STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **11**, January 2023, pp 116–124.
Published online in International Academic Press (www.IAPress.org)

Visual servoing of a 3R robot by metaheuristic algorithms

Inssaf HARRADE, Achraf DAOUI, Zakaria CHALH, Mhamed SAYYOURI*

National School of Applied Sciences, Sidi Mohamed Ben Abdellah-Fez University, Engineering Systems and Applications Laboratory, Fez, Morocco

Abstract In planar robot control, the singularity problem is frequently encountered when determining the geometric or kinematic robot model. To overcome this problem, a new method is presented in this paper. Indeed, this method is used to compute the joint positions of 3R robots with high accuracy. The main idea of the proposed method is to use a metaheuristic algorithm for retrieving the coordinates of four reference points in the object image that is captured by a hand-eye camera of the 3R robot. Then, the joint positions are estimated by using a metaheuristic algorithm. The resulting positions are then used for controlling the 3R robot for moving to the desired position. The simulation experiments are conducted by using several metaheuristic algorithms with the same population size (N = 400). The obtained results show the high accuracy of the proposed method in terms of determining the exact positions of the 3R robot joints, which leads to finding the optimal robot trajectory.

Keywords Metaheuristic algorithms, 3R robot, joint position, visual servoing, optimal trajectory.

DOI: 10.19139/soic-2310-5070-1552

1. Introduction

From a certain point of view, visual servoing is an alternative to traditional control techniques; it is a new technology that consists in integrating into the control loop of robots visual information extracted from images provided by onboard or off-board vision sensors (camera) to control the movement of a robot to perform the desired action. Inoue and Shirai developed the first use of the closed-loop vision concept in 1973[1, 2], who described how a vision sensor could increase the accuracy of positioning in robotics.
The goal is to localize the dynamic system (usually a robot) concerning objects with unknown locations in its workspace and track moving objects between the design and the object. In recent years, visual servoing has played a vital role in the industry (especially robotics).

Robots are defined by their inverse geometric model from mathematical functions with several unknown and complex variables, posing the singularity problem [3]. This problem can generate an infinite number of solutions or no solutions.
Our study case figures two sign solutions (bend up or bend down) that determine the desired bend configuration[4, 5]. For this reason, researchers have proposed several methods to overcome this problem [6, 7].

The singularity problem is commonly encountered in the modeling of planar robots, especially in the computation of their inverse geometric model or inverse Jacobian. There are two singularities; these singular

*Correspondence to: Inssaf HARRADE (Email: inssaf.harrade@usmba.ac.ma). National School of Applied Sciences, Sidi Mohamed Ben Abdellah-Fez University, Engineering Systems, and Applications Laboratory (LISA). Avenue My Abdallah Km 5, Route d'Imouzzer, Fez BP 72, Morocco.

ISSN 2310-5070 (online) ISSN 2311-004X (print)

configurations for the inverse geometric model are also distinct for the robot's inverse Jacobian, limiting the interior of the robot's workspace. To overcome this problem, we propose in this paper a novel method for controlling a robot by using metaheuristic algorithms. Moreover, these algorithms reduce the computational complexity, minimize the robot control block diagram and find the optimal trajectory. In addition, the proposed method is simple to implement, effective in finding the values of the joint angles, fast to implement using some metaheuristic algorithms, and it only requires the specification of the joint angle intervals.

The rest of this document is organized as follows. Section 2 presents a brief overview of the 3R robot. In section 3, we offer the proposed visual kinematic block diagram serving. The proposed method based on metaheuristic algorithms is presented in section 4. In section 5, the results of the performed numerical simulations are presented. Finally, section 6 concludes the paper.

2. 3R Robot modeling

This paper proposes using a simple planar robotic system modeled according to the Denavit-Hartenberg (D-H) convention [8] as shown in Figure 1. This planar robot has three parallel axes of rotation. It is a robot with 3 degrees of freedom: 2 for the positioning and one for the orientation of the gripper (end device). The camera is connected to the end device. The reference $\mathbb{R}_c(x_c, y_c, z_c)$, is linked to the camera with its origin is located at the optical center, as shown in Fig. 1. The 3R robot uses positioning control, i.e., it uses the data provided by the camera, which has the following intrinsic parameters: $G_x = G_y = 1000, u_0 = v_0 = 0$ in pixels.

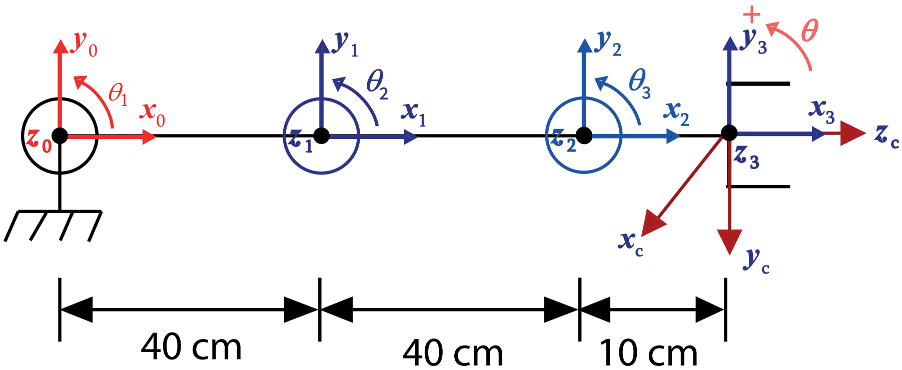

Figure 1. Modilization of the 3R plan robot.

The object consists of four coplanar points arranged in a square. The plane of the thing is perpendicular to the plane of the robot and perpendicularly intersects the viewing plane of the robot (Fig. 1). The thing is located at the yellow mark (Fig. 5). When the visual servoing converges, the depth along the z_c axis of the reference frame of all object points is 0.3 m.

3. Visual kinematic control

In this section, we briefly present the visual servoing. Vision control or visual servoing consists in controlling a robot in a closed loop by integrating the information provided by the camera.

3.1. Classic visual servoing techniques

Visual servoing techniques generally require a prior mapping of visual primitives (initialization) and their follow-up throughout the servoing phase. These techniques are effective for many applications but encounter some difficulties [9].
The significant difficulty that limits the potential of these techniques is the use of geometric primitives (points, lines, curves, ...).

3.2. Vision sensor configuration

The vision sensor configurations are defined by the following two forms [9] :

- **On-board sensor configuration:** Its name indicates that a visual sensor is glued to the robot's terminal organ. This configuration is called "eye-in-hand" in the literature.
- **Remote sensor configuration:** For this "eye-to-hand" configuration, the vision sensor is not on the robots. It is located in the robot work environment.

3.3. The quantities to be controlled

Different quantities can be considered input to the visual servo control block diagram. They can be of a three-dimensional 3D nature. A measure of the object poses expressed in the camera frame, or from 2D type when they directly represent geometric vision primitives extracted from the 2D image plane, Or can be 2D1/2 type that takes advantage of 3D and 2D representations [10].

3.4. The robot control architecture

There are mainly three types of controllers that can control a robot with looped visual feedback [11]:

- **Sequence control:** *It is used when the imaging device cannot provide images at an acquisition rate higher than 1 Hz. This type then performs a positional control of the joint coordinates. If necessary, one can repeat this sequence until reaching the visual servoing goal [12].*
- **Kinematic control:** *The motion control is a closed-loop servo control; it includes the periodic sending of joint speed commands to the lower-level controllers of the robot. It is the most common one and if the robots low-level controller can generate and apply the joint position trajectory according to the joint velocity command, it is easy to implement [12].*
- **Dynamic control:** *Dynamic control considers more realistic modeling of the robot behavior [5]. It is based on identifying dynamic parameters (mass, inertia, etc.) to obtain a speed or torque control, which linearizes the system's dynamic behavior.*

3.5. Visual kinematic control

In this work, a kinematic visual servoing is used to control the 3R robot. For this purpose, the block diagram shown in Fig.2 is designed.

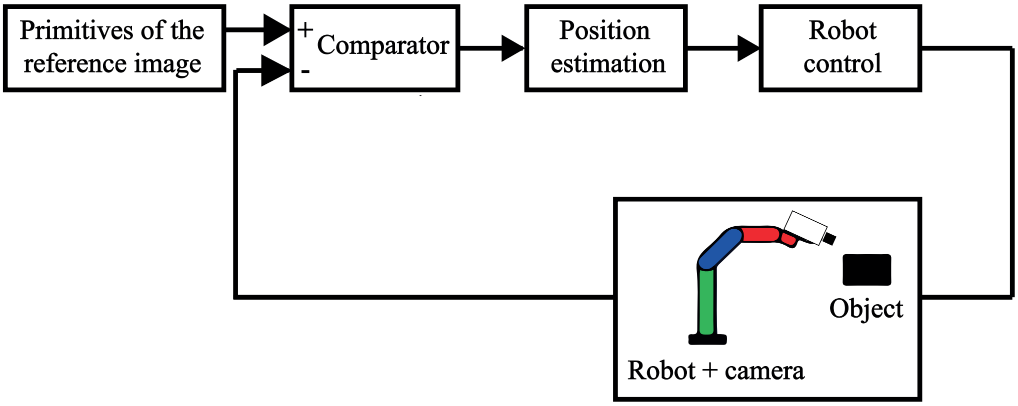

Figure 2. Block diagram of the visual servo.

The instruction is given in pixels: these are the coordinates of the 4 points of the image at the vertices of a square of 200 pixels (Fig.3) on each side. When the observed image converges to the desired one, the camera's optical axis becomes perpendicular to the plane of the target, which is a necessary condition for obtaining a rectangle in the image.
It is worth mentioning that the controllable degrees of freedom of the 3R robot used in this paper is the translation speed along the y_c and Z axes, and the rotation speed around the x_c axis.

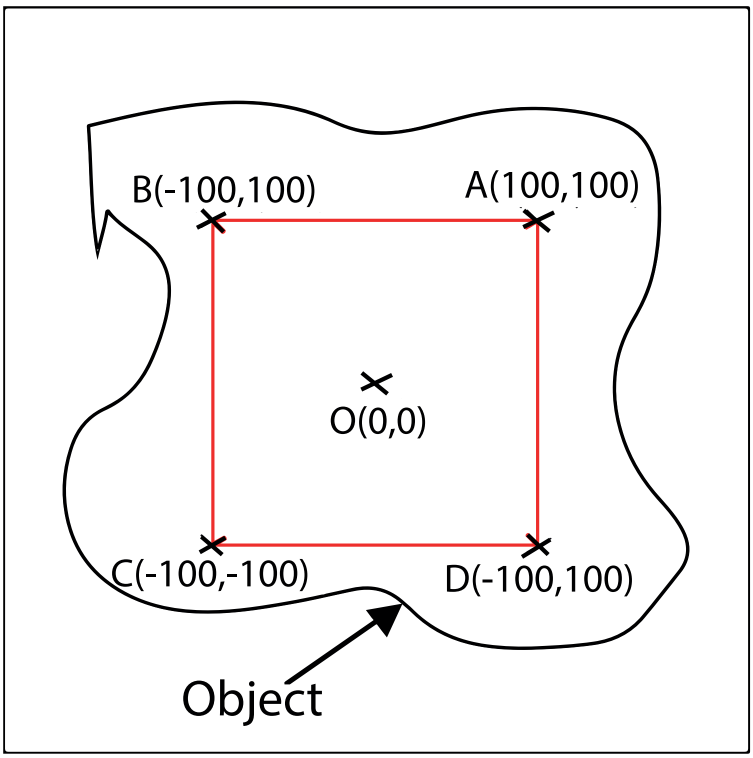

Figure 3. The coordinates of four points of the object in the image.

4. Metaheuristic algorithms for 3R robot control

Metaheuristics is a decision process that is part of nature and human life; it con-stitutes a set of methods that provide good quality solutions in a reasonable time. It can be considered a decision, making a problem to reach a resolution as good as possible, to improve the performance of a thought system. We can assume that each decision attempts to face an optimal or quasi-optimal situation. Generally, MAs evolve towards the global optimum by evaluating objective function. They behave similarly to search algorithms [13],[14] trying to learn the characteristics of the problem to find the approximate value of the best solution. In a technical sense, MAs attempt to combine basic heuristic methods into higher-level frameworks to explore the search space efficiently.

Many MAs range from simple local search to more complex global search algorithms [15]. Indeed, there are nature-based metaheuristics that have been developed. For example, we can refer to Harmony Search (HS) [16], [17], Artificial Bee Colony algorithm (ABC) [18], Teaching Learning Based Optimization (TLBO) [19], Cuckoo Search (CS) [20], are defined as some of the best known recent MAs. The following flowchart(Fig.4) summarizes the general functioning of MAs:

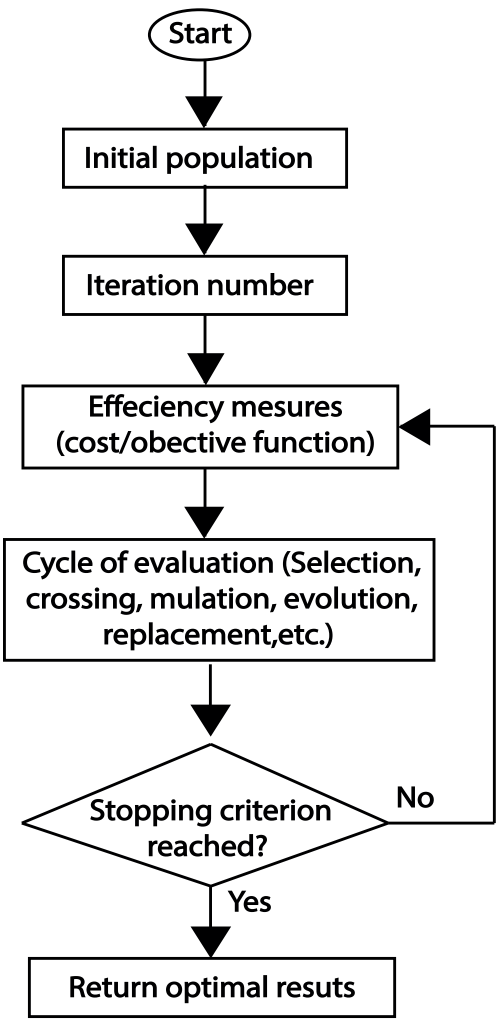

Figure 4. Functioning of metaheuristic algorithms.

The suggested MAs-based algorithm to control the 3R robot outlined in Algorithm 1 a is shown in this section as a pseudo-code that outlines the steps of the proposed approach to detect the desired primitives in the object. In the following equation, the objective function to be evaluated by the MAs is defined:

$$E = \sum_{i=1}^{8} |P_o(i) - P(i)| \tag{1}$$

Where: $P_o(i)$ is the desired primitives and $P(i)$ will be the primitives are computed by the metaheuristic algorithms.

Algorithm 1 (The proposed MAs based algorithm in pseudo-code for 3R robot control.)
Inputs: The coordinates of the four points in the image in pixels(P_o: A(x_A, y_A),B(x_B, y_B),C(x_C, y_C),D(x_D, y_D)), the homogeneous matrix of the 3R robot.

Output: the values of the articulation angles(θ_1,θ_2,θ_3),the estimated primitives (P: Â(x_A, y_A), B̂(x_B, y_B), Ĉ(x_C, y_C),D̂(x_D, y_D)).

Initialization of MAs

SA= 400 (Number of search agents)
IT= 200 (The number of iterations that can be performed at maximum)
Dim=3 (Dimension of our problem): because we're seeking for the articulation angles' values (θ_1,θ_2,θ_3),of the 3R robot.
Lb=[$\frac{-\pi}{2}$, $\frac{-\pi}{2}$,$-\pi$](The minimum values of (θ_1,θ_2,θ_3))
Ub=[$\frac{\pi}{2}$, $\frac{\pi}{2}$,π](The maximum values of (θ_1,θ_2,θ_3))

Initialize a MA set of search agents.

Do
Calculate,for each search agent, the values of Â(x_A, y_A), B̂(x_B, y_B), Ĉ(x_C, y_C) and D̂(x_D, y_D).
Evaluate the objective function for each SA given by Eq.(1).
Run the instructions of each MA.
While (t <IT)
End Do
Return the best solution until the global optimum is obtained.

To evaluate the performance of the proposed Algorithm 1, we simulated our system to obtain the desired results, which are presented in the next section.

5. Simulation and Results

This section presents the simulation results of a 3R robot with a camera embedded at the end-organ using MAs for joint angle optimization. We considered a fixed number of iterations and population for all optimization algorithms used in the performed test in the first part. The second part of the simulation concerns the position of a manipulator robot with three degrees of freedom.

5.1. Optimization of joint nails

In this part, we use some MAs to compute the primitives that allow us to find the optimal position of the robot joint angles and prove the robustness of the proposed algorithm 1.
To compare and accurately interpret the results between these algorithms, we have performed some tests with the same number of population and iteration. So that the computation time is relative to each type of algorithm. Moreover, we have calculated the error in absolute value between the desired primitives and the primitives calculated by Eq.(1).

The following Tab. One illustrates the obtained results using the proposed MAs-based algorithm 1.

Table 1.

Comparison between eight MAs.

Primitives calculated by	Desired primitives(pixels) 100 100 100 -100 -100 -100 -100 100								Errors (E)	Time of calculation (seconds)
SCA	101	101	101	-101	-101	-101	-101	101	8	0,5161
ABC	97	97	97	-97	-97	-97	-97	97	24	130,7343
MFO	100	100	100	-100	-100	-100	-100	100	0	0,5245
$TLBO$	100	100	100	-100	-100	-100	-100	100	0	2,1081
ACO	100	100	100	-100	-100	-100	-100	100	0	2,8609
BA	100	100	100	-100	-100	-100	-100	100	0	0,6455
HS	100	100	100	-100	-100	-100	-100	100	0	1,4599
$Cuckoo$	101	75	102	-128	-102	-128	-101	75	112	25,0629

The results obtained (Table 1.) clearly show that the computation of the primitives by some of these eight algorithms (HS, BA, TLBO, ACO, MFO) is numerically close to the desired primitives, for the 3R robot control. This indicates that the use of MAs overcomes the problem of singularities and the proposed method optimizes the trajectory by optimizing the angles of articulation. The results obtained in Table 1 also show the speed of some MAs in optimizing the circuit (BA, MFO).
When running the simulation, we obtain an animation of the robot and the image seen by the camera represented as shown in Fig.5.
The arm moves towards the object via the onboard virtual camera in the graphical interface, represented by four red dots that move with the arm. The reference target is black and corresponds to the reference indicated at the beginning (reference image primitives). According to the achieved results, the red points converge towards the center of the black issues. This suggests that the control law is effectively applied.
These exciting results justify the efficiency of the proposed MAs method for joint computing angles.

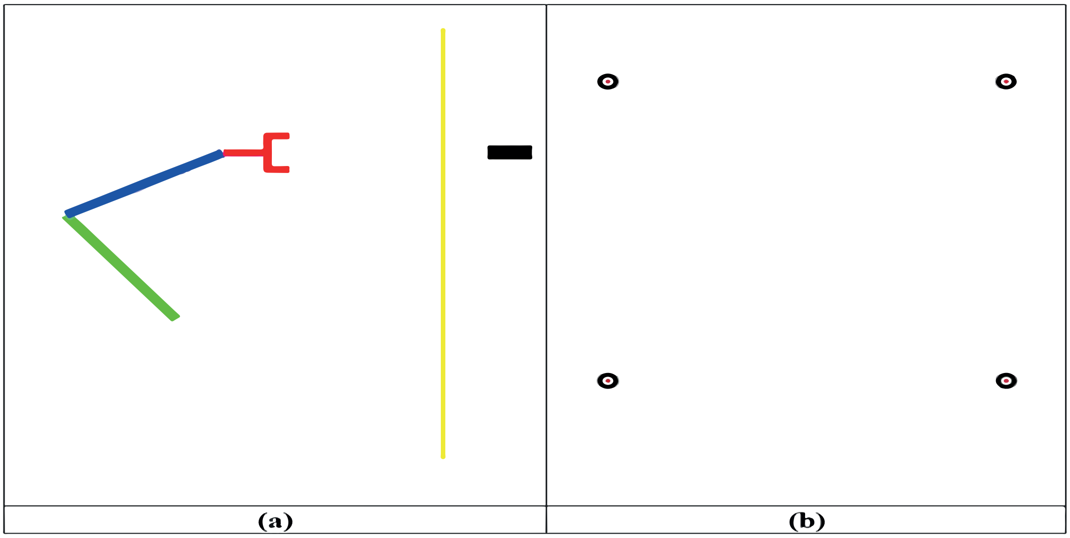

Figure 5. (a): Graphical interface of the robot animation, (b): camera visualization.

6. Conclusion

In this paper, we have employed MAs computed by the primitives for the joint angles. The efficiency of our proposed optimization method is illustrated in terms of computation rate and convergence trajectory of the robot with low errors of finding the desired primitives. The simulations' results justified the proposed computational method over other classical computational methods. The proposed metaheuristic algorithm method will be used in the future for other more complex robot examples, such as robots with 6 degrees of freedom and more.

REFERENCES

1. S.Wang, K.Zhang, G.Herrmann, *An Adaptive Controller for Robotic Manipulators with Unknown Kinematics and Dynamics*, IFAC-PapersOnLine 53:8796–8801. https://doi.org/10.1016/j.ifacol.2020.12.1385, 2020.
2. Y.Shirai, H.Inoue , *Guiding a robot by visual feedback in assembling tasks*, Pattern recognition 5:99–108, 1973.
3. J.Landuré , *Étude du lieu des singularités d'un manipulateur parallèle sphérique redondant*, PhD Thesis, Université Laval, 2017.
4. M.Furet , *Analyse cinétostatique de mécanismes de tenségrité: Application à la modélisation de cous d'oiseaux et de manipulateurs bio-inspirés*, PhD Thesis, Ecole Centrale de Nantes, 2020.
5. A.Liégeois , *Modélisation et commande des robots manipulateurs*, Techniques de l'Ingénieur, S 7:730, 2000.
6. D.Chablat, R.Jha, F.Rouillier, G.Moroz, *Non-singular assembly mode changing trajectories in the workspace for the 3-RPS parallel robot*, In: Advances in Robot Kinematics. Springer, pp 149–159, 2014.
7. D.Chablat, P.Wenger , *Séparation des solutions aux modèles géométriques direct et inverse pour les manipulateurs pleinement parallèles*, Mechanism and Machine Theory 36:763–783, 2001.
8. AC.Reddy , *Difference between Denavit-Hartenberg (DH) classical and modified conventions for forwarding kinematics of robots with a case study*, In: International Conference on Advanced Materials and Manufacturing Technologies (AMMT). JNTUH College of Engineering Hyderabad Chandigarh, India, pp 267–286, 2014.
9. R.Fleurmond , *Asservissement visuel coordonné de deux bras manipulateurs*, PhD Thesis, Universite Toulouse III Paul Sabatier, 2015.
10. AY.Tamtsia , *Nouvelles contributions à l'application des moments en asservissement visuel*, PhD Thesis, Université Blaise Pascal-Clermont-Ferrand II, 2013.
11. IH.Aguilar , *Commande des bras manipulateurs et retour visuel pour des applications à la robotique de service*, PhD Thesis, Thèse de doctorat, Université Toulouse III, 2007.
12. A.Krupa , *Contributions à l'asservissement visuel échographique*, SPhD Thesis, Université Rennes 1, 2012.
13. Z.Li, Q.Chen, V.Koltun, *Combinatorial Optimization with Graph Convolutional Networks and Guided Tree Search*, In Advances in Neural Information Processing Systems. Curran Associates, Inc, 2018.
14. F.Gueye , *Algorithmes de recherche d'itinéraires en transport multimodal*, PhD Thesis, Toulouse, INSA, 2010.

15. T.Dokeroglu, E.Sevinc, T.Kucukyilmaz, A.Cosar , *A survey on new generation metaheuristic algorithms*, Computers & Industrial Engineering 137:106040. https://doi.org/10.1016/j.cie.2019.106040, 2019.
16. X-S.Yang, *Harmony search as a metaheuristic algorithm*, In Music-inspired harmony search algorithm. Springer, pp 1–14, 2009.
17. SO.Degertekin , *Improved harmony search algorithms for sizing optimization of truss structures* Computers & Structures 92:229–241, 2012.
18. D.Karaboga, B.Basturk, *A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm*, Journal of global optimization 39:459–471, 2007.
19. R.Rao , *Review of applications of TLBO algorithm and a tutorial for beginners to solve the unconstrained and constrained optimization problems*, Decision science letters 5:1–30, 2016.
20. AH.Gandomi, X-S.Yang, AH.Alavi, *Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems*, Engineering with computers 29:17–35, 2013.