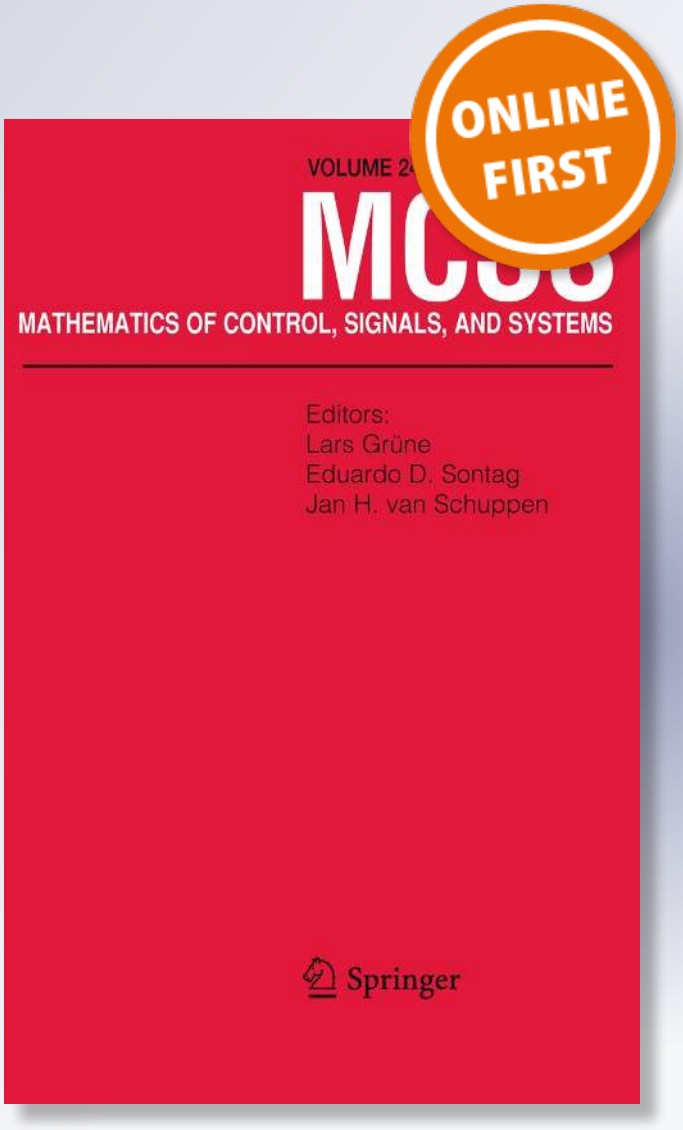
Design of a flight control architecture using a non-convex bundle method

Marion Gabarrou, Daniel Alazard & Dominikus Noll

Mathematics of Control, Signals, and Systems

ISSN 0932-4194

Math. Control Signals Syst.
DOI 10.1007/s00498-012-0093-z



Springer

Math. Control Signals Syst.
DOI 10.1007/s00498-012-0093-z

ORIGINAL ARTICLE

Design of a flight control architecture using a non-convex bundle method

Marion Gabarrou · Daniel Alazard · Dominikus Noll

Received: 20 January 2012 / Accepted: 21 September 2012

Abstract We design a feedback control architecture for longitudinal flight of an aircraft. The multi-level architecture includes the flight control loop to govern the short-term dynamics of the aircraft, and the autopilot to control the long-term modes. Using H_∞ performance and robustness criteria, the problem is cast as a non-convex and non-smooth optimization program. We present a non-convex bundle method, prove its convergence, and show that it is apt to solve the longitudinal flight control problem.

Keywords Non-smooth optimization · Non-convex bundle method · Feedback control · Multi-objective H_∞-control · Flight-controller and autopilot for longitudinal flight

1 Introduction

Automatic control of aircraft generally follows a scheme known as *guidance, navigation, and control* (GNC), which stipulates the use of architectures with interconnected control loops at different levels [1,2]. Figure 1 presents such a multi-level control architecture for the case of longitudinal flight.

M. Gabarrou · D. Noll (✉)
Université Paul Sabatier, Institut de Mathématiques, Toulouse, France
e-mail: noll@mip.ups-tlse.fr

M. Gabarrou
e-mail: Marion.Gabarrou@onera.fr

D. Alazard
Institut Supérieur de l'Aéronautique et de l'Espace, Toulouse, France
e-mail: alazard@isae.fr

Published online: 30 October 2012

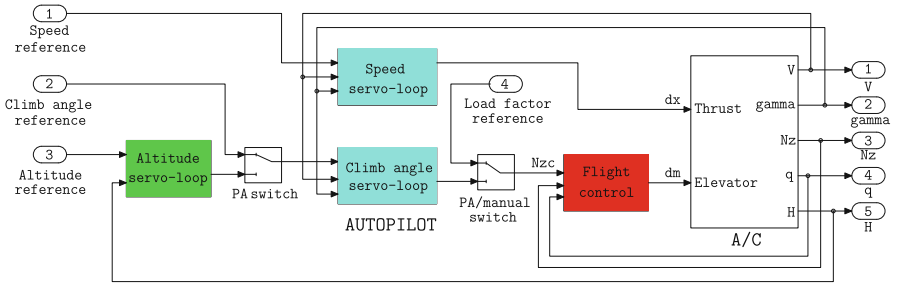


Fig. 1 Longitudinal control of an aircraft. The flight control loop (*red box*) controls the short-term dynamics in high frequency. The autopilot (*cyan boxes*) controls the long-term dynamics in low frequency (color figure online)

The inner loop (the control loop) governs the short-term dynamics in high frequency. It is represented by the *flight controller* in the red box. The outer loop (the guidance loop) serves to control the long-term dynamics in low frequency, represented by the *autopilot* shown in the cyan boxes. Roughly, GNC can therefore be understood as a frequency decoupling strategy. In the case of longitudinal flight, this decoupling dissociates short term rotational dynamics from long-term translational modes.

An important feature in longitudinal flight is the switch between automatic and manual mode on the input of the low-level control loop. The pilot can at any moment de-activate the autopilot and switch to manual mode. Autopilot and flight controller, therefore, operate together in cruising mode, but in manual mode, the commands of the pilot through the side-stick are interpreted as vertical load factor input references N_{z_c} and sent directly to the flight controller, which must then operate independently. In consequence, the two controllers have to be considered as decentralized units, but designed simultaneously to work satisfactory in automatic and manual mode. Due to lack of appropriate design techniques, current practice is to tune the two controller blocks independently, which leads to a lack of performance and robustness. The present work proposes a method which allows simultaneous synthesis of the full architecture.

The way we proceed is by translating simultaneous synthesis of both controller blocks into a non-smooth non-convex optimization program. We then present a non-smooth optimization method, prove its convergence, and use it to solve the control problem. Our algorithm expands on previous work [3–5] and develops the non-convex bundling technique originally put forward in [3,6]. Here, we use a progress function technique, which is motivated by older ideas for smooth problems in [7], and expands on the non-smooth approach in [8]. We propose a new form of the non-convex cutting plane oracle, referred to as *down-shifted tangents*, which offers several advantages over previously used methods.

The structure of the paper is as follows. In Sect. 2, we present the longitudinal control problem. Sections 3–4 present the non-convex bundle method and prove convergence. Section 5 goes back to the motivating application, gives specific information on how to compute Clarke subgradients, how to adapt the cutting plane strategy to the situation, and concludes with numerical results in longitudinal flight control.

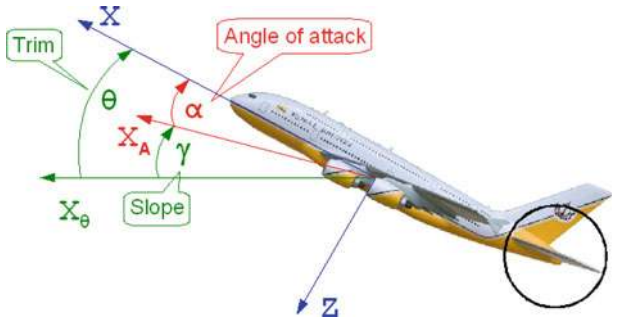


Fig. 2 Frame for longitudinal flight

2 Longitudinal flight

In this section, we present the control application, going gradually from a concrete class of examples to a more abstract setting. Section 2.2 indicates how performance and robustness criteria are found, and Sect. 2.3 presents a general setting which could be valid for other multi-objective H_∞-control problems.

2.1 Open-loop model

We consider an aircraft moving in the vertical plane (Fig. 2). Its aerodynamic behavior, linearized around one particular flight point (Mach= 0.7, Altitude= 5,000 ft), is described by a set of eqnarrays of the form

$$\begin{bmatrix} \dot{x}_P \\ y_P \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} x_P \\ u \end{bmatrix} \tag{1}$$

where numerical data are given in the Appendix. Here, the state is $x_P = [V,\ \gamma,\ \alpha,\ q,\ H]^{\mathrm{T}}$, the control $u = [d_x,\ d_m]^{\mathrm{T}}$, and the output is $y_P = [V,\ \gamma,\ N_z,\ q,\ H]^{\mathrm{T}}$. In particular,

- The states are aerodynamic speed V (m/s), climb angle (or slope) γ (rad), angle of attack α (rad), pitch rate $q = \dot{\theta} = \dot{\alpha} + \dot{\gamma}$ (rad/s), and altitude H $[m]$.
- The controls are engine thrust d_x (% of the maximal thrust) and elevator deflection d_m (rad).
- The measurements are vertical load factor N_z $[\mathrm{m/s^2}]$, and $[V, \gamma, q, H]$.

The longitudinal dynamics are characterized by 5 eigenvalues, which for the specific flight point chosen are

- $\lambda_{1,2} = -0.56 \pm 1.61\, j$ (i.e., pulsation: 1.7 rad/s and damping ratio: 0.33) is the angel-of-attack (AoA) oscillation, also called short-term mode. This mode mainly affects the states α and q,

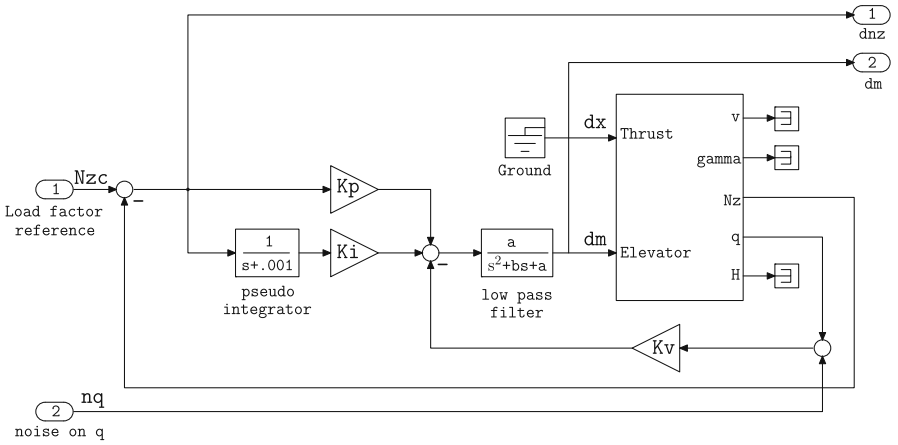


Fig. 3 Functional scheme of the flight control loop

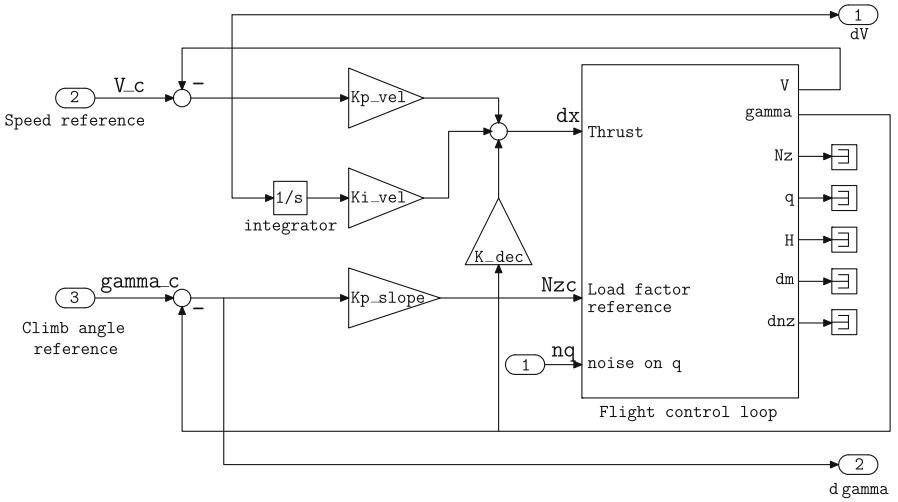


Fig. 4 Functional scheme of the guidance loop

- $\lambda_{3,4} = -0.0039 \pm 0.064\, j$ (i.e., pulsation: 0.064 rad/s and damping ratio: 0.06) is the phugoïd mode, also called long-term mode. It mainly affects the states γ and V,
- $\lambda_5 = -0.0026$ is the altitude convergence mode (a very long-term mode). It mainly impacts the state H.

The structures of the command laws are presented in Figs. 3 and 4. Practitioners prefer simple controller structures in order to address issues like saturation, interpolation of the controller according to flight operating conditions, and feedforward compensation adapted to the various aircraft configurations.

The autopilot generates engine thrust d_x and the vertical load factor input reference N_{z_c}

$$K^{(1)}: \begin{bmatrix} d_x(s) \\ N_{z_c}(s) \end{bmatrix} = \begin{bmatrix} K_{p_{\text{vel}}} + \frac{K_{i_{\text{vel}}}}{s} & K_{\text{dec}} \\ 0 & K_{p_{\text{slope}}} \end{bmatrix} \begin{bmatrix} dV(s) \\ d\gamma(s) \end{bmatrix} \tag{2}$$

and involves a P-feedback to servo-loop the speed V, a PI-feedback to control the slope γ, and a P feedback for γ in order to decouple V from γ.

The flight-control law governing the elevator deflection d_m reads

$$K^{(2)}: \; d_m(s) = F(s) \begin{bmatrix} K_p + \frac{K_i}{s+\varepsilon} & -K_v \end{bmatrix} \begin{bmatrix} N_{z_c}(s) - N_z(s) \\ q(s) \end{bmatrix} \tag{3}$$

and combines a PI feedback to servo-loop the vertical load factor N_z with a P-feedback on the pitch rate q to damp the angle-of-attack (AoA) oscillation. In addition, the role of the low-pass filter $F(s) = a/(s^2 + bs + a)$ is to prevent spill-over of unmodeled dynamics, caused mainly by flexible structural modes [9].

Remark 1 PDE-based models for flexible aircraft are currently developed, so future approaches might give better insight into the presently unmodeled structural modes. Validating such a model is outside the scope of the present contribution.

The goal is to optimize the controller gains grouped in the optimization variable

$$\mathbf{x} = [K_p; K_i; K_v; b; a; K_{p_{\text{slope}}}; K_{p_{\text{vel}}}; K_{i_{\text{vel}}}; K_{\text{dec}}],$$

in order to synthesize the two controller blocks $K^{(1)}$ and $K^{(2)}$ in such a way that performance and robustness requirements are met in automatic and manual mode.

Remark 2 In a conventional approach, we would fix the low-pass filter F beforehand, and then design $K^{(1)}$ and the remaining parameters in $K^{(2)}$ separately. Our approach shows that it is preferable to design all elements simultaneously, as this leads to better performance. The conventional block-by-block design can then still be useful to initialize the optimization algorithm.

2.2 Controller specifications

Performance and robustness criteria are defined by introducing frequency weights on specific closed-loop transfer functions $T_i(\mathbf{x}, s) := T_{w_i \to z_i}(\mathbf{x}, s)$ between suitably chosen inputs w_i and outputs z_i. In this study, we consider the six transfers $V_c \to dV$, $\gamma_c \to d\gamma$, $\gamma_c \to dV$, $V \to d\gamma$, $N_{z_c} \to dN_z$, $(N_{z_c}, n_q) \to dm$. For each of these channels $w_i \to z_i$, we construct a state-space representation

$$P_i(s): \begin{bmatrix} \dot{x}_i \\ z_i \\ y_i \end{bmatrix} = \begin{bmatrix} A^i & B_1^i & B_2^i \\ C_1^i & D_{11}^i & D_{12}^i \\ C_2^i & D_{21}^i & D_{22}^i \end{bmatrix} \begin{bmatrix} x_i \\ w_i \\ u_i \end{bmatrix}, \quad i = 1, \ldots, 6, \tag{4}$$

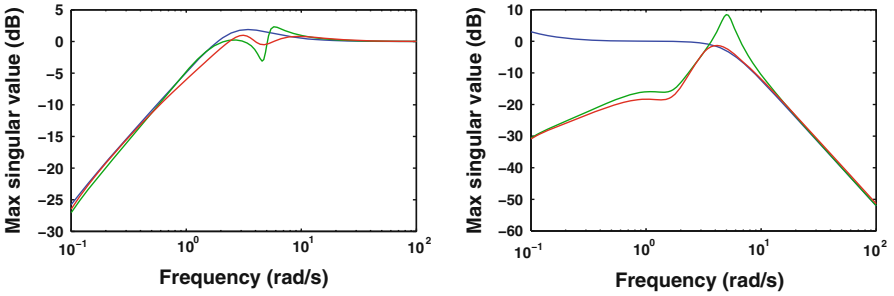


Fig. 5 Criteria for flight controller. Performance channel $T_{N_z \to dN_z}$ on the left assures good tracking of vertical load factor in the range $[10^{-1}, 10^{0}]$. Robustness channel $T_{nq \to dm}$ on the right limits influence of noise on elevator deflection in the range $> 10^{1}$. *Blue* is template, *green* initial guess, *red* optimized. Both criteria are not relevant for frequencies below 10^{-1} (color figure online)

where $x_i \in \mathbb{R}^{n_i}$ is the state of representation P_i, $u_i \in \mathbb{R}^{m_2^i}$ the control input and $y_i \in \mathbb{R}^{p_2^i}$ the measured output. Observe that channels $i = 1, \ldots, 4$ concern the autopilot (2). Therefore, $\dim(u_1) = \cdots = \dim(u_4) = 2$, $\dim(y_1) = \cdots = \dim(y_4) = 2$, and we connect the same controller

$$u_i(s) = K^{(1)}(\mathbf{x}, s) y_i(s), \quad i = 1, \ldots, 4$$

to the first four channels. Similarly, channels $i = 5, 6$ concern the flight controller (3), so that $\dim(u_5) = \dim(u_6) = 1$ and $\dim(y_5) = \dim(y_6) = 2$, and we connect the same controller

$$u_i(s) = K^{(2)}(\mathbf{x}, s) y_i(s), \quad i = 5, 6$$

to the last two channels. Notice that $K^{(1)}$ depends on all nine parameters in $\mathbf{x}$, whereas $K^{(2)}$ depends only on the flight control gains $(\mathbf{x}_1, \ldots, \mathbf{x}_5) = (K_p, K_i, K_v, b, a)$. This reflects the fact that we want $K^{(2)}$ independent of the autopilot in order to guarantee closed-loop performances during manual mode.

The rationale of these channels is as follows. The first specification for flight control is tracking of the load factor N_z. We use a template $W_1(s) = \left(s^2 + 4s\right) / \left(s^2 + 4s + 7\right)$ for $T_{N_{zc} \to dN_z}(\mathbf{x}, s)$, where dN_z is the vertical load factor tracking error. In other words, we want $T_5 := W_1^{-1} T_{N_{zc} \to dN_z}$ to be close to 1. The situation can be seen in Fig. 5 left.

The second specification concerns robustness with regard to unmodeled dynamics. We want to cut off the command signal $d_m(s)$ in high frequency (roll-off). To do this, we impose the low-pass template $W_2(s) = 25/(s^2 + \sqrt{2}\,5\,s + 25)$, which aims at shaping a second order roll-off beyond 5 rad/s, on $T_{(N_{zc}, n_q) \to dm}(\mathbf{x}, s)$, where n_q is the pitch rate measurement noise. That means we want $T_6 = W_2^{-1} T_{(N_{zc}, n_q) \to dm}$ close to 1, and this channel is visualized in Fig. 5 right.

Remark 3 One can notice in Fig. 5 that frequency-domain templates for T_5, T_6 need not be satisfied for pulsations under 0.1 rad/s. For the flight controller, we are only

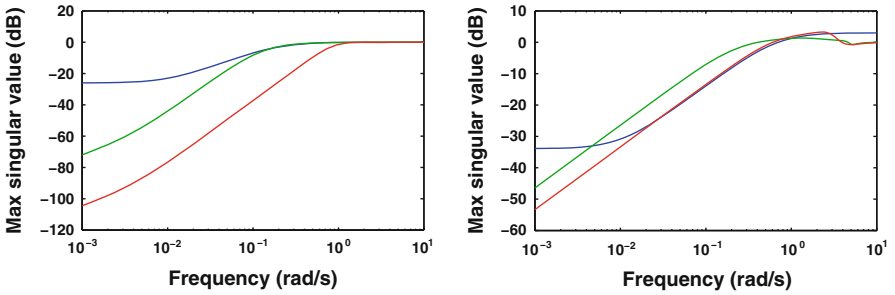


Fig. 6 Performance channels for autopilot. Velocity tracking error $T_{V\to dV}$ left and climb angle (*slope*) tracking error $T_{\gamma\to d\gamma}$ right are kept small for frequencies below 10^{-1}. *Blue* template, *green* before optimization, *red* after optimization (color figure online)

interested in the high frequency band $\Omega_{\text{high}} = [0.1, 10]\text{rad/s}$, as its performances concern the short-term dynamics only and are not affected even when templates are violated in very low frequency.

The specifications for the autopilot include tracking of speed and slope (climb angle). For that we introduce a template $W_3(s) = (s + 0.01)/(s + 0.2)$, which we use for both $T_{V\to dV}(\mathbf{x})$ and $T_{\gamma\to d\gamma}(\mathbf{x})$, where dV, $d\gamma$ are the tracking errors of speed V and slope γ. We put $T_1 = W_3^{-1}T_{V_c\to dV}$ and $T_2 = W_3^{-1}T_{\gamma\to d\gamma}$, visualized in Fig. 6, which we want as small as possible. Furthermore, we want to decouple speed and slope, and for that we impose the template $0.05 \times W_3(s)$ on $T_{\gamma\to dV}(\mathbf{x}, s)$ and $T_{V\to d\gamma}(\mathbf{x}, s)$. This defines T_3 and T_4, shown in Fig. 7, which again should be small.

Remark 4 The autopilot controls the low-frequency range, which means frequency-domain templates for $T_1, \ldots, T_4$ have only to be satisfied for frequencies within the low-frequency band $\Omega_{\text{low}} = [0, 0.1]\text{rad/s}$.

2.3 Optimization program

The performance and robustness specifications are now cast as an optimization program:

$$\begin{aligned} \text{minimize} \quad & f(x) := \max_{i=1,\ldots,4} \left\| W_i^{-1} T_{w_i\to z_i}(\boldsymbol{x}) \right\|^2_{\infty,\Omega_{\text{low}}} \\ \text{subject to} \quad & c(x) := \max_{i=5,6} \left\| W_i^{-1} T_{w_i\to z_i}(\boldsymbol{x}) \right\|^2_{\infty,\Omega_{\text{high}}} - r^2 \le 0 \\ & x \in \mathbb{R}^n \end{aligned} \tag{5}$$

where objective f and constraint c represent weighted H_∞-norms on different frequency bands Ω_{low} and Ω_{high}, and where $r \approx 1$. Notice that in each case W^{-1} is a transfer function, so $W^{-1}T_{w\to z} = T_{w\to\widetilde{z}}$ with $\widetilde{z} = W^{-1}z$ is just another

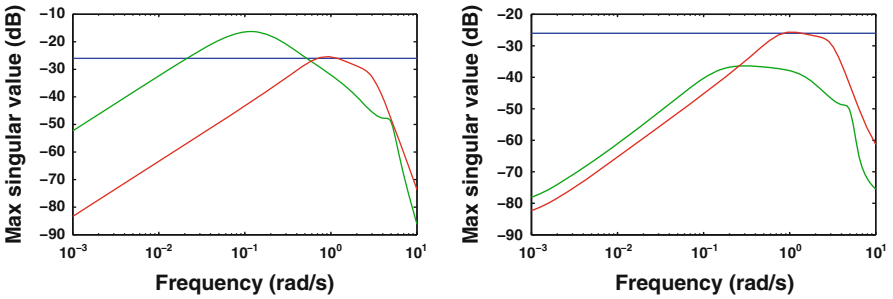


Fig. 7 Cross channels $\gamma \to dV$ (*left*) and $V \to d\gamma$ (*right*) for autopilot. The template -26 dB is given in *blue*. Smallness of these responses assures decoupling of climb angle and velocity. The constant template indicates simply a weighting of the H_∞-norms. Decoupling increases the overall robustness of the design (color figure online)

closed-loop transfer channel from w to $\widetilde{z}$. Each norm $\| W_i^{-1} T_{w_i \to z_i} \|_{\infty,\Omega}^2$ contributing to the maximum in f or c has therefore the abstract form

$$f(\mathbf{x}) = \|\mathcal{F}(\mathbf{x}, \cdot)\|_{\infty,\Omega}^2 = \sup_{\omega \in \Omega} \lambda_1[\mathcal{F}(\mathbf{x}, \omega)] = \sup_{\omega \in \Omega} f(\mathbf{x}, \omega) \tag{6}$$

where $\lambda_1(X)$ is the maximum eigenvalue of the Hermitian matrix X, and where the mapping

$$\mathcal{F}(\mathbf{x}, \omega) = T(\mathbf{x}, j\omega) T(\mathbf{x}, j\omega)^H \tag{7}$$

is smooth in $\mathbf{x}$, jointly continuous in $(\mathbf{x}, \omega)$, and takes values in a space $\mathbb{H}$ of appropriately sized complex Hermitian matrices. This is due to the fact that $K(\mathbf{x})$ depends smoothly on the design parameter $\mathbf{x}$ (see Lemma 4, Sect. 5). Given the fact that the H_∞-norm is only defined for stable transfer functions, f and c are only defined on the set S of those parameters $\mathbf{x}$ where all $T_{w_i \to z_i}(\mathbf{x})$ are stable. In other words, program (5) has the hidden constraint $\mathbf{x} \in S$.

The salient point is that (5) is highly non-smooth due to the presence of the semi-infinite maximum eigenvalue function (6). We therefore develop a non-smooth progress function method to solve such programs algorithmically. A similar rationale was previously applied to mixed H_2/H_∞-control [8], where in contrast with (5) the objective function f was smooth. H_∞/H_∞-control with structured control laws $K(\mathbf{x})$ was pioneered in [10]. Optimization methods for the band-limited H_∞-norm were first discussed in [11].

Remark 5 In classical H_∞-loopshaping, the use of the banded H_∞-norm is avoided mainly due to lack of methods to deal with it algorithmically. The advantage of working with banded norms is that the state-space dimension of the channel representations (4) is kept small. If one tries to adapt the templates W_i so that their effect is negligible outside the band Ω of interest, the state space dimension of the plants P^i increases.

Remark 6 Simple control architectures like (2), (3) are preferred by practitioners for various reasons. The building blocks are well-understood, and they are easier to hardware embed. It is therefore important to stress that it is precisely this need for simplicity which renders controller design difficult. Namely, computing advanced but unstructured full-order H_∞-controller e.g. by solving algebraic Riccati equations (AREs) or linear matrix inequalities (LMIs), would be easier.

Remark 7 The gap between abstract H_∞-theory based on AREs on the one hand, and the need for practical controller structures to solve real problems on the other, has created a paradoxal situation, where controllers are tuned using heuristics, while the sophisticated techniques of H_∞-control cannot be brought to work. Our contribution helps to close this gap, as it allows to apply the H_∞-paradigm to structured controllers. We mention that this requires optimization techniques like (5), because even for a relatively simple structure like (2), (3), it is impossible to simply throw the blocks $K^{(1)}$, $K^{(2)}$ by hand, as there are six concurring performance and robustness specifications to satisfy.

3 Non-convex bundle method

In this section, we present our non-smooth algorithm, discuss its constituents and rationale, and prove convergence. We consider an abstract version of (5),

$$\min\{f(\mathbf{x}) : c(\mathbf{x}) \leq 0, \mathbf{x} \in \mathbb{R}^n\}, \tag{8}$$

where $f, c : \mathbb{R}^n \to \mathbb{R}$ are locally Lipschitz functions. To solve (8) algorithmically, we assume that for every $\mathbf{x} \in \mathbb{R}^n$ we have the function value $f(\mathbf{x})$ and a Clarke subgradient $g \in \partial f(\mathbf{x})$ at our disposal, and similarly $c(\mathbf{x}), h \in \partial c(\mathbf{x})$. In cases where several subgradients are available, the method can be adapted to include this information.

3.1 Progress function and optimality conditions

We address program (8) by introducing a *progress function* $F(\cdot, \mathbf{x})$ at the current iterate $\mathbf{x}$,

$$F(\cdot, \mathbf{x}) = \max\{f(\cdot) - f(\mathbf{x}) - \mu c(\mathbf{x})_+, c(\cdot) - c(\mathbf{x})_+\}, \tag{9}$$

where $\mu > 0$ is fixed and $c(\mathbf{x})_+ = \max(c(\mathbf{x}), 0)$. The idea is as follows. Notice that $F(\mathbf{x}, \mathbf{x}) = 0$, where either the left branch $f(\cdot) - f(\mathbf{x}) - \mu c(\mathbf{x})_+$ or the right branch $c(\cdot) - c(\mathbf{x})_+$ of (9) is active at $\mathbf{x}$, i.e., attains the maximum, depending on whether $\mathbf{x}$ is feasible for (8) or not. If $c(\mathbf{x}) > 0$, meaning that $\mathbf{x}$ is infeasible, then the right-hand term in (9) is active at $\mathbf{x}$, whereas the left-hand term equals $-\mu c(\mathbf{x}) < 0$ at $\mathbf{x}$. Reducing $F(\cdot, \mathbf{x})$ below its value 0 at the current $\mathbf{x}$ therefore reduces constraint violation. The period when iterates $\mathbf{x}$ are infeasible is called phase I.

On the other hand, if $c(\mathbf{x}) \leq 0$, meaning that $\mathbf{x}$ is feasible, then the left-hand term in $F(\cdot, \mathbf{x})$ becomes dominant, so reducing $F(\cdot, \mathbf{x})$ below its current value 0 at $\mathbf{x}$ now reduces f, while maintaining feasibility. This is phase II, where the true optimization of f takes place.

The following lemma, whose proof can be found in [8], gives an optimality test for program (8) based on the progress function. Recall that $\mathbf{x}^*$ satisfies the John necessary optimality conditions for program (8) if there exist $\lambda_0^* \geq 0$, $\lambda_1^* \geq 0$ with $\lambda_0^* + \lambda_1^* = 1$ such that $0 \in \lambda_0^* \partial f(\mathbf{x}^*) + \lambda_1^* \partial c(\mathbf{x}^*)$, $\lambda_1^* c(\mathbf{x}^*) = 0$, and $c(\mathbf{x}^*) \leq 0$. If in addition $\lambda_0^* > 0$, then $\mathbf{x}^*$ satisfies the Karush–Kuhn–Tucker conditions with associated Lagrange multiplier $\lambda_1^*/\lambda_0^* \geq 0$.

Lemma 1 (Compare [8, Lemma 5.1]). *Suppose* $0 \in \partial_1 F(\mathbf{x}^*, \mathbf{x}^*)$ *for some* $\mathbf{x}^* \in \mathbb{R}^n$, *where* ∂_1 *is the subdifferential with respect to the first coordinate. Then we have the following possibilities:*

1. *Either* $c(\mathbf{x}^*) > 0$, *in which case* $\mathbf{x}^*$ *is a critical point of c, called a critical point of constraint violation.*
2. *Or* $c(\mathbf{x}^*) \leq 0$, *in which case* $\mathbf{x}^*$ *satisfies the John necessary optimality conditions for program (8). In addition, there are two sub-cases:*
   (a) *Either* $\mathbf{x}^*$ *is a Karush–Kuhn–Tucker point of (8).*
   (b) *Or* $\mathbf{x}^*$ *fails to be a Karush–Kuhn–Tucker point. The latter can only happen when* $c(\mathbf{x}^*) = 0$ *and at the same time* $0 \in \partial c(\mathbf{x}^*)$.

We plan to solve program (8) by constructing a sequence of iterates $\mathbf{x}^j$, such that $\mathbf{x}^{j+1}$ is a descent step for $F(\cdot, \mathbf{x}^j)$ away from $\mathbf{x}^j$. That is $F(\mathbf{x}^{j+1}, \mathbf{x}^j) < F(\mathbf{x}^j, \mathbf{x}^j) = 0$ in a qualified way. We expect $\mathbf{x}^j$ to converge to a point $\mathbf{x}^*$ satisfying $0 \in \partial_1 F(\mathbf{x}^*, \mathbf{x}^*)$. Lemma 1 tells us that $\mathbf{x}^*$ is a KKT point of program (8) as a rule. The exceptions from that rule are conditions 1. and 2b. Condition 1 gives the case where iterates $\mathbf{x}^j$ get stuck at a limit point $\mathbf{x}^*$ with value $c(\mathbf{x}^*) > 0$ in phase I. This is a critical point of constraint violation. (Condition 2b is the limiting case, where $c(\mathbf{x}^*) = 0$. This case was never observed in our experiments and appears unlikely in practice.) A first order method may indeed get trapped at such points, and in classical mathematical programming second order techniques are used to avoid them. Here we are working with a non-smooth program, where second order elements are not available. When critical points of constraint violation are encountered, we restart our method at a different initial guess.

When reducing constraint violation in phase I, a controlled increase in f not exceeding $\mu c(\mathbf{x})$ is granted. This helps the algorithm in not being trapped at infeasible critical points of f alone. For the theoretical justification see Sect. 4.

The algorithm used to compute solutions to (8) is shown schematically in Fig. 8, and stated formally as Algorithm 1. We subsequently describe its essential features.

3.2 Working model

We denote the current serious iterate of the algorithm by $\mathbf{x}$, or $\mathbf{x}^j$ if the counter j of the outer loop is used. If a new serious iterate is found, it will be denoted by $\mathbf{x}^+$, or $\mathbf{x}^{j+1}$. Serious iterates refer to the outer loop colored blue in Fig. 8.

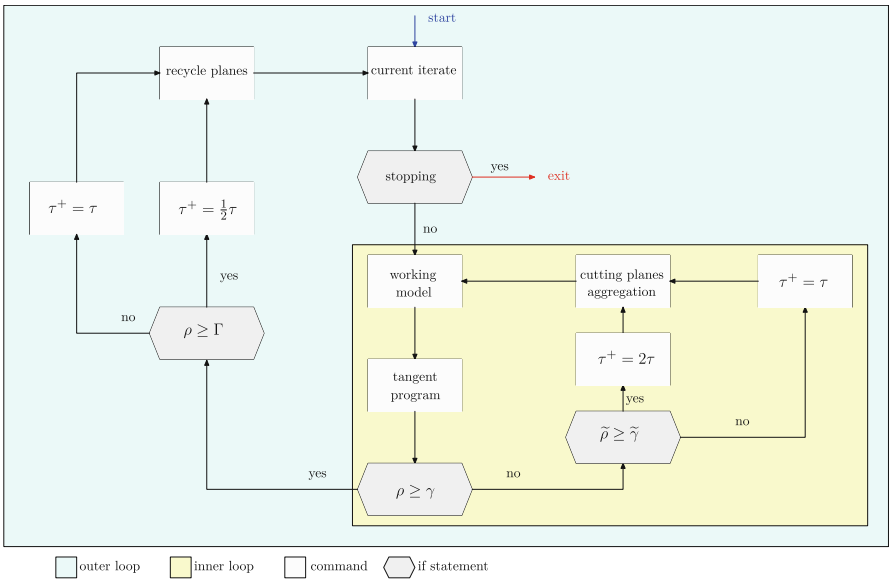


Fig. 8 Flowchart of proximity control algorithm (color figure online)

At the current iterate $\mathbf{x}$, we use approximations $F_k(\cdot, \mathbf{x})$ of the progress function $F(\cdot, \mathbf{x})$ called working models. Every working model satisfies $F_k(\mathbf{x}, \mathbf{x}) = 0$ and $\partial_1 F_k(\mathbf{x}, \mathbf{x}) \subset \partial_1 F(\mathbf{x}, \mathbf{x})$. Moreover, the F_k decompose into a polyhedral convex possibly non-smooth first-order part, $F_k^{[1]}(\cdot, \mathbf{x}) = \max_{(a,g)\in\mathcal{G}_k} a + g^\top(\cdot - \mathbf{x})$, and a nonconvex but smooth second-order part $F^{[2]}(\cdot, \mathbf{x}) = \frac{1}{2}(\cdot - \mathbf{x})^\top Q(\mathbf{x})(\cdot - \mathbf{x})$:

$$F_k(\cdot, \mathbf{x}) = \max_{(a,g)\in\mathcal{G}_k} a + g^\top(\cdot - \mathbf{x}) + \tfrac{1}{2}(\cdot - \mathbf{x})^\top Q(\mathbf{x})(\cdot - \mathbf{x}). \tag{10}$$

Here $\mathcal{G}_k \subset \mathbb{R}^n \times \mathbb{R}^n$ is a finite set, which we update continuously during the inner loop with counter k, colored yellow in Fig. 8. In contrast, the second order term $F^{[2]}(\cdot, \mathbf{x}) = \frac{1}{2}(\cdot - \mathbf{x})^\top Q(\mathbf{x})(\cdot - \mathbf{x})$ is held fixed during the inner loop and only updated between serious steps $\mathbf{x} \to \mathbf{x}^+$. We allow $Q(\mathbf{x}) \in \mathbb{S}^n$ to be indefinite, and we assume that the operator $\mathbf{x} \mapsto Q(\mathbf{x})$, $\mathbb{R}^n \to \mathbb{S}^n$, is bounded on bounded sets. Our notation $F_k(\cdot, \mathbf{x}) = F_k^{[1]}(\cdot, \mathbf{x}) + F^{[2]}(\cdot, \mathbf{x})$ highlights that the second order part does not depend on k.

3.3 Tangent program

In the inner loop at serious iterate $\mathbf{x}$, we generate trial steps $\mathbf{y}^k$ indexed by the counter k of the inner loop, which are candidates to be elected as the new serious iterate $\mathbf{x}^+$. The trial step $\mathbf{y}^k$ is obtained by solving the convex tangent program

$$\min_{\mathbf{y}\in\mathbb{R}^n} F_k(\mathbf{y},\mathbf{x}) + \tfrac{\tau_k}{2}\,\|\mathbf{y}-\mathbf{x}\|^2\,. \tag{11}$$

Here, τ_k is the proximity control parameter, which is updated during the inner loop. Convexity of (11) is assured because we require $Q(\mathbf{x}) + \tau_k I \succ 0$ for every k, where $\succ 0$ means positive definite. Observe that (11) is equivalent to the convex quadratic program (CQP)

$$\begin{aligned} \text{minimize}\quad & t + \frac{1}{2}(\mathbf{y}-\mathbf{x})^\top (Q(\mathbf{x}) + \tau_k I)(\mathbf{y}-\mathbf{x}) \\ \text{subject to}\quad & a + g^\top(\mathbf{y}-\mathbf{x}) \le t \\ & (a,g) \in \mathcal{G}_k \end{aligned} \tag{12}$$

with unknown variable $(t,\mathbf{y}) \in \mathbb{R}^{1+n}$, which can be conveniently solved with standard CQP solvers.

The necessary optimality condition for (11) is $\tau_k(\mathbf{x}-\mathbf{y}^k) \in \partial_1 F_k(\mathbf{y}^k,\mathbf{x})$, or equivalently,

$$g_k^* := (Q(\mathbf{x}) + \tau_k I)(\mathbf{x}-\mathbf{y}^k) \in \partial_1 F_k^{[1]}(\mathbf{y}^k,\mathbf{x}), \tag{13}$$

and we call g_k^* the aggregate subgradient. Equivalently, there exist pairs $(a_1,g_1),\ldots,(a_r,g_r) \in \mathcal{G}_k$ and $\lambda_i > 0$, $\sum_{i=1}^r \lambda_i = 1$, such that

$$a_i + g_i^\top(\mathbf{y}^k - \mathbf{x}) = t_k,\ i = 1,\ldots,r \qquad (Q(\mathbf{x}) + \tau_k I)(\mathbf{x}-\mathbf{y}^k) = \sum_{i=1}^r \lambda_i g_i, \tag{14}$$

where $t_k = F_k^{[1]}(\mathbf{y}^k,\mathbf{x})$. Putting $a_k^* = \sum_{i=1}^r \lambda_i a_i$, we call $m_k^*(\cdot,\mathbf{x}) = a_k^* + g_k^{*\top}(\cdot-\mathbf{x})$ the aggregate plane. We say that the subgradients $g_1,\ldots,g_r$ are *called* by the aggregate subgradient, and that the planes $a_i + g_i^\top(\cdot-\mathbf{x})$ are called by the aggregate plane. An equivalent way to define the aggregate plane is to use (13) and choose a_k^* such that $m_k^*(\cdot,\mathbf{x}) = a_k^* + g_k^{*\top}(\cdot-\mathbf{x})$ has value $t_k = F_k^{[1]}(\mathbf{y}^k,\mathbf{x})$ at $\mathbf{y}^k$.

When building the new set $\mathcal{G}_{k+1}$ after a null step $\mathbf{y}^k$, we assure that $(a_k^*, g_k^*) \in \mathcal{G}_{k+1}$. This allows us to drop any of the older $(a_i,g_i) \in \mathcal{G}_k$.

3.4 Acceptance test

In order to decide whether the solution $\mathbf{y}^k$ of (11) is acceptable to become the new serious iterate $\mathbf{x}^+$ in the outer loop, we use the test

$$\rho_k = \frac{F(\mathbf{y}^k,\mathbf{x})}{F_k(\mathbf{y}^k,\mathbf{x})} \overset{?}{\ge} \gamma, \tag{15}$$

where $0 < \gamma < 1$ is fixed throughout. As usual, this test compares actual decrease and predicted decrease at $\mathbf{y}^k$. If F_k represents F accurately at $\mathbf{y}^k$, we expect $\rho_k \approx 1$, but

we accept $\mathbf{y}^k$ as the new $\mathbf{x}^+$ already when $\rho_k \geq \gamma$. According to standard terminology in bundle methods, $\mathbf{y}^k$ is called a null step if $\rho_k < \gamma$, while the case $\rho_k \geq \gamma$, when $\mathbf{x}^+ = \mathbf{y}^k$, is referred to as a serious step.

3.5 Cutting planes

If the trial step $\mathbf{y}^k$ fails the acceptance test (15), then agreement between F and F_k at $\mathbf{y}^k$ was bad. In this case, the inner loop has to continue, but we have to improve the quality of the next working model $F_{k+1}(\cdot, \mathbf{x})$ in order to do better at the next trial. Since the second order part $F^{[2]}(\cdot, \mathbf{x})$ of the model does not change during the inner loop k, we have to improve the first-order part $F^{[1]}_{k+1}(\cdot, \mathbf{x})$. In traditional bundle methods this is achieved by including a cutting plane into the new working model, whose role is to cut away the unsuccessful trial step $\mathbf{y}^k$. In the convex case, cutting planes are simply tangents to the first-order part $F^{[1]}(\cdot, \mathbf{x})$ of the progress function $F(\cdot, \mathbf{x})$ at $\mathbf{y}^k$. Without convexity it is more delicate to obtain a suitable cutting plane. In this study, we use downshifted tangents as substitutes for the traditional convex cutting planes. Here is the construction.

In accordance with the decomposition of the working model $F_k(\cdot, \mathbf{x})$, we decompose the progress function

$$F(\cdot, \mathbf{x}) = F^{[1]}(\cdot, \mathbf{x}) + F^{[2]}(\cdot, \mathbf{x}),$$

where $F^{[2]}(\cdot, \mathbf{x}) = \frac{1}{2}(\cdot - \mathbf{x})^\top Q(\mathbf{x})(\cdot - \mathbf{x})$ is the second-order part, and $F^{[1]} = F - F^{[2]}$ is the first-order part.

Given the null step $\mathbf{y}^k$, pick a subgradient $g_k \in \partial_1 F^{[1]}(\mathbf{y}^k, \mathbf{x})$. Then the affine function $t_k(\cdot) = F^{[1]}(\mathbf{y}^k, \mathbf{x}) + g_k^\top(\cdot - \mathbf{y}^k)$ is a tangent to $F^{[1]}(\cdot, \mathbf{x})$ at $\mathbf{y}^k$. Without convexity we may not use $t_k(\cdot)$ directly as a cutting plane. We do not even know whether $t_k(\mathbf{x}) \leq F^{[1]}(\mathbf{x}, \mathbf{x}) = F(\mathbf{x}, \mathbf{x}) = 0$, as would be the minimum requirement for a plane contributing to the new model $F^{[1]}_{k+1}(\cdot, \mathbf{x})$. We therefore define the downshift as

$$s_k = [t_k(\mathbf{x})]_+ + c\|\mathbf{y}^k - \mathbf{x}\|^2, \tag{16}$$

where $c > 0$ is some small constant fixed at the beginning. Now we define the cutting plane as

$$m_k(\cdot, \mathbf{x}) = t_k(\cdot) - s_k. \tag{17}$$

Notice that $\nabla m_k(\cdot, \mathbf{x}) = \nabla t_k(\cdot) = g_k$, while $m_k(\mathbf{x}, \mathbf{x}) \leq -c\|\mathbf{y}^k - \mathbf{x}\|^2 \leq 0$. The cutting plane can also be written as $m_k(\cdot, \mathbf{x}) = a_k + g_k^\top(\cdot - \mathbf{x})$, where

$$a_k = t_k(\mathbf{x}) - s_k = t_k(\mathbf{x}) - [t_k(\mathbf{x})]_+ - c\|\mathbf{y}^k - \mathbf{x}\|^2.$$

In particular, the cutting plane depends on the full information $\mathbf{x}$, $\mathbf{y}^k$, and $g_k \in \partial_1 F^{[1]}(\mathbf{y}^k, \mathbf{x})$, whereas $t_k(\cdot)$ only depends on $\mathbf{y}^k$ and the specific subgradient g_k at $\mathbf{y}^k$. We assure that $\mathcal{G}_{k+1}$ contains the newly generated pair (a_k, g_k).

3.6 Exploiting the structure of the progress function

The construction of the cutting plane in Sect. 3.5 does not fully exploit the structure of the first-order part $F^{[1]}$ of the progress function F. Namely, observe that

$$\begin{aligned} F^{[1]}(\cdot, \mathbf{x}) &= \max\left\{ f(\cdot) - f(\mathbf{x}) - \mu c(\mathbf{x})_+ - F^{[2]}(\cdot, \mathbf{x}), c(\cdot) - c(\mathbf{x})_+ - F^{[2]}(\cdot, \mathbf{x}) \right\} \\ &=: \max\left\{ F^{[11]}(\cdot, \mathbf{x}), F^{[12]}(\cdot, \mathbf{x}) \right\}, \end{aligned} \tag{18}$$

and so far our construction only includes a down-shifted tangent to that part $F^{[1i]}$ of $F^{[1]}$ which is active at $\mathbf{y}^k$. It is beneficial to include also a down-shifted tangent to the inactive part. Indeed, suppose for instance $F_k^{[11]}(\mathbf{y}^k, \mathbf{x}) < F_k^{[12]}(\mathbf{y}^k, \mathbf{x})$. Then in section 3.5 we included a downshifted tangent to $F_k^{[12]}$ into $\mathcal{G}_{k+1}$. Now let $\widetilde{t}_k(\cdot)$ be a tangent to the inactive part $F_k^{[11]}$ at $\mathbf{y}^k$. Then we build $\widetilde{m}_k(\cdot, \mathbf{x}) = \widetilde{t}_k(\cdot) - \widetilde{s}_k$, where $\widetilde{s}_k = \left[\widetilde{t}_k(\mathbf{x})\right]_+ + c\|\mathbf{y}^k - \mathbf{x}\|^2$ just as in (16), that is, we down-shift with respect to the value $F(\mathbf{x}, \mathbf{x}) = 0$ at $\mathbf{x}$, and not with respect to the potentially lower value $F^{[11]}(\mathbf{x}, \mathbf{x})$. This generalized cutting plane $\widetilde{m}_k$, when added into $\mathcal{G}_{k+1}$, may have some beneficial secondary effect. Even though it is inactive at $\mathbf{y}^k$, it may become active elsewhere, just as the branch $F^{[i]}$ of F inactive at $\mathbf{x}$ may become active as we move away from $\mathbf{x}$. The inactive plane $\widetilde{m}_k$ has therefore an anticipative effect, and we sometimes call these planes anticipated cutting planes.

3.7 Exactness and recycling

In order to guarantee $\partial_1 F_k(\mathbf{x}, \mathbf{x}) \subset \partial_1 F(\mathbf{x}, \mathbf{x})$ we keep at least one plane of the form $m_0(\cdot, \mathbf{x}) = g_0^\top(\cdot - \mathbf{x})$ in the model at all times k. We call m_0 an exactness plane, because it assures $F_k(\mathbf{x}, \mathbf{x}) = 0$. Formally $(0, g_0) \in \mathcal{G}_k$ for all k. As it may happen that $\partial_1 F(\mathbf{x}, \mathbf{x})$ is not singleton, we are free to add other exactness planes $(0, g')$, $g' \in \partial_1 F(\mathbf{x}, \mathbf{x})$ into $\mathcal{G}_k$, for instance, one at each inner loop step k.

When a serious step $\mathbf{x} \to \mathbf{x}^+$ is made, the old working model is lost, and we will have to start $\mathcal{G}_1$ anew when the inner loop starts. This is in contrast with convex bundle methods, where all planes accumulated on the way may stay in $\mathcal{G}$ forever. The only reason to not keep them all is to avoid overflow. In contrast, in the nonconvex case we lose planes from previous serious steps for the following reason: the plane $m(\cdot, \mathbf{x}) = a + g^\top(\cdot - \mathbf{x})$ stored in $\mathcal{G}$ will in general be useless at $\mathbf{x}^+$, because we may have $m(\mathbf{x}^+, \mathbf{x}) \geq F(\mathbf{x}^+, \mathbf{x}^+) = 0$. We therefore propose to recycle the old plane

$m(\cdot, \mathbf{x})$ as

$$m(\cdot, \mathbf{x}^+) = m(\cdot, \mathbf{x}) - s^+,$$

with s^+ the downshift at $\mathbf{x}^+$. That is

$$s^+ = \big[m(\mathbf{x}^+, \mathbf{x})\big]_+ + c\|\mathbf{x}^+ - \mathbf{x}\|^2.$$

Formally, if $(a, g) \in \mathcal{G}_{k_j}$ at the end of the jth inner loop occurring at counter $k = k_j$, then let $a^+ = a - s^+$ and put $(a^+, g) \in \mathcal{G}_1$ at the beginning of the $(j+1)^{\text{st}}$ inner loop.

3.8 Management of the proximity parameter

At the core of Algorithm 1 is the management of τ during the inner loop. According to step 7, the τ-parameter is never decreased during the inner loop. It is increased when $\rho_k < \gamma$, $\widetilde{\rho}_k \geq \widetilde{\gamma}$, and held constant when $\rho_k < \gamma$, $\widetilde{\rho}_k < \widetilde{\gamma}$. The test

$$\widetilde{\rho}_k = \frac{F_{k+1}(\mathbf{y}^k, \mathbf{x})}{F_k(\mathbf{y}^k, \mathbf{x})} \overset{?}{\geq} \widetilde{\gamma},$$

where $\gamma < \widetilde{\gamma} < 1$ is fixed throughout, compares working models $F_{k+1}(\cdot, \mathbf{x})$ and $F_k(\cdot, \mathbf{x})$ at $\mathbf{y}^k$. If $\widetilde{\rho}_k < \widetilde{\gamma}$, then agreement between the two is bad, while $\widetilde{\rho}_k \geq \widetilde{\gamma}$ means it is not bad. The interpretation of step 7 is that $\rho_k < \gamma$ in tandem with $\widetilde{\rho}_k \geq \widetilde{\gamma}$ means F_k is far from F at $\mathbf{y}^k$, but at the same time F_k is reasonably close to F_{k+1} at $\mathbf{y}^k$. Now as F_{k+1} is supposed to make progress toward F, this constellation ($\rho_k < \gamma$, $\widetilde{\rho}_k \geq \widetilde{\gamma}$) tells us that the intended progress is too marginal. This is where we increase $\tau_{k+1} = 2\tau_k$ to force smaller steps at the next sweep $k+1$. The opposite situation $\rho_k < \gamma$ and $\widetilde{\rho}_k < \widetilde{\gamma}$ is considered as still open. Keeping $\tau_{k+1} = \tau_k$ fixed, we rely on improving F_{k+1} by adding cutting planes and the aggregate plane.

Observe that $F_{k+1}(\mathbf{y}^k, \mathbf{x}) \geq F_k(\mathbf{y}^k, \mathbf{x})$, because the aggregate plane, which contributes to F_{k+1}, knows the value of F_k at $\mathbf{y}^k$. Since $F_k(\mathbf{y}^k, \mathbf{x}) < 0$, the quotient $\widetilde{\rho}_k$ satisfies $\widetilde{\rho}_k \leq 1$.

3.9 Management of the proximity parameter between serious steps

As soon as a serious step $\mathbf{x} \to \mathbf{x}^+$ is made, we need to pass the τ-parameter on to the next inner loop. This is done via the memory element $\tau^\sharp$. We proceed as follows. If $\rho_k \geq \Gamma$, where $0 < \gamma < \Gamma < 1$, then we decrease the τ-parameter, as agreement between model and reality is *good*. If $\gamma \leq \rho_k < \Gamma$, then agreement is *not bad*, and we keep τ as is. This is organized in step 8. We re-set $\tau^\sharp = T$ if the preceding inner loop terminates with $\tau > T$. One can also dispense with this re-set, see [5] for details.

Algorithm 1. Proximity control algorithm for (8)

Parameters: $0 < \gamma < \widetilde{\gamma} < 1$, $0 < \gamma < \Gamma < 1$, $0 < q < \infty$, $0 < c < \infty$, $q < T \leq \infty$.

1: **Initialize outer loop**. Choose initial serious iterate $\mathbf{x}^1$ and initial matrix $Q_1 = Q_1^\top$ with $-qI \preceq Q_1 \preceq qI$. Initialize memory control parameter $\tau_1^\sharp$ such that $Q_1 + \tau_1^\sharp I \succ 0$. Put $j = 1$.

2: **Stopping test**. At outer loop counter j and serious iterate $\mathbf{x}^j$, stop if $0 \in \partial_1 F(\mathbf{x}^j, \mathbf{x}^j)$. Otherwise goto inner loop.

3: **Initialize inner loop**. Put inner loop counter $k = 1$ and initialize $\tau_1 = \tau_j^\sharp$. Build working model $F_1(\cdot, \mathbf{x}^j)$ by using initial set $\mathcal{G}_1$ and matrix Q_j.

4: **Trial step generation.** At inner loop counter k solve tangent program

$$\min_{\mathbf{y} \in \mathbb{R}^n} F_k(\mathbf{y}, \mathbf{x}^j) + \tfrac{\tau_k}{2} \|\mathbf{y} - \mathbf{x}^j\|^2.$$

The solution is the new trial step $\mathbf{y}^k$.

5: **Acceptance test.** Check whether

$$\rho_k = \frac{F(\mathbf{y}^k, \mathbf{x}^j)}{F_k(\mathbf{y}^k, \mathbf{x}^j)} \geq \gamma.$$

If this is the case put $\mathbf{x}^{j+1} = \mathbf{y}^k$ (serious step), quit inner loop and goto step 8. If this is not the case (null step) continue inner loop with step 6.

6: **Update working model**. Generate a cutting plane $m_k(\cdot, \mathbf{x}^j) = a_k + g_k^\top(\cdot - \mathbf{x}^j)$ at null step $\mathbf{y}^k$ and counter k using downshift (17). Compute aggregate plane $m_k^*(\cdot, \mathbf{x}^j) = a_k^* + g_k^{*\top}(\cdot - \mathbf{x}^j)$ at $\mathbf{y}^k$. Build $\mathcal{G}_{k+1} = \mathcal{G}_k \cup \{(a_k, g_k), (a_k^*, g_k^*)\}$. In order to keep the size of $\mathcal{G}_{k+1}$ reasonable allow removing some of the elements of $\mathcal{G}_k$. Build new working model $F_{k+1}(\cdot, \mathbf{x}^j)$.

7: **Update proximity control parameter.** Compute secondary control parameter

$$\widetilde{\rho}_k = \frac{F_{k+1}(\mathbf{y}^k, \mathbf{x}^j)}{F_k(\mathbf{y}^k, \mathbf{x}^j)}.$$

Then decide as follows. Put

$$\tau_{k+1} = \begin{cases} \tau_k, & \text{if } \widetilde{\rho}_k < \widetilde{\gamma} \qquad \text{(bad)} \\ 2\tau_k, & \text{if } \widetilde{\rho}_k \geq \widetilde{\gamma} \qquad \text{(too bad)} \end{cases}$$

Then increase inner loop counter k and continue inner loop with step 4.

8: **Update Q_j and memory element**. Update matrix $Q_j \to Q_{j+1}$ respecting $Q_{j+1} = Q_{j+1}^\top$ and $-qI \preceq Q_{j+1} \preceq qI$. Then store new memory element

$$\tau_{j+1}^\sharp = \begin{cases} \tau_{k+1}, & \text{if } \gamma \leq \rho_k < \Gamma \qquad \text{(not bad)} \\ \frac{1}{2}\tau_{k+1}, & \text{if } \rho_k \geq \Gamma \qquad \text{(good)} \end{cases}$$

Increase $\tau_{j+1}^\sharp$ if necessary to ensure $Q_{j+1} + \tau_{j+1}^\sharp I \succ 0$. If $\tau_{j+1}^\sharp > T$ then re-set $\tau_{j+1}^\sharp = T$. Increase outer loop counter j by 1 and loop back to step 2.

4 Convergence analysis

In this section, we state and prove a convergence result for Algorithm 1. We shall require the notion of lower C^1-functions introduced by Spingarn [12]. More generally, following [13], a locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$ is called lower C^k at $\mathbf{x}_0$ if there exists a compact space K and a continuous function $F : B(\mathbf{x}_0, \delta) \times K \to \mathbb{R}$ for which all partial derivatives of order $\leq k$ with respect to $\mathbf{x}$ are also continuous, such that

$$f(\mathbf{x}) = \max_{\mathbf{y} \in K} F(\mathbf{x}, \mathbf{y}) \tag{19}$$

for every $\mathbf{x} \in B(\mathbf{x}_0, \delta)$. The function f is called lower C^k if it is lower C^k at every $\mathbf{x} \in \mathbb{R}^n$. According to [13] lower C^2-functions are lower C^k for every $k \geq 2$. On

the other hand, the class of lower C^1-functions is strictly larger than lower C^2, and sufficiently large to include all practical situations.

Theorem 1 *Suppose the program data f and c in (8) are locally Lipschitz lower C^1-functions. In addition, let the following conditions be satisfied:*

(a) f is weakly coercive on the constraint set $\Omega = \{\mathbf{x} \in \mathbb{R}^n : c(\mathbf{x}) \leq 0\}$, i.e., if $\mathbf{x}^j$ is a sequence of feasible iterates with $\|\mathbf{x}^j\| \to \infty$, then $f(\mathbf{x}^j)$ is not monotonically decreasing.

(b) c is weakly coercive, i.e., if $\|\mathbf{x}^j\| \to \infty$, then $c(\mathbf{x}^j)$ is not monotonically decreasing.

Then the sequence of serious steps $\mathbf{x}^j$ generated by Algorithm 1 is bounded. It either ends finitely with $0 \in \partial_1 F(\mathbf{x}^j, \mathbf{x}^j)$, or it is infinite, in which case every accumulation point $\mathbf{x}^*$ of $\mathbf{x}^j$ satisfies $0 \in \partial_1 F(\mathbf{x}^*, \mathbf{x}^*)$. In particular, $\mathbf{x}^*$ is either a critical point of constraint violation, or a KKT-point of (8).

Here, motivated by Lemma 1, we shall call $\mathbf{x}^*$ a critical point of constraint violation, if $0 \in \partial c(\mathbf{x}^*)$ in tandem with $c(\mathbf{x}^*) \geq 0$. The proof is divided into several Lemmas. The first step is to prove that the inner loop ends finitely. We write $\mathbf{x}$ for the current serious iterate $\mathbf{x}^j$, and Q for the matrix $Q(\mathbf{x}^j)$.

Lemma 2 *Suppose the inner loop at serious iterate $\mathbf{x}$ turns infinitely, i.e., $\rho_k < \gamma$ for all $k \in \mathbb{N}$. Then there exists $k_0 \in \mathbb{N}$ such that $\tau_k = \tau_{k_0}$ for all $k \geq k_0$.*

Proof i) Suppose on the contrary that the control parameter is increased infinitely often. Then, as it is never decreased in the inner loop, we must have $\tau_k \to \infty$. We will show that this implies $0 \in \partial_1 F(\mathbf{x}, \mathbf{x})$, contradicting step 2 of the algorithm. Indeed, the inner loop is only entered when $0 \notin \partial_1 F(\mathbf{x}, \mathbf{x})$. Notice that when $\tau_k \to \tau_{k+1}$ is increased, we have $\widetilde{\rho}_k \geq \widetilde{\gamma}$, so we have an infinity of counters $k \in \mathcal{K}$ where this happens.
ii) Recall that by (13) the aggregate subgradient satisfies $g_k^* = (Q + \tau_k I)(\mathbf{x} - \mathbf{y}^k) \in \partial_1 F_k^{[1]}(\mathbf{y}^k, \mathbf{x})$. By the subgradient inequality we have

$$(\mathbf{x} - \mathbf{y}^k)^\top (Q + \tau_k I)(\mathbf{x} - \mathbf{y}^k) \leq F_k^{[1]}(\mathbf{x}, \mathbf{x}) - F_k^{[1]}(\mathbf{y}^k, \mathbf{x}) = -F_k^{[1]}(\mathbf{y}^k, \mathbf{x}). \tag{20}$$

Recall that $m_0(\cdot, \mathbf{x}) \leq F_k^{[1]}(\cdot, \mathbf{x})$, where $m_0(\cdot, \mathbf{x}) = g_0^\top(\cdot - \mathbf{x})$ is the exactness plane at $\mathbf{x}$. Substituting this in (20) implies

$$(\mathbf{x} - \mathbf{y}^k)^\top (Q + \tau_k I)(\mathbf{x} - \mathbf{y}^k) \leq g_0^\top (\mathbf{x} - \mathbf{y}^k) \leq \|g_0\| \|\mathbf{x} - \mathbf{y}^k\|.$$

Since $\tau_k \to \infty$, the left hand side behaves asymptotically like $\tau_k \|\mathbf{x} - \mathbf{y}^k\|^2$. In other words, fixing $0 < \zeta < 1$, we may assume that it is minorized by $(1-\zeta)\tau_k \|\mathbf{x} - \mathbf{y}^k\|^2$ for k large enough. After dividing a factor $\|\mathbf{x} - \mathbf{y}^k\|$ we obtain $(1 - \zeta)\tau_k \|\mathbf{x} - \mathbf{y}^k\| \leq \|g_0\|$, which implies boundedness of $\tau_k(\mathbf{x} - \mathbf{y}^k)$, and therefore also boundedness of the sequence g_k^*. As $\tau_k \to \infty$, we deduce $\mathbf{y}^k \to \mathbf{x}$ and $(\mathbf{x} - \mathbf{y}^k)^\top (Q + \tau_k I)(\mathbf{x} - \mathbf{y}^k) \to 0$.
iii) Subtracting $\frac{1}{2}(\mathbf{x} - \mathbf{y}^k)^\top Q(\mathbf{x} - \mathbf{y}^k)$ on both sides of (20) gives

$$\frac{1}{2}(\mathbf{x} - \mathbf{y}^k)^\top Q(\mathbf{x} - \mathbf{y}^k) + \tau_k \|\mathbf{x} - \mathbf{y}^k\|^2 \leq -F_k(\mathbf{y}^k, \mathbf{x}).$$

Fix $0 < \zeta < 1$. As $\tau_k \to \infty$, we have for $k \in \mathcal{K}$ sufficiently large

$$(1-\zeta)\tau_k\|\mathbf{x}-\mathbf{y}^k\| \le \|g_k^*\| \le (1+\zeta)\tau_k\|\mathbf{x}-\mathbf{y}^k\|.$$

Indeed, by the definition (13) of the aggregate subgradient g_k^* we have $\|g_k^*\|/(\tau_k\|\mathbf{x}-\mathbf{y}^k\|) = \|(\tau_k^{-1}Q+I)\|\mathbf{x}-\mathbf{y}^k\|/\|\mathbf{x}-\mathbf{y}^k\| \to 1$, in view of $\tau_k^{-1} \to 0$, hence $1-\zeta < \|g_k^*\|/(\tau_k\|\mathbf{x}-\mathbf{y}^k\|) < 1+\zeta$ for k large enough. A similar argument shows

$$\tfrac{1}{2}(\mathbf{x}-\mathbf{y}^k)^\top Q(\mathbf{x}-\mathbf{y}^k) + \tau_k\|\mathbf{x}-\mathbf{y}^k\|^2 \ge (1-\zeta)\tau_k\|\mathbf{x}-\mathbf{y}^k\|^2$$

for $k \in \mathcal{K}$ large enough. Combining these estimates gives

$$-F_k(\mathbf{y}^k,\mathbf{x}) \ge \tfrac{1-\zeta}{1+\zeta}\|g_k^*\|\|\mathbf{x}-\mathbf{y}^k\|. \tag{21}$$

iv) Now we argue that $F_k(\mathbf{y}^k,\mathbf{x}) \to F(\mathbf{x},\mathbf{x}) = 0$. Going back to the subgradient inequality (20), we see that the left hand side tends to 0 by iii). Hence $0 \le \liminf(-F_k^{[1]}(\mathbf{y}^k,\mathbf{x}))$, or equivalently, $\limsup F_k^{[1]}(\mathbf{y}^k,\mathbf{x}) \le 0$. It therefore remains to prove $\liminf F_k^{[1]}(\mathbf{y}^k,\mathbf{x}) \ge 0$. To prove this, observe that $F_k^{[1]}(\mathbf{y}^k,\mathbf{x}) \ge m_0(\mathbf{y}^k,\mathbf{x})$ for the exactness plane $m_0(\cdot,\mathbf{x})$ at $\mathbf{x}$. Since $m_0(\mathbf{y}^k,\mathbf{x}) \to m_0(\mathbf{x},\mathbf{x}) = 0$ due to iii), the claim follows.

v) Now let $\eta_k := \operatorname{dist}(g_k^*, \partial_1 F(\mathbf{x},\mathbf{x}))$. We prove $\eta_k \to 0$. Using the subgradient inequality we have for a fixed vector $\mathbf{y}$

$$g_k^{*\top}(\mathbf{y}-\mathbf{y}^k) + F_k^{[1]}(\mathbf{y}^k,\mathbf{x}) \le F_k^{[1]}(\mathbf{y},\mathbf{x}) = m_{\mathbf{z}_k(\mathbf{y})}(\mathbf{y},\mathbf{x}),$$

where $m_{\mathbf{z}_k(\mathbf{y})}(\cdot,\mathbf{x})$ is a cutting plane at $\mathbf{z}_k(\mathbf{y}) \in \{\mathbf{y}^1,\dots,\mathbf{y}^k\}$ with respect to serious iterate $\mathbf{x}$, contributing to the build-up of model $F_k^{[1]}(\cdot,\mathbf{x})$, and exact at $\mathbf{y}$. In other words

$$m_{\mathbf{z}_k(\mathbf{y})}(\cdot,\mathbf{x}) = F^{[1]}(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) + g_{\mathbf{z}_k(\mathbf{y})}^\top(\cdot-\mathbf{z}_k(\mathbf{y})) - s$$

where $g_{\mathbf{z}_k(\mathbf{y})} \in \partial_1 F^{[1]}(\mathbf{z}_k(\mathbf{y}),\mathbf{x})$ and where $s = s(\mathbf{z}_k(\mathbf{y}),\mathbf{x})$ is the downshift at $\mathbf{z}_k(\mathbf{y})$ with respect to $\mathbf{x}$. That is $s(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) = t_{\mathbf{z}_k(\mathbf{y})}(\mathbf{x})_+ + c\|\mathbf{z}_k(\mathbf{y})-\mathbf{x}\|^2$. Here $t_{\mathbf{z}_k(\mathbf{y})}(\mathbf{x}) = F^{[1]}(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) + g_{\mathbf{z}_k(\mathbf{y})}^\top(\mathbf{x}-\mathbf{z}_k(\mathbf{y}))$. Substituting this gives

$$\begin{aligned} g_k^{*\top}(\mathbf{y}-\mathbf{y}^k) + F_k^{[1]}(\mathbf{y}^k,\mathbf{x}) &\le F^{[1]}(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) + g_{\mathbf{z}_k(\mathbf{y})}^\top(\mathbf{y}-\mathbf{z}_k(\mathbf{y})) - s(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) \\ &= F^{[1]}(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) + g_{\mathbf{z}_k(\mathbf{y})}^\top(\mathbf{y}-\mathbf{z}_k(\mathbf{y})) \\ &\quad - \Big[F^{[1]}(\mathbf{z}_k(\mathbf{y}),\mathbf{x}) - g_{\mathbf{z}_k(\mathbf{y})}^\top(\mathbf{x}-\mathbf{z}_k(\mathbf{y}))\Big]_+ - c\|\mathbf{z}_k(\mathbf{y})-\mathbf{x}\|^2. \end{aligned} \tag{22}$$

There are two cases to discuss, $[\dots]_+ > 0$ and $[\dots]_+ = 0$. Consider $[\dots]_+ > 0$ first. Then

$$g_k^{*\top}(\mathbf{y}-\mathbf{y}^k) + F_k^{[1]}(\mathbf{y}^k,\mathbf{x}) \le g_{\mathbf{z}_k(\mathbf{y})}^{\top}(\mathbf{y}-\mathbf{x}) - c\|\mathbf{z}_k(\mathbf{y})-\mathbf{x}\|^2.$$

Due to boundedness of the g_k^* and of the set of trial steps we may pass to a subsequence $\mathcal{K}'$ of $\mathcal{K}$ where $g_k^* \to g^*$ and $\mathbf{z}_k(\mathbf{y}) \to \mathbf{z}(\mathbf{y})$ for some $\mathbf{z}(\mathbf{y})$. From part iv) we know $F_k(\mathbf{y}^k,\mathbf{x}) \to F(\mathbf{x},\mathbf{x}) = 0$. Hence, passing to the limit in the above estimate gives

$$g^{*\top}(\mathbf{y}-\mathbf{x}) \le g_{\mathbf{z}(\mathbf{y})}^{\top}(\mathbf{y}-\mathbf{x}) - c\|\mathbf{z}(\mathbf{y})-\mathbf{x}\|^2 \le g_{\mathbf{z}(\mathbf{y})}^{\top}(\mathbf{y}-\mathbf{x}). \tag{23}$$

One can see from this relation that $\mathbf{y} \to \mathbf{x}$ implies $\mathbf{z}(\mathbf{y}) \to \mathbf{x}$, because the $g_{\mathbf{z}(\mathbf{y})}$ are bounded. Using this information in (23), and writing $\mathbf{e}(\mathbf{y}) = (\mathbf{y}-\mathbf{x})/\|\mathbf{y}-\mathbf{x}\|$, we have

$$g^{*\top}\mathbf{e}(\mathbf{y}) \le g_{\mathbf{z}(\mathbf{y})}^{\top}\mathbf{e}(\mathbf{y}).$$

Fixing an arbitrary unit vector $\mathbf{e}$, we arrange convergence $\mathbf{y} \to \mathbf{x}$ in such a way that $\mathbf{e}(\mathbf{y}) = (\mathbf{y}-\mathbf{x})/\|\mathbf{y}-\mathbf{x}\| \to \mathbf{e}$. Passing to a subsequence, we may in addition have $g_{\mathbf{z}(\mathbf{y})} \to g_{\mathbf{x}}$ for some $g_{\mathbf{x}}$, where $g_{\mathbf{x}} \in \partial_1 F^{[1]}(\mathbf{x},\mathbf{x})$ by upper semicontinuity of the Clarke subdifferential. That shows $g^{*\top}\mathbf{e} \le \max\{g^{\top}\mathbf{e} : g \in \partial_1 F^{[1]}(\mathbf{x},\mathbf{x})\}$, and by the Hahn–Banach theorem we deduce $g^* \in \partial_1 F^{[1]}(\mathbf{x},\mathbf{x})$. That shows $\eta_k \le \|g_k^* - g^*\| \to 0$ in the case $[\dots]_+ > 0$.

It remains to discuss the case where $[\dots]_+ = 0$. Going back to (22), we may again pass to the limit $k \in \mathcal{K}'$ such that $g_k^* \to g^*$ and $\mathbf{z}_k(\mathbf{y}) \to \mathbf{z}(\mathbf{y})$ to obtain

$$\begin{aligned} g^{*\top}(\mathbf{y}-\mathbf{x}) &\le F^{[1]}(\mathbf{z}(\mathbf{y}),\mathbf{x}) + g_{\mathbf{z}(\mathbf{y})}^{\top}(\mathbf{y}-\mathbf{z}(\mathbf{y})) - c\|\mathbf{z}(\mathbf{y})-\mathbf{x}\|^2 \\ &= F^{[1]}(\mathbf{z}(\mathbf{y}),\mathbf{x}) + g_{\mathbf{z}(\mathbf{y})}^{\top}(\mathbf{x}-\mathbf{z}(\mathbf{y})) + g_{\mathbf{z}(\mathbf{y})}^{\top}(\mathbf{y}-\mathbf{x}) - c\|\mathbf{z}(\mathbf{y})-\mathbf{x}\|^2 \\ &\le g_{\mathbf{z}(\mathbf{y})}^{\top}(\mathbf{y}-\mathbf{x}) - c\|\mathbf{z}(\mathbf{y})-\mathbf{x}\|^2 \qquad (\text{using } t_{\mathbf{z}(\mathbf{y})}(\mathbf{x}) \le 0). \end{aligned}$$

This shows again that $\mathbf{z}(\mathbf{y}) \to \mathbf{x}$ when $\mathbf{y} \to \mathbf{x}$. Now the proof proceeds as above, and we deduce $g^* \in \partial F^{[1]}(\mathbf{x},\mathbf{x})$ in the case $[\dots]_+ = 0$, too. That ends the proof of $\eta_k \to 0$.

vi) Let $\eta := \operatorname{dist}(0, \partial_1 F^{[1]}(\mathbf{x},\mathbf{x}))$. We have to prove $\eta = 0$. Assume on the contrary that $\eta > 0$. Using the definition of η_k choose $\tilde{g}_k \in \partial_1 F^{[1]}(\mathbf{x},\mathbf{x})$ such that $\|g_k^* - \tilde{g}_k\| = \eta_k$. Then $\|\tilde{g}_k\| \ge \eta$, hence $\|g_k^*\| \ge \eta - \eta_k > (1-\zeta)\eta$ for k large enough, given that $\eta_k \to 0$ by v) and $\eta > 0$. (Here $\zeta \in (0,1)$ is the parameter chosen in part iii)). Going back with this to (21) gives

$$-F_k(\mathbf{y}^k,\mathbf{x}) \ge \tfrac{(1-\zeta)^2}{1+\zeta}\eta\|\mathbf{x}-\mathbf{y}^k\|. \tag{24}$$

vi) Choose $\epsilon > 0$ such that

$$\epsilon < \frac{\eta(\widetilde{\gamma}-\gamma)(1-\zeta)^2}{(1+\zeta)^2}. \tag{25}$$

We claim that there exists $k(\epsilon)$ such that $F(\mathbf{y}^k, \mathbf{x}) \leq F_{k+1}(\mathbf{y}^k, \mathbf{x}) + (1+\zeta)\epsilon\|\mathbf{x} - \mathbf{y}^k\|$ for all $k \in \mathcal{K}$, $k \geq k(\epsilon)$.

Indeed, let $m_k(\cdot, \mathbf{x})$ be the cutting plane at $\mathbf{y}^k$, $M_k(\cdot, \mathbf{x}) = m_k(\cdot, \mathbf{x}) + \frac{1}{2}(\cdot - \mathbf{x})^\top Q(\cdot - \mathbf{x})$. Then $F_{k+1}(\mathbf{y}^k, \mathbf{x}) = M_k(\mathbf{y}^k, \mathbf{x})$ by construction. Moreover, $m_k(\cdot, \mathbf{x}) = t_k(\cdot) - s_k$, where $t_k(\cdot)$ is the tangent to $F^{[1]}(\cdot, \mathbf{x})$ at $\mathbf{y}^k$, and s_k is the corresponding downshift (16). That means

$$\begin{aligned} m_k(\cdot, \mathbf{x}) &= F^{[1]}(\mathbf{y}^k, \mathbf{x}) + g_k^\top(\cdot - \mathbf{y}^k) - s_k \\ &= F^{[1]}(\mathbf{y}^k, \mathbf{x}) + g_k^\top(\cdot - \mathbf{y}^k) - c\|\mathbf{x} - \mathbf{y}^k\|^2 - [t_k(\mathbf{x})]_+ . \end{aligned}$$

There are two cases to discuss, $[\dots]_+ > 0$ and $[\dots]_+ = 0$. Assuming first $t_k(\mathbf{x}) > 0$, we have

$$F^{[1]}(\mathbf{y}^k, \mathbf{x}) - m_k(\mathbf{y}^k, \mathbf{x}) = F^{[1]}(\mathbf{y}^k, \mathbf{x}) - F^{[1]}(\mathbf{x}, \mathbf{x}) - g_k^\top(\mathbf{y}^k - \mathbf{x}) + c\|\mathbf{x} - \mathbf{y}^k\|^2.$$

According to [14, Thm. 2], a lower C^1-function is approximately convex in the following sense. For a sequence $\mathbf{y}^k \to \mathbf{x}$ there exists $k(\epsilon)$ such that $g_k^\top(\mathbf{x} - \mathbf{y}^k) \leq F^{[1]}(\mathbf{x}, \mathbf{x}) - F^{[1]}(\mathbf{y}^k, \mathbf{x}) + \epsilon\|\mathbf{x} - \mathbf{y}^k\|$ for all $k \geq k(\epsilon)$. Substituting this gives

$$F^{[1]}(\mathbf{y}^k, \mathbf{x}) - m_k(\mathbf{y}^k, \mathbf{x}) \leq \epsilon\|\mathbf{x} - \mathbf{y}^k\| + c\|\mathbf{x} - \mathbf{y}^k\|^2.$$

Re-arranging $F^{[1]}(\mathbf{y}^k, \mathbf{x}) - m_k(\mathbf{y}^k, \mathbf{x}) = \big(F^{[1]}(\mathbf{y}^k, \mathbf{x}) + (\mathbf{y}^k - \mathbf{x})^\top Q(\mathbf{y}^k - \mathbf{x})\big) - \big(m_k(\mathbf{y}^k, \mathbf{x}) + (\mathbf{y}^k - \mathbf{x})^\top Q(\mathbf{y}^k - \mathbf{x})\big) = F(\mathbf{y}^k, \mathbf{x}) - M_k(\mathbf{y}^k, \mathbf{x})$, we have

$$F(\mathbf{y}^k, \mathbf{x}) - M_k(\mathbf{y}^k, \mathbf{x}) \leq \epsilon\|\mathbf{x} - \mathbf{y}^k\| + c\|\mathbf{x} - \mathbf{y}^k\|^2 \leq (1+\zeta)\epsilon\|\mathbf{x} - \mathbf{y}^k\| \tag{26}$$

for k large enough. This ends the case $[\dots]_+ > 0$. Notice that in the second case $t_k(\mathbf{x}) \leq 0$ we get an even better estimate $F(\mathbf{y}^k, \mathbf{x}) - M_k(\mathbf{y}^k, \mathbf{x}) = c\|\mathbf{x} - \mathbf{y}^k\|^2 \leq \epsilon\|\mathbf{x} - \mathbf{y}^k\|$ for large k, so (26) holds in both cases. vii) Using (24) and (26), we now expand the parameter $\widetilde{\rho}_k$ as

$$\begin{aligned} \widetilde{\rho}_k &= \rho_k + \frac{F(\mathbf{y}^k, \mathbf{x}) - M_k(\mathbf{y}^k, \mathbf{x})}{F(\mathbf{x}, \mathbf{x}) - F_k(\mathbf{y}^k, \mathbf{x})} \\ &\leq \rho_k + \frac{(1+\zeta)^2\epsilon\|\mathbf{x} - \mathbf{y}^k\|}{(1-\zeta)^2\eta\|\mathbf{x} - \mathbf{y}^k\|} = \rho_k + \frac{(1+\zeta)^2\epsilon}{(1-\zeta)^2\eta} \\ &< \rho_k + \widetilde{\gamma} - \gamma < \widetilde{\gamma} \end{aligned}$$

using the choice (25) of ϵ and $\rho_k < \gamma$. But this contradicts $\widetilde{\rho}_k \geq \widetilde{\gamma}$ for the infinitely many $k \in \mathcal{K}$. Hence $\eta > 0$ was an incorrect hypothesis, and we have shown $\eta = 0$. This ends the proof. □

Lemma 3 *Under the hypotheses of the theorem, the inner loop at serious iterate* $\mathbf{x}$ *ends finitely.*

Proof From the previous Lemma 2, we deduce that if the inner loop turns infinitely, then $\rho_k < \gamma$ and $\tau_k = \tau$ for $k \geq k_0$. By step 7 of the algorithm this implies $\widetilde{\rho}_k < \widetilde{\gamma}$ for all $k \geq k_0$, so that we are in the situation analyzed in [4, Lemma 6.3], and the conclusion is that we must have $0 \in \partial_1 F(\mathbf{x}, \mathbf{x})$. As this contradicts step 2 of the algorithm, the inner loop must be finite. □

Proof of Theorem 1 i) We first prove $F(\mathbf{x}^{j+1}, \mathbf{x}^j) \to 0$ $(j \to \infty)$, along with boundedness of the sequence $\mathbf{x}^j$. Notice that by construction, $F(\mathbf{x}^{j+1}, \mathbf{x}^j) \leq 0$ for every j. There are two cases to discuss.

Case I $c(\mathbf{x}^j) > 0$ for every $j \in \mathbb{N}$. Here the sequence of serious iterates never becomes feasible, and the algorithm remains in phase I. Here we expect to converge to a critical point of constraint violation. Notice that in case I, we have

$$F(\mathbf{x}^{j+1}, \mathbf{x}^j) = \max\{f(\mathbf{x}^{j+1}) - f(\mathbf{x}^j) - \mu c(\mathbf{x}^j), c(\mathbf{x}^{j+1}) - c(\mathbf{x}^j)\} \leq 0,$$

which shows $c(\mathbf{x}^j)$ is monotonically decreasing. Therefore $c(\mathbf{x}^j) \to c(\mathbf{x}^*)$ for every accumulation point $\mathbf{x}^*$ of the $\mathbf{x}^j$, and from $c(\mathbf{x}^{j+1}) - c(\mathbf{x}^j) \leq F(\mathbf{x}^{j+1}, \mathbf{x}^j) \leq 0$ we obtain $F(\mathbf{x}^{j+1}, \mathbf{x}^j) \to 0$. We use hypothesis (b) to deduce that the sequence $\mathbf{x}^j$ is bounded.

Case II There exists $j_0 \in \mathbb{N}$ such that $c(\mathbf{x}^{j_0}) \leq 0$. Then from that index j_0 onward we have

$$F(\mathbf{x}^{j+1}, \mathbf{x}^j) = \max\{f(\mathbf{x}^{j+1}) - f(\mathbf{x}^j), c(\mathbf{x}^{j+1})\} \leq 0,$$

hence $f(\mathbf{x}^{j+1}) \leq f(\mathbf{x}^j)$ and $c(\mathbf{x}^{j+1}) \leq 0$. The iterates therefore stay feasible for $j \geq j_0$, and the objective f is optimized, so that we are in phase II. In particular, the sequence $f(\mathbf{x}^j)$, $j \geq j_0$, is monotonically decreasing. Therefore, for every accumulation point $\mathbf{x}^*$ of the $\mathbf{x}^j$, we have $f(\mathbf{x}^j) \to f(\mathbf{x}^*)$. Then $\liminf_{j\to\infty} F(\mathbf{x}^{j+1}, \mathbf{x}^j) \geq \lim_{j\to\infty} f(\mathbf{x}^{j+1}) - f(\mathbf{x}^j) = 0$ in tandem with $F(\mathbf{x}^{j+1}, \mathbf{x}^j) \leq 0$ proves $F(\mathbf{x}^{j+1}, \mathbf{x}^j) \to 0$. Here we use hypothesis (a) to deduce that the sequence $\mathbf{x}^j$ is bounded.

ii) Suppose in the jth inner loop the serious step is accepted at inner loop counter k_j, that is, $\mathbf{x}^{j+1} = \mathbf{y}^{k_j}$. We show that $\tau_{k_j}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2 \to 0$ and also $\|\mathbf{x}^j - \mathbf{x}^{j+1}\|_{Q_j+\tau_{k_j}I} \to 0$. To see this, observe that by the optimality condition (13) we have $g_j^* = (Q_j + \tau_{k_j} I)(\mathbf{x}^j - \mathbf{x}^{j+1}) \in \partial_1 F_{k_j}^{[1]}(\mathbf{x}^{j+1}, \mathbf{x}^j)$, hence by the subgradient inequality

$$\begin{aligned}(\mathbf{x}^j - \mathbf{x}^{j+1})^\top (Q_j + \tau_{k_j} I)(\mathbf{x}^j - \mathbf{x}^{j+1}) &\leq F_{k_j}^{[1]}(\mathbf{x}^j, \mathbf{x}^j) - F_{k_j}^{[1]}(\mathbf{x}^{j+1}, \mathbf{x}^j) \\ &= -F_{k_j}^{[1]}(\mathbf{x}^{j+1}, \mathbf{x}^j).\end{aligned}$$

Subtracting $F^{[2]}(\mathbf{x}^{j+1}, \mathbf{x}^j) = \frac{1}{2}(\mathbf{x}^j - \mathbf{x}^{j+1})^\top Q_j(\mathbf{x}^j - \mathbf{x}^{j+1})$ on both sides gives

$$\tfrac{1}{2}(\mathbf{x}^j - \mathbf{x}^{j+1})^\top Q_j(\mathbf{x}^j - \mathbf{x}^{j+1}) + \tau_{k_j}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2 \le -F_{k_j}(\mathbf{x}^{j+1}, \mathbf{x}^j).$$

Now by the acceptance test, $-F_{k_j}(\mathbf{x}^{j+1}, \mathbf{x}^j) \le -\gamma^{-1}F(\mathbf{x}^{j+1}, \mathbf{x}^j)$, we have

$$\tfrac{1}{2}(\mathbf{x}^j - \mathbf{x}^{j+1})^\top Q_j(\mathbf{x}^j - \mathbf{x}^{j+1}) + \tau_{k_j}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2 \le -\gamma^{-1}F(\mathbf{x}^{j+1}, \mathbf{x}^j).$$

Next we use the fact that $Q_j + \tau_{k_j} I \succ 0$, which allows us to regroup the portion $\frac{1}{2}(\mathbf{x}^j - \mathbf{x}^{j+1})^\top Q_j(\mathbf{x}^j - \mathbf{x}^{j+1}) + \frac{1}{2}\tau_{k_j}\|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2$ on the left into the norm $\frac{1}{2}\|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2_{Q_j+\tau_{k_j}I}$, so that altogether the left hand side is the sum of two squared norms:

$$\tfrac{1}{2}\|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2_{Q_j+\tau_{k_j}I} + \tfrac{1}{2}\tau_{k_j}\|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2 \le -\gamma^{-1}F(\mathbf{x}^{j+1}, \mathbf{x}^j).$$

But the term on the right converges to 0 by part i), and this proves simultaneously $\tau_{k_j}\|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2 \to 0$ and $\|\mathbf{x}^{j+1} - \mathbf{x}^j\|^2_{Q_j+\tau_{k_j}I} \to 0$, as claimed.

iii) Let $\mathbf{x}^*$ be an accumulation point of the sequence $\mathbf{x}^j$ of serious iterates. We have to prove $0 \in \partial_1 F(\mathbf{x}^*, \mathbf{x}^*)$. Select an infinite subsequence $J \subset \mathbb{N}$ such that $\mathbf{x}^j \to \mathbf{x}^*$, $j \in J$. Recall that $g_j^* = (Q_j + \tau_{k_j} I)(\mathbf{x}^j - \mathbf{x}^{j+1})$ is the aggregate subgradient belonging to $\mathbf{x}^{j+1}$ in the jth inner loop. We distinguish two cases. *Case 1:* There exists $\theta > 0$ such that $\|g_j^*\| \ge \theta > 0$ for all $j \in J$. *Case 2:* There exists an infinite $J' \subset J$ such that $g_{j'}^* \to 0$, $j' \in J'$. Case 1 will be discussed in paragraphs iv)–vii). Case 2 is considered in part viii).

iv) We discuss the first case $\|g_j^*\| \ge \theta > 0$ for all $j \in J$. We first show that this working hypothesis implies $\tau_{k_j} \to \infty$ $(j \in J)$. Indeed, suppose there exists an infinite subset $J' \subset J$ such that the τ_{k_j}, $j \in J'$, are bounded. Then, using boundedness of the Q_j and of the set of serious steps proved in i), we could extract a subsequence $J'' \subset J'$ such that $Q_j \to \bar{Q}$, $\mathbf{x}^j - \mathbf{x}^{j+1} \to \delta\mathbf{x}$, $\tau_{k_j} \to \bar{\tau}$ and therefore $g_j^* \to (\bar{Q} + \bar{\tau} I)\delta\mathbf{x}$, where consequently $\|(\bar{Q} + \bar{\tau} I)\delta\mathbf{x}\| \ge \theta > 0$. But also $(\mathbf{x}^j - \mathbf{x}^{j+1})^\top(Q_j + \tau_{k_j} I)(\mathbf{x}^j - \mathbf{x}^{j+1}) \to \delta\mathbf{x}^\top(\bar{Q} + \bar{\tau} I)\delta\mathbf{x} = 0$ as a consequence of ii). Since $\bar{Q} + \bar{\tau} I$ is symmetric and $\succeq 0$, this contradicts $\|(\bar{Q} + \bar{\tau} I)\delta\mathbf{x}\| > 0$. Hence the τ_{k_j}, $j \in J'$ could not be bounded. This shows $\tau_{k_j} \to \infty$, $j \in J$.

So far we know that $\mathbf{x}^j \to \mathbf{x}^*$ and $\tau_{k_j} \to \infty$ $(j \in J)$. Now let J^+ be the set of those indices $j \in J$ where the τ-parameter was increased at least once during the jth inner loop, J^- the other indices in J, where τ remained unchanged. In other words, in view of step 3 of the algorithm,

$$J^+ = \{j \in J : \tau_{k_j} > \tau_j^\sharp\}, \qquad J^- = \{j \in J : \tau_{k_j} = \tau_j^\sharp\}.$$

Then J^- must be finite. Indeed, $\tau_{k_j} \to \infty$, $(j \in J)$, but $\tau_j^\sharp \le T < \infty$ according to step 8 of the algorithm.

v) Working on the set J^+, let us assume that the τ-parameter was increased for the last time at stage $k_j - \nu_j$, where $\nu_j \geq 1$. That is

$$\tau_{k_j} = \tau_{k_j-1} = \cdots = \tau_{k_j-\nu_j+1} = 2\tau_{k_j-\nu_j}.$$

According to step 7 of the algorithm, we have

$$\rho_{k_j-\nu_j} < \gamma, \qquad \widetilde{\rho}_{k_j-\nu_j} \geq \widetilde{\gamma}.$$

Since $\tau_{k_j-\nu_j} \to \infty$, $(j \in J^+)$, boundedness of the subgradients $\widetilde{g}_j = (Q_j + \frac{1}{2}\tau_{k_j} I)(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})$ shows $\mathbf{y}^{k_j-\nu_j} - \mathbf{x}^j \to 0$. Here boundedness of the $\widetilde{g}_j$ can be seen as follows. By the subgradient inequality,

$$\begin{aligned}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})^\top (Q_j + \tfrac{1}{2}\tau_{k_j} I)(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) &\leq F^{[1]}_{k_j-\nu_j}(\mathbf{x}^j, \mathbf{x}^j) - F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) \\ &= -F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j).\end{aligned}$$

Now the exactness plane at $\mathbf{x}^j$ has the form $m_0(\cdot, \mathbf{x}^j) = g_{0j}^\top(\cdot - \mathbf{x}^j)$ for some $g_{0j} \in \partial_1 F^{[1]}(\mathbf{x}^j, \mathbf{x}^j)$, and we have $m_0(\cdot, \mathbf{x}^j) \leq F^{[1]}_{k_j-\nu_j}(\cdot, \mathbf{x}^j)$ by construction of the working model. Using this we have

$$\begin{aligned}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})^\top (Q_j + \tfrac{1}{2}\tau_{k_j} I)(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) &\leq g_{0j}^\top(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) \\ &\leq \|g_{0j}\|\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|.\end{aligned}$$

As $\tau_{k_j} \to \infty$ and the Q_j are bounded, the left hand side behaves asymptotically like $\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2$. So after dividing one factor, we have $\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\| \leq C\|g_{0j}\|$ for some constant $C > 0$. Since the $\mathbf{x}^j$ are bounded, so are the g_{0j}, and we deduce boundedness of $\tau_{k_j}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})$. This shows boundedness of the $\widetilde{g}_j$ and also $\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j} \to 0$ because of $\tau_{k_j} \to \infty$.

vi) As $\mathbf{x}^j \to \mathbf{x}^*$, part v) implies $\mathbf{y}^{k_j-\nu_j} \to \mathbf{x}^*$, $j \in J^+$. Passing to a subsequence, we may assume $\widetilde{g}_j \to \widetilde{g}$ for some $\widetilde{g}$. We show $\widetilde{g} \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$. From the subgradient inequality,

$$\widetilde{g}_j^\top \mathbf{h} \leq F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j} + \mathbf{h}, \mathbf{x}^j) - F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j). \tag{27}$$

From $\widetilde{\rho}_{k_j-\nu_j} \geq \widetilde{\gamma}$ we obtain

$$-\widetilde{\gamma}^{-1} F_{k_j-\nu_j+1}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) \geq -F_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j).$$

Adding $\frac{1}{2}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})^\top Q_j(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})$ on both sides gives

$$\begin{aligned}-\widetilde{\gamma}^{-1} F_{k_j-\nu_j+1}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) + \tfrac{1}{2}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})^\top Q_j(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) \\ \geq -F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j).\end{aligned}$$

Combining this with (27) gives

$$\widetilde{g}_j^\top \mathbf{h} \le F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}+\mathbf{h},\mathbf{x}^j) - \widetilde{\gamma}^{-1}F_{k_j-\nu_j+1}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j) + \tfrac{1}{2}(\mathbf{x}^j-\mathbf{y}^{k_j-\nu_j})^\top Q_j(\mathbf{x}^j-\mathbf{y}^{k_j-\nu_j}). \quad (28)$$

Since $\mathbf{y}^{k_j-\nu_j}-\mathbf{x}^j\to 0$, the rightmost term converges to 0 by boundedness of the Q_j. We claim that the term $\widetilde{\gamma}^{-1}F_{k_j-\nu_j+1}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)$ converges to $\widetilde{\gamma}^{-1}F(\mathbf{x}^*,\mathbf{x}^*)=0$.

It suffices to show $F^{[1]}_{k_j-\nu_j+1}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\to 0$, because we already know that $F^{[2]}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)=\frac{1}{2}(\mathbf{y}^{k_j-\nu_j}-\mathbf{x}^j)^\top Q_j(\mathbf{y}^{k_j-\nu_j}-\mathbf{x}^j)$ converges to 0. Now recall $F^{[1]}_{k_j-\nu_j+1}(\mathbf{y}^{k_j+\nu_j},\mathbf{x}^j)=m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)$ for a cutting plane $m_{k_j-\nu_j}(\cdot,\mathbf{x}^j)$ at $\mathbf{y}^{k_j-\nu_j}$ with regard to serious iterate $\mathbf{x}^j$. That means we have $m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\le F^{[1]}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\to F^{[1]}(\mathbf{x}^*,\mathbf{x}^*)=0$, because cutting planes are downshifted tangents. Hence $\limsup m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\le 0$. It therefore suffices to show $\liminf m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\ge F^{[1]}(\mathbf{x}^*,\mathbf{x}^*)=0$. Now $m_{k_j-\nu_j}(\cdot,\mathbf{x}^j)=t_{k_j-\nu_j}(\cdot)-s_j$, where $t_{k_j-\nu_j}$ is a tangent to $F^{[1]}(\cdot,\mathbf{x}^j)$ at $\mathbf{y}^{k_j-\nu_j}$, and $s_j\ge 0$ is the down-shift. Clearly $t_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j})=F^{[1]}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\to F^{[1]}(\mathbf{x}^*,\mathbf{x}^*)=0$ by joint continuity of F and the fact that the second order term also goes to 0, so we can concentrate on proving $s_j\to 0$. Now

$$s_j=\left[t_{k_j-\nu_j}(\mathbf{x}^j)\right]_+ + c\|\mathbf{x}^j-\mathbf{y}^{k_j-\nu_j}\|^2\to 0$$

because $t_{k_j-\nu_j}(\mathbf{x}^j)=F^{[1]}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)+\widetilde{g}_j^\top(\mathbf{x}^j-\mathbf{y}^{k_j-\nu_j})\to 0$ by the argument just used. This proves our claim $F_{k_j-\nu_j+1}(\mathbf{y}^{k_j-\nu_j},\mathbf{x}^j)\to 0$.

Going back with this information to (28), passing to the limit gives $\widetilde{g}^\top\mathbf{h}$ on the left hand side and $\ell:=\limsup F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}+\mathbf{h},\mathbf{x}^j)$ on the right, we have $\widetilde{g}^\top\mathbf{h}\le\ell$, and we proceed to analyze the terms $F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}+\mathbf{h},\mathbf{x}^j)$ occurring on the right of (28).

Observe that $F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}+\mathbf{h},\mathbf{x}^j)=m_{\mathbf{z}_j(\mathbf{h})}(\mathbf{y}^{k_j-\nu_j}+\mathbf{h},\mathbf{x}^j)$ for one of the cutting planes contributing to the buildup of $F^{[1]}_{k_j-\nu_j}(\cdot,\mathbf{x}^j)$. By construction, $m_{\mathbf{z}_j(\mathbf{h})}(\cdot,\mathbf{x}^j)=t_{\mathbf{z}_j(\mathbf{h})}(\cdot)-s_j$, where $t_{\mathbf{z}_j(\mathbf{h})}(\cdot)$ is a tangent to $F^{[1]}(\cdot,\mathbf{x}^j)$ at a null step $\mathbf{z}_j(\mathbf{h})$, and s_j is the downshift is with regard to this tangent and serious iterate $\mathbf{x}^j$. The $\mathbf{z}_j(\mathbf{h})$ are among the previous null steps which form a bounded set. We may therefore extract a subsequence with $\mathbf{z}_j(\mathbf{h})\to\mathbf{z}(\mathbf{h})$ for some $\mathbf{z}(\mathbf{h})$. The tangent is of the form $t_{\mathbf{z}_j(\mathbf{h})}(\cdot)=F^{[1]}(\mathbf{z}_j(\mathbf{h}),\mathbf{x}^j)+g^\top_{\mathbf{z}_j(\mathbf{h})}(\cdot-\mathbf{z}_j(\mathbf{h}))$, where $g_{\mathbf{z}_j(\mathbf{h})}\in\partial_1F^{[1]}(\mathbf{z}_j(\mathbf{h}),\mathbf{x}^j)$. Passing to another subsequence, we may assume $g_{\mathbf{z}_j(\mathbf{h})}\to g_{\mathbf{z}(\mathbf{h})}\in\partial_1F^{[1]}(\mathbf{z}(\mathbf{h}),\mathbf{x}^*)$ by upper semi-continuity of the Clarke subdifferential.

Next observe that for this subsequence the downshift also converges $s_j\to s^*$, where s^* is the downshift of tangent $t_{\mathbf{z}(\mathbf{h})}(\cdot)$ at $\mathbf{z}(\mathbf{h})$ with subgradient $g_{\mathbf{z}(\mathbf{h})}$ at serious step $\mathbf{x}^*$. That shows $t_{\mathbf{z}_j(\mathbf{h})}(\mathbf{y}^{k_j-\nu_j}+\mathbf{h})-s_j\to t_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^*+\mathbf{h})-s^*=m_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^*+\mathbf{h},\mathbf{x}^*)=\ell$. As usual there are two cases for s^*.

First consider the case $s^* = \left[t_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^*)\right]_+ + c\|\mathbf{x}^* - \mathbf{z}(\mathbf{h})\|^2 = t_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^*) + c\|\mathbf{x}^* - \mathbf{z}(\mathbf{h})\|^2$. Then $\widetilde{g}^\top \mathbf{h} \le \ell = g_{\mathbf{z}(\mathbf{h})}^\top \mathbf{h} - c\|\mathbf{x}^* - \mathbf{z}(\mathbf{h})\|^2 \le g_{\mathbf{z}(\mathbf{h})}^\top \mathbf{h}$. This shows that if $\mathbf{h} \to 0$, then $\mathbf{z}(\mathbf{h}) \to \mathbf{x}^*$. Consequently, $g_{\mathbf{z}(\mathbf{h})} \to g_{\mathbf{x}^*}$ for some $g_{\mathbf{x}^*} \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$. Now fixing a unit vector $\mathbf{e}$, we can steer $\mathbf{h} \to 0$ in such a way that $\mathbf{h}/\|\mathbf{h}\| \to \mathbf{e}$. That implies $\widetilde{g}^\top \mathbf{e} \le g_{\mathbf{x}^*}^\top \mathbf{e} \le \max\{g^\top \mathbf{e} : g \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)\}$. The expression on the right is the support function of $\partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$, and by Hahn–Banach, $\widetilde{g} \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$.

Next consider the case $s^* = c\|\mathbf{x}^* - \mathbf{z}\|^2$. Then $\widetilde{g}^\top \mathbf{h} \le F^{[1]}(\mathbf{z}, \mathbf{x}^*) + g_{\mathbf{z}}^\top(\mathbf{x}^* + \mathbf{h} - \mathbf{z}) - c\|\mathbf{x}^* - \mathbf{z}\|^2 \le g_{\mathbf{z}}^\top \mathbf{h} - c\|\mathbf{x}^* - \mathbf{z}\|^2 \le g_{\mathbf{z}}^\top \mathbf{h}$ using $\left[t_{\mathbf{z}}(\mathbf{x}^*)\right]_+ = 0$. That gives the same estimate as before, so the conclusion in both cases is $\widetilde{g} \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$.

vii) Let $\eta := \operatorname{dist}(0, \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*))$. We have to prove $\eta = 0$. Assume on the contrary that $\eta > 0$. Then $\|\widetilde{g}\| \ge \eta > 0$ for $\widetilde{g}$ found in part vi). Fix $0 < \zeta < 1$. Using $\widetilde{g}_j \to \widetilde{g}$ we have $\|\widetilde{g}_j\| \ge (1-\zeta)\eta$ for j large enough. Now, assuming first that $[\dots]_+ > 0$, we have

$$\begin{aligned} m_{k_j - \nu_j}(\cdot, \mathbf{x}^j) &= F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) + \widetilde{g}_j^\top(\cdot - \mathbf{y}^{k_j-\nu_j}) - s_j \\ &= F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) + \widetilde{g}_j^\top(\cdot - \mathbf{y}^{k_j-\nu_j}) - t_{k_j-\nu_j}(\mathbf{x}^j) - c\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2 \\ &= \widetilde{g}_j^\top(\cdot - \mathbf{x}^j) - c\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2. \end{aligned} \tag{29}$$

Therefore

$$\begin{aligned} F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) &- m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) \\ &= F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) - \widetilde{g}_j^\top(\mathbf{y}^{k_j-\nu_j} - \mathbf{x}^j) + c\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2. \end{aligned}$$

Now choose $\epsilon > 0$ such that

$$\epsilon < \frac{(1-\zeta)^2(\widetilde{\gamma} - \gamma)\eta}{(1+\zeta)^2}. \tag{30}$$

Since f and g are lower C^1, the $F(\cdot, \mathbf{x}^j)$ are uniformly ϵ-convex in the sense that there exists $j(\epsilon)$ such that $\widetilde{g}_j^\top(\mathbf{y}^{k_j-\nu_j} - \mathbf{x}^j) \le F^{[1]}(\mathbf{x}^j, \mathbf{x}^j) - F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) + \epsilon\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|$ for all $j \ge j(\epsilon)$, cf. [14, Thm. 2]. Substituting this in (29) at $\mathbf{y}^{k_j-\nu_j}$ gives

$$\begin{aligned} F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) - m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) &\le \epsilon\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\| + c\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2 \\ &\le (1+\zeta)\epsilon\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\| \end{aligned} \tag{31}$$

for j large enough. The case $[\dots]_+ = 0$ in (29) leads to the even stronger estimate $F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) - m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) = c\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2$, so we may continue with (31). Now, recall that $\widetilde{g}_j = (Q_j + \frac{1}{2}\tau_{k_j} I)(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) \in \partial_1 F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j)$ gives

$$\widetilde{g}_j^\top(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) \le F^{[1]}_{k_j-\nu_j}(\mathbf{x}^j, \mathbf{x}^j) - F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) = -F^{[1]}_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j).$$

Subtracting a quadratic term from both sides, we get

$$\tfrac{1}{2}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})^\top Q_j(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) + \tfrac{1}{2}\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2 \le -F_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j).$$

As $\tau_{k_j} \to \infty$, we have

$$(1-\zeta)\tfrac{1}{2}\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\| \le \|\widetilde{g}_j\| \le (1+\zeta)\tfrac{1}{2}\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|$$

and also

$$\begin{aligned}\tfrac{1}{2}(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j})^\top Q_j(\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}) + \frac{1}{2}\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2 \\ \ge (1-\zeta)\frac{1}{2}\tau_{k_j}\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|^2.\end{aligned}$$

Combining these gives

$$-F_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) \ge \frac{(1-\zeta)^2}{1+\zeta}\eta\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|. \tag{32}$$

Combining (31) and (32) leads to

$$\begin{aligned}\widetilde{\rho}_{k_j-\nu_j} &= \rho_{k_j-\nu_j} + \frac{F^{[1]}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j) - m_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j)}{-F_{k_j-\nu_j}(\mathbf{y}^{k_j-\nu_j}, \mathbf{x}^j)} \\ &\le \rho_{k_j-\nu_j} + \frac{(1+\zeta)^2\epsilon\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|}{(1-\zeta)^2\eta\|\mathbf{x}^j - \mathbf{y}^{k_j-\nu_j}\|} \qquad \text{use (31) and (32)} \\ &\le \rho_{k_j-\nu_j} + \widetilde{\gamma} - \gamma < \widetilde{\gamma}, \qquad \text{use (30)}\end{aligned}$$

contradicting $\widetilde{\rho}_{k_j-\nu_j} \ge \widetilde{\gamma}$ for the infinitely many $j \in J^+$. This shows that the hypothesis $\eta > 0$ was incorrect, hence $\eta = 0$, which ends the convergence proof in the case started in part iv).

viii) It remains to deal with the case $g_j^* \to 0$, $j \in J'$. Since g_j^* is a subgradient of $F_{k_j}(\cdot, \mathbf{x}^j)$ at $\mathbf{x}^{j+1}$, the subgradient inequality gives for any test vector $\mathbf{h}'$:

$$\begin{aligned}g_j^{*\top}\mathbf{h}' &\le F_{k_j}^{[1]}(\mathbf{x}^{j+1} + \mathbf{h}', \mathbf{x}^j) - F_{k_j}^{[1]}(\mathbf{x}^{j+1}, \mathbf{x}^j) \\ &= F_{k_j}^{[1]}(\mathbf{x}^{j+1} + \mathbf{h}', \mathbf{x}^j) - F_{k_j}(\mathbf{x}^{j+1}, \mathbf{x}^j) + \tfrac{1}{2}(\mathbf{x}^j - \mathbf{x}^{j+1})^\top Q_j(\mathbf{x}^j - \mathbf{x}^{j+1}) \\ &= F_{k_j}^{[1]}(\mathbf{x}^{j+1} + \mathbf{h}', \mathbf{x}^j) - F_{k_j}(\mathbf{x}^{j+1}, \mathbf{x}^j) \\ &\qquad + \tfrac{1}{2}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2_{Q_j+\tau_{k_j}I} - \tfrac{1}{2}\tau_{k_j}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2 \\ &\le F_{k_j}^{[1]}(\mathbf{x}^{j+1} + \mathbf{h}', \mathbf{x}^j) - F_{k_j}(\mathbf{x}^{j+1}, \mathbf{x}^j) + \tfrac{1}{2}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2_{Q_j+\tau_{k_j}I} \\ &\le F_{k_j}^{[1]}(\mathbf{x}^{j+1} + \mathbf{h}', \mathbf{x}^j) - \gamma^{-1}F(\mathbf{x}^{j+1}, \mathbf{x}^j) + \tfrac{1}{2}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2_{Q_j+\tau_{k_j}I}.\end{aligned}$$

Fixing another test vector $\mathbf{h}$, we put $\mathbf{h}' = \mathbf{x}^j - \mathbf{x}^{j+1} + \mathbf{h}$ and substitute it to obtain

$$\tfrac{1}{2}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2_{Q_j+\tau_{k_j}I} + g_j^{*\top}\mathbf{h} \leq F^{[1]}_{k_j}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j) - \gamma^{-1}F(\mathbf{x}^{j+1}, \mathbf{x}^j).$$

Since $g_j^* \to 0$ by hypothesis, and $\|\mathbf{x}^j - \mathbf{x}^{j+1}\|_{Q_j+\tau_{k_j}I} \to 0$, $F(\mathbf{x}^{j+1}, \mathbf{x}^j) \to 0$ by part i), and we may therefore condense the above to

$$\epsilon_j \leq F^{[1]}_{k_j}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j)$$

for every test vector $\mathbf{h}$, where $\epsilon_j = \tfrac{1}{2}\|\mathbf{x}^j - \mathbf{x}^{j+1}\|^2_{Q_j+\tau_{k_j}I} + g_j^{*\top}\mathbf{h} + \gamma^{-1}F(\mathbf{x}^{j+1}, \mathbf{x}^j) \to 0$.

Now recall that in the jth inner loop $F^{[1]}_{k_j}(\cdot, \mathbf{x}^j)$ is constructed as a maximum of cutting planes, so there exists a null step $\mathbf{z}_j(\mathbf{h}) \in \{\mathbf{y}^1, \ldots, \mathbf{y}^{k_j-1}\}$ such that $F^{[1]}_{k_j}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j) = m_{\mathbf{z}_j(\mathbf{h})}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j)$ for the cutting plane at trial $\mathbf{z}_j(\mathbf{h})$ for serious iterate $\mathbf{x}^j$. Next recall that $m_{\mathbf{z}_j(\mathbf{h})}(\cdot, \mathbf{x}^j) = t_{\mathbf{z}_j(\mathbf{h})}(\cdot) - s_j$, where $t_{\mathbf{z}_j(\mathbf{h})}$ is a tangent to $F^{[1]}(\cdot, \mathbf{x}^j)$ at $\mathbf{z}_j(\mathbf{h})$, and s_j is the corresponding downshift. Since $t_{\mathbf{z}_j(\mathbf{h})}(\cdot) = F^{[1]}(\mathbf{z}_j(\mathbf{h}), \mathbf{x}^j) + g^\top_{\mathbf{z}_j(\mathbf{h})}(\cdot - \mathbf{z}_j(\mathbf{h}))$ for some $g_{\mathbf{z}_j(\mathbf{h})} \in \partial_1 F^{[1]}(\mathbf{z}_j(\mathbf{h}), \mathbf{x}^j)$, we have

$$\epsilon_j \leq F^{[1]}_{k_j}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j) = F^{[1]}(\mathbf{z}_j(\mathbf{h}), \mathbf{x}^j) + g^\top_{\mathbf{z}_j(\mathbf{h})}(\mathbf{x}^j + \mathbf{h} - \mathbf{z}_j(\mathbf{h})) - s_j. \quad (33)$$

Here we have to discuss the two cases $s_j = t_{\mathbf{z}_j(\mathbf{h})} + c\|\mathbf{z}_j(\mathbf{h}) - \mathbf{x}^j\|^2$ and $s_j = c\|\mathbf{z}_j(\mathbf{h}) - \mathbf{x}^j\|^2$.

Starting with the first case, as the set of all trial steps visited during the run of the algorithm is bounded, we may extract a subsequence of J such that $\mathbf{z}_j(\mathbf{h}) \to \mathbf{z}(\mathbf{h})$ and $g_{\mathbf{z}_j(\mathbf{h})} \to g_{\mathbf{z}(\mathbf{h})}$. As $\mathbf{x}^j \to \mathbf{x}^*$, upper semi-continuity of the Clarke subdifferential gives $g_{\mathbf{z}(\mathbf{h})} \in \partial_1 F^{[1]}(\mathbf{z}(\mathbf{h}), \mathbf{x}^*)$. Moreover, as the downshift procedure is continuous in the data used, $s_j \to s$, where s is the downshift for tangent $t_{\mathbf{z}(\mathbf{h})}(\cdot)$ to $F^{[1]}(\cdot, \mathbf{x}^*)$ at $\mathbf{z}(\mathbf{h})$. In other words, $m_{\mathbf{z}(\mathbf{h})}(\cdot, \mathbf{x}^*) = t_{\mathbf{z}(\mathbf{h})}(\cdot) - s$ is the cutting plane which our method would compute at null step $\mathbf{z}$ for serious iterate $\mathbf{x}^*$ if the corresponding tangent used the subgradient $g_{\mathbf{z}(\mathbf{h})}$. Altogether, this implies

$$0 \leq F^{[1]}(\mathbf{z}(\mathbf{h}), \mathbf{x}^*) + g^\top_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^* + \mathbf{h} - \mathbf{z}(\mathbf{h})) - s,$$

where $s = t_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^*) + c\|\mathbf{z}(\mathbf{h}) - \mathbf{x}^*\|^2$. We obtain

$$\begin{aligned} 0 \leq F^{[1]}(\mathbf{z}(\mathbf{h}), \mathbf{x}^*) &+ g^\top_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^* + \mathbf{h} - \mathbf{z}(\mathbf{h})) - F^{[1]}(\mathbf{z}(\mathbf{h}), \mathbf{x}^*) \\ &- g^\top_{\mathbf{z}(\mathbf{h})}(\mathbf{x}^* - \mathbf{z}(\mathbf{h})) - c\|\mathbf{z}(\mathbf{h}) - \mathbf{x}^*\|^2, \end{aligned}$$

which can be re-arranged as

$$0 \leq c\|\mathbf{z}(\mathbf{h}) - \mathbf{x}^*\|^2 \leq g^\top_{\mathbf{z}(\mathbf{h})}\mathbf{h}. \quad (34)$$

Since the set of all possible $g_{\mathbf{z}(\mathbf{h})}$ is bounded, the estimate shows that $\mathbf{z}(\mathbf{h}) \to \mathbf{x}^*$ when $\mathbf{h} \to \mathbf{0}$. Dividing by $\|\mathbf{h}\|$, we now have

$$0 \leq g_{\mathbf{z}(\mathbf{h})}^\top \frac{\mathbf{h}}{\|\mathbf{h}\|}.$$

Now fix a unit vector $\mathbf{e}$ and let $\mathbf{h} \to \mathbf{0}$ in such a way that $\mathbf{h}/\|\mathbf{h}\| \to \mathbf{e}$. From the previous we know that $\mathbf{z}(\mathbf{h}) \to \mathbf{x}^*$. Therefore, using the upper semi-continuity of the Clarke subdifferential, we may extract a subsequence such that $g_{\mathbf{z}(\mathbf{h})} \to g_{\mathbf{x}^*}$ for some $g_{\mathbf{x}^*} \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$. We have therefore shown $0 \leq g_{\mathbf{x}^*}^\top \mathbf{e} \leq \max\{g^\top \mathbf{e} : g \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)\}$. But the expression on the right is the Clarke directional derivative of $F^{[1]}(\cdot, \mathbf{x}^*)$ at $\mathbf{x}^*$ in direction $\mathbf{e}$. As $\mathbf{e}$ was arbitrary, we have shown that the Clarke directional derivative of $F^{[1]}(\cdot, \mathbf{x}^*)$ is non-negative in every direction, and this implies $0 \in \partial_1 F^{[1]}(\mathbf{x}^*, \mathbf{x}^*)$. This ends the proof in the case $[\dots]_+ > 0$.

It remains to discuss the case $[\dots]_+ = 0$. Going back to estimate (33), we observe that the downshift is $s_j = c\|\mathbf{z}_j(\mathbf{h}) - \mathbf{x}^j\|^2$. As before, $F_{k_j}^{[1]}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j) = m_{\mathbf{z}_j(\mathbf{h})}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j)$, and we now represent the cutting plane as $m_{\mathbf{z}_j(\mathbf{h})}(\cdot, \mathbf{x}^j) = m_{\mathbf{z}_j(\mathbf{h})}(\mathbf{x}^j, \mathbf{x}^j) + g_{\mathbf{z}_j(\mathbf{h})}^\top(\cdot - \mathbf{x}^j)$ for the same $g_{\mathbf{z}_j(\mathbf{h})} \in \partial_1 F^{[1]}(\mathbf{z}_j(\mathbf{h}), \mathbf{x}^j)$. Now as the tangent at $\mathbf{x}^j$ satisfies $t_{\mathbf{z}_j(\mathbf{h})}(\mathbf{x}^j) \leq 0$, we have $m_{\mathbf{z}_j(\mathbf{h})}(\mathbf{x}^j, \mathbf{x}^j) \leq -c\|\mathbf{z}_j(\mathbf{h}) - \mathbf{x}^j\|^2$. Therefore

$$\epsilon_j \leq F_{k_j}^{[1]}(\mathbf{x}^j + \mathbf{h}, \mathbf{x}^j) \leq -c\|\mathbf{x}^j - \mathbf{z}_j(\mathbf{h})\|^2 + g_{\mathbf{z}_j(\mathbf{h})}^\top \mathbf{h}.$$

Passing to the limits $\epsilon_j \to 0$, $\mathbf{x}^j \to \mathbf{x}^*$, $\mathbf{z}_j(\mathbf{h}) \to \mathbf{z}(\mathbf{h})$, $g_{\mathbf{z}_j(\mathbf{h})} \to g_{\mathbf{z}(\mathbf{h})}$ as in the previous case, we get $0 \leq -c\|\mathbf{z}(\mathbf{h}) - \mathbf{x}^*\|^2 + g_{\mathbf{z}(\mathbf{h})}^\top \mathbf{h}$. But now we are back in the situation (34), and the conclusion is the same. This ends the proof in case $[\dots]_+ = 0$. □

5 Application to flight control

In this section, we switch back from the abstract optimization program to (5), discussing the elements needed to apply our algorithm.

5.1 The banded H_∞–norm

We start by discussing the banded H_∞-norm $f(\mathbf{x})$ in (6). The first observation is that f is lower C^1. We have the even stronger

Lemma 4 *Let f be a squared H_∞-norm (6) on a closed frequency band Ω. Then f is lower C^2 on the open set $S = \{\mathbf{x} \in \mathbb{R}^n : T_{w\to z}(\mathbf{x}, \cdot)$ is internally stable$\}$.*

Proof The mapping $\mathcal{F} : \mathbb{R}^n \times \mathbb{S}^1 \to \mathbb{H}$ defined by (7) is of class C^2 in $\mathbf{x}$ and analytic in s for $\mathbf{x} \in S$. Indeed, the closed-loop matrices $A_{\mathrm{cl}} = A + BKC$, and similarly B_{cl}, C_{cl}, D_{cl}, are affine functions of K, so that $\mathcal{F}(K, s)$ depends rationally on K and s. By construction (2), (3), the controller $K = K(\mathbf{x})$ depends rationally on $\mathbf{x}$, hence $\mathcal{F}(\mathbf{x}, s)$

depends rationally on $\mathbf{x}, s$. Since matrix inversion is allowed for $\mathbf{x} \in S$, the claim follows.

Writing the maximum eigenvalue as

$$\lambda_1(X) = \max\{Z \bullet X : Z \succeq 0, \mathrm{Trace}(Z) = 1\},$$

we have

$$f(\mathbf{x}) = \max_{\omega \in \Omega} f(\mathbf{x}, \omega) = \max_{\omega \in \Omega} \max_{Z \succeq 0, \mathrm{Tr}(Z)=1} Z \bullet \mathcal{F}(\mathbf{x}, \omega),$$

which is a representation of the form (19) with $(Z, \omega) \mapsto Z \bullet \mathcal{F}(\mathbf{x}, \omega)$ of class C^2. The compact space is $K = \{Z \in \mathbb{H} : Z \succeq 0, \mathrm{Trace}(Z) = 1\} \times \Omega$. □

Computation of the H_∞-norm is based on the algorithm of Boyd et al. [15]. Computation of Clarke subgradients $g \in \partial f(\mathbf{x})$ was discussed in [6]. Notice that the peak frequencies $\Omega(\mathbf{x}) = \{\omega \in \Omega : f(\mathbf{x}) = f(\mathbf{x}, \omega)\}$, obtained along with the function value $f(\mathbf{x})$, are needed to compute subgradients. Recall that the set $\Omega(\mathbf{x})$ of peak frequencies has a very special structure. We have

Lemma 5 (Compare [15],[16, Lemma 1]). *The set* $\Omega(\mathbf{x})$ *is either finite, or* $\Omega(\mathbf{x}) = \Omega$.

If $\mathbf{y}^k$ is a null step at serious step $\mathbf{x}$, then it is reasonable to enrich the working model $F_k(\cdot, \mathbf{x})$ by adding several cutting planes or near cutting planes of objective f and constraint c simultaneously. This may be done by building a finite set $\Omega_e(\mathbf{y}^k)$ of near active frequencies at $\mathbf{y}^k$, i.e., frequencies ω satisfying $f(\mathbf{y}^k) - \theta \leq f(\mathbf{y}^k, \omega) < f(\mathbf{y}^k)$ for some threshold $\theta > 0$, and computing tangents to $f(\cdot, \omega)$ at $\mathbf{y}^k$. By Lemma 5 we assure that $\Omega_e(\mathbf{y}^k) \supset \Omega(\mathbf{y}^k)$ when $\Omega(\mathbf{y}^k)$ is finite, which it always is in practice. Similarly for tangents arising from the constraint c. These near tangents to F are then downshifted with respect to the current value $F(\mathbf{x}, \mathbf{x}) = 0$ just as the regular tangent. Ways to select an extended set of frequencies $\Omega_e(\mathbf{y}^k)$ containing $\Omega(\mathbf{y}^k)$ are given in [6]. It is for instance wise to include the finitely many secondary peaks, that is, the local maxima of the curve $\omega \mapsto f(\mathbf{y}^k, \omega)$, because secondary peaks are candidates to become active at the next iteration. Ways to compute those are for instance given in [16].

5.2 Internal stability

The last issue we have to discuss before applying our algorithm to (5) concerns the hidden constraint $\mathbf{x} \in S = \{\mathbf{x} \in \mathbb{R}^n : T_i(\mathbf{x}, \cdot), i = 1, \ldots, 6$ are internally stable$\}$, which is not dealt with explicitly in (8). Notice that S is an open set, so $\mathbf{x} \in S$ is not a constraint in the usual sense of optimization. The closed-loop channels T_i in (5) are obtained by substituting controllers $K^{(1)}$, $K^{(2)}$ into the corresponding plants (4), which provides closed-loop system matrices $A_i(\mathbf{x})$ whose stability we have to guarantee. Using the spectral abscissa $\alpha(A) = \max\{\mathrm{Re}(\lambda) : \lambda$ eigenvalue of $A\}$, we

can replace internal stability by the inequality constraint

$$\max_{i=1,\ldots,6} \alpha\left(A_i(\mathbf{x})\right) \leq -\epsilon \tag{35}$$

for some small $\epsilon > 0$. In order to maintain stability of the iterates, we add the constraint (35) to program (5).

Notice that in our application the open-loop system is stable, and it is not too hard to tune the three blocks autopilot, flight controller, low-pass filter independently to find a stabilizing choice of parameters $\mathbf{x}_1$. In other situations, it may be necessary to compute an initial stabilizing iterate $\mathbf{x}_1$ satisfying (35) by solving an optimization program. Here one may use the method of Burke et al. [17], which consists in optimizing

$$\min_{\mathbf{x}\in\mathbb{R}^n} \max_{i=1,\ldots,6} \alpha\left(A_i(\mathbf{x})\right) \tag{36}$$

using a descent method until $\mathbf{x}_1$ satisfying (35) is found.

Remark 8 As a rule it is easy to find a stabilizing controller for practical systems, those being designed to work correctly. However, from a purely mathematical point of view, *deciding* whether or not a stabilizing structured controller exists is NP-complete for most practical controller structures [18]. That means if one fails to find a stabilizing controller, e.g. with program (36), or by using specific knowledge about the given application, then a proof that no stabilizing controller of the given structure exists will take exponential time (in the system order), and will therefore be difficult or even impossible to obtain.

5.3 Numerical results

In this section, we present numerical tests obtained with our algorithm. In a first phase an initial stabilizing controller $\mathbf{x}_1 = [-0.1, -0.15, -1.0, 5\sqrt{2}, 25, -5.0, -0.05, -0.0035, 0]$ is found by a traditional design, where each of the blocks (PI, P, filter) is tuned manually and independently. The corresponding closed-loop channels are shown in blue in Figs. 5, 6, 7. The six H_∞-norms involved are $\|T\|_\infty := (\|T_1\|_\infty, \ldots, \|T_6\|_\infty)$ with $\|T\|_\infty = [1.0336e+00, 2.2775e+00, 3.0700e+00, 3.0359e-01, 1.1345e+00, 3.8147e+00]$, which means $f(\mathbf{x}_1) = 3.07^2$, $c(\mathbf{x}_1) = 3.8147^2$. The algorithm is now run with the constraint $c(\mathbf{x}) = \max_{i=5,6} \|W_i^{-1} T_i(\mathbf{x}, \cdot)\|_\infty^2 - r^2 \leq 0$ with $r = 1.08$. We used the following two-stage stopping test. If the inner loop at $\mathbf{x}^j$ finds a serious iterate $\mathbf{x}^{j+1}$ satisfying

$$\frac{\|\mathbf{x}^j - \mathbf{x}^{j+1}\|}{1 + \|\mathbf{x}^j\|} < \text{tol}, \tag{37}$$

then $\mathbf{x}^{j+1}$ is accepted as the final solution. On the other hand, if the inner loop is unable to find a serious step and provides three consecutive unsuccessful trial steps $\mathbf{y}^k$ satisfying

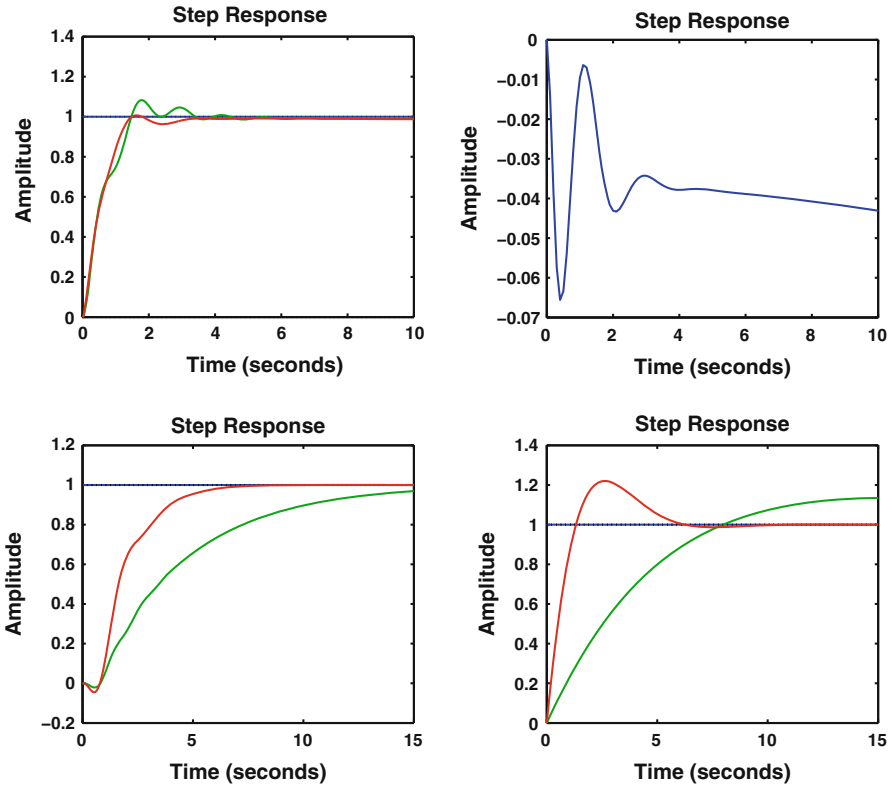


Fig. 9 Closed-loop step responses for $\mathbf{x}^*$. *Top left* $T_{N_z \to dN_z}$, *top right* $T_{nq \to dm}$, *bottom left* $T_{\gamma \to d\gamma}$, *bottom right* $T_{V \to dV}$

$$\frac{\|\mathbf{x}^j - \mathbf{y}^k\|}{1 + \|\mathbf{x}^j\|} < \text{tol}, \tag{38}$$

or if a maximum number of 20 allowed steps k in the inner loop is reached, then we decide that $\mathbf{x}^j$ is already optimal. The second stopping criterion (38) is rarely invoked in our experiments. Both tests are based on the observation that $0 \in \partial_1 F(\mathbf{x}^j, \mathbf{x}^j)$ if and only if $\mathbf{y}^k = \mathbf{x}^j$ is solution of the tangent program (11), and on Lemmas 2, 3.

In our flight control example we use tol $= 2.0 \times 10^{-4}$, which induces the algorithm to stop based on (37) after 72 iterations within 379 seconds CPU. The relative progress of function and constraint at that stage are

$$|f(\mathbf{x}^{j+1}) - f(\mathbf{x}^j)|/(1 + |f(\mathbf{x}^j)|) = 1.3 \times 10^{-5},$$
$$|c(\mathbf{x}^{j+1}) - c(\mathbf{x}^j)|/(1 + |c(\mathbf{x}^j)|) = 6.9 \times 10^{-5}.$$

The optimal controller was $\mathbf{x}^* = [-0.0937, -0.108, -0.648, 10.743, 34.335, -10.968, -0.218, -0.142, 0.0258]$ with $\|T\|_\infty = [1.0181, 1.0890, 1.0257, 1.0890, 1.0273, 1.0800]$ meaning $f(\mathbf{x}^*) = 1.089^2$, $c(\mathbf{x}^*) = 1.08^2$. In particular, the constraint

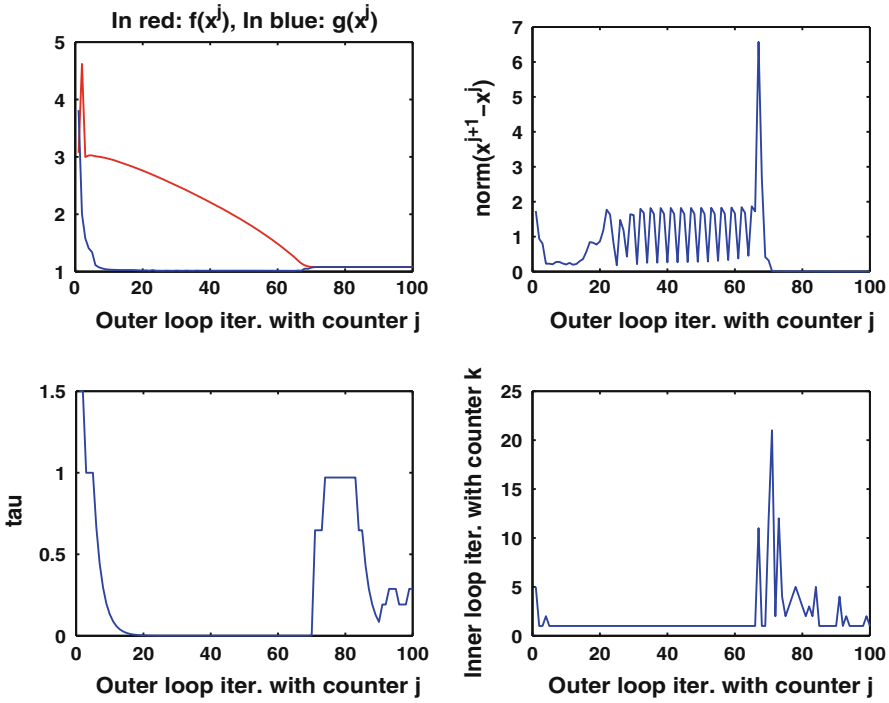


Fig. 10 Bearing of the algorithm. *Top left* shows $j \mapsto f(\mathbf{x}^j)$ (red) and $j \mapsto c(\mathbf{x}^j)$ (blue). *Top right* $j \mapsto \|x^{j+1} - x^j\|$ shows length of accepted serious step. *Lower left* shows $j \mapsto k_j$, the number of iterates of the inner loop. *Lower right* shows $j \mapsto \tau_j^\sharp$, the τ-parameter at serious steps. From iteration 72 onward progress is slight, the inner loop takes more time to find serious steps, and τ behaves more irregularly

is active, as it should be. The performance and robustness curves of $\mathbf{x}^*$ are shown in red in Figs. 5–7. Time domain responses of $\mathbf{x}^*$ are shown in Fig. 9.

For the purpose of testing, we considered smaller values of the tolerance tol in order to see how many iterations the algorithm needs to reach this precision. For instance, tol $= 1.12 \times 10^{-4}$ leads already to 100 iterations, reached in 713 s CPU, tol $= 1.1 \times 10^{-4}$ leads to 119, tol $= 1.09 \times 10^{-4}$ to 138, tol $= 1.06 \times 10^{-4}$ to 169 iterations, highlighting the well-known fact that stopping is a delicate problem in non-smooth methods.

Figure 10 displays typical parameters of the algorithm during the first 100 iterations. From iteration 73 onwards the algorithm essentially stagnates, which leads to an increase in τ and k_j. Steplength at that stage becomes small, and progress is slight.

The final experiment consists in inspecting step responses in closed loop (see Fig. 9).

6 Conclusion

We have applied a nonconvex bundle algorithm to solve a multi-objective H_∞-control design problem (5), where the controller is structured. Convergence of the algorithm has been proved in the sense that every accumulation point $\mathbf{x}^*$ of the sequence of serious

iterates $\mathbf{x}^j$ is either a critical point of constraint violation, or a Karush–Kuhn–Tucker point. We have shown that the algorithm allows to solve the problem of simultaneous synthesis of flight controller and autopilot in longitudinal flight of aircraft.

The proposed technique has two advantages over the model-based bundle technique of [4], where an ideal model is used to compute cutting planes. In the case of the composite H_∞-norm (6), this ideal model is of the form $\phi(\cdot, \mathbf{x}) = \max_{\omega\in\mathbb{S}^1} \lambda_1 \left(\mathcal{F}(\mathbf{x}, \omega) + \mathcal{F}'(\mathbf{x}, \omega)(\cdot - \mathbf{x})\right)$ and has therefore the same structure as (6), but may be costly to compute if the system gets sizable. In [3] it was shown that computing $\phi(\mathbf{y}, \mathbf{x})$ at a trial step $\mathbf{y}$ can be up to 27 times more expensive than computing the objective $f(\mathbf{y})$ itself. A second observation is that the new method seems less prone to rapid increase of the τ-parameter in the inner loop, which on average allows larger steps.

Acknowledgments Financial support by Fondation de Recherche pour l'Aéronautique et l'Espace (FNRAE) under research grant *Survol* and by Fondation d'Entreprise EADS (FEADS) under research grant *Technicom* is gratefully acknowledged.

Appendix

The numerical data for the specific flight point Mach= 0.7, Altitude= 5000 ft used in (5) are

$$A = \begin{bmatrix} -0.0120 & -9.8040 & -14.8800 & 0 & 0 \\ 0.0004 & 0 & 0.8524 & 0 & -0.0000 \\ -0.0004 & 0 & -0.8524 & 1.0000 & 0.0000 \\ 0 & 0 & -2.6650 & -0.2783 & 0 \\ 0 & 234.1000 & 0 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 4.9580 & 0 \\ 0 & 0.3113 \\ 0 & -0.3113 \\ 0 & -4.9360 \\ 0 & 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 1.0000 & 0 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 & 0 \\ 0.0085 & 0 & 13.5409 & -0.7092 & -0.0001 \\ 0 & 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & & 1.0000 \end{bmatrix},$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & -5.1535 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}.$$

References

1. Tischler MB (1996) Advances in aircraft flight control. Taylor & Francis, Bristol
2. Stevens BL, Lewis FL (1992) Aircraft control and simulation. Wiley, New York
3. Apkarian P, Noll D, Prot O (2008) A trust region spectral bundle method for nonconvex eigenvalue optimization. SIAM J Optim 19(1):281–306
4. Noll D, Prot O, Rondepierre A (2008) A proximity control algorithm to minimize nonsmooth and nonconvex functions. Pacific J Optim 4(3):569–602
5. Noll D (2010) Cutting plane oracles to minimize nonsmooth and nonconvex functions. J Set-Valued Variat Anal 18(3–4):531–568
6. Apkarian P, Noll D (2006) Nonsmooth H_∞ synthesis. IEEE Trans Autom Control 51(1):71–86
7. Polak E (1997) Optimization: algorithms and consistent approximations. Spinger, Berlin
8. Apkarian P, Noll D, Rondepierre A (2008) Mixed H_2/H_∞ control via nonsmooth optimization. SIAM J Control Optim 47(3):1516–1546
9. Alazard D (2002) Robust H_2 design for lateral flight control of a highly flexible aircraft. J Guidance Control Dyn 25(6):502–509
10. Apkarian P, Noll D (2006) Nonsmooth optimization for multidisk H_∞ synthesis. Eur J Control 12(3):229–244
11. Apkarian P, Noll D (2007) Nonsmooth optimization for multiband frequency domain control design. Automatica 43(4):724–731
12. Spingarn JE (1981) Submonotone subdifferentials of Lipschitz functions. Trans Am Math Soc 264:77–89
13. Rockafellar RT, Wets RJ-B (1998) Variational analysis. Springer, Berlin
14. Georgiev P, Daniilidis A (2004) Approximate convexity and submonotonicity. J Math Anal Appl 291:117–144
15. Boyd S, Balakrishnan V (1990) A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its L_∞-norm. Syst Control Lett 15:1–7
16. Bompart V, Noll D, Apkarian P (2007) Second order non smooth optimization for feedback control. Numer Math 107(3):433–454
17. Burke JV, Lewis AS, Overton ML (2002) Two numerical methods for optimizing matrix stability. Linear Algebra Appl 351–352:117–145
18. Blondel V, Tsitsiklis J (1997) NP-hardness of some linear control design problems. SIAM J Control Optim 35(6):2118–2127