Delft University of Technology

Delft Center for Systems and Control

Technical report 08-015

A Hybrid Steepest Descent Method for Constrained Convex Optimization*

M. Gerard, B. De Schutter, and M. Verhaegen

To cite this work, please refer to the published version:

M. Gerard, B. De Schutter, and M. Verhaegen, "A hybrid steepest descent method for constrained convex optimization," *Automatica*, vol. 45, no. 2, pp. 525–531, Feb. 2009. doi:10.1016/j.automatica.2008.08.018

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

* This report can also be downloaded via https://dspub.eu/08-015

A Hybrid Steepest Descent Method for Constrained Convex Optimization

Mathieu Gerard [1,*], Bart De Schutter [2], Michel Verhaegen [1]

Delft University of Technology

Abstract

This paper describes a hybrid steepest descent method to decrease over time any given convex cost function while keeping the optimization variables into any given convex set. The method takes advantage of properties of hybrid systems to avoid the computation of projections or of a dual optimum. The convergence to a global optimum is analyzed using Lyapunov stability arguments. A discretized implementation and simulation results are presented and analyzed. This method is of practical interest to integrate real-time convex optimization into embedded controllers thanks to its implementation as a dynamical system, its simplicity, and its low computation cost.

Key words: Real-time optimization, Convex optimization, Gradient methods, Steepest descent method, Hybrid systems.

1 Introduction

Optimization problems play a role of increasing importance in many engineering domains, and specially in control theory. While some design procedures require finding the optimum of a complex problem only once, other real-time control techniques want to track the optimum of a time-varying cost function with time-varying constraints. Some applications based on real-time optimization are the following:

- **Model Predictive Control** [15], [10]. The cost function measures the error between the predicted outputs of the controlled plant and the desired outputs, over a prediction horizon. The desired outputs can be changed in real-time, for example by a human operator, which makes the optimization problem time varying. The variables to be continuously optimized are the future control inputs to the plant over a control horizon.
- **Control Allocation** [4]. Here the task is to optimally distribute some desired control input to a set of actuators, based on actuator cost and constraints. The changing character of the generic control input makes the problem time-varying.
- **Lyapunov-based control**. Stabilization is performed by making a Lyapunov function or an objective function to decrease over time down to a minimum. A good summary of such methods applied to the field of motion coordination can be found in [16].
- **Model Reference Adaptive Control** [19], [17], [12]. The objective is to adapt the controller parameters so that the controlled plant behaves like the reference one. The time-varying cost function is computed from the output error between the controlled and the reference plants.

In the literature, one can distinguish 3 main approaches to deal with these problems:

- **Repeated Optimization** [15] where the new optimization problem is solved at each time step;
- **Precomputed Optimization** [1] where all the possible problems are solved off-line and stored in a look-up table, which can become very large;
- **Update Laws** [19] where the optimization variables are taken as states of a dynamical system and are given a certain dynamics.

In the field of optimization, many efficient techniques exist to solve constrained convex optimization problems [5]. They are designed to output, after several iterations, an accurate value of the optimum. Therefore they are

* Corresponding author.
Email address: m.p.gerard@tudelft.nl (Mathieu Gerard).
[1] Mathieu Gerard and Michel Verhaegen are with the Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands.
[2] Bart De Schutter is with the Delft Center for Systems and Control and with the Marine & Transport Technology department of Delft University of Technology.

very well suited for off-line optimization, where a problem should be solved accurately once. However, the iterative character and the complexity of the algorithm do not always make it suitable for on-line implementation. This is probably why those techniques are not often, and very cautiously, integrated into embedded controllers.

On the other hand, update laws are very easy to implement and require very few computation power. Since many years, research on this topic has been going on within the field of adaptive control [19], [17], [12]. For example, one interesting method to adapt the parameters on-line in Model Reference Adaptive Control is to use a gradient descent method for a cost function defined as the square of the output error between the controlled real plant and the reference. This technique works well in the unconstrained case.

However, when it comes to adding constraints, solutions in the control field are much more limited. One possible method found in [17] and [12] is the gradient method with projection. The idea is to project the gradient on the active constraints when the state is on the boundary of the feasible set. In that way, the descent direction is always such that the state stays in the feasible set. This method works in continuous time but the discrete-time implementation is much more intricate, specially in the case of nonlinear constraints. Moreover, the way to compute the projection is not obvious and rather complex. Therefore it is only worked out and used for very simple cases.

In case of discrete-time implementation, which is always the case when using digital controllers, the only currently available method, proposed in [12], uses scaling to project back any new value of the state into the feasible set if necessary. However, the idea is worked out only in case of a very simple feasible set (a ball centered at the origin) and no proof of convergence is given.

This paper bridges those two worlds of control and optimization by developing an update law to deal with the "general" case of a convex cost function and convex constraints while enabling easy integration into traditional controllers. Moreover, the method is simple, which allows good insight, has low computational cost and is easily discretizable for discrete-time implementation.

The proposed technique takes the form of a traditional ordinary differential equation

$$\dot{x} = f(x)$$

In [6], they are designed to sort lists and to diagonalize matrices. References [13] and [3] present plenty of examples where methods coming from the control area can be used to synthesize and analyze numerical algorithms. In this paper, the vector field f will be designed to solve constrained convex optimization problems. If the optimization problem is time-varying, f can obviously be expected to also depend on time. However, the rest of the paper will focus on an invariant problem for simplicity. The formal requirements are described in Section 2. The proposed system is described and analyzed in Section 3. Section 4 will show that the technique can still be used after a simple discretization. Finally a simulation example is presented in Section 5.

2 Problem formulation

Let us consider the convex optimization problem

$$\begin{aligned} \min_x \quad & q(x) \\ \text{subject to} \quad & g(x) \leq 0 \end{aligned} \tag{1}$$

with $x \in \Re^n$, $q : \Re^n \to \Re$ a differentiable convex function and $g : \Re^n \to \Re^m$ such that each g_i is a differentiable convex function for $i = 1, .., m$. The constraint $g(x) \leq 0$ defines a convex set that we will call the feasible set, for consistency with the optimization terminology. Its complement is called the infeasible set. The two following assumptions are made:

Assumption 1 The feasible set is not empty, i.e. $\exists x_f$ s.t. $g(x_f) \leq 0$.

The optimal value of the cost function in the feasible set is then denoted q^*, i.e. $q^* = \min_x \{q(x) | g(x) \leq 0\}$

Assumption 2 q^* is finite

The objective is to find a vector function $f : \Re^n \to \Re^n$ such that the dynamical system

$$\dot{x}(t) = f(x(t)) \tag{2}$$

has the following properties:

- for $x(\bar{t})$ outside the feasible set at some time $\bar{t}$, the trajectory $x(t)$ enters into the feasible set, i.e. $\exists t_\text{f} > \bar{t}$ s.t. $g(x(t_\text{f})) \leq 0$.
- the trajectory $x(t)$ remains in the feasible set as soon as $x(t_\text{f})$ is in the set, i.e. $g(x(t)) \leq 0 \quad \forall t > t_\text{f}$ s.t. $g(x(t_\text{f})) \leq 0$,
- for $x(t_\text{f})$ in the feasible set, the trajectory $x(t)$ decreases the cost function $q(x(t))$ at all time until $q(x(t)) = q^*$, i.e. $q(x(t_1)) > q(x(t_2))$ $\forall (t_1, t_2)$ with $t_\text{f} \leq t_1 < t_2$ s.t. $q(x(t_1)) > q^*$, and $\lim_{t \to \infty} q(x(t)) = q^*$.

3 Hybrid steepest descent solution

One efficient way to decrease an unconstrained cost function is to use a gradient descent method, as used traditionally in adaptive control [17]. Therefore the basis of this method is similar. It can also be noted that the gradient of a function is not its only descent direction. Other directions have been proposed in the literature, like Newton's direction [5]. The investigation of alternative directions to improve the convergence while limiting the increase of computation complexity is left for future work.

The original idea of the current paper is based on the way the constraints are considered. Because of the computational complexity, a direct projection of the gradient on the constraints is discarded. But on the other hand, each constraint is seen as a kind of barrier. More precisely, each constraint which would not be satisfied at a certain time instant will push the trajectory toward the feasible set. In that way, the trajectory will never leave the feasible set. Furthermore, if $x(t)$ is on the boundary of the feasible set, it will be pushed alternatively by both the gradient and the constraint. If they are pushing in opposite directions, $x(t)$ will naturally slide along the border and the projection will appear indirectly. Compared to interior point methods, this technique has the advantage to have descent directions defined outside the feasible set, which can be useful in case of time-varying constraints.

The proposed hybrid feedback law is therefore:

$$f(x) = \begin{cases} -\nabla q(x) & \text{if } g_j(x) \leq 0 \ \ \forall j \\ -\sum_{i\in L(x)} \nabla g_i(x) & \text{if } \exists j: \ g_j(x) > 0 \end{cases} \tag{3}$$

with $L(x) = \{l : g_l(x) \geq 0\}$.

The rest of this section is dedicated to the analysis of the behavior of this system using hybrid systems techniques and Lyapunov arguments.

3.1 Filippov solutions and sliding modes

The vector field $f(x)$ is measurable and essentially locally bounded but discontinuous. Therefore the study of the solution of the vector differential equation $\dot{x}(t) = f(x(t))$ requires the use of a particular solution concept. We make use of the Filippov solution concept [9], [18] recalled in the following definition:

Definition (Filippov) A vector function $x(.)$ is called a solution of (2) on $[t_1, t_2]$ if $x(.)$ is absolutely continuous on $[t_1, t_2]$ and for almost all $t \in [t_1, t_2]$: $\dot{x} \in K[f](x)$, where $K[f](x)$ is the convex hull of the limits of $f(y)$ for $y \to x$ while y stays out of a set of zero Lebesgue measure where f is not defined [9], [18].

At a point x around which $f(x)$ is continuous, $K[f](x)$ reduces to $f(x)$. However, on a switching surface, $K[f](x)$ will contain a set of possible values for $\dot{x}$.

So, at all time, $\dot{x}$ has the following form:

$$\dot{x} = -\gamma_0(x)\nabla q(x) - \sum_{i=1}^{m} \gamma_i(x)\nabla g_i(x) \tag{4}$$

for some $\gamma_j(x) \geq 0, j \in \{0, ..., m\}$. Depending the situation, the values of the $\gamma_j(x)$ will be different:

- for x strictly in the feasible set, $\gamma_0 = 1$ and $\gamma_i = 0 \ \forall i$,
- for x in the infeasible set, $\gamma_j = 1 \ \forall j \in L(x)$ and 0 otherwise (so $\gamma_0 = 0$),
- for x on a boundary, the values of the γ_j will depend on a possible sliding motion as defined by the Filippov solution concept.

Following the Filippov solution concept, for x on the switching surface between the feasible and infeasible sets, either a sliding motion can take place, i.e. a motion along the switching surface, or a motion toward one of the sets [9]. Since the $-\nabla g_i(x)$ are always pointing toward the feasible region, a sliding mode will appear only if $-\nabla q(x)$ is pointing toward the infeasible region. In that case, the sliding motion will require $\gamma_0(x) > 0$ in (4). In case there is no sliding motion, then due to (3), we have $f(x) = -\nabla q(x)$ and $\gamma_0(x) = 1$ in (4). In conclusion, $\gamma_0(x) > 0$ on the boundary between the feasible and infeasible sets.

3.2 Stationary points of the update law (2)-(3)

The most interesting property of the update law (2)-(3) is that the globally stable equilibria of the dynamical system precisely coincide with the optimal points of the constrained optimization problem. First it will be shown that the stationary points of the systems are optimal, and vice versa. The stability of those points is proved in the next subsection.

Definition [9] A point $x = p$ is called stationary if it is a trajectory, that is, if $x(t) \equiv p$ is a solution of (2).

Following the definition, it can be concluded that a point p is stationary if and only if $0 \in K[f](p)$, [9]. Further, a point will be called an equilibrium if it is stationary and stable.

Theorem 1 below states that, if the convex optimization problem is feasible, the stationary points lie in the feasible set. The "optimality" of the stationary points is considered in Theorem 2. Further, the next section demonstrates the asymptotic stability.

Theorem 1 *If the functions $g_i(x) : \Re^n \to \Re$ are convex ($i = 1, ..., m$) and if there exists an x_f such that $g_i(x_\mathrm{f}) \leq 0 \ \ \forall i \in \{1, ..., m\}$ then*

$$\sum_{i\in L} \nabla g_i(\bar{x}) \neq 0$$

for any subset L of $\{1,...,m\}$ and any $\bar{x}$ such that $g_i(\bar{x}) > 0$ for some $i \in L$.

Proof *Let us define $T^{g_i}_{\bar{x}}(x)$ the tangent hyperplane to the function g_i at $\bar{x}$:*

$$T^{g_i}_{\bar{x}}(x) = \nabla g_i^T(\bar{x})\,(x-\bar{x}) + g_i(\bar{x}) \triangleq G_i(\bar{x})x - h_i(\bar{x}) \quad (5)$$

where $G_i(\bar{x}) = \nabla g_i^T(\bar{x})$ and $h_i(\bar{x}) = \nabla g_i^T(\bar{x})\bar{x} - g_i(\bar{x})$. Due to the convexity of g_i, we know that

$$g_i(x) \geq T^{g_i}_{\bar{x}}(x) \qquad \forall \bar{x}, \forall x \quad (6)$$

The proof is done by contradiction. Assume there exist a point $\bar{x}$ and a set L such that

$$\begin{cases} g_i(\bar{x}) = G_i(\bar{x})\bar{x} - h_i(\bar{x}) > 0 \;\; \forall i \in L \\ \sum_{i\in L} G_i(\bar{x}) = 0 \end{cases} \quad (7)$$

This directly leads to

$$0 = \sum_{i\in L} G_i(\bar{x})\bar{x} > \sum_{i\in L} h_i(\bar{x}) \quad (8)$$

Furthermore, by the hypothesis of the theorem, there exists an x_f such that $g_i(x_\mathrm{f}) \leq 0 \;\; \forall i \in L$ and therefore by (5) and (6)

$$0 \geq \sum_{i\in L} g_i(x_\mathrm{f}) \geq \sum_{i\in L}(G_i(\bar{x})x_\mathrm{f} - h_i(\bar{x})) = -\sum_{i\in L} h_i(\bar{x}) \quad (9)$$

Equations (8) and (9) clearly lead to a contradiction. It can therefore be concluded that a combination $(\bar{x}, L)$ does not exist , which proves the theorem.

Using this result, the "optimality" of the stationary points can be assessed.

Theorem 2 *If (1) is feasible then a point p is a stationary point of (2)-(3) if and only if it is an optimal point of (1).*

Proof *Due to Theorem 1, it can be concluded that $f(x)$ is always different from 0 for x in the infeasible set and therefore the stationary points lie in the feasible set.*

For any feasible x, the dynamics takes the form of equation (4) with $\gamma_0(x) > 0$. If there exists a stationary point p such that $\dot{x}(p) = 0$, and by defining

$$\lambda_i = \frac{\gamma_i(p)}{\gamma_0(p)} \quad (10)$$

it is easy to check that the following set of equations is satisfied:

$$\begin{aligned} g_i(p) &\leq 0 \;\; \forall i \\ \lambda_i &\geq 0 \;\; \forall i \\ \lambda_i g_i(p) &= 0 \;\; \forall i \\ \nabla q(p) + \textstyle\sum_{i=1}^m \lambda_i \nabla g_i(p) &= 0 \end{aligned}$$

These equations are the well-known Karush-Kuhn-Tucker (KKT) conditions, which prove that the stationary point p is an optimal solution of the convex problem (1) while the λ's are the Lagrange multipliers [5] [2].

Moreover, if p is optimal, it satisfies the KKT conditions. The Lagrange multipliers λ will define a suitable dynamics of the form (4), which belongs to $K[f](p)$. Therefore, 0 belongs to $K[f](p)$ and, following the definition, p is a stationary point.

By Assumptions 1 and 2, there always exists at least one such stationary point p. Furthermore, $q(p) = q^*$

3.3 Asymptotic stability

Finally, it can be shown that (2)-(3) is asymptotically stable and converges toward one of the stationary points found above. In view of the structure of (3), we propose the following Lyapunov function:

$$V(x) = \max(q(x), q^*) - q^* + \beta \sum_{i=1}^m \max(g_i(x), 0) \quad (11)$$

with β a strictly positive parameter.

It is obvious that this Lyapunov function is strictly positive everywhere except at stationary points where $V(p) = 0$:

- in the infeasible set, we have $V(x) > 0$ since at least one $g_i(x) > 0$
- in the feasible set (or at the boundary), we have $V(x) > 0$ since $q(x) > q^*$, except at the stationary points where $q(x) = q^*$

Unfortunately, this Lyapunov function is not differentiable everywhere. To handle this case, the theory developed in [18] will be used. The main stability theorem is recalled below. But first, to go more smoothly through the technicalities, let us recall some definitions.

Definition [11] A function $f(x)$ is said to be essentially bounded on X if the function is unbounded only on a set of measure zero, i.e. $\mu\{x \in X : |f(x)| > a\} =$

0 for some real number $a \geq 0$ where μ is the Lebesgue measure.

Definition [7] A function $V(x)$ is said to be regular when the usual directional derivative exists in any direction. Examples of regular functions include smooth functions, convex Lipschitz functions, and functions that can be written as the pointwise maximum of a set of smooth functions.

Therefore it can be concluded that $V(x)$ is regular.

Definition [14] A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$.

Definition [7] The Clarke's generalized gradient $\partial V(x)$ of a locally Lipschitz function $V(x)$ is the convex hull of the limits of the gradients of the function around the points where the gradient of V is not defined.

$$\partial V(x) = \bar{\text{co}}\{\lim_{y \to x, y \notin \Omega_V} \nabla V(y)\} \tag{12}$$

for Ω_V a set of measure zero where the gradient of V is not defined.

Definition [18] The set $\dot{\tilde{V}}(x)$, which is proved in [18] to contain $\frac{d}{dt}V(x(t))$, when this last quantity exists, is defined as

$$\dot{\tilde{V}}(x) \triangleq \bigcap_{\xi \in \partial V(x)} \xi^T K[f](x) \tag{13}$$

In the smooth case, where all the functions involved are differentiable, $\dot{\tilde{V}}(x)$ reduces to the time derivative of the function $V(x(t))$: $\dot{\tilde{V}}(x) = \nabla V^T \dot{x}$.

Definition A set S is said to be negative if all the elements of the set are negative: $S < 0 \quad \Leftrightarrow \quad s < 0 \; \forall s \in S$

Theorem 3 *[18] Let $\dot{x} = f(x)$ be essentially locally bounded in a region $Q \supset \{x \in \Re^n | \, \|x - p\| < r\}$ for some real number r and $0 \in K[f](p)$. Also let $V : \Re^n \to \Re$ be a regular function satisfying $V(p) = 0$ and $0 < V_1(\|x - p\|) \leq V(x) \leq V_2(\|x - p\|)$ for $x \neq p$ in Q for some $V_1, V_2 \in$ class $\mathcal{K}$. Then*

(1) $\dot{\tilde{V}}(x) \leq 0$ in Q implies that $x(t) \equiv p$ is a uniform stable function.
(2) If in addition, there exists a class $\mathcal{K}$ function $w(.)$ in Q with the property $\dot{\tilde{V}}(x) \leq -w(x) < 0$ for $x \neq p$ then the solution $x(t) \equiv p$ is uniformly asymptotically stable.

Now Theorem 3 is used to prove the stability of (2)-(3).

Theorem 4 *The system defined by (2)-(3) is uniformly asymptotically stable if it has one unique finite stationary point.*

Proof *The vector field f (3) satisfies the requirements of Theorem 3 for any set Q of the form $\{x \in \Re^n | \, \|x - p\| < r\}$ with a finite r and $0 \in K[f](p)$. Moreover, the Lyapunov function V (11) is continuous, since it is the sum of continuous functions; regular and convex, since it can be written as the pointwise maximum of a set of smooth convex functions; and positive. Then due to the convexity of the functions and the unique finite stationary point p such that $V(p) = 0$, V can be bounded from below by the function $V_1(||x - p||)$ with V_1 of class $\mathcal{K}$. With the same arguments, V can also be bounded from above by a function $V_2(||x - p||)$ with V_2 of class $\mathcal{K}$.*

The stability conclusion coming from Theorem 3 now depends on the values of the generalized time derivative of V. The 3 main regions are first considered before going to a more general case including the boundaries.

- *For x in the feasible set we have: $V(x) = q(x) - q^*$, $\nabla V = \nabla q$, $\dot{x} = -\nabla q$ and therefore*

$$\dot{\tilde{V}}(x) = -\nabla q(x)^T \nabla q(x) \leq 0$$

  which is always negative as long as $\nabla q(x) \neq 0$, and this can only happen at the stationary point. Moreover, as we move away from the optimal point in the feasible set, the norm of $\nabla q(x)$ cannot decrease, because of the convexity of $q(x)$, and therefore $\dot{\tilde{V}}(x)$ cannot increase. This will be important when showing that $\dot{\tilde{V}}(x)$ can be bounded from below and from above by class $\mathcal{K}$ functions.
- *For x in the infeasible set for $q(x) < q^*$ we have: $V(x) = \beta \sum_{i \in L} g_i(x)$, $\nabla V = \sum_{i \in L} \nabla g_i(x)$, $\dot{x} = -\sum_{i \in L} \nabla g_i(x)$ and therefore*

$$\dot{\tilde{V}}(x) = -\beta \left\| \sum_{i \in L} \nabla g_i(x) \right\|^2 < 0$$

  which is always strictly negative and never increasing as we move away from the stationary point by the same convexity arguments as before.
- *For x in the infeasible set for $q(x) \geq q^*$ we have: $V(x) = q(x) - q^* + \beta \sum_{i \in L} g_i(x)$, $\nabla V = \nabla q + \sum_{i \in L} \nabla g_i(x)$, $\dot{x} = -\sum_{i \in L} \nabla g_i(x)$ and therefore*

$$\dot{\tilde{V}}(x) = -\nabla q(x)^T \left(\sum_{i \in L} \nabla g_i(x) \right) - \beta \left\| \sum_{i \in L} \nabla g_i(x) \right\|^2 < 0$$

  Since $\sum_{i \in L} \nabla g_i(x) \neq 0$ (see Theorem 1), it is always possible to find a β large enough such that $\dot{\tilde{V}}(x)$ is *strictly negative. Moreover, again thanks to convexity, the norm of $\nabla g_i(x)$ cannot decrease as we move away from the stationary point in the infeasible set. Therefore, a large enough β can also render $\dot{\tilde{V}}(x)$ non-increasing. The interpretation of β is here to create a large enough barrier such that the constraints dominate the cost function in the infeasible set.*

- *In the general case, $\dot{\tilde{V}}(x)$ is not anymore a number but a set of values for each x. To show that $\dot{\tilde{V}}(x) < 0$, the fact that a subset of a negative set is also a negative set will be used twice.*

  Since $K[f](x)$ is a subset of the "headless" cone

$$\bar{X} = \{-\varphi_0 \nabla q(x) - \sum_{i \in L(x)} \varphi_i \nabla g_i(x) | \varphi_j \geq 0, \sum_j \varphi_j > 0\}$$

  and since ∂V contains

$$\bar{V} = \{\nabla q(x) + \beta \sum_{i \in L(x)} \gamma_i \nabla g_i(x) | \gamma_j \geq 0, \sum_j \gamma_j \leq 1\}$$

  $\dot{\tilde{V}}(x)$ is included in the set $\bar{W} = \{\bigcap_{\xi \in \bar{V}} \xi^T \bar{X}\}$. So if $\bar{W}$ is negative, then $\dot{\tilde{V}}$ is negative as well.

  If $K[f](x)$ does not contain 0, i.e. if x is not the stationary point, then 0 is not in the convex "headless" cone $\bar{X}$ neither and the entire set is situated in a half-space defined by the separating hyperplane passing through 0 and with normal vector v. Such hyperplane is not unique and therefore a normal vector v can be chosen such that v belongs to $\bar{X}$ with $\varphi_0 > 0$ and $\sum_j \varphi_j = 1$. Then it is obvious that the vector $-\frac{1}{\varphi_0} v$ belongs to $\bar{V}$ for $\beta \geq \frac{1}{\varphi_0}$. Therefore, there exists a vector $\xi = -\frac{1}{\varphi_0} v$ belonging to $\bar{V}$ such that $\xi^T \bar{X} < 0$, which implies $\bar{W} < 0$ and finally $\dot{\tilde{V}}(x) < 0$. In case $K[f](p)$ contains zero, the stationary point is reached and the value of the Lyapunov function will remain zero.

Finally, Theorem 3 holds which proves the theorem.

In the case where the system (2)-(3) has many stationary points (for a convex optimization problem they will represent a convex set : a sublevel set of a convex function [5]), the equivalent of LaSalle's theorem for non-smooth systems presented in [18] can be used to show that the system (2)-(3) will converge to a point in the largest invariant set of $\{x | 0 \in \dot{\tilde{V}}(x)\}$, which in our case is the complete set of optimal points.

4 Discretization

Because of the sliding mode, i.e. the infinite number of switches, the dynamical system cannot be simulated directly. For practical implementation, it is a good idea to sample the system at a given sampling frequency [20]. In that way, the computation can easily be scheduled on time-driven microcontrollers and the number of switches is limited to one every period.

Obviously, because of the sampling, the accuracy of the system is going to be reduced. Two main drawbacks can be foreseen, whatever the sampling technique used:

- during a sliding mode, the trajectory will not stay perfectly on the boundary but will meander slightly around it. The amplitudes of the oscillations will depend on the sampling period and on the norms of the gradients $\nabla q(x)$ and $\nabla g_i(x)$.
- the trajectory will not precisely converge toward the precise optimal point but will oscillate around it.

In this paper, the sampling is done by approximating the derivative by a forward difference, i.e. Euler's method. For a sampling time Δt, the difference equation is given by

$$x_{k+1} = x_k + f(x_k) \Delta t \quad (14)$$

Other methods exist and will be investigated in future research.

Further, in order to reduce oscillations during the sliding mode, smooth transitions can be implemented between the feasible and the infeasible set [8]. In practice, the gradient of the constraints are weighted based on the value of the constraint via a smoothened step s, for example:

$$s(g(x)) = \frac{1}{\pi} \text{atan}(\frac{g(x)}{\epsilon}) + 0.5 \quad (15)$$

where ϵ is a tuning parameter which should be small. Then the gradient of the cost function is weighted by a complementary function. Finally, the smooth version of f is given by

$$f_s = -\left(1 - \frac{1}{m} \sum_{i=1}^{m} s(g_i(x))\right) \nabla q(x) - \sum_{i=1}^{m} s(g_i(x)) \nabla g_i(x) \quad (16)$$

5 A simulation example

To illustrate the essence of the developed method, we consider a simple convex optimization problem in 2 dimensions. The cost function and one of the constraints are chosen to be linear while the second constraint is taken nonlinear, although convex.

Let us consider the following optimization problem for $x = \begin{pmatrix} x_1 & x_2 \end{pmatrix}^T$:

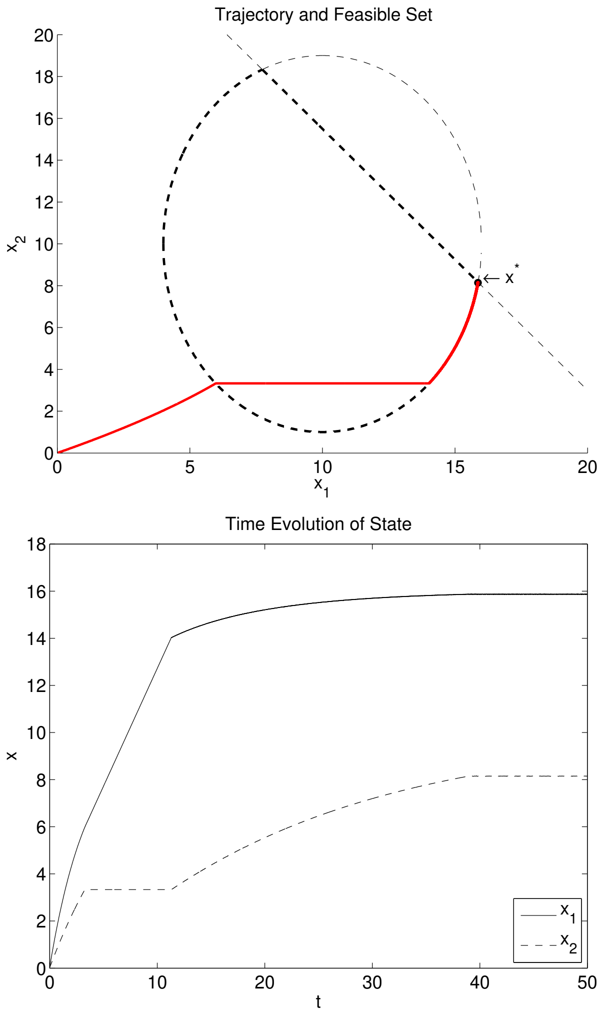

Figure 1. Simulation with sampling time $\Delta t = 0.01$. The oscillations in the sliding mode are not visible and the trajectory converges toward the optimal point. Just after $t = 3$ the trajectory reaches the feasible set; after $t = 11$ the trajectory starts sliding along the nonlinear constraint; and at $t = 40$ the optimal point is reached.

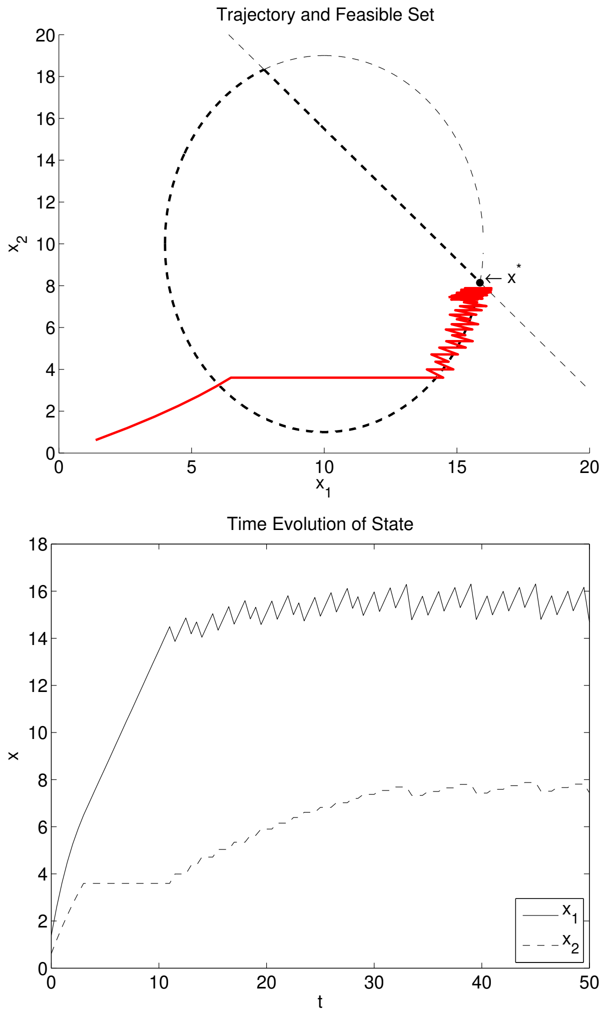

Figure 2. Simulation with sampling time $\Delta t = 0.5$. The oscillations in the sliding mode are now visible but the trajectory still evolves in the right direction until oscillating around the optimal point.

$$\min_x \; -x_1 \tag{17}$$

$$\text{subject to } \frac{(x_1-10)^2}{36} + \frac{(x_2-10)^2}{81} - 1 \leq 0$$

$$\frac{10}{8}x_1 + x_2 - 28 \leq 0$$

We have:

$$\nabla q(x) = \begin{pmatrix} -1 & 0 \end{pmatrix}^T \tag{18}$$

$$\nabla g_1(x) = \begin{pmatrix} \frac{1}{18}(x_1-10) & \frac{2}{81}(x_2-10) \end{pmatrix}^T \tag{19}$$

$$\nabla g_2(x) = \begin{pmatrix} \frac{10}{8} & 1 \end{pmatrix}^T \tag{20}$$

Figure 1 presents the results of the simulation for a small sampling time $\Delta t = 0.01$ and initial condition $x_0 = (0\ 0)^T$. While x is outside the feasible set (until $t = 3$), the trajectory converges toward it. Then the gradient of the cost function is followed until reaching a constraint at $t = 11$. Afterwards, the trajectory slides along the constraint to the optimal point, which is reached at $t = 40$. Thanks to the small sampling time, the oscillations in the sliding mode are hardly visible. It can be checked that the Lyapunov function (with $\beta = 3$) always decreases over time. Note that on the trajectory picture, the dashed curves represent the zero level-sets of the constraints and therefore the interior of the bold dashed curve is the feasible set.

Figure 2 presents the same results for a larger sampling time $\Delta t = 0.5$. Here the oscillations are very large but the trajectory is still evolving in the right direction.

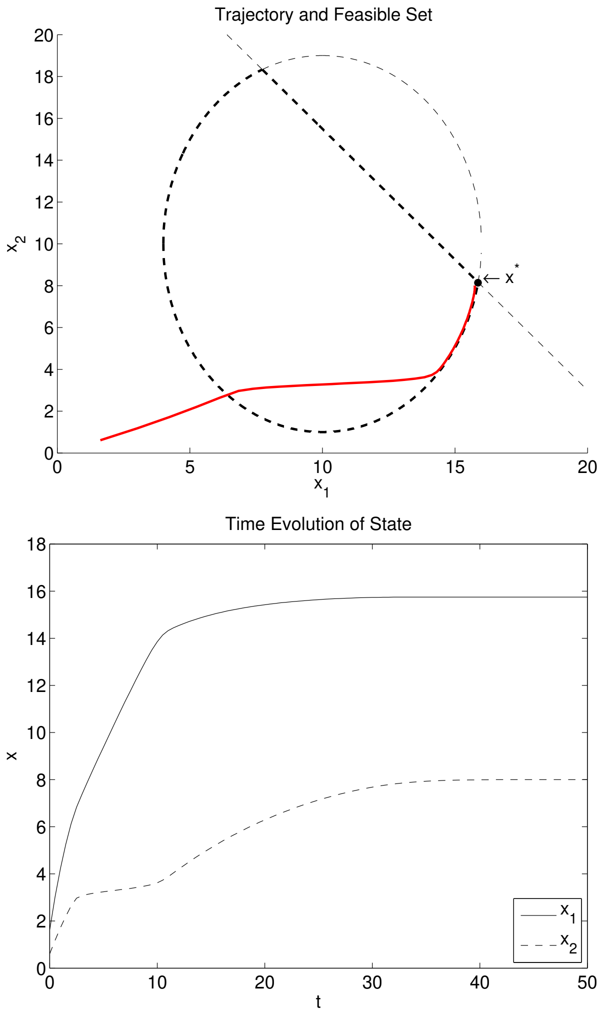

Figure 3. Simulation with sampling time $\Delta t = 0.5$ and smooth transitions. The oscillations in the sliding mode are completely removed.

In order to reduce the oscillations induced by the sliding mode, smooth transitions are implemented between the feasible and infeasible sets. The results are shown on Figure 3 . The oscillations are completely removed, unfortunately at a cost of slight decrease of the accuracy.

6 Conclusion

A very simple hybrid system implementing a convex optimization algorithm has been presented. The main idea is to follow the steepest descent direction for the objective function in the feasible set and for the constraints in the infeasible set. The continuous hybrid system guarantees that its trajectory enters the feasible set of the related optimization problem and next converges asymptotically to the set of optimal points. Furthermore, a possible discretization has been proposed for practical implementation. After sampling, the trajectory is then meandering around the border of the feasible set and finally oscillates around the optimal point.

In this paper, convex problems have been considered in order to be able to draw conclusions about the convergence to a global optimum. In case of non-convex problems, the convergence can still be assured towards a local optimum. However, nothing will guarantee that the global optimum is attained.

The scope of application of this kind of method will not be in the off-line optimization arena. However, in particular for on-line application, this method will be of interest. This is because of its implementation as a dynamical system, its simplicity, its low computation cost, and its capacity to be implemented in discrete-time.

7 Acknowledgment

The authors would like to thank the Competence Centre for Automotive Research (CCAR) for funding this research.

References

[1] A. Bemporad, F. Borrelli, and M. Morari. Model predictive control based on linear programming - the explicit solution. *IEEE Transactions on Automatic Control*, 47:1974– 1985, 2002.

[2] D.P. Bertsekas. *Nonlinear Programming*. Athena Scientific, 1999.

[3] A. Bhaya and E. Kaszkurewicz. *Control Perspectives on Numerical Algorithms and Matrix Problems*. Society for Industrial and Applied Mathematic, 2006.

[4] M. Bodson. Evaluation of optimization methods for control allocation. *Journal of Guidance, Control, and Dynamics*, 25(4):703–711, 2002.

[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[6] R. Brockett. Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems. *Linear Algebra and its Applications*, 146:79–91, 1991.

[7] F. Clarke. *Optimization and Nonsmooth Analysis*. Wiley, New-York, 1983.

[8] C. Edwards and S. Spurgeon. *Sliding Mode Control, Theory and Applications*. Taylor & Francis, 1998.

[9] A.F. Filippov. *Differential Equations with Discontinuous Right-hand Side*. Kluwer Academic Publishers, The Netherlands, 1988.

[10] C.E. Garcia, D.M. Prett, and M. Morari. Model predictive control: theory and practice - a survey. *Automatica*, 25(3):335–348, 1989.

[11] E. Hewitt and K. Stromberg. *Real and Abstract Analysis*. Springer, 1975.

[12] P. Ioannou and J. Sun. *Robust Adaptive Control*. Prentice Hall, 1996.

[13] K. Kashimaa and Y. Yamamoto. System theory for numerical analysis. *Automatica*, 43(7):1156–1164, 2007.

[14] H. Khalil. *Nonlinear Systems*. Prentice Hall, 2002.

[15] J.M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, Harlow, England, 2002.

[16] S. Martinez, J. Cortes, and F. Bullo. Motion coordination with distributed information. *IEEE Control System Magazine*, 27:75–88, 2007.

[17] S. Sastry and M. Bodson. *Adaptive Control - Stability, Convergence, and Robustness*. Prentice Hall, 1989.

[18] D. Shevitz and B. Paden. Lyapunov stability of nonsmooth systems. *IEEE Transactions on Automatic Control*, 39(9):1910–1914, 1994.

[19] K. Åström and B. Wittenmark. *Adaptive Control*. Prentice Hall, 1994.

[20] K. Åström and B. Wittenmark. *Computer-Controlled Systems: Theory and Design*. Prentice Hall, 1997.