Min-Max MPC using a tractable QP Problem [⋆]

T. Alamo [a], D.R. Ramirez [a], D. Muñoz de la Peña [a], E.F. Camacho [a]

[a] *Departamento de Ingeniería de Sistemas y Automática. Universidad de Sevilla.*
Camino de los Descubrimientos s/n. 41092 Sevilla. Spain.

Abstract

Min-Max MPC (MMMPC) controllers suffer from a great computational burden that is often circumvented by using approximate solutions or upper bounds of the worst possible case of a performance index. These upper bounds are usually computed by means of linear matrix inequalities (LMI) techniques. In this paper a more efficient approach is shown. This paper proposes a computationally efficient MMMPC control strategy in which a close approximation of the solution of the min-max problem is computed using a quadratic programming problem. The overall computational burden is much lower than that of the min-max problem and the resulting control is shown to have guaranteed stability. A Simulation example is given in the paper.

Key words: Robust control; Uncertain systems; Min-max; Model predictive control

1 Introduction

Min-max model predictive controllers (MMMPC) have been limited to a narrow field of applications due to its great computational burden. The computation of the control signal to be applied implies the minimization of the worst case of a performance index. Solving this problem can be very time consuming as it is of the NP-hard kind (Veres and Norton (1993); Lee and Yu (1997); Scokaert and Mayne (1998)). This computational burden has led to a lack of experimental results. For moderate fast dynamics the min-max problem can be solved numerically only when the number of extreme realizations of the uncertainty is relatively low, i.e., when the prediction horizon is small. Also a complexity reduction strategy like that of Alamo et al. (2005b) can be used, but, when fast dynamics are to be controlled the only choice is to use approximate solutions (Ramírez et al. (2004); Alamo et al. (2005a)).

MMMPC controllers can be divided into two types: open-loop and closed-loop min-max predictive controllers. In the first type, predictions are computed in an open-loop manner (although the resulting controller is a feedback controller). Closed-loop min-max predictive controllers (see Lee and Yu (1997); Scokaert and Mayne (1998); Bemporad et al. (2003); Kerrigan and

[⋆] This paper is a revised and expanded version of a paper presented at the joint conference CDC-ECC 2005.
Email addresses: alamo@cartuja.us.es (T. Alamo), danirr@cartuja.us.es (D.R. Ramirez), davidmps@cartuja.us.es (D. Muñoz de la Peña), eduardo@cartuja.us.es (E.F. Camacho).

Maciejowski (2004) and references in Ramirez and Camacho (2006)) take into account that the control law is actually applied in a feedback manner when computing the predictions. Open-loop MMMPC is known to be very conservative whereas closed-loop MMMPC is known to suffer from a much greater computational burden. The results presented in the paper can be applied to open-loop prediction MMMPC or to MMMPC using a semi-feedback strategy (Rossiter et al. (1998)). In this, some kind of feedback is introduced into the predictions because the system is pre-controlled using an inner feedback gain. This technique is known to reduce the conservatism of open-loop predictive controllers (Bemporad (1998); Löfberg (2003)) without having to increase the computational burden.

Multi-parametric programming has been applied to show that the MMMPC control law is piecewise affine when a quadratic (Ramirez and Camacho (2006)) or 1-norm based criterion (Bemporad et al. (2003); Kerrigan and Maciejowski (2004)) is used as the cost function. Thus, explicit forms of the control law can be built. Such explicit forms can be evaluated very fast provided that the complexity of the state space partition is moderate, which it is the case for many applications. If the process model or the controller tuning parameters change, however, the computation of the explicit form of the control law has to be done again.

A common solution to the computational burden issue is to use an upper bound of the worst case cost instead of computing it explicitly. This upper bound can be efficiently computed by using linear matrix inequalities (LMI) techniques such as in Kothare et al. (1996); Casavola et al. (2000); Lu and Arkun (2000); Wan and Kothare (2003). The LMI problems have a computational burden that cannot be neglected in certain applications. In Ramirez et al. (2006) is presented a different approach based on a computationally cheap upper-bound of the worst case cost. It proved to be very close to the original MMMPC but it requires nonlinear optimization of a non convex cost function. Moreover, the cost function, although computationally efficient, still requires more computations that in the case of a conventional constrained MPC. As a result, the computational burden is much lower than that of the exact MMMPC but it is still much higher than that of a conventional constrained MPC. Moreover, stability results were not provided.

This paper proposes a different strategy in which the min-max problem is replaced by a quadratic programming (QP) problem that provides a close approximation to the solution of the original min-max problem. The computational burden is much lower than that of the min-max problem and also lower than that of Ramirez et al. (2006). Moreover it is comparable to that of a standard constrained MPC based on a quadratic cost. Thus, it can be easily implemented in almost any platform capable to run a constrained MPC. Stability of the proposed approach is guaranteed as it is shown in the paper.

The paper is organized as follows: section 2 presents the MMMPC controller and some of its properties. Section 3 presents the efficient approximation of the solution of the min-max problem based on a QP problem. Robust stability of the proposed controller is shown in section 4. The strategy is illustrated by means of a simulation example in section 5. Finally, section 6 presents some conclusions.

2 Min-Max MPC with bounded additive uncertainties

Consider the following state space model with bounded additive uncertainties (Camacho and Bordóns (2004)):

$$x(t+1) = Ax(t) + Bu(t) + D\theta(t+1) \tag{1}$$

with $x(t) \in \mathbb{R}^{dimx}$ the state vector, $u(t) \in \mathbb{R}^{dimu}$ the input vector and $\theta(t) \in \{\theta \in \mathbb{R}^{dim\theta} : \|\theta\|_\infty \leq \varepsilon\}$ the uncertainty, that is supposed to be bounded. The system is subject to p state and input time invariant constraints.

$$F_u u(t) + F_x x(t) \leq g.$$

where $F_u \in \mathbb{R}^{p\times dimu}$ and $F_x \in \mathbb{R}^{p\times dimx}$.

Although the results presented in this paper are not valid in general for closed-loop MMMPC with a quadratic cost (see Mayne et al. (2000) and references therein), they are valid when using a semi-feedback approach in which the control input is given by

$$u(t) = -Kx(t) + v(t), \tag{2}$$

where the feedback matrix K is chosen to achieve some desired property such as nominal stability or LQR optimality without constraints. The MMMPC controller will compute the optimal sequence of correction control inputs $v(t)$. The state equation of system (1) can be rewritten as

$$x(t+1) = A_{CL}x(t) + Bv(t) + D\theta(t+1), \tag{3}$$

with $A_{CL} = (A - BK)$. In the following it will be assumed that the semi-feedback approach is used.

The cost function is a quadratic performance index of the form:

$$V(x,\mathbf{v},\boldsymbol{\theta}) = \sum_{j=0}^{N-1} x(t+j|t)^T Qx(t+j|t) + \sum_{j=0}^{N-1} u(t+j|t)^T Ru(t+j|t) + x(t+N|t)^T Px(t+N|t) \tag{4}$$

where $x(t+j|t)$ is the prediction of the state for $t+j$ made at t and $u(t+j|t) = -Kx(t+j|t) + v(t+j|t)$. Note that these values depend on the future values of the uncertainty. Matrices $Q, P \in \mathbb{R}^{dimx\times dimx}$ and $R \in \mathbb{R}^{dimu\times dimu}$ are symmetric positive definite matrices used as weighting parameters.

Min-Max MPC (Campo and Morari (1987)) is based on finding the control correction sequence $\mathbf{v} = \left[v(t|t)^T, \cdots, v(t+N-1|t)^T\right]^T$ that minimizes the cost function for the worst possible case of the predicted future evolution of the process state or output signal, while guaranteeing robust constraint satisfaction. This is accomplished through the solution of the following min-max problem:

$$\begin{aligned} \mathbf{v}^*(x) = \ & \arg\min_{\mathbf{v}} \max_{\boldsymbol{\theta}\in\Theta} \quad V(x,\mathbf{v},\boldsymbol{\theta}) \\ & \qquad \text{s.t. } F_u u(t+j|t) + F_x x(t+j|t) \leq g, \quad j = 0,\ldots,N, \ \forall \boldsymbol{\theta} \in \Theta, \\ & \qquad\quad x(t+N|t) \in \Omega, \ \forall \boldsymbol{\theta} \in \Theta, \end{aligned} \tag{5}$$

with $x(t|t) = x$, where $\boldsymbol{\theta} = \left[\theta(t+1)^T, \cdots, \theta(t+N)^T\right]^T$ is a sequence of future values of $\theta(t)$ over a prediction horizon N, and $\boldsymbol{\Theta} = \{\boldsymbol{\theta} \in \mathbb{R}^{N\cdot dim\theta} : \|\boldsymbol{\theta}\|_\infty \leq \varepsilon\}$ is the set of possible uncertainty trajectories. A terminal region constraint $x(t+N|t) \in \Omega$, where

Ω is a polyhedron, is included to assure stability of the control law (Mayne et al. (2000)) [1] . As usual in all predictive control schemes, the solution of problem (5) is applied in a feedback manner using a receding horizon strategy.

The predictions $x(t+j|t)$ and $u(t+j|t)$ depend linearly on x, $\mathbf{v}$ and $\boldsymbol{\theta}$. This means that it is possible to find a vector $d \in \mathbb{R}^p$ and matrices G_x, G_v and G_θ such that all the robust linear constraints of problem (5) can be rewritten as (Camacho and Bordóns (2004)):

$$G_x^i x + G_v^i \mathbf{v} + G_\theta^i \boldsymbol{\theta} \le d_i, \quad i = 1\ldots, p, \;\; \forall \boldsymbol{\theta} \in \Theta,$$

where G_x^i, G_v^i, G_θ^i denote the i-th rows of G_x, G_v and G_θ respectively and d_i is the i-th component of $d \in \mathbb{R}^p$. Denote now $\|G_\theta^i\|_1$ the sum of the absolute values of row G_θ^i. Taking into account that $\max_{\boldsymbol{\theta}\in\Theta} G_\theta^i \boldsymbol{\theta} = \max_{\|\boldsymbol{\theta}\|_\infty \le \varepsilon} G_\theta^i \boldsymbol{\theta} = \varepsilon \|G_\theta^i\|_1$, the robust fulfillment of the constraints is satisfied if and only if: $G_x^i x + G_v^i \mathbf{v} + \varepsilon \|G_\theta^i\|_1 \le d_i, \;\; i = 1, \ldots, p$. Therefore, to guarantee robust constraint satisfaction, the following set of linear constraint must be satisfied:

$$G_x x + G_v \mathbf{v} \le d_\varepsilon.$$

where the i-th component of d_ε is equal to $d_i - \varepsilon \|G_\theta^i\|_1$. Note that this is a necessary and sufficient condition.

Taking into account (3),(2) and the quadratic nature of the performance index, the cost function can be evaluated as a quadratic function on the initial state, the control correction vector and the uncertain trajectory.

$$V(x, \mathbf{v}, \boldsymbol{\theta}) = \mathbf{v}^T M_{vv} \mathbf{v} + \boldsymbol{\theta}^T M_{\theta\theta} \boldsymbol{\theta} + 2\boldsymbol{\theta}^T M_{\theta v} \mathbf{v} + 2x^T M_{vf}^T \mathbf{v} + 2x^T M_{\theta f}^T \boldsymbol{\theta} + x^T M_{ff} x \tag{6}$$

where the matrices can be obtained from the system and the control parameters (Camacho and Bordóns (2004)).

It can be seen that $M_{\theta\theta}$ is a Gram matrix and therefore at least positive semidefinite. On the other hand, M_{vv} is positive definite because $R > 0$. Note that as $M_{\theta\theta} \ge 0$, $V(x, \mathbf{v}, \boldsymbol{\theta})$ is convex on $\boldsymbol{\theta}$, and because $M_{vv} > 0$ strictly convex on $\mathbf{v}$. Therefore the solution of (5) will be unique Camacho and Bordóns (2004). Moreover, due to the convexity on $\boldsymbol{\theta}$ the maximum is attained at least at one of the vertices of the hypercube described by $\boldsymbol{\Theta}$. Taking this into account, problem (5) is equivalent to

$$\begin{aligned} \mathbf{v}^*(x) = \arg\, & \min_{\mathbf{v}} \max_{\boldsymbol{\theta}\in \text{vert}(\boldsymbol{\Theta})} V(x, \mathbf{v}, \boldsymbol{\theta}) \\ & \text{s.t.} \;\; G_x x + G_v \mathbf{v} \le d_\varepsilon \end{aligned} \tag{7}$$

where vert($\boldsymbol{\Theta}$) is the set of vertices of $\boldsymbol{\Theta}$.

The terminal region Ω is assumed to satisfy the following conditions:

- **C1**: If $x \in \Omega$ then $A_{CL}x + D\theta \in \Omega$, for every $\theta \in \{\theta \in \mathbb{R}^{dim\theta} : \|\theta\|_\infty \le \varepsilon\}$.

[1] In this paper we have used standard assumptions in order to prove stability. There are other frameworks to ensure stability such as the recent works El-Farra et al. (2004b,a); Mhaskar et al. (2005), that use a hybrid predictive control structure which brings together MPC and Lyapunov-based bounded control techniques and allows for an explicit characterization of the stability region in which any MPC formulation can be embedded with guaranteed closed-loop stability.

- **C2**: If $x \in \Omega$ then $u(x) = -Kx \in U$, where $U \triangleq \{u : F_u u + F_x x \leq g\}$.

Moreover, matrix P that characterizes the terminal cost is assumed to satisfy

- **C3**: $P - A_{CL}^T P A_{CL} > Q + K^T R K$.

The stability of A_{CL} guarantees the existence of a positive definite matrix P satisfying C3.

The maximum cost for a given x and $\mathbf{v}$ is denoted as

$$V^*(x,\mathbf{v}) = \max_{\boldsymbol{\theta} \in \text{vert}(\boldsymbol{\Theta})} V(x,\mathbf{v},\boldsymbol{\theta}) = V(x,\mathbf{v},0) + \max_{\boldsymbol{\theta} \in \text{vert}(\Theta)} \boldsymbol{\theta}^T H \boldsymbol{\theta} + 2\boldsymbol{\theta}^T q(x,\mathbf{v}) \tag{8}$$

where $H = M_{\theta\theta}$, $q(x,\mathbf{v}) = M_{\theta v}\mathbf{v} + M_{\theta f} x$ and $V(x,\mathbf{v},0) = \mathbf{v}^T M_{vv} \mathbf{v} + 2x^T M_{vf}^T \mathbf{v} + x^T M_{ff} x$ is the part of the cost that does not depend on the uncertainty. With this definition, problem (7) can be rewritten as

$$\begin{aligned} \mathbf{v}^*(x) = \; & \arg\min_{\mathbf{v}} V^*(x,\mathbf{v}) \\ & \text{s.t. } G_x x + G_v \mathbf{v} \leq d_\varepsilon, \end{aligned} \tag{9}$$

and the system is controlled by $K_{MPC}(x(t)) = -Kx(t) + v^*(t|t)$, where $\mathbf{v}^*(x(t)) = \left[v^*(t|t)^T, \cdots, v^*(t+N-1|t)^T\right]^T$.

In order to evaluate $V^*(x,\mathbf{v})$ it is necessary to evaluate the function for all the vertices of Θ. Taking into account that the number of vertices is $2^{N*dim\theta}$ it is clear that the problem cannot be solved in real time beyond a certain dimension of $\boldsymbol{\theta}$ (because this is a well known NP-hard problem).

3 A QP approach to Min-Max MPC

In this section it is shown how the min-max problem (9) can be replaced by a tractable QP problem which provides a close approximation of the solution of the original problem. The strategy can be summarized in the following steps:

(1) Obtain an initial guess of the solution of (9), denoted by $\tilde{\mathbf{v}}^*$. As seen later, this can be achieved solving a QP problem.
(2) Using $\tilde{\mathbf{v}}^*$, obtain a quadratic function of $\mathbf{v}$ that bounds the worst case cost.
(3) Compute the control law. This involves the solution of a QP problem.

In the following all these steps will be detailed.

3.1 Computing $\tilde{\mathbf{v}}^*$

Given H defined as in equation (8), denote $T_i = \sum_{j=1}^{N \cdot dim\theta} |H_{ij}|$, where H_{ij} denotes the (i,j)-th component of matrix H. Then, define the diagonal matrix T as

$$T = \begin{bmatrix} T_1 & & \\ & \ddots & \\ & & T_n \end{bmatrix} \tag{10}$$

Because of how matrix T is defined, $T - H$ is a symmetric diagonally dominant real matrix with nonnegative diagonal entries, thus $T - H \geq 0$ which implies that $T \geq H$. Let $\tilde{V}(x, \mathbf{v}, \boldsymbol{\theta})$ be:

$$\tilde{V}(x, \mathbf{v}, \boldsymbol{\theta}) = V(x, \mathbf{v}, 0) + \boldsymbol{\theta}^T T \boldsymbol{\theta} + 2q^T(x, \mathbf{v})\boldsymbol{\theta} \tag{11}$$

From the inequality $T \geq H$ it is inferred that $\tilde{V}(x, \mathbf{v}, \boldsymbol{\theta}) \geq V(x, \mathbf{v}, \boldsymbol{\theta})$. The maximum of $\tilde{V}(x, \mathbf{v}, \boldsymbol{\theta})$ can be computed as

$$\begin{aligned} \tilde{V}^*(x, \mathbf{v}) &= \max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \tilde{V}(x, \mathbf{v}, \boldsymbol{\theta}) \\ &= V(x, \mathbf{v}, 0) + \text{trace}(T)\varepsilon^2 + 2\varepsilon \left\| q(x, \mathbf{v}) \right\|_1 = V(x, \mathbf{v}, 0) + \left\| H \right\|_s \varepsilon^2 + 2\varepsilon \left\| q(x, \mathbf{v}) \right\|_1 \end{aligned} \tag{12}$$

where $\|H\|_s$ denotes the sum of the absolute values of the elements of H. Then an initial guess of the solution of (9) can be obtained as

$$\begin{aligned} \tilde{\mathbf{v}}^*(x) = \arg\min_{\tilde{\mathbf{v}}} \ & \tilde{V}^*(x, \tilde{\mathbf{v}}) \\ \text{s.t. } & G_x x + G_v \tilde{\mathbf{v}} \leq d_\varepsilon, \end{aligned} \tag{13}$$

It is clear that this problem can be casted as a QP problem.

3.2 Obtaining an upper bound of the worst case cost

The upper-bound of the maximum will be obtained in two steps. In the first one we compute a set of parameters from $\tilde{\mathbf{v}}^*$ that allows us later, in the second step, to compute the bound as a quadratic function of $\mathbf{v}$. The first step is based on the diagonalization procedure of Ramirez et al. (2006).

3.2.1 Computing the parameter vector $\alpha(\mathbf{v})$

Note that:

$$V^*(x, \mathbf{v}) = \max_{\boldsymbol{\theta} \in \text{vert}(\boldsymbol{\Theta})} \begin{bmatrix} \boldsymbol{\theta} \\ 1 \end{bmatrix}^T M(\mathbf{v}) \begin{bmatrix} \boldsymbol{\theta} \\ 1 \end{bmatrix} = \max_{\|z\|_\infty \leq 1} z^T M(\mathbf{v}) z \tag{14}$$

with $z = \begin{bmatrix} \frac{\boldsymbol{\theta}^T}{\varepsilon} & 1 \end{bmatrix}^T$ and $M(\mathbf{v}) = \begin{bmatrix} \varepsilon^2 H & \varepsilon q(x,\mathbf{v}) \\ \varepsilon q^T(x,\mathbf{v}) & V(x,\mathbf{v},0) \end{bmatrix} \in \mathbb{R}^{n\times n}$, where $n = N \cdot dim\theta + 1$.

The following procedure, which is based on that presented in Ramirez et al. (2006), provides an upper bound of the worst case cost for a given $\mathbf{v}$. It computes $\alpha(\mathbf{v}) = [\alpha_1(\mathbf{v}),\dots,\alpha_{n-1}(\mathbf{v})]^T$ and a diagonal matrix $\Gamma(\mathbf{v}) \geq M(\mathbf{v})$ such that its trace is an upper bound of the worst case cost for $\mathbf{v}$ (see property 1).

Procedure 1 *Computation of $\alpha(\mathbf{v}) = [\alpha_1(\mathbf{v}),\dots,\alpha_{n-1}(\mathbf{v})]^T$ and $\Gamma(\mathbf{v})$.*

```
(1) Let $S^{(0} = M(\mathbf{v}) \in \mathbb{R}^{n\times n}$.
(2) For $k = 1$ to $n-1$
(3)     Let $M_{sub}^{(k} = [S_{ij}^{(k-1}]$ for $i,j = k\cdots n$.
(4)     Obtain the partition $M_{sub}^{(k} = \begin{bmatrix} a & b^T \\ b & M_r \end{bmatrix}$, where $a \in \mathbb{R}$, $b \in \mathbb{R}^{n-k}$ and $M_r \in \mathbb{R}^{(n-k)\times(n-k)}$ .
(5)     Make $\alpha_k(\mathbf{v}) = \sqrt{\|b\|_1}$.
(6)     If $\alpha_k(\mathbf{v}) = 0$ then $S^{(k} = S^{(k-1}$, else $S^{(k} = S^{(k-1} + \begin{bmatrix} \mathbf{0}_{k-1,1} \\ \alpha_k(\mathbf{v}) \\ \frac{-b}{\alpha_k(\mathbf{v})} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{k-1,1} \\ \alpha_k(\mathbf{v}) \\ \frac{-b}{\alpha_k(\mathbf{v})} \end{bmatrix}^T$ .
(7) end for
(8) Make $\Gamma(\mathbf{v}) = S^{(n-1}$.
```

Note that in previous procedure, $\mathbf{0}_{m,n}$ denotes a $(m \times n)$ matrix of zeros. The following property shows that the trace of $\Gamma(\mathbf{v})$ constitutes an improved upper bound of $V^*(x,\mathbf{v})$. That is, $V^*(x,\mathbf{v}) \leq \text{trace}\,(\Gamma(\mathbf{v})) \leq \tilde{V}^*(x,\mathbf{v})$.

Property 1 *Matrices $S^{(0}$, $S^{(1}$, ..., $S^{(n-1}$, obtained by means of procedure 1 satisfy:*

(i) $S^{(k}$ *is a partially diagonalized matrix. That is, there is a diagonal matrix* $T^{(k} \in \mathbb{R}^{k\times k}$ *such that* $S^{(k} = \begin{bmatrix} T^{(k} & 0 \\ 0 & M_{sub}^{(k} \end{bmatrix}$.

(ii) $S^{(n-1} = \Gamma(\mathbf{v})$ *is a diagonal matrix.*

(iii) $V^*(x,\mathbf{v}) \leq \text{trace}\,(\Gamma(\mathbf{v}))$.

(iv) $\|S^{(k}\|_s \leq \|S^{(k-1}\|_s$.

(v) $\text{trace}\,(\Gamma(\mathbf{v})) \leq \tilde{V}^*(x,\mathbf{v})$, $\forall \mathbf{v}$.

PROOF: SEE APPENDIX A

Remark 1 *This procedure is the base to obtain a QP problem that provides a solution with a worst case cost that it is close to the optimal worst case cost but with the advantage of the lower computational burden of a QP problem (see section 3.2.2).*

3.2.2 Obtaining the bound as a quadratic function on $\mathbf{v}$

The diagonalization process shown in 3.2.1 can be used to obtain a matrix denoted by $\hat{\Gamma}(\mathbf{v})$, which allows to obtain a bound of the maximum that can be computed as a quadratic function of $\mathbf{v}$. This is achieved by means of the following procedure:

Procedure 2 *Obtaining the matrix* $\hat{\Gamma}(\mathbf{v})$.

```
(1) Obtain $\tilde{\mathbf{v}}^*$ from the QP problem defined in equation (13).
(2) Compute $\alpha(\tilde{\mathbf{v}}^*)$ by procedure 1.
(3) Let $\hat{S}^{(0}(\mathbf{v}) = M(\mathbf{v}) \in \mathbb{R}^{n\times n}$.
(4) For $k = 1$ to $n-1$
(5)     Let $\hat{M}_{sub}(\mathbf{v}) = [\hat{S}_{ij}^{(k-1}(\mathbf{v})]$ for $i,j = k\cdots n$.
(6)     Obtain the partition $\hat{M}_{sub}(\mathbf{v}) = \begin{bmatrix} a(\mathbf{v}) & b^T(\mathbf{v}) \\ b(\mathbf{v}) & M_r(\mathbf{v}) \end{bmatrix}$, where $a(\mathbf{v}) \in \mathbb{R}$.
(7)     If $\alpha_k(\tilde{\mathbf{v}}^*) = 0$ then $\hat{S}^{(k}(\mathbf{v}) = \hat{S}^{(k-1}(\mathbf{v})$, else $\hat{S}^{(k}(\mathbf{v}) = \hat{S}^{(k-1}(\mathbf{v}) + \begin{bmatrix} \mathbf{0}_{k-1,1} \\ \alpha_k(\tilde{\mathbf{v}}^*) \\ \frac{-b(\mathbf{v})}{\alpha_k(\tilde{\mathbf{v}}^*)} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{k-1,1} \\ \alpha_k(\tilde{\mathbf{v}}^*) \\ \frac{-b(\mathbf{v})}{\alpha_k(\tilde{\mathbf{v}}^*)} \end{bmatrix}^T$.
(8) end for
(9) Make $\hat{\Gamma}(\mathbf{v}) = \hat{S}^{(n-1}(\mathbf{v})$.
```

Theorem 1 *Denote* $\hat{V}^*(x,\mathbf{v}) = \text{trace}\,(\hat{\Gamma}(\mathbf{v}))$. *Then*

(1) $\hat{\Gamma}(\tilde{\mathbf{v}}^*) = \Gamma(\tilde{\mathbf{v}}^*)$.

(2) $\hat{V}^*(x,\mathbf{v})$ *is a quadratic function on* $\mathbf{v}$.

(3) $V^*(x,\mathbf{v}) \leq \hat{V}^*(x,\mathbf{v})$.

PROOF: SEE APPENDIX A

3.3 Computing the control law

The value of the control signal is obtained solving the following QP optimization problem

$$\begin{aligned} \hat{\mathbf{v}}^*(x) = \arg\min_{\hat{\mathbf{v}}}\ & \hat{V}^*(x,\hat{\mathbf{v}}) \\ & \text{s.t. } G_x x + G_v \hat{\mathbf{v}} \leq d_\varepsilon, \end{aligned} \tag{15}$$

and the system is controlled by $\hat{K}_{MPC}(x(t)) = -Kx(t) + \hat{v}^*(t|t)$, where $\hat{v}^*(t|t)$ is the first element of $\hat{\mathbf{v}}^*(x)$.

Remark 1 *Note that, in order to reduce the difference between the solution of the exact min-max problem and* $\hat{\mathbf{v}}^*(x)$ *the whole procedure can be applied twice or more times using at each subsequent step the solution obtained in the previous step as the*

dimx	6	10	14	18
4	5.37×10^3	1.94×10^4	4.62×10^4	1.31×10^5
8	2.67×10^4	1.03×10^5	2.74×10^5	5.3×10^5

Table 1

Average computational burden (measured in flops) required to compute the control law for different values of the controller horizons and dimension of x. For each entry, 10 simulations (each of 100 samples) with random systems have been computed. The prediction and control horizons are equal.

initial guess used at procedure 1. Thus for the first time two QP problems must be solved but for each subsequent step only one QP is necessary.

The computational burden of the proposed strategy is clearly much lower than that of the exact MMMPC. It is mostly due to the QP problems that must be solved to obtain the initial guess and the proposed solution itself. Note that in both cases, the complexity of each problem is the same as that of a standard constrained MPC using a quadratic cost function. The complexity is strongly related to the size of $\mathbf{v}$, that is, the number of control moves that must be computed. The dimension of the state vector also affects, but less than in the exact MMMPC in which the number of vertices to be considered in the maximization problem is $2^{dimx \times N}$. To illustrate this, consider table 1, in which it is presented the computational burden as a function of the horizons and the size of x of a set of random systems.

4 Stability of the proposed control law

In this section the stability properties of the control $\hat{K}_{MPC}(x(t))$ are shown. First some properties are presented and then stability is proved. Recall that $\mathbf{v}^*$, $\tilde{\mathbf{v}}^*$ and $\hat{\mathbf{v}}^*$ are the solutions of (9), (13) and (15) respectively. Denote also $J(x) = V^*(x, \mathbf{v}^*)$, $\tilde{J}(x) = \tilde{V}^*(x, \tilde{\mathbf{v}}^*)$ and $\hat{J}(x) = \hat{V}^*(x, \hat{\mathbf{v}}^*)$. Note that problems (9), (13) and (15) have the same feasibility region as the constraints are the same. The following property will be used to proof the stability of the control law.

Property 2

$$\hat{J}(x) \leq \tilde{J}(x)$$

Proof:

As $\hat{\mathbf{v}}^*$ is the minimizer of $\hat{V}^*(x, \mathbf{v})$ it follows that:

$$\hat{J}(x) = \hat{V}^*(x, \hat{\mathbf{v}}^*) \leq \hat{V}^*(x, \tilde{\mathbf{v}}^*) \tag{16}$$

Thus, in order to prove that $\hat{J}(x) \leq \tilde{J}(x)$ it suffices to show that $\hat{V}^*(x, \tilde{\mathbf{v}}^*) \leq \tilde{V}^*(x, \tilde{\mathbf{v}}^*) = \tilde{J}(x)$. As it was shown in theorem 1, $\hat{V}^*(x, \tilde{\mathbf{v}}^*) = \text{trace}\,(\hat{\Gamma}(\tilde{\mathbf{v}}^*)) = \text{trace}\,(\Gamma(\tilde{\mathbf{v}}^*))$. Moreover, from property 1, $\text{trace}\,(\Gamma(\tilde{\mathbf{v}}^*)) \leq \tilde{V}^*(x, \tilde{\mathbf{v}}^*)$. Thus,

$$\hat{V}^*(x, \tilde{\mathbf{v}}^*) = \text{trace}\,(\hat{\Gamma}(\tilde{\mathbf{v}}^*)) = \text{trace}\,(\Gamma(\tilde{\mathbf{v}}^*)) \leq \tilde{V}^*(x, \tilde{\mathbf{v}}^*) = \tilde{J}(x)$$

It is then proved that $\hat{V}^*(x, \tilde{\mathbf{v}}^*) \leq \tilde{J}(x)$ and this completes the proof. □

It is clear that the optimal solution $\hat{\mathbf{v}}^*$ of problem (15) is a suboptimal feasible solution of the original min-max problem (9). As it is claimed in the following property, the difference between the optimal value of the original objective function and the value obtained with $\hat{\mathbf{v}}^*$ is bounded by $\|H\|_s\,\varepsilon^2$ (see (8) for the definition of H). Note that this result gives an implicit measure of how well $\hat{\mathbf{v}}^*$ approximates the solution of the original min-max problem (9).

Property 3 *It holds that:*

$$V^*(x,\hat{\mathbf{v}}^*) - \sigma\varepsilon^2 \leq J(x),$$

where $\sigma = \|H\|_s$.

PROOF: SEE APPENDIX A

The following property, which is proved in Alamo et al. (2005a) will be used in the proof of the stability of the proposed approach (see theorem 2 below).

Property 4 *Consider that assumptions C1, C2 and C3 are satisfied. Let $\mathbf{v} = \left[v(t|t)^T, \cdots, v(t+N-1|t)^T\right]^T$ and $\mathbf{v}_s$ a shifted version of $\mathbf{v}$ computed as $\mathbf{v}_s = \left[v(t+1|t)^T, v(t+2|t)^T, \cdots, v(t+N-1|t)^T, 0\right]^T$. If $\mathbf{v}$ is feasible for problem (9) at $x(t)$ then $\mathbf{v}_s$ is also feasible at $x(t+1)$ and there is $\gamma > 0$ such that:*

$$V^*(x(t+1),\mathbf{v}_s) \leq V^*(x(t),\mathbf{v}) - x(t)^T Q x(t) + \gamma\varepsilon^2$$

The following theorem states the stabilizing properties of the proposed control law.

Theorem 2 *Under assumptions C1, C2 and C3, the control law given by $\mathbf{u}(x(t)) = -Kx(t) + \hat{v}^*(t|t)$ stabilizes system (1).*

Proof: Let $\hat{\mathbf{v}}_s^*$ be the shifted version (as in property 4) of $\hat{\mathbf{v}}^*$. Due to the non optimality of $\hat{\mathbf{v}}_s^*$ for problem (9) it holds that

$$J(x(t+1)) \leq V^*(x(t+1),\hat{\mathbf{v}}_s^*) \tag{17}$$

Note that $\hat{\mathbf{v}}_s^*$ is feasible for both (15) and (9), thus by property 4

$$V^*(x(t+1),\hat{\mathbf{v}}_s^*) \leq V^*(x(t),\hat{\mathbf{v}}^*) - x(t)^T Q x(t) + \gamma\varepsilon^2 \tag{18}$$

Furthermore, by property (3)

$$V^*(x(t),\hat{\mathbf{v}}^*) \leq J(x(t)) + \sigma\varepsilon^2$$

Substituting this inequality in (18) and using (17):

$$J(x(t+1)) \leq J(x(t)) - x(t)^T Q x(t) + (\gamma+\sigma)\varepsilon^2$$

which can be rewritten

$$J(x(t+1)) - J(x(t)) \leq -x(t)^T Q x(t) + (\gamma+\sigma)\varepsilon^2 \tag{19}$$

Note that $(\gamma+\sigma)\varepsilon^2 > 0$ and that $-x(t)^T Qx(t) \leq 0$, thus it is ensured that the optimal worst case cost will decrease, i.e. $J(x(t+1)) - J(x(t)) < 0$, as long as $x(t)^T Qx(t) > (\gamma+\sigma)\varepsilon^2$. Define

$$\Phi_\varepsilon = \left\{ x \in \mathbb{R}^{dimx} : (9) \text{ is feasible and } x^T Qx \leq (\gamma+\sigma)\varepsilon^2 \right\}$$

It is clear that the system state is steered into the set Φ_ε (which contains the origin) from any arbitrary $x(t)$, but whenever it enters into Φ_ε it may evolve out of it or remain inside, because it is not ensured that the optimal worst case cost decrease.

Taking into account that $-x(t)^T Qx(t) \leq 0$ it follows that

$$J(x(t+1)) - J(x(t)) \leq -x(t)^T Qx(t) + (\gamma+\sigma)\varepsilon^2 \leq (\gamma+\sigma)\varepsilon^2$$

which yields:

$$J(x(t+1)) \leq J(x(t)) + (\gamma+\sigma)\varepsilon^2 \tag{20}$$

Suppose that $x(t) \in \Phi_\varepsilon$, then

$$J(x(t)) + (\gamma+\sigma)\varepsilon^2 \leq \max_{x\in\Phi_\varepsilon} J(x) + (\gamma+\sigma)\varepsilon^2$$

thus, taking into account (20) it follows that $\forall x(t) \in \Phi_\varepsilon$ it holds that

$$J(x(t+1)) \leq \beta$$

where $\beta = \max_{x\in\Phi_\varepsilon} J(x) + (\gamma+\sigma)\varepsilon^2$. Thus whenever the system state enters into Φ_ε it evolves into the set

$$\Omega_\beta = \left\{ x \in \mathbb{R}^{dimx} : J(x) \leq \beta \right\}.$$

Once the state is in Φ_ε it can evolve outside of Φ_ε, but it will remain inside Ω_β. From Ω_β it will be steered again into Φ_ε and so on. Then the system state is always confined into Ω_β from the moment it enters for the first time in Φ_ε. Thus the state system is ultimately bounded which means that the system is stabilized by the control law $\hat{K}_{MPC}(x(t)) = -Kx(t) + \hat{v}^*(t|t)$. □

5 Example

To illustrate the results presented in this paper, consider the two-tank network example given in chapter 20 of Ogunnaike and Ray (1994). Using the parameters given in Alamo et al. (2005b) the following continuous time state-space model can be obtained:

$$\dot{x} = \begin{bmatrix} -\frac{0.5}{3} & \frac{0.2}{3} \\ \frac{0.5}{2} & -\frac{0.5}{2} \end{bmatrix} x + \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} u \tag{21}$$

$$y = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x$$

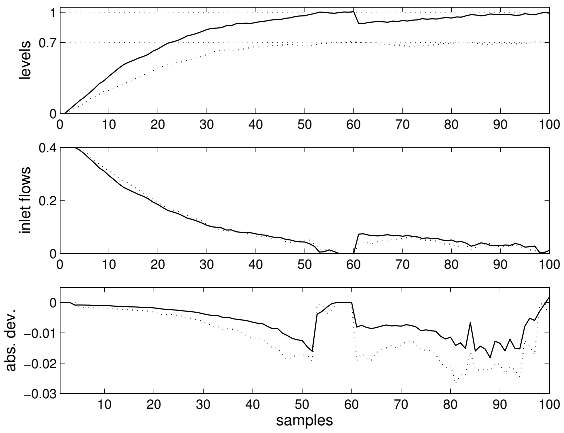

Fig. 1. Liquid levels (system state), inlet flows (control signal values) and absolute deviation (from the exact MMMPC) of the solution obtained by the proposed strategy (tank 1 solid plot, tank 2 dotted plot).

Table 2. Mean flops for the original Min-Max MPC and the proposed strategy for different values of the prediction and control horizon (N) in the simulation example of section 5.

N	Avg. flops (min-max)	Avg. flops (proposed)
4	4.77×10^6	4.28×10^4
5	3.73×10^7	7.6×10^4
6	3.43×10^8	1.28×10^5
7	1.84×10^9	1.42×10^5

Constraints are imposed on both states and control actions such that $\|x(k)\|_\infty \leq 1.5$ and $\|u(k)\|_\infty \leq 0.4$. A discrete time model has been obtained from (21) sampling at 0.2 minutes using a zero-order holder. Figure 1 shows the results of the proposed controller applied to the two-tank model. The set-point for the liquid level of each tank was 1 m and 0.7 m respectively. The prediction horizon was $N = 7$. Identity matrices were chosen as Q and R. An uncertainty of ± 0.025 meters is considered to affect both liquid levels. In the simulation a random noise of ± 0.01 meters has been added to both levels and an unexpected loss of liquid in tank 1 is introduced at sampling time $t = 60$.

The absolute deviation of the solution of (15) from that of (9) (computed as $\hat{\mathbf{v}}^*(x) - \mathbf{v}^*(x)$) is also shown in figure 1. It can be seen that it is very small throughout the simulation.

Finally, the lower computational burden of the proposed strategy is illustrated in table 2 in which are listed the average flops that problem (9) took to be solved as well as the flops needed to compute $\hat{\mathbf{v}}^*(x)$ (including those needed to compute $\tilde{\mathbf{v}}^*(x)$, $\alpha(\tilde{\mathbf{v}}^*(x))$ and $\hat{\Gamma}(\mathbf{v})$). The computational burden is much lower in the proposed strategy and the gap broadens exponentially as prediction horizon grows.

6 Conclusions

An MMMPC based on an tractable QP problem has been presented in this paper. The solution of this QP problem is close to that of the min-max problem whereas it has a much lower computational burden. As it is based on a QP problem, it can be implemented almost in any industrial hardware capable to run a constrained MPC controller. Thus it extends very much the fields of application of MMMPC controllers. Low computational burden is not the only advantage of this control strategy over other proposed MMMPC. The proposed controller is shown to be stable, which together with the explicit consideration of the uncertainty in the computation of the control law guarantees performance.

References

Alamo, T., Muñoz de la Peña, D., Limón Marruedo, D., Camacho, E., 2005a. Constrained min max predictive control: Modifications of the objective function leading to polynomial complexity. IEEE Transactions on Automatic Control 50, 710–714.

Alamo, T., Ramirez, D., Camacho, E., 2005b. Efficient implementation of constrained min-max model predictive control with bounded uncertainties: A vertex rejection approach. Journal of Process Control 15, 149–158.

Bemporad, A., December 1998. Reducing conservativeness in predictive control of constrained systems with disturbances. In: Proceedings of the 37th IEEE Conference on Decision and Control. pp. 1384–1389.

Bemporad, A., Borrelli, F., Morari, M., 2003. Min-max Control of Constrained Uncertain Discrete-Time Linear Systems. IEEE Transactions on Automatic Control 48 (9), 1600–1606.

Camacho, E., Bordóns, C., 2004. Model Predictive Control, 2nd Edition. Springer-Verlag.

Campo, P., Morari, M., June 10-12 1987. Robust Model Predictive Control. In: Proc. American Control Conference. pp. 1021–1026.

Casavola, A., Giannelli, M., Mosca, E., 2000. Min-Max Predictive Control strategies for input-saturated polytopic uncertain systems. Automatica 36, 125–133.

El-Farra, N. H., Mhaskar, P., Christofides, P. D., 2004a. Hybrid predictive control of nonlinear systems: Method and applications to chemical processes. International Journal of Robust and Nonlinear Control 14, 199–225.

El-Farra, N. H., Mhaskar, P., Christofides, P. D., 2004b. Uniting bounded control and mpc for stabilization of constrained linear systems. Automatica 40, 101–110.

Kerrigan, E., Maciejowski, J., 2004. Feedback min-max model predictive control using a single linear program: Robust stability and the explicit solution. International Journal of Robust Nonlinear Control 14, 295–413.

Kothare, M., Balakrishnan, V., Morari, M., 1996. Robust constrained model predictive control using linear model inequalities. Automatica 32 (10), 1361–1379.

Lee, J., Yu, Z., 1997. Worst-case formulations of model predictive control for systems with bounded parameters. Automatica 33 (5), 763–781.

Löfberg, J., Apr 2003. Minimax approaches to robust model predictive control. Ph.D. thesis.

Lu, Y., Arkun, Y., 2000. Quasi-Min-Max MPC algorithms for LPV systems. Automatica 36 (4), 527–540.

Mayne, D., Rawlings, J., Rao, C., Scokaert, P., 2000. Constrained model predictive control: Stability and optimality. Automatica 36, 789–814.

Mhaskar, P., El-Farra, N. H., Christofides, P. D., 2005. Robust hybrid predictive control of nonlinear systems. Automatica 41, 209–217.

Ogunnaike, B., Ray, W., 1994. Process dynamics, modeling, and control. Oxford University Press.

Ramirez, D., Alamo, T., Camacho, E., de la Peña, D. M., 2006. Min-Max MPC based on a computationally efficient upper-bound of the worst case cost. Journal of Process Control 16, 511–519.

Ramírez, D., Arahal, M., Camacho, E., 2004. Min-Max Predictive Control of a Heat Exchanger using a Neural Network Solver. IEEE Trans. on Control Systems Technology 12 (5), 776–786.

Ramirez, D., Camacho, E., 2006. Piecewise affinity of min-Max MPC with bounded additive uncertainties and a quadratic criterion. Automatica 42, 295–302.

Rossiter, J., Kouvaritakis, B., Rice, M., 1998. A numerically robust state-space approach to stable-predictive control strategies. Automatica 34, 65–73.

Scokaert, P., Mayne, D., 1998. Min-max feedback model predictive control for constrained linear systems. IEEE Transactions on Automatic Control 43 (8), 1136–1142.

Veres, S., Norton, J., 1993. Predictive Self-Tuning Control by Parameter Bounding and Worst Case Design. Automatica 29 (4), 911–928.

Wan, Z., Kothare, M., 2003. An efficient off-line formulation of robust model predictive control using linear matrix inequalities. Automatica 39, 837–846.

A Complex o auxiliary proofs

A.1 Proof of property 1:

(i) $S^{(k}$ is a partially diagonalized matrix:

Let us suppose that $S^{(k-1}$ is a partially diagonalized matrix. That is,

$$S^{(k-1} = \begin{bmatrix} T^{(k-1} & 0 \\ 0 & M_{sub}^{(k-1} \end{bmatrix} = \begin{bmatrix} T^{(k-1} & 0 & 0 \\ 0 & a & b^T \\ 0 & b & M_r \end{bmatrix}. \tag{A.1}$$

where $T^{(k-1} \in \mathbb{R}^{(k-1)\times(k-1)}$ is a diagonal matrix. Two cases must be taken into account: if $\alpha_k(\mathbf{v})$ is equal to zero then $b = \mathbf{0}_{n-k,1}$ and trivially, $S^{(k} = S^{(k-1}$ is a partially diagonalized matrix. If $\alpha_k(\mathbf{v}) \neq 0$ then:

$$S^{(k} = S^{(k-1} + \begin{bmatrix} \mathbf{0}_{k-1,1} \\ \alpha_k(\mathbf{v}) \\ \frac{-b}{\alpha_k(\mathbf{v})} \end{bmatrix} \begin{bmatrix} \mathbf{0}_{k-1,1} \\ \alpha_k(\mathbf{v}) \\ \frac{-b}{\alpha_k(\mathbf{v})} \end{bmatrix}^T = \begin{bmatrix} T^{(k} & 0 \\ 0 & M_{sub}^{(k} \end{bmatrix} \tag{A.2}$$

where $T^{(k} = \begin{bmatrix} T^{(k-1} & 0 \\ 0 & a + \alpha_k^2(\mathbf{v}) \end{bmatrix}$ and $M_{sub}^{(k} = M_r + \frac{bb^T}{\alpha_k^2(\mathbf{v})}$.

(ii) $\Gamma(\mathbf{v})$ is a diagonal matrix:

Note that $\Gamma(\mathbf{v}) = S^{(n-1}$. From the previous claim, $S^{(n-1} = \begin{bmatrix} T^{(n-1} & 0 \\ 0 & M_{sub}^{(n-1} \end{bmatrix} \in \mathbb{R}^{n\times n}$, where $T^{(n-1} \in \mathbb{R}^{(n-1)\times(n-1)}$. Thus, $M_{sub}^{(n-1} \in \mathbb{R}$. It follows that $\Gamma(\mathbf{v})$ is a diagonal matrix.

(iii) $V^*(x,\mathbf{v}) \leq \text{trace}(\Gamma(\mathbf{v}))$: By construction, $M(\mathbf{v}) = S^{(0} \leq S^{(1} \leq S^{(2} \leq \ldots \leq S^{(n-1} = \Gamma(\mathbf{v})$. Thus, $\Gamma(\mathbf{v})$ is a diagonal matrix that satisfies $\Gamma(\mathbf{v}) \geq M(\mathbf{v})$. From this and equation (14):

$$V^*(x,\mathbf{v}) = \max_{\|z\|_\infty \leq 1} z^T M(\mathbf{v}) z \leq \max_{\|z\|_\infty \leq 1} z^T \Gamma(\mathbf{v}) z = \text{trace}(\Gamma(\mathbf{v}))$$

(iv) $\|S^{(k}\|_s \leq \|S^{(k-1}\|_s$:

If $\alpha_k(\mathbf{v}) = \sqrt{\|b\|_1} = 0$ then $S^{(k} = S^{(k-1}$ and the claim is trivially satisfied. Suppose now that $\alpha_k(\mathbf{v}) \neq 0$ then, using equations (A.1,A.2):

$$\begin{aligned} \|S^{(k}\|_s &= \|T^{(k-1}\|_s + a + \alpha_k^2(\mathbf{v}) + \|M_r + \frac{bb^T}{\alpha_k^2(\mathbf{v})}\|_s = \|T^{(k-1}\|_s + a + \|b\|_1 + \|M_r + \frac{bb^T}{\|b\|_1}\|_s \\ &\leq \|T^{(k-1}\|_s + a + \|b\|_1 + \|M_r\|_s + \|\frac{bb^T}{\|b\|_1}\|_s = \|T^{(k-1}\|_s + a + \|b\|_1 + \|M_r\|_s + \|b\|_1 = \|S^{(k-1}\|_s \end{aligned}$$

Note that in order to obtain the inequality $\|S^{(k}\|_s \leq \|S^{(k-1}\|_s$, the equality $\|bb^T\|_s = \|b\|_1^2$ has been used. Also note that in the previous equation a must be nonnegative as it is one of the entries of the diagonal of a positive semidefinite matrix.

(v) $\text{trace}(\Gamma(\mathbf{v})) \leq \tilde{V}^*(x,\mathbf{v}), \forall \mathbf{v}$:

Note that $V(x,\mathbf{v},0) \geq 0$, thus, it results that (see equation 12):

$$\tilde{V}^*(x,\mathbf{v}) = \|M(\mathbf{v})\|_s = \left\| \begin{bmatrix} \varepsilon^2 H & \varepsilon q(x,\mathbf{v}) \\ \varepsilon q^T(x,\mathbf{v}) & V(x,\mathbf{v},0) \end{bmatrix} \right\|_s = \|S^{(0}\|_s$$

Moreover, $\Gamma(\mathbf{v})) = S^{(n-1}$ is a diagonal matrix positive semidefinite, thus $\text{trace}\,(\Gamma(\mathbf{v})) = \|S^{(n-1}\|_s$. On the other hand, as it has been shown, $\|S^{(k}\|_s \leq \|S^{(k-1}\|_s$. This implies that $\|S^{(n-1}\|_s \leq \|S^{(0}\|_s$, that is, $\text{trace}\,(\Gamma(\mathbf{v})) \leq \tilde{V}^*(x,\mathbf{v}), \forall \mathbf{v}$. □

A.2 Proof of theorem 1:

(i) $\hat{\Gamma}(\tilde{\mathbf{v}}^*) = \Gamma(\tilde{\mathbf{v}}^*)$.

Note that the computation of $\hat{\Gamma}(\tilde{\mathbf{v}}^*)$ relies on $\alpha(\tilde{\mathbf{v}}^*)$. It is clear from procedures 1 and 2 that if $\mathbf{v} = \tilde{\mathbf{v}}^*$ then $\hat{S}^{(k}(\tilde{\mathbf{v}}^*) = S^{(k}(\tilde{\mathbf{v}}^*)$. This implies that $\hat{S}^{(n-1}(\tilde{\mathbf{v}}^*) = S^{(n-1}(\tilde{\mathbf{v}}^*)$, thus $\hat{\Gamma}(\tilde{\mathbf{v}}^*) = \Gamma(\tilde{\mathbf{v}}^*)$.

(ii) $\hat{V}^*(x,\mathbf{v})$ is a quadratic function on $\mathbf{v}$.

Consider matrix $\hat{S}^{(0}(\mathbf{v}) = M(\mathbf{v})$ defined in (14), which can be partitioned as

$$\hat{S}^{(0}(\mathbf{v}) = \begin{bmatrix} \varepsilon^2 H_{11} & \varepsilon^2 H_{1r}^T & \varepsilon q_1(x,\mathbf{v}) \\ \varepsilon^2 H_{1r} & \varepsilon^2 H_{rr} & \varepsilon q_r(x,\mathbf{v}) \\ \varepsilon q_1(x,\mathbf{v}) & \varepsilon q_r^T(x,\mathbf{v}) & V(x,\mathbf{v},0) \end{bmatrix}$$

where H_{11}, $q_1(x,\mathbf{v})$ and $V(x,\mathbf{v},0) \in \mathbb{R}$, H_{1r}, $q_r(x,\mathbf{v}) \in \mathbb{R}^{(N\cdot dim\theta)-1}$ and $H_{rr} \in \mathbb{R}^{\{(N\cdot dim\theta)-1\}\times\{(N\cdot dim\theta)-1\}}$. Note that $q_1(x,\mathbf{v})$ and $q_r(x,\mathbf{v})$ have an affine dependence on $\mathbf{v}$ whereas $V(x,\mathbf{v},0)$ is a quadratic function on $\mathbf{v}$. Suppose that $\alpha_1(\tilde{\mathbf{v}}^*) \neq 0$ (the case of $\alpha_1(\tilde{\mathbf{v}}^*) = 0$ is trivial). Using $\alpha_1 = \alpha_1(\tilde{\mathbf{v}}^*)$, matrix $\hat{S}^{(0}(\mathbf{v})$ is partially diagonalized by adding the term $c_1(\mathbf{v})c_1^T(\mathbf{v})$ where

$$c_1(\mathbf{v}) = \begin{bmatrix} \alpha_1 \\ -\frac{\varepsilon^2 H_{1r}}{\alpha_1} \\ -\frac{\varepsilon q_1(x,\mathbf{v})}{\alpha_1} \end{bmatrix}$$

which yields

$$\hat{S}^{(1}(\mathbf{v}) = \begin{bmatrix} \varepsilon^2 H_{11} + \alpha_1^2 & 0 & 0 \\ 0 & \varepsilon^2 H_{rr} + \frac{\varepsilon^4 H_{1r} H_{1r}^T}{\alpha_1^2} & \varepsilon q_r(x,\mathbf{v}) + \frac{\varepsilon^3 H_{1r} q_1(x,\mathbf{v})}{\alpha_1^2} \\ 0 & \varepsilon q_r^T(x,\mathbf{v}) + \frac{\varepsilon^3 H_{1r}^T q_1(x,\mathbf{v})}{\alpha_1^2} & V(x,\mathbf{v},0) + \frac{\varepsilon^2 q_1^2(x,\mathbf{v})}{\alpha_1^2} \end{bmatrix}$$

Note that the sub-matrix

$$M_r(x,\mathbf{v}) = \begin{bmatrix} \varepsilon^2 H_{rr} + \frac{\varepsilon^4 H_{1r} H_{1r}^T}{\alpha_1^2} & \varepsilon q_r(x,\mathbf{v}) + \frac{\varepsilon^3 H_{1r} q_1(x,\mathbf{v})}{\alpha_1^2} \\ \varepsilon q_r^T(x,\mathbf{v}) + \frac{\varepsilon^3 H_{1r}^T q_1(x,\mathbf{v})}{\alpha_1^2} & V(x,\mathbf{v},0) + \frac{\varepsilon^2 q_1^2(x,\mathbf{v})}{\alpha_1^2} \end{bmatrix}$$

has the same structure as $M(\mathbf{v})$, that is, the last element is a quadratic function of $\mathbf{v}$, the remaining elements of the last row and column are affine functions of $\mathbf{v}$ and all the other elements are constants. That is, as the structure is preserved, a new iteration of procedure 2 supposes a further diagonalization in which only the last element has a quadratic dependence on $\mathbf{v}$. At the end of procedure 2 the diagonal matrix $\hat{S}^{(n-1}(\mathbf{v}) = \hat{\Gamma}(\mathbf{v})$ is obtained with all its elements constant (i.e., they do not depend on $\mathbf{v}$) except the last one which has the form

$$\hat{\Gamma}_{nn}(\mathbf{v}) = V(x,\mathbf{v},0) + \frac{\varepsilon^2 q_1^2(x,\mathbf{v})}{\alpha_1^2} + \cdots \tag{A.3}$$

Once $\hat{\Gamma}(\mathbf{v})$ has been obtained, the bound of the maximum can be computed as

$$\hat{V}^*(x,\mathbf{v}) = \text{trace}(\hat{\Gamma}(\mathbf{v})) \tag{A.4}$$

which is a quadratic function of $\mathbf{v}$.

(iii) $V^*(x,\mathbf{v}) \leq \hat{V}^*(x,\mathbf{v})$.

This follows from the fact that by construction $\hat{\Gamma}(\mathbf{v}) = M(\mathbf{v}) + c_1(\mathbf{v})c_1^T(\mathbf{v}) + \cdots + c_{n-1}(\mathbf{v})c_{n-1}^T(\mathbf{v})$. Thus, $M(\mathbf{v}) \leq \hat{\Gamma}(\mathbf{v})$ and, together with the fact that by construction $\hat{\Gamma}(\mathbf{v})$ is diagonal, implies that

$$V^*(x,\mathbf{v}) = \max_{\|z\|_\infty \leq 1} z^T M(\mathbf{v}) z \leq \max_{\|z\|_\infty \leq 1} z^T \hat{\Gamma}(\mathbf{v}) z = \text{trace}(\hat{\Gamma}(\mathbf{v})) = \hat{V}^*(x,\mathbf{v})$$

□

A.3 Proof of property 3:

Note that $J(x) = V^*(x,\mathbf{v}^*)$. From equations (8) and (11) it results that:

$$\tilde{V}(x,\mathbf{v},\boldsymbol{\theta}) = V(x,\mathbf{v},\boldsymbol{\theta}) + \boldsymbol{\theta}^T (T - H)\boldsymbol{\theta}.$$

where T is a diagonal matrix defined as in (10). Taking into account that $T \geq H \geq 0$, $\|\boldsymbol{\theta}\|_\infty \leq \varepsilon$ and that T is diagonal:

$$\tilde{V}(x,\mathbf{v},\boldsymbol{\theta}) \leq V(x,\mathbf{v},\boldsymbol{\theta}) + \boldsymbol{\theta}^T T \boldsymbol{\theta} \leq V(x,\mathbf{v},\boldsymbol{\theta}) + \text{trace}(T)\varepsilon^2 = V(x,\mathbf{v},\boldsymbol{\theta}) + \|H\|_s \varepsilon^2$$

thus it can be inferred that

$$V^*(x,\mathbf{v}^*) \geq \tilde{V}^*(x,\mathbf{v}^*) - \sigma\varepsilon^2$$

with $\sigma = \|H\|_s$. As $\tilde{\mathbf{v}}^*$ is the minimizer of $\tilde{V}^*(x,\tilde{\mathbf{v}})$:

$$V^*(x,\mathbf{v}^*) \geq \tilde{V}^*(x,\tilde{\mathbf{v}}^*) - \sigma\varepsilon^2$$

which in turn can be rewritten as

$$J(x) \geq \tilde{J}(x) - \sigma\varepsilon^2$$

Recall that from property 2: $\hat{J}(x) \leq \tilde{J}(x)$; thus:

$$J(x) \geq \hat{J}(x) - \sigma\varepsilon^2 = \hat{V}^*(x,\hat{\mathbf{v}}^*) - \sigma\varepsilon^2$$

From theorem 1 $\hat{V}^*(x,\mathbf{v}) \geq V^*(x,\mathbf{v})$ thus

$$J(x) \geq V^*(x,\hat{\mathbf{v}}^*) - \sigma\varepsilon^2$$

and this completes the proof. □