Sciences, Engineering and Technology*, 4(15):2310–2315, 2012.

[18] S. Quinlan. *Real-Time Modification of Collision-Free Paths*. Doctoral Dissertation, Stanford University Stanford, CA, 1995.

[19] A. Ravankar, A. Ravankar, Y. Kobayashi, Y. Hoshino, and C.-C. Peng. Path smoothing techniques in robot navigation: State-of-the-art, current and future challenges. *Sensors*, 18(9):3170, 2018.

[20] Y. S. Silveira and P. J. Alsina. A New Robot Path Planning Method Based on Probabilistic Foam. In *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)*, pages 217–222, Oct. 2016.

[21] A. H. Wright. Genetic algorithms for real parameter optimization. In *Foundations of genetic algorithms*, volume 1, pages 205–218. Elsevier, 1991.

[22] X.-S. Yang. Harmony search as a metaheuristic algorithm. In *Music-inspired harmony search algorithm*, pages 1–14. Springer, 2009.