Expert Systems with Applications 39 (2012) 7895–7907

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Multiobjective evolutionary algorithms for multivariable PI controller design

Gilberto Reynoso-Meza *, Javier Sanchis, Xavier Blasco, Juan M. Herrero

Grupo de Control Predictivo y Optimización Heurística (CPOH), Instituto Universitario de Automática e Informática Industrial, Universitat Politècnica de València, Camino de Vera 14, 46022-Valencia, Spain

A R T I C L E I N F O

Keywords:
Multiobjective optimisation
Controller tuning
PID tuning
Multiobjective evolutionary optimisation
Decision making

A B S T R A C T

A multiobjective optimisation engineering design (MOED) methodology for PI controller tuning in multivariable processes is presented. The MOED procedure is a natural approach for facing multiobjective problems where several requirements and specifications need to be fulfilled. An algorithm based on the differential evolution technique and spherical pruning is used for this purpose. To evaluate the methodology, a multivariable control benchmark is used. The obtained results validate the MOED procedure as a practical and useful technique for parametric controller tuning in multivariable processes.

1. Introduction

PI and PID controllers currently represent a reliable digital control solution because of their simplicity and efficacy (Åström & Hägglund, 2005). They are often used in industrial applications and there is ongoing research on new techniques for robust tuning in single-input single-output (SISO) systems, as well as multiple-input multiple-output (MIMO) systems. MIMO systems are very common in industrial processes, and their complexity relies on the dynamic interaction between inputs and outputs.

New PI-PID controller tuning techniques mainly search for a trade-off solution among several control and operational requirements. Some approaches state the design problem as an analytical/numerical optimisation procedure (Astrom, Panagopoulos, & Hagglund, 1998; Ge, Chiu, & Wang, 2002; Goncalves, Palhares, & Takahashi, 2008; Panagopoulos, Astrom, & Hagglund, 2002; Toscano, 2005), or as an evolutionary optimisation statement (Iruthayarajan & Baskar, 2009, 2010; Kim, Maruta, & Sugie, 2008). In both cases, a variety of specifications with several requirements and specifications must be faced. Such problems involving multiple objectives are known as multiobjective problems (MOP).

In an MOP, the designer (control engineer) has to deal with a list of requirements and searches for a solution with a desired trade-off (preferences) among objectives. A traditional approach to handle preferences in an MOP is to translate it into a single-objective problem using weighting factors. More elaborate methods have been developed (Marler & Arora, 2004), such as goal programming, lexicographic methods, physical programming (Messac, Gupta, & Akbulut, 1996), and recently, global physical programming (Martínez, Sanchis, & Blasco, 2006, Sanchis, Martínez, Blasco, & Reynoso-Meza, 2010).

Multiobjective optimisation (MOO) can handle these issues in a simpler manner because of its simultaneous optimisation approach. In MOO, all of the objectives and constraints are significant from the designer's point of view. Consequently, each is optimised to obtain a set of optimal non-dominated solutions. In this set of solutions, no solution is better than the others in every objective – but each solution offers different balances between design objectives. As a result, the decision maker (DM) can obtain a better insight into the trade-off for different solutions and can analyse the tendencies. This approach produces more information for selecting the most preferable solution that meets the DM's preferences.

The difficulty involved in the PI-PID tuning process based on optimisation increases when:

- MIMO systems are considered instead of SISO systems.
- The number of engineering requirements (objectives) increases.
- MOO is required instead of single objective optimisation.
- Constrained problems are treated instead of unconstrained problems.

It is, therefore, worthwhile searching for new algorithms and strategies to tackle constrained MOO for PI-PID tuning in multivariable processes. Therefore, this paper proposes an MOP statement for constrained MIMO PI tuning that demonstrates its viability in an easy and intuitive way. This is fulfilled by defining the MOO statement with well-known performance indexes and a graphical visualisation of the Pareto front. This is a very important issue since the DM requires a useful and interpretable approximation for the decision making stage.

The remainder of this paper is organised as follows: in Section 2 a review of MOO is presented; in Section 3 a multiobjective optimisa-

* Corresponding author. Tel.: +34 963877007; fax: +34 963879579.
E-mail address: gilreyme@posgrado.upv.es (G. Reynoso-Meza).
URL: http://ctl-predictivo.upv.es (G. Reynoso-Meza).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2012.01.111

tion engineering design (MOED) methodology for multivariable PI controller tuning is explained. In Section 4 the MOED methodology is evaluated in a multivariable benchmark process. Finally, some concluding remarks are given.

2. Multiobjective optimisation review

An MOP, without loss of generality,[1] can be stated as follows:

$$\min_{\theta \in \Re^n} \boldsymbol{J}(\boldsymbol{\theta}) = [J_1(\boldsymbol{\theta}), \ldots, J_m(\boldsymbol{\theta})] \in \Re^m \tag{1}$$

where $\boldsymbol{\theta} \in \Re^n$ is defined as the decision vector and $\boldsymbol{J}(\boldsymbol{\theta})$ as the objective vector (see Fig. 1). A unique solution does not generally exist for an MOP because no solution is better than the others for all the objectives. Let $\boldsymbol{\Theta_P}$ be defined as the Pareto set, or set of solutions of the MOP, and $\boldsymbol{J_P}$ be defined as the Pareto front or the projection of $\boldsymbol{\Theta_P}$ in the objective space. Each point in the Pareto front is said to be a non-dominated solution (see Fig. 2).

Definition 1 (*Dominance relation*). Given a solution $\theta^{\mathbf{1}}$ with objective vector $J(\theta^{\mathbf{1}})$ dominates a second solution $\theta^{\mathbf{2}}$ with objective vector $J(\theta^{\mathbf{2}})$ if and only if:

$$\{\forall i \in [1, 2, \ldots, m], J_i(\boldsymbol{\theta^1}) \leqslant J_i(\boldsymbol{\theta^2})\}$$
$$\wedge$$
$$\{\exists q \in [1, 2, \ldots, m] : J_q(\boldsymbol{\theta^1}) < J_q(\boldsymbol{\theta^2})\}$$

which is denoted as $\theta^{\mathbf{1}} \prec \theta^{\mathbf{2}}$.

Two useful vectors can be defined: the ideal solution $\boldsymbol{J}^{min}$ and the nadir solution $\boldsymbol{J}^{max}$:

$$J^{ideal} = \boldsymbol{J}^{min} = \left[\min_{J(\boldsymbol{\theta})\in J_P^*} J_1(\boldsymbol{\theta}), \ldots, \min_{J(\boldsymbol{\theta})\in J_P^*} J_m(\boldsymbol{\theta})\right]$$
$$J^{nadir} = \boldsymbol{J}^{max} = \left[\max_{J(\boldsymbol{\theta})\in J_P^*} J_1(\boldsymbol{\theta}), \ldots, \max_{J(\boldsymbol{\theta})\in J_P^*} J_m(\boldsymbol{\theta})\right] \tag{2}$$

MOO techniques search for a discrete approximation $\boldsymbol{\Theta_P^*}$ of the Pareto set $\boldsymbol{\Theta_P}$ capable of generating a good quality description $\boldsymbol{J_P^*}$ of the Pareto front $\boldsymbol{J_P}$ (see Fig. 3). In this way, the DM has a set of solutions for a given problem and more flexibility for choosing a particular or desired solution. There are several widely used algorithms for calculating this Pareto front approximation (normal boundary intersection method (Das & Dennis, 1998; Miettinen, 1998), normal constraint method (Martínez, Herrero, Sanchis, Blasco, & García-Nieto, 2009; Martínez, García-Nieto, Sanchis, & Blasco, 2009; Messac & Ismail-Yahaya, 2003; Sanchis, Martínez, Blasco, & Salcedo, 2008), and the successive Pareto front optimisation (Ruzika1 & Wiecek, 2009)). Recently, multiobjective evolutionary algorithms (MOEAs) have been used due to their flexibility in dealing with non-convex and highly constrained functions (Coello, Veldhuizen, & Lamont, 2002; Coello & Lamont, 2004). For this reason, MOEAs are considered in this work.

A general framework is required to successfully incorporate the MOO approach into any engineering process. A multiobjective optimisation engineering design (MOED) methodology is shown in Fig. 4. It consists in three main steps:

MOP definition: at this stage the following are defined: the design concept (how to tackle the problem at hand); the engineering requirements (what it is important to optimise); and the constraints (which solutions are not practical/allowed). The design concept implies the existence of a parametric model that defines the parameter values (the decision space) that leads to a particular design alternative and its performance (Mattson & Messac, 2005).

MOO process: at this stage, the MOO statement, as well as the MOEA, are defined. It is important to select an MOEA that assures reasonable diversity, spread, and convergence to the Pareto front and is an efficient constraint handling mechanism.

Decision making stage: finally, with the calculated approximation $\boldsymbol{J_P^*}$, the DM can analyse the trade-off along the Pareto front. The DM will select the best vector solution according to his/her needs. A reliable tool or methodology is required for this final step, since it is not a trivial task to perform an analysis on m-dimensional Pareto fronts.

The MOED methodology for PI tuning the multivariable process is then defined.

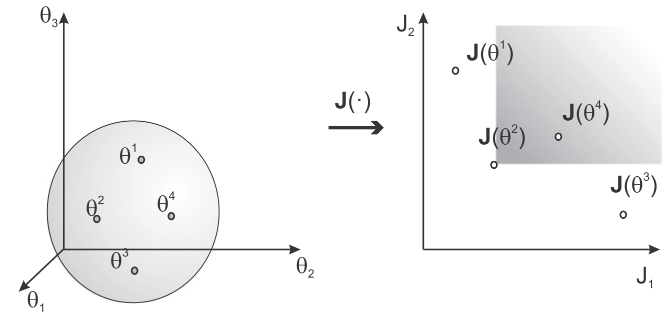

Fig. 1. Pareto set (left) and Pareto front (right). Objective vector $\boldsymbol{J}(\boldsymbol{\theta^4})$ is dominated by $\boldsymbol{J}(\boldsymbol{\theta^2})$.

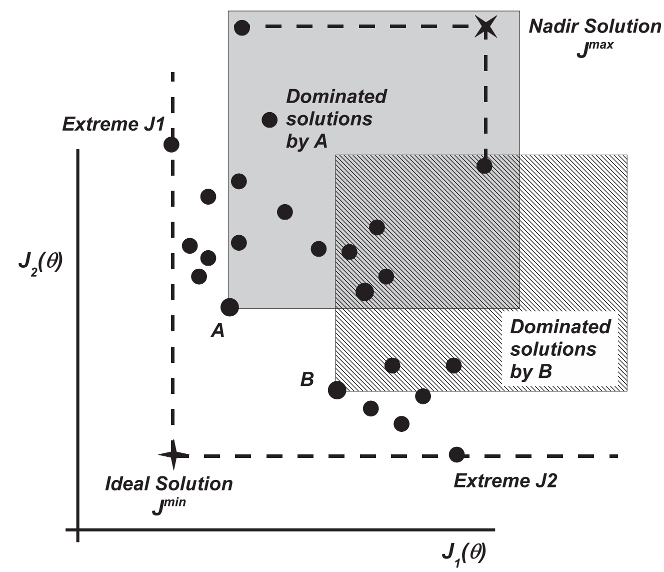

Fig. 2. Dominance concept. A given objective vector $\boldsymbol{A}$ dominates the objective vectors with a better or equal cost value in all objectives (with, at least, one of them being better). Two important points are defined: the ideal solution and the nadir solution (see Eq. 2).

3. Multiobjective optimisation engineering design applied to multivariable PI controller tuning

MIMO systems are common in industry. Their complexity is due to their coupling effects between inputs and outputs. Consider a $N \times N$ multivariable process modelled by the following transfer matrix:

[1] A maximisation problem can be converted to a minimisation problem taking into account that $\max J_i(\theta) = min(-J_i(\theta))$ is applied.