Improving productivity using a multi-objective optimization of robotic trajectory planning

Carlos Llopis-Albert[1], Francisco Rubio[2], Francisco Valero[2]

[1]Departamento de Ingeniería Mecánica y de Materiales (DIMM)

Universitat Politècnica de València – Camino de Vera s/n, 46022 – Valencia, Spain

Email: cllopisa@upvnet.upv.es

[2]Centro de Investigación en Ingeniería Mecánica (CIIM)

Universitat Politècnica de València – Camino de Vera s/n, 46022 – Valencia, Spain

Emails: frubio@mcm.upv.es; fvalero@mcm.upv.es

Abstract

This study presents a methodology to tackle robot tasks in a cost-efficient way. It poses a multi-objective optimization problem for trajectory planning of robotic arms that an efficient algorithm will solve. The method finds the minimum time to perform robot tasks while considering the physical constraints of the real working problem and the economic issues participating in the process. This process also considers robotic system dynamics and the presence of obstacles to avoid collisions. It generates an entire set of equally optimal solutions for each process, the Pareto-optimal frontiers. They provide information about the trade-offs between the different decision variables of the multi-objective optimization problem. This procedure can help managers in decision-making

processes regarding performing tasks, items to be manufactured or robotic services performed to meet with the current demand, and also, to define an efficient scheduling. It improves productivity and allows firms to stay competitive in rapid changing markets.

Key words: service robots, multi-objective optimization, robotics, trajectory planning, Pareto frontier, trade-offs, scheduling

1. Introduction

An industrial robot is an automatically controlled, reprogrammable, and multipurpose manipulator that industrial automation applications use. A service robot is a robot that operates semi or fully autonomously to perform useful services for humans and equipment. World robots are rapidly growing in number in recent years. Process complexity deriving from automation requires efficient algorithms that control them to provide cost-efficient solutions (e.g., Kelly, Johnson, Dorsey, & Blodgett, 2004). Specifically, in recent years researchers are working hard in the trajectory planning of robot arms (e.g., Chen & Zhao, 2013; Chettibi, Lehtihet, Haddad, & Hanchi, 2002; Cho, Choi, & Lee, 2006; Gasparetto & Zanotto, 2010; Huang, Xu, & Liang, 2006; Suñer et al., 2007; Rubio et al., 2010; Rubio, Llopis-Albert, Valero, & Suñer, 2015). Furthermore, mathematical optimization techniques solve many engineering problems (e.g., Llopis-Albert & Capilla, 2010a, 2010b).

This study presents a new robotic technology to address robotic systems' cost-effectiveness through a multi-objective optimization problem for robotic arm trajectory planning, which an efficient algorithm solves. The method finds the minimum time trajectory to perform robot tasks while considering the physical constraints and the economic issues participating in the process. The methodology also allows analyzing

the trade-off between the different decision variables through the Pareto-optimal frontiers. A solution belongs to the Pareto optimal frontier if an objective does not improve without adversely affecting at least one other objective. This methodology allows an immediate change, a quality improvement of the products, an increase in productivity, and a reduction of cycle times, which may increase opportunities to react to market developments and receptivity. The procedure overcomes the limitations of economic analysis methods that can currently assess robotic systems cost-effectiveness in production lines and robot services.

2. Multi-objective optimization

Many real-world design tasks involve complex multi-objective optimization problems of various competing design specifications and constraints that make a single design highly improbable. Therefore, a trade-off among the conflicting design objectives is necessary. A multi-objective optimization affects several non-commensurable and often competing objectives, cost functions, or performance functions within a feasible decision variable space. This study follows above optimization model because, for example, a minimum time trajectory to produce an item leads to lower costs in energy consumption. Therefore, a trade-off exists between executable time and costs. The multi- objective optimization problem solves the collision-free trajectory-planning problem of robotic arms while considering the economic issues participating in the process. The algorithm, according to previous works (Rubio, Valero, Suñer, & Mata, 2009; Rubio, Valero, Suñer, & Cuadrado, 2012; Rubio et al., 2015; Valero, Mata, & Besa, 2006), returns robot's minimum total traveling time. This time has to do with productivity and flexibility, because it accelerates operation or execution time of the process. Problem constraints are the torque, power, jerk (variables to do with work

quality, accuracy, and equipment maintenance), and energy consumption (related to savings). Optimization problem constraints require a fulfillment because minimum-time algorithms have discontinuous values of acceleration and torques leading to dynamic problems during trajectory performance. The imposition of smooth trajectories can solve the problem by using spline functions in path and trajectory planning. The jerk constraint is crucial for working with precision and without vibrations, and affects control system and joints and bars' wearing. These methods enable the errors, the stresses (in robot's actuators and mechanical structure), and the resonance frequencies to shrink during trajectory tracking.

The economic objective function is the following:

$$\text{Max B} = \frac{1}{(1+r)^T} \left[\sum_{p=1}^{n} (P_p - C_p) \cdot N_p \right] \tag{4}$$

where B is the objective function to be maximized and represents the current value of the net benefit from a generic service task (€) defined as the revenue of the services performed minus total costs; P_p is the market unitary price of the service p (€); r is the annual discount rate; T represents number of years; C_p stands for the unitary cost to perform the service p (€), ranging from costs of raw materials, energy, amortization, labor force, maintenance, taxes to direct and indirect costs; and $N_p(t)$ is a function accounting for the number of services carried out per hour:

$$N_p(t) = \frac{K}{t(S_k)^\mu} \tag{5}$$

Tasks' sets S_k to perform the item or service (p) constitutes the work load, where k represents the number of tasks. The cumulated task time $t(S_k) = \sum_{j \in S_k} t_j$ is called the service time, being K a constant related to the current

number of working hours per year. The parameter μ refers to the economic environment and the market seasonality.

Each one of these tasks is carried out by the robot arm, which uses a certain time to describe the optimal trajectory. As above mentioned, the developed algorithm returns the minimum time to carry out the task of the robot arm in order to perform the service p (t_{minp}), while considering the time of the other tasks as constant. The lower the time used by the robot to perform its task, the greater the number of services carried out per hour. Then, the cumulative time of all tasks can be defined as follows:

$$t(S_k) = t_{minp} + \sum_{j \notin S_{robot}}^{k} t_j \qquad (6)$$

3. Results of the application of the methodology to different examples

This study applies multi-objective optimization methodology to different examples following those by Rubio et al. (2012). This study uses as a model the PUMA 560 robot, which stands for Programmable Universal Machine for Assembly.

Five examples provide positive results with sequences between 32 and 57 intermediate configurations between the initial and final ones, using different physical working environments (see Rubio et al., 2012). The robot uses different working constraint values for each actuator.

Table 1 presents algorithm results, that is, the execution time for the robot to perform robot task trajectory.

Table 1. Execution times (s) for the different examples solved with physical constraints

Case	Execution time (s)	Case	Execution time (s)
1_s_s	3.79	4_5_s	18.28
1_s_75	22.55	4_10_s	14.51
1_5_s	19.27	4_25_s	10.69
1_5_75	25.76	4_5_s	18.28
2_s_s	5.14	4_10_s	14.51
2_s_200	5.15	4_25_s	10.69
2_s_175	5.3	4_50_s	8.49
2_s_150	5.62	4_100_s	6.74
2_s_125	6.42	4_1000_s	3.21
2_s_100	12.25	4_s_s	2.41
2_s_95	21.08	4_5_40	18.65
2_5_s	23.05	4_s_40	9.94
2_5_95	26.35	5_s_s	3.08
3_s_s	2.27	5_s_40	9.18
3_s_50	7.34	5_5_s	15.91
3_5_s	14.82	5_5_40	15.93
3_5_50	17.94		

(Nomenclature used. Case: numberexample_X_Y. The first number indicates the example solved, the X position indicates the value of a physical constraint -jerk- and the Y position indicates the value of energy consumed. Letter s in any position means without that constrain)

Subsequently, the economic issues associated to robot service tasks are analyzed. We suppose a cost of the service considered of 0.8 € (without considering the cost of the energy consumed) and a price of 1€ for the five examples. When the cost of the energy

consumed is considered, the different cases have different costs. A cost of 0.0676 €/kWh has been defined, which has been added to cost of 0.8 €. For reasons of clarity, the service tasks is provided in only one shift of 8 hours (365 working days in a year), and the benefits B are presented for a period of one year. Different number of service tasks performed per year are obtained for each case, because they present different minimum execution times (t_{min}). The time of the other tasks needed to perform the service (i.e., the summation of times shown in Eq. (6), $\sum_{j \notin S_{robot}}^{k} t_j$) has been defined as 90 s. Therefore, the different cases also present different benefits. For instance, the case 3_s_s, which has no constraints in both the jerk and the energy consumed, presents the maximum benefits per year (23243 €). Contrary, the case 2_5_95, with severe physical constraints, shows the minimum benefits (22962 €).

Now we consider that three different services are performed. This exercise is intended to illustrate the loss of benefits on account of not using efficient algorithms. This loss of benefits is represented by the Pareto fronts for three different services. The services differ in their cumulative time to be performed but share the same execution time of the robot arm (t_{minp}). Then the minimum trajectory time for the case 3_s_s is used for all items, i.e., 2.27 s. The cumulative time of the Service 1=90 s; Service 2=100 s; and Service 3=80 s. These services also differ in the total costs (without considering the energy costs), prices and values of the parameter μ, which is intended to simulate different economic environments and market seasonality. Then the total cost of Service 1=0.8 €; Service 2=0.82 €; and Service 3=0.84 €, while the prices are Service 1=1.0 €; Service 2=1.05 €; and Service 3=1.02 €. The parameter μ takes the values for each service of 0.6, 0.5 and 0.55, respectively. In this case, t(S$_k$) has been defined as a cubic function of t_{minp}.

Then, if the market conditions do not change and the efficient algorithm is not used, the minimum trajectory time is not obtained. In this scenario, there is a benefit loss due to the fact that robot arm may present higher execution times. The multi-objective optimization problem allows obtaining the Pareto frontiers, which provides information about the trade-offs of the decision variables. The trade-off between the benefits and the execution time for the case 3_s_s (i.e., the Pareto frontier) is presented in Fig. 1.

Then the algorithm allows quantifying the benefit loss because of no using this robot programming technology. Each solution in the front will have an optimal objective function value, an optimal value of variables and constraints. All constraints will be satisfied by any solution in the Pareto optimal front.

Note that for the cases defined, the differences between their annual energy costs are almost negligible compared with the other costs.
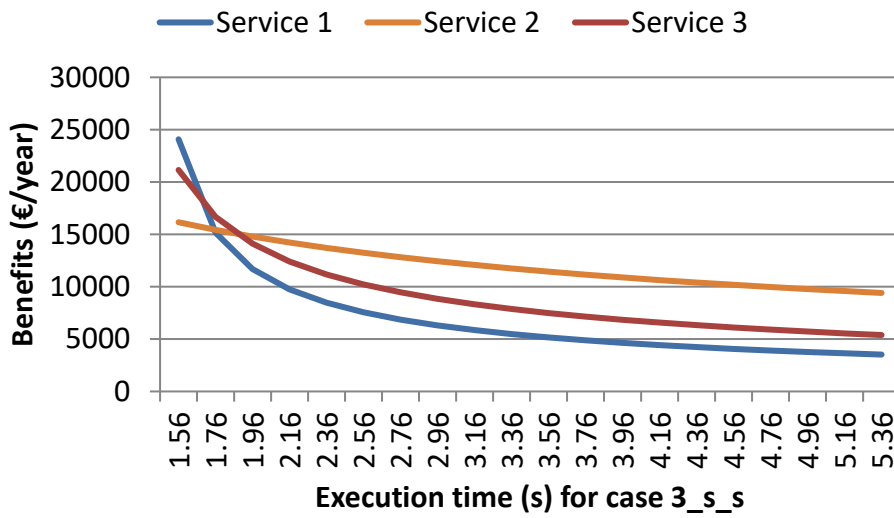

Figure 1. Pareto frontiers for the case 3_s_s when considering different services tasks.

4. Conclusions

This study presents a new robot programming technology applicable to many domestic and professional service robots. It consists of an efficient algorithm, which solves the kinematics and dynamics of robot arms, to obtain minimum time trajectories to perform service tasks subjected to physical constraints while avoiding collisions. This is performed by taking under consideration characteristics of the real working problem and the economic issues involved in the process. The problem has been posed as a multi-objective optimization problem that provides the trade-offs between the decision variables by means of the technique based on Pareto frontiers. With the optimal execution times calculated in a cost-effective manner, the results can be used for improving a wide variety of robot service tasks.

The proposed procedure has been successfully assessed for different examples of service tasks. These examples have proven the worth of the algorithm on account of the higher benefits obtained if compared when it is not applied. Furthermore, the Pareto frontiers of the two conflicting objectives analyzed (benefits and execution times of the service tasks) are illustrated for three different services. They can help managers in the decision making process, regarding which services should be performed, and to define an efficient scheduling of the services. Pareto frontiers allow service firms to stay competitive in rapidly changing markets, which also entail high levels of quality and efficiency. Therefore, the design and planning of the robot service tasks is considerably improved by the proposed algorithm.

References

Chettibi, T., Lehtihet, H.E., Haddad, M. and Hanchi, S. (2002). Minimum cost trajectory planning for industrial robots. Mechanism and Machine Theory, Vol. 37 No. 10, pp. 1063-86.

Cho, B.H., Choi, B.S. and Lee, J.M. (2006). Time-optimal trajectory planning for a robot system under torque and impulse constraints. International Journal of Control, Automation, and Systems, Vol. 4 No. 1, pp. 10-16.

Gasparetto, A. and Zanotto, V. (2010). Optimal trajectory planning for industrial robots. Advances in Engineering Software, Vol. 41 No. 4, pp. 548-56.

Huang, P., Xu, Y. and Liang, B. (2006). Global minimum jerk trajectory planning of space manipulator. International Journal of Control, Automation and Systems, Vol. 4, pp. 405-13.

International Federation of Robotics, IFR (2012). http://www.ifr.org

Rubio, F.J., Valero, F.J., Suñer, J.L. and Mata, V. (2009). Direct step-by-step method for industrial robot path planning. Industrial Robot: An International Journal, Vol. 36(6), 594-607.

Rubio, F.J., Valero, F.J., Suñer, J.L., Garrido, A. (2010). The simultaneous algorithm and the best interpolation function for trajectory planning. Industrial Robot: An International Journal, Vol. 37(5), 441-451.

Rubio, F.J., Valero, F.J., Suñer, J.L., Cuadrado, J. (2012). Optimal time trajectories for industrial robots with torque, power, jerk and energy consumed constraints. Industrial Robot: An International Journal, Vol. 39(1), 92–100.

Suñer, J.L., Valero, F.J., Ródenas, J.J. and Besa, A. (2007). Comparison between settlement procedures of spline interpolation for path planning of industrial robots. VIII Iberoamerican congress on mechanical engineering, Cusco (Peru).

Valero, F.J., Mata, V., Besa, A. (2006). Trajectory planning in workspaces with obstacles taking into account the dynamic robot behavior. Mechanism and Machine Theory (ISSN 0094-114X) V 41, 525–536.