CMS
DOI 10.1007/s10287-007-0062-z

ORIGINAL PAPER

A multi-parametric programming approach for multilevel hierarchical and decentralised optimisation problems

Nuno P. Faísca · Pedro M. Saraiva · Berç Rustem · Efstratios N. Pistikopoulos

Abstract In this paper, we outline the foundations of a general global optimisation strategy for the solution of multilevel hierarchical and general decentralised multilevel problems, based on our recent developments on multi-parametric programming and control theory. The core idea is to recast each optimisation subproblem, present in the hierarchy, as a multi-parametric programming problem, with parameters being the optimisation variables belonging to the remaining subproblems. This then transforms the multilevel problem into single-level linear/convex optimisation problems. For decentralised systems, where more than one optimisation problem is present at each level of the hierarchy, Nash equilibrium is considered. A three person dynamic optimisation problem is presented to illustrate the mathematical developments.

Keywords Hierarchical decision making · Multilevel programming · Multi-parametric programming · Discrete-time systems · Closed-loop optimal control

1 Introduction

The development of a general theory to solve multi-person objective decision problems is of great importance for decision making and control theory (Başar 1975). Multi-person objective decision problems have attracted numerous investigations

N. P. Faísca · B. Rustem · E. N. Pistikopoulos (✉)
Centre for Process Systems Engineering,
Imperial College London, London SW7 2AZ, UK
e-mail: e.pistikopoulos@imperial.ac.uk

P. M. Saraiva
Gepsi-PSE Group, Department of Chemical Engineering,
University of Coimbra, 3030-290 Coimbra, Portugal

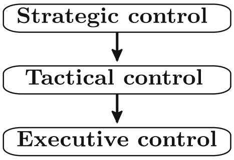

Fig. 1 Hierarchical control of an automatic vehicle (Rodić and Vukobratović 1999)

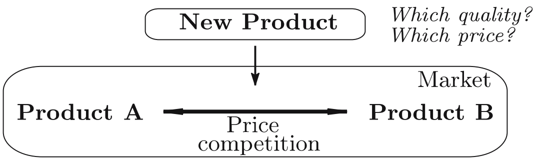

Fig. 2 Product positioning under price competition (Choi et al. 1990)

(Başar 1975, 1978; Tolwinski 1981; Başar and Olsder 1982; Anandalingman 1988; Liu 1998; Li et al. 2002; Shih et al. 2004), with diverse applications in engineering (Morari et al. 1980; Clark 1983; Stephanopoulos and Ng 2000), financial problems (Anandalingman 1988; Nie et al. 2006) and in other areas, two examples of such applications are depicted in Figs. 1 and 2.

In this work we focus on multilevel decentralised optimisation problems, where the objectives (optimisation subproblems) are organised in a hierarchy of decisions. In this hierarchy, each optimisation subproblem controls a subset of the full set of optimisation variables; the latter is completely controlled by the unique optimisation problem positioned at the top level.

The multi-layer nature in such problems results in non-linearities and non-convexities (Vicente and Calamai 1994); hence, it is not surprising that general solution strategies for solving such complex problems are rather limited. Moreover, the possible presence of logical decisions further increases the problems' complexity. Therefore, it is widely accepted that a global optimisation approach is needed for the solution of such multilevel problems (Floudas 2000).

Recently, Pistikopoulos and co-workers have been developing a general theory, algorithms and computation tools for the solution of general classes of multi-parametric programming problems (Pistikopoulos et al. 2007a) and multi-parametric control (Pistikopoulos et al. 2007b). The application of parametric programming theory to multi-level problems (Faísca et al. 2007b) makes possible the development of a unified strategy for their solution to global optimality. The core idea behind this approach is to recast each optimisation subproblem as a multi-parametric programming problem. Computing the rational reaction set for each subproblem in the entire feasible space, and subsequently, computing the corresponding equilibria within the hierarchical network, disassembles the complexity of the original problem. For instance, in an optimisation level with two subproblems or more, these explicit expressions are used to compute the Nash equilibrium between them. In our previous work (Faísca et al. 2007b; Pistikopoulos et al. 2007a) we have addressed the bilevel programming problem, a hierarchy of two optimisation subproblems organised in two levels. In this paper we extend the methodology proposed in Faísca et al. (2007b) to cope with multilevel decentralised optimisation problems. Furthermore, the methodology is applied to an optimal control problem of multi-level nature, where the foundations of a general theory for multi-level hierarchical and decentralised problems are established.

This paper is organised as follows. Section 2 introduces the multi-level mathematical formulation, which is used throughout the paper, and respective definitions of feasible and rational reaction set. It also briefly introduces the relevant multi-parametric programming theory and algorithms. The proposed multi-parametric programming approach for the solution of tri-level programming problems and bilevel programming with multi-followers problems is then described in detail in Sect. 3, and illustrated with example problems. Sect. 4 outlines the application of the proposed approach to multilevel optimal control of dynamic systems.

2 Preliminaries

2.1 Problem formulation

The general multilevel decentralised optimisation problem can be described as follows:

$$
\begin{aligned}
&\min_{x, y_1^i, y_2^k, \ldots, y_m^l} \quad f_1(x, y_1^i, y_2^k, \ldots, y_m^l), && \text{(1st level)}\\
&\quad \mathbf{s.t.}\ g_1(x, y_1^i, y_2^k, \ldots, y_m^l) \le 0,\\
&\qquad \text{where } \left[y_1^i, y_2^k, \ldots, y_m^l\right] \text{ solve},\\
&\quad \ldots, \min_{y_1^i, y_2^k, \ldots, y_m^l} \quad f_2^i(x, y_1^i, y_2^k, \ldots, y_m^l), \ldots && \text{(2nd level)}\\
&\qquad \mathbf{s.t.}\ g_2^i(x, y_1^i, y_2^k, \ldots, y_m^l), \le 0,\\
&\qquad\quad \text{where } \left[y_2^k, \ldots, y_m^l\right] \text{ solve},\\
&\qquad\quad \vdots\\
&\qquad\quad \ldots, \min_{y_m^l} f_m^l(x, y_1^i, y_2^k, \ldots, y_m^l), \ldots && \text{(mth level)}\\
&\qquad\qquad \mathbf{s.t.}\ g_m^l(x, y_1^i, y_2^k, \ldots, y_m^l) \le 0.
\end{aligned}
\tag{1}
$$

Here, f are real convex functions, g are vectorial real functions defining convex sets and x, y are sets of variables belonging to the group of real numbers; $i \in \{1, 2, \ldots, I\}, k \in \{1, 2, \ldots, K\}, l \in \{1, 2, \ldots, L\}$, implying that (2nd level) has I optimisation subproblems (3rd level) K optimisation subproblems and (mth level) has L optimisation subproblems, respectively.

For the sake of simplicity and without loss of generality, we analyse the relations in Problem (1) using two particular classes of multilevel programming problems: the tri-level programming problem, which organises vertically in three levels, and the bilevel programming problem with multi-followers, in a horizontal structure at the second level.

2.1.1 Tri-level programming

The tri-level programming problem can be stated as follows:

$$\begin{aligned}
&\min_{x,y_1,y_2} f_1(x, y_1, y_2), && \text{(1st level)}\\
&\quad \textbf{s.t.}\ g_1(x, y_1, y_2) \le 0,\\
&\qquad \text{where } [y_1, y_2] \text{ solve},\\
&\qquad \min_{y_1,y_2} f_2(x, y_1, y_2), && \text{(2nd level)}\\
&\qquad\ \textbf{s.t.}\ g_2(x, y_1, y_2), \le 0,\\
&\qquad\quad \text{where } [y_2] \text{ solve},\\
&\qquad\quad \min_{y_2} f_3(x, y_1, y_2), && \text{(3rd level)}\\
&\qquad\quad \textbf{s.t.}\ g_3(x, y_1, y_2) \le 0,
\end{aligned} \tag{2}$$

with the following definitions:

– feasible set for the third level,

$$\Omega_2(x, y_1) = \{y_2 \in Y_2 : g_3(x, y_1, y_2) \le 0\}, \tag{3}$$

– rational reaction set for the third level,

$$\phi_2(x, y_1) = \{y_2 \in Y_2 : y_2 \in \text{argmin}\{f_2(x, y_1, y_2) : y_2 \in \Omega_2(x, y_1)\}\}, \tag{4}$$

– feasible set for the second level,

$$\Omega_1(x) = \{y_1, y_2 \in Y_1, Y_2 : g_2(x, y_1, y_2) \le 0, g_3(x, y_1, y_2) \le 0\}, \tag{5}$$

– rational reaction set for the second level,

$$\begin{aligned}\phi_1(x) = \{y_1, y_2 \in Y_1, Y_2 : y_1 \in \text{argmin}\\ \times\{f_2(x, y_1, y_2) : y_1 \in \Omega_1(x), y_2 \in \phi_2(x, y_1)\}\}.\end{aligned} \tag{6}$$

Note the parametric nature of the rational reaction sets, Eqs. (4) and (6), which reflects the dependence of the decisions taken at the upper levels on the decisions taken at the lower levels. This in fact, evidences that in multilevel programming problems the relations between the levels differ from the well-known Stackelberg game, where the decisions made by the followers don't affect the decision, already taken by the leader (Vicente 1992).

2.1.2 Bilevel programming with multi-followers

Bilevel programming problems with multi-followers involve two optimisation levels with several optimisation subproblems at the lower (2nd) level:

$$
\begin{aligned}
&\min_{x,y_1,y_2,\ldots,y_m} F(x, y_1, y_2, \ldots, y_m), && \text{(1st level)}\\
&s.t.\ G(x, y_1, y_2, \ldots, y_m) \le 0,\\
&\quad x \in X,\\
&\quad y_i \in \operatorname{argmin}\{f_i(x, y_1, y_2, \ldots, y_m) : g_i(x, y_1, y_2, \ldots, y_m)\\
&\quad \le 0, y_i \in Y_i\}, && \text{(2nd level)}\\
&\quad i \in \{1, 2, \ldots, m\},
\end{aligned}
\tag{7}
$$

with the following definitions:

- feasible set for the ith follower,

$$
\Omega_i(x, y_1, y_2, \ldots, y_{i-1}, y_{i+1}, \ldots, y_m) = \{y_i \in Y_i : g_i(x, y_1, y_2, \ldots, y_m) \le 0\},
\tag{8}
$$

- rational reaction set for the ith follower,

$$
\begin{aligned}
&\phi_i(x, y_1, y_2, \ldots, y_{i-1}, y_{i+1}, \ldots, y_m)\\
&\quad = \{y_i \in Y_i : y_i \in \operatorname{argmin}\{f_i(x, y_1, y_2, \ldots, y_m) : y_i \in \Omega_i(x)\}\}.
\end{aligned}
\tag{9}
$$

Since one assumption is that followers may exchange information, conflicts naturally occur. The Nash equilibrium is often a preferred strategy to coordinate such decentralised systems (Başar 1975, 1978; Cruz 1978; Başar and Selbuz 1979; Choi et al. 1990; Liu 1998). Consequently, the optimisation subproblems positioned in the lower level are assumed to reach a Nash equilibrium point $(x, y_1^*, y_2^*, \ldots, y_m^*)$ (Başar and Olsder 1982):

$$
\begin{cases}
f_1(x, y_1^*, y_2^*, \ldots, y_m^*) \le f_1(x, y_1, y_2^*, \ldots, y_m^*), \forall y_1 \in Y_1,\\
f_2(x, y_1^*, y_2^*, \ldots, y_m^*) \le f_2(x, y_1^*, y_2, \ldots, y_m^*), \forall y_2 \in Y_2,\\
\vdots\\
f_m(x, y_1^*, y_2^*, \ldots, y_m^*) \le f_m(x, y_1^*, y_2^*, \ldots, y_m), \forall y_m \in Y_m.
\end{cases}
\tag{10}
$$

Once more observe the parametric nature of the followers' rational reaction set, Eq. (9). In this case, however, each rational reaction set is a function of both the upper level decision variables and the decision variables of the other subproblems located in the same hierarchical level. Additionally, the priority remains to solve the leader's objective function to global optimality. Thus, we aim to compute the set $\{x, y_1, \ldots, y_m\}$ which optimises globally the leader objective:

$$
\min_{x,y_1,\ldots,y_m} \{F(x, y_1, \ldots, y_m) : G(x, y_1, \ldots, y_m) \le 0, y_i \in \phi_i, i = 1, \ldots, m\},
\tag{11}
$$

and the set $\{y_1, \ldots, y_m\}$ which corresponds to a Nash equilibrium point, Eq. (10).

2.2 Multi-parametric programming

Consider the general multi-parametric non-linear programming problem:

$$\begin{aligned} &\min_x f(x,\theta), \\ s.t.\ & g_i(x,\theta) \le 0, \quad \forall\ i = 1,\ldots,p, \\ & h_j(x,\theta) = 0, \quad \forall\ j = 1,\ldots,q, \\ & x \in X \subseteq \mathbb{R}^n, \\ & \theta \in \Theta \subseteq \mathbb{R}^m, \end{aligned} \tag{12}$$

where f, g and h are twice continuously differentiable in x and θ. Assume also that f is a convex function and g, h define a convex set. Therefore, the first-order Karush–Kuhn–Tucker (KKT) optimality conditions for (12) are given as follows:

$$\begin{aligned} &\mathscr{L} = f(x,\theta) + \sum_{i=1}^{p} \lambda_i g_i(x,\theta) + \sum_{j=1}^{q} \mu_j h_j(x,\theta), \\ &\nabla_x \mathscr{L} = 0, \\ &\lambda_i g_i(x,\theta) = 0, \quad \lambda_i \ge 0, \quad \forall\, i = 1,\ldots,p, \\ &h_j(x,\theta) = 0, \quad \forall\, j = 1,\ldots,q. \end{aligned} \tag{13}$$

The main sensitivity result for (12) derives directly from system (13), as shown in Theorem 1.

Theorem 1 *Basic sensitivity theorem (Fiacco 1976): let θ_0 be a vector of parameter values and (x_0, λ_0, μ_0) a KKT triple corresponding to (13), where λ_0 is nonnegative and x_0 is feasible in (12). Also assume that (i) strict complementary slackness (SCS) holds, (ii) the binding constraint gradients are linearly independent (LICQ: Linear Independence Constraint Qualification), and (iii) the second-order sufficiency conditions (SOSC) hold. Then, in the neighbourhood of θ_0, there exists a unique, once continuously differentiable function, $z(\theta) = [x(\theta), \lambda(\theta), \mu(\theta)]$, satisfying (13) with $z(\theta_0) = [x(\theta_0), \lambda(\theta_0), \mu(\theta_0)]$, where $x(\theta)$ is a unique isolated minimiser for (12), and*

$$\begin{pmatrix} \frac{dx(x_0)}{d\theta} \\ \frac{d\lambda(x_0)}{d\theta} \\ \frac{d\mu(x_0)}{d\theta} \end{pmatrix} = -(M_0)^{-1} N_0, \tag{14}$$

where, M_0 and N_0 are the Jacobian of system (13) with respect to z and θ:

$$M_0 = \left(\begin{array}{c|ccc|ccc} \nabla^2_{xx}\mathscr{L} & \nabla_x g_1 & \cdots & \nabla_x g_p & \nabla_x h_1 & \cdots & \nabla_x h_q \\ \hline -\lambda_1 \nabla_x^T g_1 & -g_1 & & & & & \\ \vdots & & \ddots & & & 0 & \\ -\lambda_p \nabla_x^T g_p & & & -g_p & & & \\ \hline \nabla_x^T h_1 & & & & & & \\ \vdots & & 0 & & & 0 & \\ \nabla_x^T h_q & & & & & & \end{array}\right),$$

$$N_0 = (\nabla^2_{\theta x}\mathcal{L}, -\lambda_1\nabla^T_\theta g_1, \ldots, -\lambda_p\nabla^T_\theta g_p, \nabla^T_\theta h_1, \ldots, \nabla^T_\theta h_q)^T. \qquad \square$$

Proof See (Fiacco 1983, pp 72).

Note that the assumptions stated in the theorem above ensure M_0 is invertible (McCormick 1976).

Dua et al. (2002) has proposed an algorithm to solve Eq. (14) in the entire range of the varying parameters for general convex problems. This algorithm is based on approximations of the non-linear *optimal* expression, $x = \gamma^*(\theta)$, by a set of first-order approximations (Corollary 1).

Corollary 1 *First-order estimation of* $x(\theta)$, $\lambda(\theta)$, $\mu(\theta)$, *near* $\theta = \theta_0$ (*Fiacco 1983*): *Under the assumptions of Theorem* 1, *a first-order approximation of* $[x(\theta), \lambda(\theta), \mu(\theta)]$ *in the neighbourhood of* θ_0 *is,*

$$\begin{bmatrix} x(\theta) \\ \lambda(\theta) \\ \mu(\theta) \end{bmatrix} = \begin{bmatrix} x_0 \\ \lambda_0 \\ \mu_0 \end{bmatrix} - (M_0)^{-1} \cdot N_0 \cdot \theta + o(||\theta||), \tag{15}$$

where $(x_0, \lambda_0, \mu_0) = [x(\theta_0), \lambda(\theta_0), \mu(\theta_0)]$, $M_0 = M(\theta_0)$, $N_0 = N(\theta_0)$, *and* $\phi(\theta) = o(||\theta||)$ *means that* $\phi(\theta)/||\theta|| \to 0$ *as* $\theta \to \theta_0$.

Each piecewise linear approximation is confined to regions defined by feasibility and optimality conditions (Dua et al. 2002). If $\breve{g}$ corresponds to the non-active constraints, and $\tilde{\lambda}$ to the Lagrangian multipliers of the active constraints:

$$\begin{cases} \breve{g}(x(\theta), \theta) \le 0 \to \text{Feasibility conditions,} \\ \tilde{\lambda}(\theta) \ge 0 \qquad\quad \to \text{Optimality conditions.} \end{cases} \tag{16}$$

Consequently, the explicit expressions are given by a conditional piecewise linear function (Dua et al. 2002):

$$\begin{cases} x = \mathbf{C}^1 + \mathbf{K}^1 \cdot \theta, \ \ \theta \in CR^1, \\ x = \mathbf{C}^2 + \mathbf{K}^2 \cdot \theta, \ \ \theta \in CR^2, \\ \qquad\vdots \\ x = \mathbf{C}^L + \mathbf{K}^L \cdot \theta, \ \theta \in CR^L, \end{cases} \tag{17}$$

where $\mathbf{K}^i$ and $\mathbf{C}^i$ are real matrices, and $CR^i \subset \mathbb{R}^m$.

3 Proposed methodology

In this section, we show how we can address tri-level programming and bilevel programming with multi-followers problems, and solve them to global optimality through the application of parametric programming. For the sake of clarity, the methodology is described using formulations with quadratic cost functions and linear constraints, however, it is applicable to general non-linear problems using suitable multi-parametric programming algorithms (Dua et al. 2004).

3.1 Tri-level programming problem

Consider the tri-level programming problem with a quadratic objective function and linear constraints:

$$
\begin{aligned}
\min_{x,y_1,y_2} f_1 = {} & L_1^1 \\
& +L_2^1 \cdot x + L_3^1 \cdot y_1 + L_4^1 \cdot y_2 \\
& +\tfrac{1}{2}x^T \cdot L_5^1 \cdot x + \tfrac{1}{2}y_1^T \cdot L_6^1 \cdot y_1 + \tfrac{1}{2}y_2^T \cdot L_7^1 \cdot y_2 \qquad \text{(1st level)} \\
& +x^T \cdot L_8^1 \cdot y_1 + y_2^T \cdot L_9^1 \cdot x + y_2^T \cdot L_{10}^1 \cdot y_1, \\
s.t. \; & \left| \begin{aligned}
& G_1^1 \cdot x + G_2^1 \cdot y_1 + G_3^1 \cdot y_2 \le 0, \\
& \min_{y_1,y_2} f_2 = L_1^2 \\
& \quad +L_2^2 \cdot x + L_3^2 \cdot y_1 + L_4^2 \cdot y_2 \\
& \quad +\tfrac{1}{2}x^T \cdot L_5^2 \cdot x + \tfrac{1}{2}y_1^T \cdot L_6^2 \cdot y_1 + \tfrac{1}{2}y_2^T \cdot L_7^2 \cdot y_2 \qquad \text{(2nd level)} \\
& \quad +x^T \cdot L_8^2 \cdot y_1 + y_2^T \cdot L_9^2 \cdot x + y_2^T \cdot L_{10}^2 \cdot y_1, \\
& s.t. \; \left| \begin{aligned}
& G_1^2 \cdot x + G_2^2 \cdot y_1 + G_3^2 \cdot y_2 \le 0, \\
& \min_{y_2} f_3 = L_1^3 \\
& \quad +L_2^3 \cdot x + L_3^3 \cdot y_1 + L_4^3 \cdot y_2 \qquad \text{(3rd level)} \\
& \quad +\tfrac{1}{2}x^T \cdot L_5^3 \cdot x + \tfrac{1}{2}y_1^T \cdot L_6^3 \cdot y_1 + \tfrac{1}{2}y_2^T \cdot L_7^3 \cdot y_2 \\
& \quad +x^T \cdot L_8^3 \cdot y_1 + y_2^T \cdot L_9^3 \cdot x + y_2^T \cdot L_{10}^3 \cdot y_1, \\
& \quad s.t. \; |G_1^3 \cdot x + G_2^3 \cdot y_1 + G_3^3 \cdot y_2 \le 0.
\end{aligned} \right.
\end{aligned} \right.
\end{aligned}
\tag{18}
$$

Problem (18) comprises three subproblems, one at each optimisation level. Each optimisation level can be recast as a multi-parametric programming problem, where the optimisation variables corresponding to the upper optimisation levels are classified as parameters. For presentation and computation purposes, (i) we group the parameters in the ith level in a single vector, ω^i and (ii) we introduce an artificial variable, v^i, to eliminate all bilinear terms.

Beginning with the (3rd level), and considering a vector,

$$\left[\omega^3\right]^T = [x \mid y_1],$$

we re-write (18) as,

$$
\begin{aligned}
\min_{y_2} f_3(y_2, \omega^3) = {} & L_1^3 + L_2^{3*} \cdot \omega^3 + L_4^3 \cdot y_2 + \frac{1}{2}\omega^{3^T} \cdot L_5^{3*} \cdot \omega^3 \\
& + \frac{1}{2} y_2^T \cdot L_7^3 \cdot y_2 + y_2^T \cdot L_8^{3*} \cdot \omega^3, \\
s.t. \; G_1^{3*} \cdot \omega^3 + G_3^3 \cdot y_2 + G_4^3 \le 0, & \quad x \in X.
\end{aligned}
\tag{19}
$$

Introducing an artificial variable, $v^3 = y_2 + \Phi \cdot \omega^3$, where Φ is an appropriate matrix, the bilinear terms, represented in (19) by matrix L_8^{3*}, are eliminated. Under the right conditions (Faísca et al. 2007b), $\Phi = L_7^{3^{-1}} L_8^{3*}$, and (19) can be rewritten as follows:

$$\min_{v_3} f_3(v_3, \omega^3) = L_1^3 + L_2^{3**} \cdot \omega^3 + \frac{1}{2}\omega^{3^T} \cdot L_5^{3**} \cdot \omega^3 + \min_{v_3}\left\{L_4^{3**} \cdot v_3 + \frac{1}{2}v_3^T \cdot L_7^{3**} \cdot v_3\right\},$$
$$s.t.\ G_3^{3**} \cdot v_3 \leq G_4^{3**} + G_1^{3**} \cdot \omega^3, v \in V, \tag{20}$$

Problem (20) can be solved with a multi-parametric programming algorithm (Dua et al. 2002), resulting in:

$$v_3^k = m_3^k + n_3^k \cdot \omega^3, \quad H_3^k \cdot \omega^3 \leq h_3^k,$$
which can be rewritten as,
$$y_2^k = m_3^k + (n_3^k - \Phi) \cdot \omega^3, \quad H_3^k \cdot \omega^3 \leq h_3^k,$$
or,
$$y_2^k = m_3^k + p_1^k \cdot x + p_2^k \cdot y_1, \quad H_{31}^k \cdot x + H_{32}^k \cdot y_1 \leq h^k, \tag{21}$$

where, $k = 1, \ldots, K_2$, with K_2 being the number of critical region, and consequently, the number of linear approximations done on the optimal rational reaction set $\phi_2(x, y_1)$ (see Corollary 1).

The expressions in (21) can then be incorporated in the second optimisation level of (18). Note that since the expressions in (21) are piecewise linear functions of y_2^k, the complexity of the original problem does not increase. Hence, the second level can be reformulated as the following K_2 optimisation problems:

$$\min_{y_1} f_2 = L_1^{2*} + L_2^{2*} \cdot x + L_3^{2*} \cdot y_1 + \frac{1}{2}x^T \cdot L_4^{2*} \cdot x + \frac{1}{2}y_1^T \cdot L_5^{2*} \cdot y_1 + y_1^T \cdot L_8^{2*} \cdot x,$$
$$s.t.\ G_1^{2*} \cdot x + G_2^{2*} \cdot y_1 + G_3^{2*} \leq 0, x \in X. \tag{22}$$

We can thus proceed with optimisation levels 1 and 2. Following this procedure, tri-level optimisation problems in (18) result in K_1 single level convex optimisation problems:

$$\begin{aligned} &\min_{x} f_1^*(x, y_1(x), y_2(x, y_1)), \\ s.t.\ &G_1(x, y_1(x), y_2(x, y_1(x))) \leq 0, \\ &x \in C_{rf}, \\ &C_{rf} = \{x \in X : \exists_{y_1, y_2} \in Y_1, Y_2, G_2(x, y_1, y_2) \leq 0, G_3(x, y_1, y_2) \leq 0\}. \end{aligned} \tag{23}$$

The number of K_1 final convex optimisation problems (23) depends on the number of critical regions obtained in each optimisation level. The algorithm is summarised in Table 1, and is illustrated with the following example.

Table 1 Parametric programming algorithm for tri-level programming problems

Step	Description
1	Recast the third level of the optimisation problem as a multi-parametric programming problem, with parameters being the upper levels optimisation variables, x and y_1 (19)
2	Solve the resulting problem using a suitable multi-parametric programming algorithm
3	Substitute each of the K_2 solutions in the 2nd optimisation level, and formulate K_2 multi-parametric problems with the variables from the leader being the parameters (22)
4	Solve the resulting problem using a suitable multi-parametric programming algorithm
5	Substitute each of the K_1 solutions in the leader's problem, and formulate the K_1 one-level optimisation problems (23)
6	Compare the K_1 optima and select the best one

3.1.1 Illustrative example 1

Consider the following linear tri-level example (Ruan et al. 2004):

$$\begin{aligned} &\min_{x,y_1,y_2} f_1 = -x - 4 \cdot y_2, \\ &\quad \text{where } [y_1, y_2] \text{ solve}, \\ &\quad \min_{y_1,y_2} f_2 = 2 \cdot y_2, \\ &\qquad \text{where } y_2 \text{ solves}, \\ &\qquad \min_{y_2} f_3 = -y_2, \\ &\qquad \textbf{s.t. } x + y_1 + y_2 \le 2.5, \\ &\qquad\quad 0 \le x, y_1, y_2 \le 1. \end{aligned} \tag{24}$$

Following the steps described in Table 1:

Step 1. Recast (3rd) level optimisation problem, f_3, as a multi-parametric programming problem, with parameters being x and y_1

$$\begin{aligned} &\min_{y_2} f_3 = -y_2, \\ &\textbf{s.t. } y_2 \le 2.5 - x - y_1, \\ &\quad 0 \le x, y_1, y_2 \le 1, \end{aligned} \tag{25}$$

solve the resulting problem using a multi-parametric optimisation algorithm (Dua et al. 2002):

$$CR^1 \begin{cases} y_2 = 1, \\ 0 \le x, y_1 \le 1, \\ x + y_1 \le 1.5, \end{cases} \quad CR^2 \begin{cases} y_2 = -x - y_1 + 2.5, \\ x, y_1 \le 1, \\ -x - y_1 \le -1.5. \end{cases} \tag{26}$$

Step 2. Incorporate rational reaction set (26) into the optimisation problem corresponding to (2nd) level;

$$\begin{aligned} &\min_{y_1,y_2} f_2^{CR^1} = 2, \\ &\textbf{s.t.}\ 0 \le x \le 1, \\ &0 \le y_1 \le 1, \\ &x + y_1 \le 1.5, \end{aligned} \qquad \begin{aligned} &\min_{y_1,y_2} f_2^{CR^2} = -2x - 2y_1 + 5, \\ &\textbf{s.t.}\ x, y_1 \le 1, \\ &-x - y_1 \le -1.5. \end{aligned} \tag{27}$$

Step 3. Solve problems (27) considering them as multi-parametric programming problems, with x being parameter;

$$CR^3 \begin{cases} y_2 = 1, \\ 0 \le x \le 1, \\ 0 \le y_1 \le 1, \\ x + y_1 \le 1.5, \end{cases} \quad CR^4 \begin{cases} y_1 = 1, \\ y_2 = -x + 1.5, \\ 0.5 \le x \le 1. \end{cases} \tag{28}$$

Step 4. Incorporate rational reaction set (28) into the optimisation problem corresponding to (1st) level;

$$\begin{aligned} &\min_{x,y_1,y_2} f_1^{CR^3} = -x - 4, \\ &\textbf{s.t.}\ 0 \le x \le 1, \\ &0 \le y_1 \le 1, \\ &x + y_1 \le 1.5, \end{aligned} \qquad \begin{aligned} &\min_{x,y_1,y_2} f_1^{CR^4} = 3x - 6, \\ &\textbf{s.t.}\ 0.5 \le x \le 1. \end{aligned} \tag{29}$$

Step 5. Solve problems in (29);

$$Solution\ 1 \begin{cases} f_1^{CR^3} = -5, \\ x = 1, \\ y_2 = 1, \\ 0 \le y_1 \le 0.5, \end{cases} \quad Solution\ 2 \begin{cases} f_1^{CR^4} = -4.5, \\ x = 0.5, \\ y_1 = 1, \\ y_2 = 1. \end{cases} \tag{30}$$

Note that in *Solution 1*, y_1 is represented by an interval. This is due to the fact that the objective function of (2nd level) does not depend on y_1.

Concluding, two solutions are obtained: *Solution 1* and *Solution 2*, which are compared with the one obtained from the literature (Ruan et al. 2004, *Solution 3*), as shown in Table 2.

From Table 2 we conclude that *Solution 1* is the global optimum for this tri-level programming problem.

3.2 Bilevel programming problem with multi-followers

Consider the bilevel programming problem with multi-followers, and assume quadratic objective functions, linear constraints and two followers:

Table 2 Solutions for problem (24)

	Parametric programming algorithm		Ruan et al. (2004)
	Solution 1	*Solution 2*	*Solution 3*
f^1	–5	−4.5	−4.5
f^2	2	2	2
f^3	1	1	1
x	1	0.5	–
y_1	0.5	0	–
y_2	1	0	–

$$\begin{aligned}
\min_{x,y_1,y_2} f_1 = {} & L_1^1 \\
& +L_2^1 \cdot x + L_3^1 \cdot y_1 + L_4^1 \cdot y_2 \text{(1st level)} \\
& +\tfrac{1}{2}x^T \cdot L_5^1 \cdot x + \tfrac{1}{2}y_1^T \cdot L_6^1 \cdot y_1 + \tfrac{1}{2}y_2^T \cdot L_7^1 \cdot y_2 \\
& +x^T \cdot L_8^1 \cdot y_1 + y_2^T \cdot L_9^1 \cdot x + y_2^T \cdot L_{10}^1 \cdot y_1, \\
s.t. \; & \left|\begin{array}{l}
G_1^1 \cdot x + G_2^1 \cdot y_1 + G_3^1 \cdot y_2 \le 0, \\
\qquad\qquad\qquad\qquad \text{(2nd level)} \\
\min_{y_1} f_2 = L_1^2 \qquad\qquad \underline{Follower\ 1} \\
\quad +L_2^2 \cdot x + L_3^2 \cdot y_1 + L_4^2 \cdot y_2 \\
\quad +\tfrac{1}{2}x^T \cdot L_5^2 \cdot x + \tfrac{1}{2}y_1^T \cdot L_6^2 \cdot y_1 + \tfrac{1}{2}y_2^T \cdot L_7^2 \cdot y_2 \\
\quad +x^T \cdot L_8^2 \cdot y_1 + y_2^T \cdot L_9^2 \cdot x + y_2^T \cdot L_{10}^2 \cdot y_1, \\
s.t. \; G_1^2 \cdot x + G_2^2 \cdot y_1 + G_3^2 \cdot y_2 \le 0, \\
\\
\min_{y_2} f_3 = L_1^3 \qquad\qquad \underline{Follower\ 2} \\
\quad +L_2^3 \cdot x + L_3^3 \cdot y_1 + L_4^3 \cdot y_2 \\
\quad +\tfrac{1}{2}x^T \cdot L_5^3 \cdot x + \tfrac{1}{2}y_1^T \cdot L_6^3 \cdot y_1 + \tfrac{1}{2}y_2^T \cdot L_7^3 \cdot y_2 \\
\quad +x^T \cdot L_8^3 \cdot y_1 + y_2^T \cdot L_9^3 \cdot x + y_2^T \cdot L_{10}^3 \cdot y_1, \\
s.t. \; G_1^3 \cdot x + G_2^3 \cdot y_1 + G_3^3 \cdot y_2 \le 0.
\end{array}\right.
\end{aligned} \tag{31}$$

The difference between Problem (31) and Problem (18) is the existence of two optimisation subproblems in a single level. Accordingly, the concept of Nash equilibrium is introduced.

As in the tri-level programming case, each optimisation subproblem in (2nd) level is recast as a multi-parametric programming problem. In this problem, the parameters are all the variables from the optimisation problem at (1st) level as well as the optimisation variables of the other subproblems at the same level, $Follower$ 1 or $Follower$ 2 in this case (31). Thus, defining vectors, $\left[\omega^2\right]^T = [x \mid y_2]$ and $\left[\omega^3\right]^T = [x \mid y_1]$, we re-write the (2nd) level optimisation subproblems as,

$$\begin{aligned}
\min_{y_1} f_2(y_1, \omega^2) = {} & L_1^2 + L_2^{2*} \cdot \omega^2 + L_3^2 \cdot y_1 + \frac{1}{2}\omega^{2^T} \cdot L_5^{2*} \cdot \omega^2 \\
& + \frac{1}{2} y_1^T \cdot L_6^2 \cdot y_1 + y_1^T \cdot L_8^{2*} \cdot \omega^2, \\
s.t. \; & G_1^{2*} \cdot \omega^2 + G_2^2 \cdot y_1 \le 0,
\end{aligned} \tag{32}$$

and,

$$\min_{y_2} f_3(y_2, \omega^2) = L_1^3 + L_2^{3*} \cdot \omega^3 + L_4^3 \cdot y_2 + \frac{1}{2}\omega^{3^T} \cdot L_5^{3*} \cdot \omega^{2a} + \frac{1}{2} y_2^T \cdot L_7^3 \cdot y_2 + y_1^T \cdot L_9^{3*} \cdot \omega^3,$$
$$s.t. \;\; G_1^{3*} \cdot \omega^3 + G_3^3 \cdot y_2 \le 0, \tag{33}$$

where ω^2 and ω^3 are the vectors of parameters. The bi-linearities can be circumvented using a strategy similar to the one used in the tri-level case. Using a multi-parametric programming algorithm (Dua et al. 2002), problems (32) and (33) result in the following parametric expressions:

$$\begin{cases} y_1 = \phi_1(x, y_2) \rightarrow \text{rational reaction set follower 1}, \\ y_2 = \phi_2(x, y_1) \rightarrow \text{rational reaction set follower 2}, \end{cases} \tag{34}$$

which are then used to compute the Nash equilibrium (x, y_1^*, y_2^*):

$$\begin{cases} f_1(x, y_1^*, y_2^*) \le f_1(x, y_1, y_2^*), \forall y_1 \in Y_1, \\ f_2(x, y_1^*, y_2^*) \le f_2(x, y_1^*, y_2), \forall y_2 \in Y_2, \end{cases} \tag{35}$$

easily computed by direct comparison (Liu 1998):

$$\phi_1'(x, y_1) = \phi_2(x, y_1), \rightarrow \; y_1 = \phi_2^*(x), \tag{36a}$$
$$\phi_1(x, y_2) = \phi_2'(x, y_2), \rightarrow \; y_2 = \phi_1^*(x). \tag{36b}$$

Finally, substituting the expressions in (36) in the leader's optimisation problem (1st) level, we end up with a single level convex optimisation problem, involving only the leader's optimisation variables, as follows:

$$\begin{aligned} &\min_x f_1^*(x, y_1(x, y_2^*(x)), y_2(x, y_1^*(x))), \\ s.t. \;\; &G_1(x, y_1(x, y_2^*), y_2(x, y_1^*)) \le 0, \\ &x \in C_{rf}, \\ &C_{rf} = \{x \in X : \exists_{y_1, y_2} \in Y, Z, G_2(x, y_1, y_2) \le 0, G_3(x, y_1, y_2) \le 0\}. \end{aligned} \tag{37}$$

The algorithm is summarised in Table 3 and is illustrated with the following example.

3.2.1 Illustrative example 2

Consider the following linear bilevel programming example involving three followers at the second level (Anandalingman 1988):

Table 3 Parametric programming algorithm for bilevel programming problems with multi-followers

Step	Description
1	Recast each of the subproblems in the lower level as a multi-parametric programming problem, with the variables out of their control being the parameters (32, 33)
2	Solve the resulting problems using the suitable multi-parametric programming algorithm
3	Compute a Nash equilibrium point by direct comparison of the rational reaction sets (35)
4	Substitute each of the K solutions in the leader's problem, and formulate the K one level optimisation problems
5	Compare the K optima points and select the best one

$$\begin{aligned}
&\min_{x,y_1,y_2,y_3} \quad F(x,y_1,y_2,y_3) = -x - y_1 - 2y_2 - y_3,\\
s.t.\ &\min_{y_1} f_1(x,y_1,y_2,y_3) = x - 3y_1 + y_2 + y_3,\\
&\min_{y_2} f_2(x,y_1,y_2,y_3) = x + y_1 - 3y_2 + y_3,\\
&\min_{y_3} f_3(x,y_1,y_2,y_3) = x + y_1 + y_2 - 3y_3,\\
&s.t. \quad 3x + 3y_1 \le 30,\\
&\qquad 2x + y_1 \le 20,\\
&\qquad y_2 \le 10,\\
&\qquad y_2 + y_3 \le 15,\\
&\qquad y_3 \le 10,\\
&\qquad x + 2y_1 + 2y_2 + y_3 \le 40,\\
&\qquad x, y_1, y_2, y_3 \ge 0.
\end{aligned} \tag{38}$$

Assume that the leader imposes all constraints to all followers. Thus, performing the steps described in Table 3:

Step 1. Recast optimisation subproblems $\min_{y_1} f_1$, $\min_{y_2} f_2$ and $\min_{y_3} f_3$ as multi-parametric programming problems, with parameters being the set of variables out of their control.

Step 2. Solve the three multi-parametric programming problems using a suitable algorithm (Dua et al. 2002).

Follower 1

$$CR_1^1 \begin{cases} y_1 = -x + 10,\\ 0 \le x, y_2, y_3 \le 10,\\ y_2 + y_3 \le 15,\\ -0.5x + y_2 + 0.5y_3 \le 10, \end{cases} \qquad CR_1^2 \begin{cases} y_1 = -0.5x - y_2 - 0.5y_3 + 20,\\ 0 \le x,\\ 0.5x - y_2 - 0.5y_3 \le -10,\\ y_2 \le 10,\\ y_2 + y_3 \le 15. \end{cases} \tag{39}$$

Follower 2

$$CR_2^1 \begin{cases} y_2 = 10, \\ 0 \le x, y_1, y_3, \\ x + y_1 \le 10, \\ y_3 \le 5, \\ 0.5x + y_1 + 0.5y_3 \le 10, \end{cases} \quad CR_2^2 \begin{cases} y_2 = -y_3 + 15, \\ 0 \le x, y_1, \\ x + y_1 \le 10, \\ 5 \le y_3 \le 10, \\ 0.5x + y_1 - 0.5y_3 \le 5, \end{cases}$$

$$CR_2^3 \begin{cases} y_2 = -0.5x - y_1 - 0.5y_3 + 20, \\ 0 \le x, \\ x + y_1 \le 10, \\ -0.5x - y_1 + 0.5y_3 \le -5, \\ -0.5x - y_1 - 0.5y_3 \le -10. \end{cases} \tag{40}$$

Follower 3

$$CR_3^1 \begin{cases} y_3 = 10, \\ 0 \le x, y_1, y_2, \\ x + y_1 \le 10, \\ y_1 \le 5, \\ 0.5x + y_1 + y_2 \le 15, \end{cases} \quad CR_3^2 \begin{cases} y_3 = -y_1 + 15, \\ 0 \le x, y_2, \\ x + y_1 \le 10, \\ 5 \le y_1, \\ 0.5x + 0.5y_1 + y_2 \le 12.5, \end{cases}$$

$$CR_3^3 \begin{cases} y_3 = -x - 2y_1 - 2y_2 + 40, \\ 0 \le x, y_1, \\ x + y_1 \le 10, \\ -0.5x - 0.5y_1 - y_2 \le -12.5, \\ 0.5x + y_1 + y_2 \le 20, \\ -0.5x - y_1 - y_2 \le -15. \end{cases} \tag{41}$$

Step 3. Compute the *Nash* equilibrium point, through direct comparison of the explicit analytical rational reaction sets (39)–(41). Through this comparison we generate 18 regions, from which, 12 have empty feasible sets. After removing empty regions:

$$CR^1 \begin{cases} y_1 = -x + 10, \\ y_2 = 10, \\ y_3 = x, \end{cases} \quad CR^2 \begin{cases} y_1 = -x + 10, \\ y_2 = -y_3 + 15, \\ y_3 = -x - 2y_1 - 2y_2 + 40, \end{cases}$$

$$CR^3 \begin{cases} y_1 = -x + 10, \\ y_2 = -0.5x - y_1 - 0.5y_3 + 20, \\ y_3 = -x - 2y_1 - 2y_2 + 40, \end{cases} \quad CR^4 \begin{cases} y_1 = -0.5x - y_2 - 0.5y_3 + 20, \\ y_2 = 10, \\ y_3 = -x - 2y_1 - 2y_2 + 40, \end{cases}$$

$$CR^5 \begin{cases} y_1 = -0.5x - y_2 - 0.5y_3 + 20, \\ y_2 = -y_3 + 15, \\ y_3 = -x - 2y_1 - 2y_2 + 40, \end{cases} \quad CR^6 \begin{cases} y_1 = -0.5x - y_2 - 0.5y_3 + 20, \\ y_2 = -0.5x - y_1 - 0.5y_3 + 20, \\ y_3 = -x - 2y_1 - 2y_2 + 40. \end{cases} \tag{42}$$

For the sake of brevity we omit here the constraints for each critical region.

Step 4. Incorporate the expressions (42) into F, and formulate 6 single level convex optimisation problems. They result in the same unique solution, as follows:

$$F = -35; \quad x = 5; \quad y_1 = 5; \quad y_2 = 10; \quad y_3 = 5.$$

The global optimum found is identical to the one reported in Anandalingman (1988).

4 An application to optimal control of multilevel systems

An important application of the proposed theory is the hierarchical control of dynamic systems (Başar and Selbuz 1979), as shown in Fig. 3.

In hierarchical control, the performance of a dynamic system is optimised within a complex structure with different objective functions at different levels, for instance as shown in Fig. 3 for a control structure involving two levels. In such a system, typically described by a discrete-time dynamic model:

$$x_{n+1} = A_n \cdot x_n + B_n^0 \cdot u_n + \sum_{i=1}^{m} B_n^i \cdot v_n^i, \tag{43}$$

we have a central controller, the leader, and m peripheral (local) controllers; x_n is the state vector of the system, u_n is the control vector of the central controller and v_n^i is the control vector of the ith local controller, all at time step n. Each local controller may have its own dynamics, which can be incorporated in Equation (43) (Başar and Selbuz 1979).

The goal is the optimisation of a quadratic objective function corresponding to the central controller:

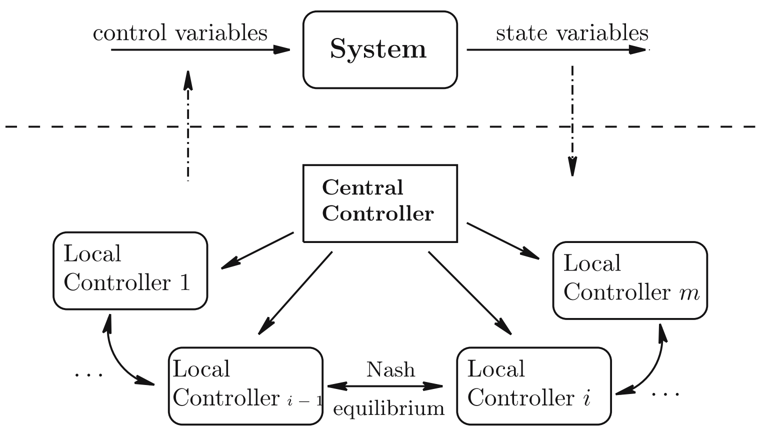

Fig. 3 Schematic representation of a hierarchical control configuration for a dynamic system

$$J_0 = (x_N)^T Q_N^0 x_N + \sum_{n=0}^{N-1} \left[(x_n)^T Q_n^0 x_n + (u_n)^T R_n^{00} u_n + \sum_{i=1}^{m} (v_n^i)^T R_n^{0i} v_n^i \right], \quad (44)$$

subject to the optimisation of each local controller's objective function:

$$J_i = (x_N)^T Q_N^i x_N + \sum_{n=0}^{N-1} \left[(x_n)^T Q_n^i x_n + (u_n)^T R_n^{i0} u_n + \sum_{k=1}^{m} (v_n^k)^T R_n^{ik} v_n^k \right]. \quad (45)$$

Expressions (43), (44) and (45) give rise to a multi-level optimisation problem formulation: the leader, central controller, has control over the complete set of optimisation variables, whereas the local controllers have access to their own optimisation set, v_n^i, and corresponding objective function. The aim is to obtain the global optimum for the central controller and optimal strategies for the local controllers. Here, we consider the general case involving constraints (where most previews strategies considered the unconstrained case—see (Cruz 1978; Başar and Selbuz 1979; Başar and Olsder 1982)).

We seek an *optimal policy*, as follows:

$$\{u_n\}^* = \{u_0^*, u_1^*, \ldots, u_N^*\} \rightarrow \gamma_0^*, \gamma_0^* \in \Gamma^0, \quad (46a)$$

$$\{v_n^1\}^* = \{(v_0^1)^*, (v_1^1)^*, \ldots, (v_N^1)^*\} \rightarrow \gamma_1^*, \gamma_1^* \in \Gamma_1, \quad (46b)$$

$$\vdots$$

$$\{v_n^i\}^* = \{(v_0^i)^*, (v_1^i)^*, \ldots, (v_N^i)^*\} \rightarrow \gamma_i^*, \gamma_i^* \in \Gamma_i, \quad (46c)$$

$$\vdots$$

$$\{v_n^m\}^* = \{(v_0^m)^*, (v_1^m)^*, \ldots, (v_N^m)^*\} \rightarrow \gamma_m^*, \gamma_2^* \in \Gamma_m. \quad (46d)$$

Then the hierarchical control problem can be recast as the following multi-level constrained optimisation problem:

$$\begin{aligned} &\min_{\gamma_0, \gamma_1, \ldots, \gamma_m} J_0(\gamma_0, \gamma_1, \ldots, \gamma_m), \qquad \text{(Central controller)},\\ &\quad \mathbf{s.t.}\ g_1(\gamma_0, \gamma_1, \ldots, \gamma_m) \le 0,\\ &\qquad \ldots, \left\{ \begin{array}{l} \min\limits_{\gamma_i} J_i(\gamma_0, \gamma_1, \ldots, \gamma_m) \\ \mathbf{s.t.}\ g_2^i(\gamma_0, \gamma_1, \ldots, \gamma_m), \le 0 \end{array} \right\}, \ldots. \ (m \text{ local controllers}). \end{aligned} \quad (47)$$

Using Eq. (43) it is possible to express each state variable as a function of the *initial state* and the *control decisions* (Pistikopoulos et al. 2000). Therefore, J_0 and J_i become functions only of the initial state: $J_0, J_i = f(x_0, \gamma_1, \gamma_2, \ldots, \gamma_m), \forall_i \in \{1, 2, \ldots, m\}$.

Since in the lower level of this two-level optimisation problem there are multiple optimisation subproblems, and there is the need to coordinate such group, it is fairly natural to assume a Nash equilibrium (Başar and Selbuz 1979):

$$J_1(\gamma_1^*, \ldots, \gamma_m^*) \le J_1(\gamma_1^*, \gamma_2, \gamma_3^*, \ldots, \gamma_K^*), \quad \forall_{\gamma_1} \in \Gamma_1, \tag{48a}$$

$$J_2(\gamma_1^*, \ldots, \gamma_m^*) \le J_1(\gamma_1^*, \gamma_2, \gamma_3^*, \ldots, \gamma_K^*), \quad \forall_{\gamma_2} \in \Gamma_2, \tag{48b}$$

$$\vdots$$

$$J_m(\gamma_1^*, \ldots, \gamma_m^*) \le J_0(\gamma_1, \gamma_2^*, \ldots, \gamma_{m-1}^*, \gamma_m^*), \quad \forall_{\gamma_m} \in \Gamma_m, \tag{48c}$$

where $\forall_{\gamma_0} \in \Gamma^0$ and $\forall_{x_0} \in X_0$, with X_0 being the feasible set of the system's initial state.

Problem (47) corresponds to a bilevel programming problem with multi-followers; the followers being the local controllers and the leader, the central controller. In contrast to Problem (31), the decisions involved in each subproblem are not only parametric relatively to the decisions of the remaining subproblems, but also depend on the initial state of the system. We refer to this class as *multi-level optimisation problems with uncertainty*. The algorithm in Table 3 can be directly applied to solve (47) only with a modification in *Step 4*, which requires 'the formulation and solution of K multi-parametric programming problems'.

A similar strategy can also be applied to tri-level optimisation problems. Moreover, if different models are involved in the subproblem, the proposed optimisation strategy is still applicable, with all control subproblems treated in a decentralised fashion. In the next section, a dynamic three person control system is described to illustrate the potential of the proposed approach.

4.1 Illustrative example 3

Consider a system which has a discrete dynamic behaviour described by the following linear state transition model (Nie et al. 2006):

$$\begin{aligned} x_{t+1} &= x_t + u_t - 2v_t^1 + v_t^2, \\ y_{t+1}^1 &= y_t^1 + 2v_t^1, \qquad\qquad t = 0, 1, 2, \\ y_{t+1}^2 &= y_t^2 + 2v_t^2, \end{aligned} \tag{49}$$

where u, v^1 and v^2 are input variables, and x, y^1 and y^2 output variables. And, with constraints on the input and state variables as follows:

$$\begin{aligned} -30 &\le v_t^1, v_t^2 \le 30, \\ -20 &\le u_t \le 20, \qquad\qquad t = 0, 1, 2, \\ -10 &\le x_0, y_0^1, y_0^2 \le 10. \end{aligned} \tag{50}$$

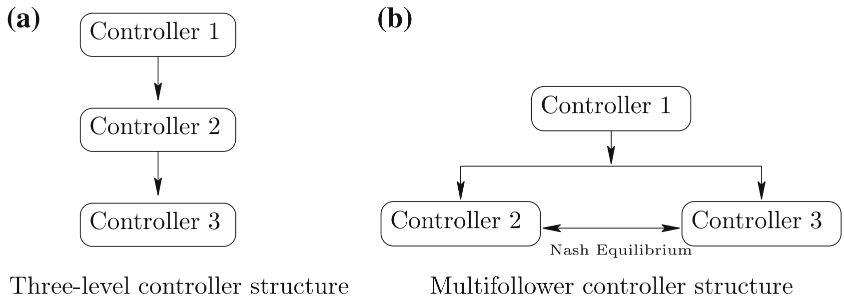

Fig. 4 Three-controller multilevel problem

Additionally, consider a three-controller system (Nie et al. 2006):

$$\mathbf{J_1} = \min_{u_0,u_1,u_2} 4x_3 + 3y_3^1 + 2y_3^2 + \sum_{t=0}^{2} \left\{ (u_t)^2 + \left(v_t^1\right)^2 - \left(v_t^2\right)^2 + 2u_t x_t + x_t^2 \right\}, \tag{51a}$$

$$\mathbf{J_2} = \min_{v_0^2,v_1^2,v_2^2} 2x_3 + 3y_3^2 + \sum_{t=0}^{2} \left\{ 2 \cdot u_t v_t^2 + \left(v_t^1 + 1\right)^2 + \left(v_t^2 + 1\right)^2 \right\}, \tag{51b}$$

$$\mathbf{J_3} = \min_{v_0^1,v_1^1,v_2^1} x_3 + 2y_3^1 - 10y_3^2 + \sum_{t=0}^{2} \left\{ -15u_t + \left(v_t^1 - 1\right)^2 - 2v_t^1 v_t^2 + \left(v_t^2\right)^2 \right\}, \tag{51c}$$

where J_1, J_2 and J_3 correspond to Controllers 1,2 and 3, respectively. Figure 4 displays two possible configurations for the control structure of the considered system.

The objective is then to derive suitable optimal strategies for the two controller structures. Case (a) of Fig. (4) corresponds to a three-level optimisation problem, whereas case (b) refers to a bilevel multi-follower optimisation problem. Therefore, using the proposed methodology, fully implemented in Matlab ®, we obtain the results summarised in Tables 4 and 5.

Table 4 Solution to the three-level optimisation problem

Critical region 1	Critical region 2	Critical region 3	Critical region 4
$u_0 = 6.84615 - 0.76928x_0$	$u_0 = -0.333333 - 1.8519x_0$	$u_0 = -1.53333 - 1.6889x_0$	$u_0 = -9 - 0.72732x_0$
$u_1 = -20$	$u_1 = -1.33333 + 2.8148x_0$	$u_1 = 8.26667 + 1.5111x_0$	$u_1 = 20$
$u_2 = 15.2308 + 0.15388x_0$	$u_2 = -2 - 2.4444x_0$	$u_2 = -20$	$u_2 = -20$
$-10 \le x_0 \le -6.63161$	$-6.63161 \le x_0 \le 7.36377$	$7.36377 \le x_0 \le 7.76466$	$7.76466 \le x_0 \le 10$

$v_0^1 = v_0^2 = -2 - 0.5u_0$; $v_1^1 = v_1^2 = -2 - 0.5u_1$; $v_2^1 = v_2^2 = -2 - 0.5u_2$

Table 5 Solution to multi-follower problem

Critical region 1
$u_0 = 1 - x_0$
$u_1 = -8 + x_0$
$u_2 = 5 - x_0$
$v_0^1 = v_0^2 = -6 + x_0$
$v_1^1 = v_1^2 = 3 - x_0$
$v_2^1 = v_2^2 = -10 + x_0$
$-10 \leq x_0 \leq 10$

5 Concluding remarks

We have described a novel global optimisation strategy for the solution of hierarchical multi-level and decentralised multi-level programs based on our recent developments in multi-parametric programming theory and algorithms (Pistikopoulos et al. 2007a,b). The algorithms proposed are suitable for problems involving general convex objective functions and convex sets of constraints.

Current research focus is towards general non-linear models, for which recent results on global multi-parametric programming (Dua et al. 2004) can be used; and general dynamic multi-level problems, for which a dynamic programming approach coupled with parametric programming can be applied (Faísca et al. 2007a).

Acknowledgments Financial support from EPSRC (GR/T02560/01) and Marie Curie European Project PRISM (MRTN-CT-2004-512233) is gratefully acknowledged.

References

Anandalingman G (1988) A mathematical programming model of decentralized multi-level systems. J Opl Res Soc 39:1021–1033

Başar T (1975) Equilibrium solutions in two-person quadratic decision problems with static information structures. IEEE Trans Automatic Control 20(3):320–328

Başar T (1978) Decentralized multicriteria optimization of linear stochastic systems. IEEE Trans Automatic Control 23(2):233–243

Başar T, Olsder GJ (1982) Dynamic Noncooperative game theory. Academic, London

Başar T, Selbuz H (1979) Closed-loop Stackelberg strategies with applications in the optimal control of multilevel systems. IEEE Trans Automatic Control 24(2):166–179

Choi SC, Desarbo WS, Harker PT (1990) Product positioning under price competition. Manage Sci 36(2):175–199

Clark PA (1983) Embedded optimization problems in chemical process design. PhD Thesis. Department of Chemical Engineering, Carnegie-Mellon University, Pittsburg

Cruz JB (1978) Leader-follower strategies for multilevel systems. IEEE Trans Automatic Control 23(2):244–255

Dua V, Papalexandri KP, Pistikopoulos EN (2004) Global optimization issues in multiparametric continuous and mixed-integer optimization problems. J Global Optim 30:59–89

Dua V, Bozinis NA, Pistikopoulos EN (2002) A multiparametric programming approach for mixed-integer quadratic engineering problems. Comput Chem Eng 26:715–733

Faísca NP, Kouramas KI, Pistikopoulos EN (2007a) Dynamic programming, Chap 7. Wiley-VCH. Weinheim

Faísca NP, Dua V, Saraiva PM, Rustem B, Pistikopoulos EN (2007b) Parametric global optimisation for bilevel programming. J Global Optim (in Press)

Fiacco AV(1976) Sensitivity analysis for nonlinear programming using penalty methods. Math Programm 10:287–311

Fiacco AV (1983) Introduction to sensitivity and stability analysis in nonlinear programming. Academic, New York

Floudas CA (2000) Deterministic global optimization. Kluwer, Dordrecht

Li M, Cruz JB, Simaan MA (2002) An approach to discrete-time incentive feedback stackelberg games. IEEE Trans Syst Man Cybern A: Syst Humans 32(4):472–481

Liu B (1998) Stackelberg–Nash equilibrium for multilevel programming with multiple followers using genetic algorithms. Comput Math Appl 36(7):79–89

McCormick GP (1976) Optimality criteria in nonlinear programming. SIAM-AMS Proc 1975(9):27

Morari M, Arkun Y, Stephanopoulos G (1980) Studies in the synthesis of control structures for chemical processes. AIChE J 26:220–232

Nie P, Chen L, Fukushima M (2006) Dynamic programming approach to discrete time dynamic feedback stackelberg games with independent and dependent followers. Euro J Oper Res 169:310–328

Pistikopoulos EN, Georgiadis MC, Dua V (2007a) Multi-parametric programming: theory, algorithms, and applications, vol 1. Wiley-VCH, Weinheim

Pistikopoulos EN, Georgiadis MC, Dua V (2007b) Multi-parametric model-based control: theory and applications, vol 2. Wiley-VCH, Weinheim

Pistikopoulos EN, Dua V, Bozinis NA, Bemporad A, Morari M (2000) On-line optimization via off-line parametric optimization tools. Comput Chem Eng 24:183–188

Rodić AD, Vukobratović MK (1999) Contribution to the integrated control synthesis of road vehicles. IEEE Trans Control Syst Technol 7(1):64–78

Ruan GZ, Wang SY, Yamakamoto Y, Zhu SS (2004) Optimality conditions and geometric properties of linear multilevel programming problem with dominated objective functions. J Optim Theory Appl 123(2):409–429

Shih HS, Wen UP, Lee ES, Lan KM, Hsiao HC (2004) A neural network approach to multiobjective and multilevel programming problems. Comput Math Appl 48:95–108

Stephanopoulos G, Ng C (2000) Perspectives on the synthesis of plant-wide control structures. J Process Control 10:97–111

Tolwinski B (1981) Closed-loop stackelberg solution to a multistage linear-quadratic game. J Optim Theory Appl 34(4):485–501

Vicente LN (1992) Bilevel programming. Master's Thesis. Department of Mathematics, University of Coimbra, Coimbra

Vicente LN, Calamai PH (1994) Bilevel and multilevel programming: a bibliography review. J Global Optim 5(3):291–306