# COMBINING STABILIZED SQP WITH THE AUGMENTED LAGRANGIAN ALGORITHM*

A. F. Izmailov†, M. V. Solodov‡, and E. I. Uskov§

July 8, 2014 (Revised November 21, 2014, and February 23, 2015)


## ABSTRACT

For an optimization problem with general equality and inequality constraints, we propose an algorithm which uses subproblems of the stabilized SQP (sSQP) type for approximately solving subproblems of the augmented Lagrangian method. The motivation is to take advantage of the well-known robust behavior of the augmented Lagrangian algorithm, including on problems with degenerate constraints, and at the same time try to reduce the overall algorithm locally to sSQP (which gives fast local convergence rate under weak assumptions). Specifically, the algorithm first verifies whether the primal-dual sSQP step (with unit stepsize) makes good progress towards decreasing the violation of optimality conditions for the original problem, and if so, makes this step. Otherwise, the primal part of the sSQP direction is used for linesearch that decreases the augmented Lagrangian, keeping the multiplier estimate fixed for the time being. The overall algorithm has reasonable global convergence guarantees, and inherits strong local convergence rate properties of sSQP under the same weak assumptions. Numerical results on degenerate problems and comparisons with some alternatives are reported.


**Key words:** stabilized sequential quadratic programming; augmented Lagrangian; superlinear convergence; global convergence.
**AMS subject classifications:** 65K05, 65K15, 90C30.

* Research of the first author is supported by the Russian Foundation for Basic Research Grant 14-01-00113 and by CNPq Grant PVE 401119/2014-9 (Brazil). The second author is supported in part by CNPq Grant 302637/2011-7, by PRONEX–Optimization, and by FAPERJ. The third author is supported by the Russian Foundation for Basic Research Grant 14-01-00113.

† Operations Research Department, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University (MSU), Uchebniy Korpus 2, Leninskiye Gory, 119991 Moscow, Russia.
Email: `izmaf@ccas.ru`

‡ IMPA – Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro, RJ 22460-320, Brazil.
Email: `solodov@impa.br`

§ Operations Research Department, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University (MSU), Uchebniy Korpus 2, Leninskiye Gory, 119991 Moscow, Russia.
Email: `ydoom@narod.ru`

# 1 Introduction

In this work, we consider the optimization problem

$$\begin{array}{ll} \text{minimize} & f(x) \\ \text{subject to} & h(x) = 0,\ g(x) \leq 0, \end{array} \tag{1.1}$$

where the objective function $f : \mathbb{R}^n \to \mathbb{R}$ and the constraints mappings $h : \mathbb{R}^n \to \mathbb{R}^l$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ are smooth enough. Recall that stationary points and the associated Lagrange multipliers for (1.1) are given by the Karush–Kuhn–Tucker (KKT) system

$$\frac{\partial L}{\partial x}(x,\, \lambda,\, \mu) = 0, \quad h(x) = 0, \quad \mu \geq 0, \quad g(x) \leq 0, \quad \langle \mu,\, g(x)\rangle = 0, \tag{1.2}$$

where $L : \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m \to \mathbb{R}$ is the Lagrangian of (1.1), i.e.,

$$L(x,\, \lambda,\, \mu) = f(x) + \langle \lambda,\, h(x)\rangle + \langle \mu,\, g(x)\rangle.$$

(In the above and in the rest of the paper, $\langle \cdot,\, \cdot\rangle$ stands for the Euclidean inner product, and $\|\cdot\|$ is the associated norm.) Of special interest for us would be the case when the constraints in the problem (1.1) are degenerate, in the sense that the multipliers associated to stationary points are not unique.

Given some $(x^k,\, \lambda^k,\, \mu^k) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$, which is the current approximation to a solution of the KKT system (1.2), the stabilized SQP (sSQP) method solves the following quadratic program (QP) in the primal-dual space:

$$\begin{array}{ll} \text{minimize}_{(x,\, \lambda,\, \mu)} & \langle f'(x^k),\, x - x^k\rangle + \dfrac{1}{2}\left\langle \dfrac{\partial^2 L}{\partial x^2}(x^k,\, \lambda^k,\, \mu^k)(x - x^k),\, x - x^k \right\rangle \\ & \quad + \dfrac{\sigma_k}{2}(\|\lambda\|^2 + \|\mu\|^2) \\ \text{subject to} & h(x^k) + h'(x^k)(x - x^k) - \sigma_k(\lambda - \lambda^k) = 0, \\ & g(x^k) + g'(x^k)(x - x^k) - \sigma_k(\mu - \mu^k) \leq 0, \end{array}$$

where $\sigma_k > 0$ is the dual stabilization parameter. We refer to [37, 22, 14, 12, 30] for the origins of the method and developments in this area; see also [31, Chapter 7]. Here, we only mention that sSQP has local superlinear convergence under the second-order sufficient optimality condition only, without any constraints qualification assumptions [12] (for equality-constrained problems, even the weaker noncriticality condition is enough [29]). This should be contrasted with the usual SQP method [6, 20] (see also [31, Chapter 4]), which in addition requires relatively strong regularity condition on the constraints (while sSQP needs nothing at all).

We note that very few globalizations of the local sSQP scheme have been proposed so far, all very recently. In [11] sSQP is combined with the inexact restoration method. In [34], for the equality-constrained case, the algorithm based on linesearch for the primal-dual two-parameter exact penalty function of [9] is proposed. Another approach is being developed in the series of papers [17, 18, 19] for problems with equality and nonnegativity constraints, where sSQP is combined with a certain primal-dual augmented Lagrangian algorithm. In [19], at each iteration the sSQP step is computed for equality-constrained problems resulting from identification of active bounds. Another approach based on identifying active constraints and applying sSQP to the corresponding equality-constrained problem has been considered earlier in [38], but with a generic globally convergent "outer algorithm" instead of a specific one, as in [19].

In this paper, we consider problems with general inequality constraints, and our goal is to globalize the sSQP method applied to *this* problem, rather than to an equality-constrained problem given by some active-set strategy. Essentially, we combine sSQP with the usual augmented Lagrangian algorithm (Aug-L) [4, 5]. Roughly speaking, we employ sSQP as inner iterations for solving the subproblems of the outer algorithm (in this sense, this is the common feature with [11, 17, 18, 19]). The hope is that few (ideally one) sSQP subproblem would be needed per outer iteration, at least asymptotically, thus giving fast convergence of the overall algorithm. Taking this point of view, it appears very natural to combine sSQP with the usual augmented Lagrangian algorithm. One reason is that Aug-L methods are very robust and have good convergence properties [7, 2, 3], including when applied to degenerate problems [13, 33]. Moreover, it is known that sSQP and Aug-L methods are related: in a sense, the former can be considered as linearization of the iterative subproblems of the latter. Thus, using sSQP for solving the Aug-L subproblems would seem to be a natural approach, which can be expected to be effective. In fact, the idea of combining some stabilized Newtonian scheme with the Aug-L method goes back at least to [21]; see also [4]. Despite that we use the usual augmented Lagrangian and not the more involved primal-dual augmented Lagrangian as in [17, 18, 19], it is interesting that *in the equality-constrained case* there are some close connections between our method and the algorithms in those works; see Section 3 below. In the general inequality-constrained case, our approach and that of [17, 18, 19] are rather different however. In Section 4, we report on numerical results for our algorithm on degenerate problems, which are the main motivation for employing sSQP.

Recall that the augmented Lagrangian for (1.1) is defined as follows: $L_\sigma : \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m \to \mathbb{R}$,

$$L_\sigma(x,\, \lambda,\, \mu) = f(x) + \frac{\sigma}{2}\left(\left\|\lambda + \frac{1}{\sigma}h(x)\right\|^2 + \left\|\max\left\{0,\, \mu + \frac{1}{\sigma}g(x)\right\}\right\|^2\right), \tag{1.3}$$

where $\sigma > 0$ is the (inverse) penalty parameter. Then, given the current dual iterate $(\lambda^k,\, \mu^k) \in \mathbb{R}^l \times \mathbb{R}^m$ and the parameter value $\sigma_k > 0$, the next primal iterate $x^{k+1}$ in the Aug-L method is obtained by solving (usually approximately) the unconstrained optimization subproblem

$$\text{minimize } L_{\sigma_k}(x,\, \lambda^k,\, \mu^k), \quad x \in \mathbb{R}^n,$$

and the next dual iterate is then given by

$$\lambda^{k+1} = \lambda^k + \frac{1}{\sigma_k}h(x^{k+1}), \quad \mu^{k+1} = \max\left\{0,\, \mu^k + \frac{1}{\sigma_k}g(x^{k+1})\right\}.$$

We refer to [3] and [13, 25] for state-of-the-art on global and local convergence properties of Aug-L methods, respectively. For many other issues, see [5].

To conclude this section, we define some further notation, and recall some facts, to be used in the sequel. Let the function $\rho : \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m \to \mathbb{R}$ be the so-called natural residual of the KKT system (1.2), i.e.,

$$\rho(x,\, \lambda,\, \mu) = \left\| \frac{\partial L}{\partial x}(x,\, \lambda,\, \mu) \right\| + \psi(x,\, \mu),$$

where $\psi : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$,

$$\psi(x,\, \mu) = \|(h(x),\, \min\{\mu,\, -g(x)\})\|$$

(here and throughout, the operations min and max are understood componentwise). By $\mathcal{M}(\bar{x})$ we denote the set of Lagrange multipliers associated to a stationary point $\bar{x}$ of problem (1.1); in particular,

$$\mathcal{M}(\bar{x}) = \{(\lambda,\, \mu) \in \mathbb{R}^l \times \mathbb{R}^m \mid \rho(\bar{x},\, \lambda,\, \mu) = 0\}.$$

We say that the second-order sufficient optimality condition (SOSC) holds at a stationary point $\bar{x}$ of problem (1.1) for a multiplier $(\bar{\lambda}, \bar{\mu}) \in \mathcal{M}(\bar{x})$, if it holds that

$$\left\langle \frac{\partial^2 L}{\partial x^2}(\bar{x},\, \bar{\lambda},\, \bar{\mu})\xi,\, \xi \right\rangle > 0 \quad \forall\, \xi \in C(\bar{x}) \setminus \{0\}, \tag{1.4}$$

where

$$C(\bar{x}) = \{\xi \in \mathbb{R}^n \mid h'(\bar{x})\xi = 0,\ \langle g_i'(\bar{x}),\, \xi\rangle \leq 0\ \forall\, i \in A(\bar{x}),\ \langle f'(\bar{x}),\, \xi\rangle = 0\},$$

is the critical cone of (1.1) at $\bar{x}$, with $A(\bar{x}) = \{i \in \{1,\, \ldots,\, m\} \mid g_i(\bar{x}) = 0\}$ being the set of inequality constraints active at $\bar{x}$.

Recall that, according to [23, Lemma 2] and [14, Theorem 2] (see also [31, Section 1.3.3]), the SOSC (1.4) implies the following error bound:

$$\|x - \bar{x}\| + \operatorname{dist}((\lambda,\, \mu),\, \mathcal{M}(\bar{x})) = O(\rho(x,\, \lambda,\, \mu)) \tag{1.5}$$

as $(x,\, \lambda,\, \mu) \to (\bar{x},\, \bar{\lambda},\, \bar{\mu})$. In fact, this error bound is equivalent to the assumption that the multiplier $(\bar{\lambda},\, \bar{\mu}) \in \mathcal{M}(\bar{x})$ is noncritical, as defined in [30]; see also [26] and [31, Section 1.3]. As in what follows we would only invoke this notion for the equality-constrained case, we give here the corresponding simpler definition. If there are no inequality constraints in (1.1), then $\bar{\lambda} \in \mathcal{M}(\bar{x})$ is said to be critical if

$$\exists\, \xi \in \ker h'(\bar{x}) \setminus \{0\} \text{ such that } \frac{\partial^2 L}{\partial x^2}(\bar{x},\, \bar{\lambda})\xi \in \operatorname{im}(h'(\bar{x}))^{\mathrm{T}}, \tag{1.6}$$

and noncritical otherwise. It can be easily seen that the version of the SOSC (1.4), corresponding to the equality-constrained case, implies noncriticality (the same is also true for general problems).

## 2 The algorithm and its convergence properties

We next describe our algorithm. As already commented, at each iteration we first solve the true sSQP subproblem; see (2.1) in Step 1 of Algorithm 2.1 below, where we first take $H_j$ as the true Hessian of the Lagrangian. If the sSQP direction is computed successfully, we next check whether it provides progress in solving the original problem (1.1); this is reflected by the test (2.2) in Step 2. If it is so, the sSQP step is accepted and we proceed to the next outer iteration. Otherwise, we check whether the primal part of the sSQP direction is of descent for the augmented Lagrangian (condition (2.5) below), and if it is so, we perform linesearch and decrease the value of the augmented Lagrangian, keeping the dual variable fixed.

If the true sSQP subproblem is not solvable, or when both tests (2.2) and (2.5) fail, we modify $H_j$ so that solving again the subproblem (2.1) with the new $H_j$ and verifying (2.5) results in a successful linesearch iteration. This is Step 4 of the algorithm. Some options for computing an appropriate $H_j$ would be commented below, after the statement of the algorithm.

In the case of a linesearch iteration, if it gives an acceptable approximate stationary point of the augmented Lagrangian, i.e., (2.7) below does not hold, we accept this point and update the dual iterates, the penalty parameter, and the stationarity tolerance the same way as the usual Aug-L methods [5] do, and proceed to the next outer iteration. Otherwise, we consider the linesearch step as an inner iteration within the process of solving the current Aug-L subproblem (i.e., within minimizing the augmented Lagrangian for fixed dual variables and fixed penalty parameter).

Our algorithm uses the bounds $\bar{\lambda}_{\min} < \bar{\lambda}_{\max}$ and $\bar{\mu}_{\max} > 0$ to safeguard the dual iterates, as does the ALGENCAN solver [1], for example (see [2, Algorithm 3.1] and also [5]).

Note that in the algorithm that follows, the index $k$ is associated to the objects that are updated on the outer iterations (referred to as sSQP and Aug-L iterations), and are fixed over sequences of consecutive inner iterations (indexed by $j$).

**Algorithm 2.1** Choose $r_0,\, \varepsilon_0,\, \sigma_0,\, \gamma > 0$ and $q,\, \theta,\, \tau,\, \varepsilon, \kappa,\, \delta \in (0,\, 1)$. Fix $\bar{\lambda}_{\min},\, \bar{\lambda}_{\max} \in \mathbb{R}^l$, $\bar{\lambda}_{\min} < \bar{\lambda}_{\max}$, $\bar{\mu}_{\max} \in \mathbb{R}^m_+$. Choose $(x^0,\, \lambda^0,\, \mu^0) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$, $\bar{\lambda}^0 \in [\bar{\lambda}_{\min},\, \bar{\lambda}_{\max}]$ and $\bar{\mu}^0 \in [0,\, \bar{\mu}_{\max}]$. Set $\hat{x}^0 = x^0$, $k = 0$ and $j = 0$.

1. Set
$$H_j = \frac{\partial^2 L}{\partial x^2}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k)$$
and find a stationary point $(\xi^j,\, \eta^j,\, \zeta^j) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$ of the QP
$$\begin{array}{ll} \text{minimize} & \langle f'(\hat{x}^j),\, \xi\rangle + \frac{1}{2}\langle H_j\xi,\, \xi\rangle + \frac{\sigma_k}{2}(\|\bar{\lambda}^k + \eta\|^2 + \|\bar{\mu}^k + \zeta\|^2) \\ \text{subject to} & h(\hat{x}^j) + h'(\hat{x}^j)\xi - \sigma_k\eta = 0, \quad g(\hat{x}^j) + g'(\hat{x}^j)\xi - \sigma_k\zeta \leq 0. \end{array} \tag{2.1}$$
If the problem (2.1) has no solution, go to step 4.

2. If
$$\bar{\lambda}^k + \eta^j \in [\bar{\lambda}_{\min},\, \bar{\lambda}_{\max}], \quad \bar{\mu}^k + \zeta^j \in [0,\, \bar{\mu}_{\max}], \quad \rho(\hat{x}^j + \xi^j,\, \bar{\lambda}^k + \eta^j,\, \bar{\mu}^k + \zeta^j) \leq r_k, \tag{2.2}$$

then set

$$x^{k+1} = \hat{x}^j + \xi^j, \quad \lambda^{k+1} = \bar{\lambda}^{k+1} = \bar{\lambda}^k + \eta^j, \quad \mu^{k+1} = \bar{\mu}^{k+1} = \bar{\mu}^k + \zeta^j, \tag{2.3}$$

$$\sigma_{k+1} = \rho(x^{k+1},\, \lambda^{k+1},\, \mu^{k+1}), \quad \varepsilon_{k+1} = \varepsilon_k, \quad r_{k+1} = q r_k, \tag{2.4}$$

set $j = 0$, $\hat{x}^0 = x^{k+1}$, increase $k$ by 1, and go to Step 1 (*sSQP iteration*).

3. If $(\xi^j,\, \eta^j,\, \zeta^j) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$ satisfies

$$\left\langle \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k),\, \xi^j \right\rangle \le -\gamma \|\xi^j\|^2, \tag{2.5}$$

   go to step 5.

4. Compute a symmetric matrix $H_j \in \mathbb{R}^{n\times n}$ such that the QP (2.1) has a stationary point $(\xi^j,\, \eta^j,\, \zeta^j)$ satisfying (2.5), and compute such a stationary point.

5. Compute $\hat{x}^{j+1} = \hat{x}^j + \tau^i \xi^j$, where $i$ is the smallest nonnegative integer such that

$$L_{\sigma_k}(\hat{x}^j + \tau^i \xi^j,\, \bar{\lambda}^k,\, \bar{\mu}^k) \le L_{\sigma_k}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k) + \varepsilon \tau^i \left\langle \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k),\, \xi^j \right\rangle. \tag{2.6}$$

6. If

$$\left\| \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^{j+1},\, \bar{\lambda}^k,\, \bar{\mu}^k) \right\| > \varepsilon_k, \tag{2.7}$$

   increase $j$ by 1 and go to Step 1 (*inner iteration*).

7. Set $\varepsilon_{k+1} = \theta \varepsilon_k$, $x^{k+1} = \hat{x}^{j+1}$ and

$$\lambda^{k+1} = \bar{\lambda}^k + \frac{1}{\sigma_k} h(\hat{x}^{j+1}), \quad \mu^{k+1} = \max\left\{0,\, \bar{\mu}^k + \frac{1}{\sigma_k} g(\hat{x}^{j+1})\right\}.$$

   If $\lambda^{k+1} \in [\bar{\lambda}_{\min},\, \bar{\lambda}_{\max}]$ and $\mu^{k+1} \in [0,\, \bar{\mu}_{\max}]$, set $\bar{\lambda}^{k+1} = \lambda^{k+1}$, $\bar{\mu}^{k+1} = \mu^{k+1}$; otherwise, choose some $\bar{\lambda}^{k+1} \in [\bar{\lambda}_{\min},\, \bar{\lambda}_{\max}]$ and $\bar{\mu}^{k+1} \in [0,\, \bar{\mu}_{\max}]$. If

$$\rho(x^{k+1},\, \lambda^{k+1},\, \mu^{k+1}) \le r_k, \tag{2.8}$$

   set $\sigma_{k+1} = \rho(x^{k+1},\, \lambda^{k+1},\, \mu^{k+1})$ and $r_{k+1} = q r_k$; otherwise, set

$$\sigma_{k+1} = \begin{cases} \sigma_k, & \text{if } \psi(x^{k+1},\, \mu^{k+1}) \le \delta \psi(x^k,\, \mu^k), \\ \kappa \sigma_k, & \text{else}, \end{cases} \tag{2.9}$$

   and $r_{k+1} = r_k$. Set $j = 0$, $\hat{x}^0 = x^{k+1}$, increase $k$ by 1 and go to Step 1 (*Aug-L iteration*).

Some comments are in order.

The KKT conditions for subproblem (2.1), after a straightforward simple transformation, can be stated in terms of its primal solution $(\xi, \eta, \zeta)$ only. This is due to the special structure of (2.1) – it can be easily seen that KKT conditions of (2.1) imply that the primal variables

$\eta$ and $\zeta$ are equal to the corresponding multipliers for the equality and inequality constraints, respectively. Taking this into account, stationary points of (2.1) are characterized by the following system:

$$\frac{\partial L}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k) + H_j\xi + (h'(\hat{x}^j))^{\mathrm{T}}\eta + (g'(\hat{x}^j))^{\mathrm{T}}\zeta = 0, \tag{2.10}$$

$$h(\hat{x}^j) + h'(\hat{x}^j)\xi - \sigma_k\eta = 0, \tag{2.11}$$

$$\bar{\mu}^k + \zeta \geq 0, \quad g(\hat{x}^j) + g'(\hat{x}^j)\xi - \sigma_k\zeta \leq 0, \quad \langle \bar{\mu}^k + \zeta,\, g(\hat{x}^j) + g'(\hat{x}^j)\xi - \sigma_k\zeta\rangle = 0. \tag{2.12}$$

We next discuss the question of how to find a matrix $H_j$ satisfying the requirements of Step 4 of Algorithm 2.1, if the attempt to use the true sSQP step fails. Note that we need to ensure solvability of the sSQP subproblem (2.1) and the directional descent condition (2.5). In what follows, we show that these requirements are satisfied for any $H_j$ such that

$$\left\langle \left(H_j + \frac{1}{\sigma_k}(h'(\hat{x}^j))^{\mathrm{T}}h'(\hat{x}^j)\right)\xi,\, \xi\right\rangle \geq \gamma\|\xi\|^2 \quad \forall\, \xi \in \mathbb{R}^n, \tag{2.13}$$

i.e., $H_j$ such that the matrix in the left-hand side is sufficiently positive definite.

The condition (2.13) is, of course, automatic if $H_j$ itself is taken sufficiently positive definite. For instance, all theoretical conclusions below are valid if we simply take $H_j = I$. However, for efficiency of the algorithm, it would be natural to take $H_j$ as a possibly small modification of the true Hessian, in particular because the latter is already computed at Step 1.

In the equality-constrained case, the requirement (2.13) automatically holds with some $\gamma > 0$ for all $k$ large enough under the SOSC (1.4), if the sequence approaches such a solution, if $\sigma_k \to 0$, and we take $H_j = \frac{\partial^2 L}{\partial x^2}(\hat{x}^j,\, \bar{\lambda}^k)$ (this can be easily seen using the Finsler–Debreau Lemma, e.g., [4, Lemma 1.25]). Note also that (2.11) is equivalent to

$$\eta = \frac{1}{\sigma_k}(h(\hat{x}^j) + h'(\hat{x}^j)\xi), \tag{2.14}$$

and in the equality-constrained case, the equality (2.10) takes the form

$$\left(H_j + \frac{1}{\sigma_k}(h'(\hat{x}^j))^{\mathrm{T}}h'(\hat{x}^j)\right)\xi = -\frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k). \tag{2.15}$$

So, if the matrix in the left-hand side is nonsingular, then the linear system (2.14), (2.15) has the unique solution $(\xi^j,\, \eta^j)$. Since equations (2.14), (2.15) are equivalent to the Lagrange system of the sSQP subproblem, $(\xi^j,\, \eta^j)$ is also the unique stationary point of this subproblem. Sufficient positive definiteness of the matrix (i.e., the fulfillment of (2.13)) can be achieved by modifying it in the process of its Cholesky factorization [36, Section 3.4], if this is the approach for computing the solution $\xi^j$ of the linear system (2.15). Then $\eta^j$ is given by the explicit formula (2.14).

In the inequality-constrained case, the choice $H_j = \frac{\partial^2 L}{\partial x^2}(\hat{x}^k,\, \bar{\lambda}^k,\, \bar{\mu}^k)$ in principle need not conform to (2.13), even locally. However, note that positive definiteness of $H_j$ (and even (2.13)) is merely sufficient but *not necessary* for the existence of stationary points of (2.1) satisfying (2.5). The later may well hold even if $H_j$ is not positive definite. In our numerical experience, quite often this is indeed the case. When it turns out that modification of the true Hessian is required, $H_j$ with the needed properties can again be obtained by the modified Cholesky factorization, or by the "convexification procedure" from [17, 18, 19]. Then no more than two QPs will have to be solved at each iteration of Algorithm 2.1, as well as when we simply take $H_j = I$ for the second QP. Alternatively, $H_j$ can be adjusted sequentially: e.g., one can keep replacing $H_j$ by $H_j + \omega I$ with some $\omega > 0$ until a stationary point satisfying (2.5) is found. This, of course, may require solving more than two QPs per iteration, but the "quality" of the resulting step can be better, since the eventually accepted $H_j$ can be closer to the true Hessian of the Lagrangian. These alternative strategies will be numerically tested in Section 4.

Given (2.13), we first show that it guarantees solvability of the sSQP subproblems (2.1).

**Lemma 2.1** *Let $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^l$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ be differentiable at the point $\hat{x}^j \in \mathbb{R}^n$. Let $H_j \in \mathbb{R}^{n\times n}$ be some symmetric matrix and $\sigma_k > 0$ be such that* (2.13) *holds with some $\gamma > 0$.*

*Then, for any $\bar{\lambda}^k \in \mathbb{R}^l$, $\bar{\mu}^k \in \mathbb{R}^m$, there exists the unique $(\xi^j, \eta^j, \zeta^j) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$ solving* (2.1)*, or equivalently, satisfying* (2.10)–(2.12)*.*

**Proof.** For any $(\xi, \eta, \zeta) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$ satisfying the equation

$$h'(\hat{x}^j)\xi - \sigma_k\eta = 0, \tag{2.16}$$

for the quadratic part of the objective function in (2.1), i.e., ignoring its linear terms, we have that

$$\begin{aligned} \langle H_j\xi, \xi\rangle + \sigma_k(\|\eta\|^2 + \|\zeta\|^2) &= \langle H_j\xi, \xi\rangle + \sigma_k\left(\left\|\frac{1}{\sigma_k^2}h'(\hat{x}^j)\xi\right\|^2 + \|\zeta\|^2\right) \\ &= \left\langle\left(H_j + \frac{1}{\sigma_k}(h'(\hat{x}^j))^{\mathrm{T}}h'(\hat{x}^j)\right)\xi, \xi\right\rangle + \sigma_k\|\zeta\|^2 \\ &> 0, \end{aligned}$$

where the inequality holds for any $(\xi, \eta, \zeta) \neq 0$, by (2.13) and (2.16). This implies strong convexity of the objective function in (2.1) on the subspace defined by the condition (2.16). This, in turn, implies strong convexity of this function on the feasible set of (2.1) (as is easily seen, strong convexity of a quadratic function on some subspace implies its strong convexity on any affine manifold parallel to this subspace). Further, as for any $\sigma_k > 0$ the feasible set in (2.1) is clearly nonempty, it then follows that this problem has the unique solution. By the linearity of its constraints, we then conclude that this solution also has the associated Lagrange multipliers; and, as stated above, the solution can be seen to be given by the system (2.10)–(2.12). This system is also sufficient for optimality, by the convexity of the problem. ■

We next show that (2.13) also implies the directional descent condition (2.5). It then also follows that the linesearch procedure on Step 5 of Algorithm 2.1 is well-defined.

**Lemma 2.2** *Let $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^l$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ be differentiable at the point $\hat{x}^j \in \mathbb{R}^n$. Let $(\xi^j,\, \eta^j,\, \zeta^j)$ satisfy (2.10)–(2.12) for some $\bar{\lambda}^k \in \mathbb{R}^l$, $\bar{\mu}^k \in \mathbb{R}^m$, some symmetric matrix $H_j \in \mathbb{R}^{n\times n}$ and $\sigma_k > 0$, such that (2.13) holds with some $\gamma > 0$.*

*Then the inequality (2.5) holds true. In particular, the Armijo condition (2.6) on Step 5 of Algorithm 2.1 is satisfied for some nonnegative integer $i$.*

**Proof.** As is easily seen from (1.3), for any $(x,\, \lambda,\, \mu) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m$ and $\sigma > 0$ it holds that

$$\begin{aligned}\frac{\partial L_\sigma}{\partial x}(x,\, \lambda,\, \mu) &= f'(x) + (h'(x))^{\mathrm{T}}\left(\lambda + \frac{1}{\sigma}h(x)\right) + (g'(x))^{\mathrm{T}} \max\left\{0,\, \mu + \frac{1}{\sigma}g(x)\right\} \\ &= \frac{\partial L}{\partial x}(x,\, \lambda,\, \mu) + \frac{1}{\rho}(h'(x))^{\mathrm{T}}h(x) + (g'(x))^{\mathrm{T}} \max\left\{-\mu,\, \frac{1}{\sigma}g(x)\right\}. \qquad (2.17)\end{aligned}$$

Hence, for any $(\xi^j,\, \eta^j,\, \zeta^j)$ satisfying (2.10)–(2.12) we obtain that

$$\begin{aligned}\left\langle \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k),\, \xi^j \right\rangle &= \left\langle \frac{\partial L}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k),\, \xi^j \right\rangle + \frac{1}{\sigma_k}\langle h(\hat{x}^j),\, h'(\hat{x}^j)\xi^j\rangle \\ &\quad + \left\langle \max\left\{-\bar{\mu}^k,\, \frac{1}{\sigma_k}g(\hat{x}^j)\right\},\, g'(\hat{x}^j)\xi^j \right\rangle \\ &= -\langle H_j\xi^j,\, \xi^j\rangle - \langle \eta^j,\, h'(\hat{x}^j)\xi^j\rangle - \langle \zeta^j,\, g'(\hat{x}^j)\xi^j\rangle \\ &\quad + \frac{1}{\sigma_k}\langle h(\hat{x}^j),\, h'(\hat{x}^j)\xi^j\rangle + \left\langle \max\left\{-\bar{\mu}^k,\, \frac{1}{\sigma_k}g(\hat{x}^j)\right\},\, g'(\hat{x}^j)\xi^j \right\rangle \\ &= -\left\langle \left(H_j + \frac{1}{\sigma_k}(h'(\hat{x}^j))^{\mathrm{T}}h'(\hat{x}^j)\right)\xi^j,\, \xi^j \right\rangle \\ &\quad - \left\langle \zeta^j - \max\left\{-\bar{\mu}^k,\, \frac{1}{\sigma_k}g(\hat{x}^j)\right\},\, g'(\hat{x}^j)\xi^j \right\rangle.\end{aligned}$$

Using the latter relation and (2.13), we then obtain that

$$\left\langle \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^j,\, \bar{\lambda}^k,\, \bar{\mu}^k),\, \xi^j \right\rangle \leq -\gamma\|\xi^j\|^2 - \sum_{i=1}^m R_i, \qquad (2.18)$$

where we set

$$\begin{aligned}R_i &= \left(\zeta_i^j - \max\left\{-\bar{\mu}_i^k,\, \frac{1}{\sigma_k}g_i(\hat{x}^j)\right\}\right)\langle g_i'(\hat{x}^j),\, \xi^j\rangle \\ &= \min\left\{\bar{\mu}_i^k + \zeta_i^j,\, \zeta_i^j - \frac{1}{\sigma_k}g_i(\hat{x}^j)\right\}\langle g_i'(\hat{x}^j),\, \xi^j\rangle, \quad i = 1,\, \ldots,\, m.\end{aligned}$$

Note that for each $i = 1,\, \ldots,\, m$, if $\bar{\mu}_i^k \leq -g_i(\hat{x}^j)/\sigma_k$ then

$$R_i = (\bar{\mu}_i^k + \zeta_i^j)\langle g_i'(\hat{x}^j),\, \xi^j\rangle.$$

And if $\bar{\mu}_i^k + \zeta_i^j = 0$ then $R_i = 0$. In addition, by (2.12), in the remaining case of $\bar{\mu}_i^k + \zeta_i^j > 0$, it holds that

$$g_i(\hat{x}^j) + \langle g_i'(\hat{x}^j),\, \xi^j \rangle - \sigma_k \zeta_i^j = 0, \tag{2.19}$$

and therefore,

$$\langle g_i'(\hat{x}^j),\, \xi^j \rangle = -g_i(\hat{x}^j) + \sigma_k \zeta_i^j = -\sigma_k \left( \bar{\mu}_i^k + \frac{1}{\sigma_k} g_i(\hat{x}^j) \right) + \sigma_k (\bar{\mu}_i^k + \zeta_i^j) > 0,$$

i.e., $R_i > 0$ in this case.

On the other hand, if $\bar{\mu}_i^k > -g_i(\hat{x}^j)/\sigma_k$ then

$$R_i = \left( \zeta_i^j - \frac{1}{\sigma_k} g_i(\hat{x}^j) \right) \langle g_i'(\hat{x}^j),\, \xi^j \rangle.$$

If $\bar{\mu}_i^k + \zeta_i^j = 0$, then

$$\zeta_i^j - \frac{1}{\sigma_k} g_i(\hat{x}^j) = -\left( \bar{\mu}_i^k + \frac{1}{\sigma_k} g_i(\hat{x}^j) \right) < 0,$$

and using also (2.12), it follows that

$$\langle g_i'(\hat{x}^j),\, \xi^j \rangle \leq \sigma_k \left( \zeta_i^j - \frac{1}{\sigma_k} g_i(\hat{x}^j) \right) < 0,$$

and thus $R_i > 0$. Moreover, by (2.12), in the remaining case of $\bar{\mu}_i^k + \zeta_i^j > 0$, again (2.19) holds, and hence,

$$R_i = \frac{1}{\sigma_k} (\langle g_i'(\hat{x}^j),\, \xi^j \rangle)^2 \geq 0.$$

We conclude that, in all the cases, $R_i \geq 0$ for all $i = 1, \ldots, m$. This and (2.18) imply (2.5). ∎

The next result is, in a sense, intermediate. It shows that the number of consecutive inner iterations must be finite, in the usual situations (for example, if level sets of the augmented Lagrangian in the primal space are bounded).

**Proposition 2.1** *Let $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^l$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ be continuously differentiable on $\mathbb{R}^n$. Suppose that for some index $k$, the Steps 1–6 of Algorithm 2.1 generate an infinite sequence $\{\hat{x}^j\}$, where $\{H_j\}$ is chosen bounded.*

*Then $\|\hat{x}^j\| \to \infty$ as $j \to \infty$.*

**Proof.** The sequence $\{\hat{x}^j\}$ is generated by a descent scheme for $L_{\sigma_k}(\cdot,\, \bar{\lambda}^k,\, \bar{\mu}^k)$, with $\bar{\lambda}^k$, $\bar{\mu}^k$ and $\sigma_k$ fixed. The assertion follows by showing that the sequence $\{\xi^j\}$ is "uniformly gradient-related" with respect to $\{\hat{x}^j\}$, in the terminology of [4].

Let $J \subset \{0,\, 1,\, \ldots\}$ be an infinite set of indices such that the subsequence $\{\hat{x}^j \mid j \in J\}$ converges to some $\hat{x}$ satisfying

$$\frac{\partial L_{\sigma_k}}{\partial x}(\hat{x},\, \bar{\lambda}^k,\, \bar{\mu}^k) \neq 0. \tag{2.20}$$

From (2.5) and (2.20), we conclude that $\{\xi^j \mid j \in J\}$ is bounded.

Suppose $\{\xi^j \mid j \in J\}$ has the zero accumulation point. Without loss of generality, we can assume that $\{\xi^j \mid j \in J\}$ converges to zero. By (2.11), for each $j \in J$ it holds that

$$\eta^j = \frac{1}{\sigma_k}(h(\hat{x}^j) + h'(\hat{x}^j)\xi^j).$$

Thus, $\{\eta^j \mid j \in J\} \to h(\hat{x})/\sigma_k$. On the other hand, by (2.12) and the fact that the number of subsets of $\{1, \ldots, m\}$ is finite, we can consider that there exists some set $I \subset \{1, \ldots, m\}$ such that, for each $j \in J$,

$$\bar{\mu}_i^k + \zeta_i^j > 0, \quad g_i(\hat{x}^j) + \langle g_i'(\hat{x}^j), \xi^j\rangle - \sigma_k\zeta_i^j = 0 \quad \forall\, i \in I,$$

$$\bar{\mu}_i^k + \zeta_i^j = 0, \quad g_i(\hat{x}^j) + \langle g_i'(\hat{x}^j), \xi^j\rangle - \sigma_k\zeta_i^j \leq 0 \quad \forall\, i \in \{1, \ldots, m\} \setminus I.$$

In particular,

$$-\bar{\mu}_i^k < \zeta_i^j = \frac{1}{\sigma_k}(g_i(\hat{x}^j) + \langle g_i'(\hat{x}^j), \xi^j\rangle) \quad \forall\, i \in I,$$

$$-\bar{\mu}_i^k = \zeta_i^j \geq \frac{1}{\sigma_k}(g_i(\hat{x}^j) + \langle g_i'(\hat{x}^j), \xi^j\rangle) \quad \forall\, i \in \{1, \ldots, m\} \setminus I.$$

It then follows that $\{\zeta_i^j \mid j \in J\} \to g_i(\hat{x})/\sigma_k \geq -\bar{\mu}_i^k$ for all $i \in I$, and $\zeta_i^j = -\bar{\mu}_i^k \geq g_i(\hat{x})/\sigma_k$ for all $i \in \{1, \ldots, m\} \setminus I$. Combining those two cases, we conclude that $\{\zeta^j \mid j \in J\} \to \max\{-\bar{\mu}^k, g(\hat{x})/\sigma_k\}$. Using this fact, passing onto the limit in (2.10) and also taking into account (2.17), we obtain that

$$0 = \frac{\partial L}{\partial x}(\hat{x}, \bar{\lambda}^k, \bar{\mu}^k) + \frac{1}{\sigma_k}(h'(\hat{x}))^{\mathrm{T}}h(\hat{x}) + (g'(\hat{x}))^{\mathrm{T}}\max\left\{-\bar{\mu}^k, \frac{1}{\sigma_k}g(\hat{x})\right\} = \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}, \bar{\lambda}^k, \bar{\mu}^k),$$

in contradiction with (2.20).

Thus, the sequence $\{\xi^j \mid j \in J\}$ cannot have the zero accumulation point. Hence, by (2.5), the sequence $\{\langle \frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}^j, \bar{\lambda}^k, \bar{\mu}^k), \xi^j\rangle \mid j \in J\}$ also cannot have the zero accumulation point.

We have thus shown that $\{\xi^j\}$ is "uniformly gradient-related", and [4, Theorem 1.8] then implies that every accumulation point $\hat{x}$ of $\{\hat{x}^j\}$, if any exist, satisfies

$$\frac{\partial L_{\sigma_k}}{\partial x}(\hat{x}, \bar{\lambda}^k, \bar{\mu}^k) = 0.$$

Then the test (2.7) in Step 6 of Algorithm 2.1 means that $\{\hat{x}^j\}$ cannot have accumulation points, which completes the proof. ∎

The next theorem establishes the key global convergence properties of Algorithm 2.1. We note that the theoretical possibility of convergence to infeasible points is something standard for augmented Lagrangian methods [5], and thus carries over also to related techniques, including what is presented here. Specifically, when all iterations from some point on are of

Aug-L type, every primal accumulation point is stationary for the infeasibility minimizing problem

$$\text{minimize } \|h(x)\|^2 + \| \max\{0,\, g(x)\}\|^2, \quad x \in \mathbb{R}^n,$$

and if this point is feasible and satisfies some weak constraints qualifications, then it is necessarily stationary in the original problem (1.1).

**Theorem 2.1** *Let $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^l$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ be continuously differentiable on $\mathbb{R}^n$. Let $\{(x^k,\, \lambda^k,\, \mu^k)\}$ be any sequence generated by Algorithm 2.1.*

*Then for every accumulation point $(\bar{x},\, \bar{\lambda},\, \bar{\mu})$ of $\{(x^k,\, \lambda^k,\, \mu^k)\}$, the following holds true:*

$$\frac{\partial L}{\partial x}(\bar{x},\, \bar{\lambda},\, \bar{\mu}) = 0, \quad \bar{\mu} \geq 0, \quad (h'(\bar{x}))^{\mathrm{T}} h(\bar{x}) + (g'(\bar{x}))^{\mathrm{T}} \max\{0,\, g(\bar{x})\} = 0, \tag{2.21}$$

*and either $\bar{x}$ is a stationary point of* (1.1) *with the associated Lagrange multiplier $(\bar{\lambda},\, \bar{\mu})$, or for any subsequence of $\{(x^k,\, \lambda^k,\, \mu^k)\}$ converging to $(\bar{x},\, \bar{\lambda},\, \bar{\mu})$, all its elements with sufficiently large indices are generated by the Aug-L iterations of Algorithm 2.1. Moreover, if the sequence $\{\sigma_k\}$ generated by Algorithm 2.1 is separated from zero, then $\bar{x}$ is a stationary point of* (1.1) *with the associated Lagrange multiplier $(\bar{\lambda},\, \bar{\mu})$.*

**Proof.** Let $K \subset \{1,\, 2,\, \ldots\}$ be an arbitrary index set for which the subsequence $\{(x^k,\, \lambda^k,\, \mu^k) \mid k \in K\}$ converges to $(\bar{x},\, \bar{\lambda},\, \bar{\mu})$. Note that for each $k \in K$, it either holds that

$$\rho(x^k,\, \lambda^k,\, \mu^k) \leq r_{k-1}, \tag{2.22}$$

or $(x^k,\, \lambda^k,\, \mu^k)$ is generated by an Aug-L iteration with $\sigma_k$ given by (2.9).

If for $k \in K$ the condition (2.22) holds infinitely often, then $r_k \to 0$, since $r_k = q r_{k-1}$ for all such $k$ and since $\{r_k\}$ is nonincreasing. But then (2.22) implies $\rho(\bar{x},\, \bar{\lambda},\, \bar{\mu}) = 0$, i.e., $\bar{x}$ is a stationary point of (1.1) with associated Lagrange multipliers $(\bar{\lambda},\, \bar{\mu})$ (in particular, (2.21) holds).

We now consider the case when (2.22) does not hold for all $k \in K$ sufficiently large. Then, according to the above, for all $k \in K$ large enough it holds that $\varepsilon_k = \theta \varepsilon_{k-1}$,

$$\left\| \frac{\partial L_{\sigma_{k-1}}}{\partial x}(x^k,\, \bar{\lambda}^{k-1},\, \bar{\mu}^{k-1}) \right\| \leq \varepsilon_{k-1}, \tag{2.23}$$

and

$$\lambda^k = \bar{\lambda}^{k-1} + \frac{1}{\sigma_{k-1}} h(x^k), \quad \mu^k = \max\left\{0,\, \bar{\mu}^{k-1} + \frac{1}{\sigma_{k-1}} g(x^k)\right\}. \tag{2.24}$$

The nonnegativity of $\bar{\mu}$ in (2.21) is implied by the second equality in (2.24). By the fact that $\{\varepsilon_k\}$ is nonincreasing, it holds that $\varepsilon_k \to 0$. By (2.17) and (2.24), we obtain that

$$\begin{aligned} \frac{\partial L_{\sigma_{k-1}}}{\partial x}(x^k,\, \bar{\lambda}^{k-1},\, \bar{\mu}^{k-1}) &= f'(x^k) + (h'(x^k))^{\mathrm{T}} \left( \bar{\lambda}^{k-1} + \frac{1}{\sigma_{k-1}} h(x^k) \right) \\ &\quad + (g'(x^k))^{\mathrm{T}} \max\left\{0,\, \bar{\mu}^{k-1} + \frac{1}{\sigma_{k-1}} g(x^k)\right\} \\ &= \frac{\partial L}{\partial x}(x^k,\, \lambda^k,\, \mu^k). \end{aligned}$$

Using (2.23) and passing onto the limit along the corresponding subsequence, we obtain the first equality in (2.21). If it also holds that $\psi(\bar{x}, \bar{\mu}) = 0$, then $\rho(\bar{x}, \bar{\lambda}, \bar{\mu}) = 0$.

Note now that for any $k$, $\sigma_{k+1}$ can be larger than $\sigma_k$ only if (2.8) holds. Also, (2.2) implies (2.8) for the corresponding $(x^{k+1}, \lambda^{k+1})$, and in this case, $\sigma_{k+1} \le r_k$ and $r_{k+1} = qr_k$. Therefore, if (2.8) holds for an infinite number of indices $k$, then $\sigma_k \to 0$. If (2.8) holds only a finite number of times, then there exists $\hat{k}$ such that for all $k \ge \hat{k}$ all the outer iterations of the algorithm are of the Aug-L type, the sequence $\{\sigma_k\}$ is nonincreasing, and for all $k \ge \hat{k}$ the value of $\sigma_{k+1}$ is set according to (2.9). In particular, $\{\sigma_k\}$ can be separated from zero only if $\psi(x^{k+1}, \mu^{k+1}) \le \delta\psi(x^k, \mu^k)$ for all $k$ large enough. From this, it follows that $\psi(x^k, \mu^k) \to 0$, so that $\rho(\bar{x}, \bar{\lambda}, \bar{\mu}) = 0$.

In particular, the above shows that if $\psi(\bar{x}, \bar{\mu}) \ne 0$, then $\sigma_k \to 0$. From (2.17) and (2.23),

$$\begin{aligned}&\Big\|\sigma_{k-1}\left(f'(x^k) + (h'(x^k))^{\mathrm{T}}\bar{\lambda}^{k-1} + (g'(x^k))^{\mathrm{T}}\bar{\mu}^{k-1}\right)\\ &+ (h'(x^k))^{\mathrm{T}}h(x^k) + (g'(x^k))^{\mathrm{T}}\max\{-\sigma_{k-1}\bar{\mu}^{k-1},\, g(x^k)\}\Big\| \;\le\; \varepsilon_{k-1}\sigma_{k-1}\end{aligned}$$

for all $k \in K$. Taking into account the boundedness of $\{(\bar{\lambda}^k, \bar{\mu}^k) \mid k \in K\}$, and passing onto the limit in the last relation above along the corresponding subsequence, we obtain the second equality in (2.21). ■

As already mentioned, in the case when the inverse penalty parameter is not separated from zero, the presented result allows the possibility of convergence to infeasible accumulation points satisfying (2.21). However, any subsequence with this property must be generated by Aug-L iterations (at least from some point on), and in particular, the iterative process then reduces to the augmented Lagrangian algorithm. More specifically, to the version used in the ALGENCAN [1] ([2, Algorithm 3.1]; see also [5]). Therefore, the tendency of Algorithm 2.1 to converge to infeasible points cannot be stronger than that of ALGENCAN. In our experience on feasible degenerate problems in [33], convergence of ALGENCAN to infeasible points is actually quite rare, even though it cannot be excluded theoretically.

The rate of convergence of Algorithm 2.1 is established in the following theorem. In particular, it shows that all the desirable local properties of sSQP are preserved. The nature and the necessity of the "localization" condition (2.25) below are discussed immediately after the proof.

**Theorem 2.2** *Let $f : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}^l$ and $g : \mathbb{R}^n \to \mathbb{R}^m$ be twice differentiable in a neighborhood of $\bar{x} \in \mathbb{R}^n$, with their second derivatives being continuous at $\bar{x}$. Let $\bar{x}$ be a stationary point of the problem* (1.1)*, and let the SOSC* (1.4) *hold with an associated Lagrange multiplier $(\bar{\lambda}, \bar{\mu}) \in \mathbb{R}^l \times \mathbb{R}^m$, satisfying also $\bar{\lambda} \in (\bar{\lambda}_{\min}, \bar{\lambda}_{\max})$, $\bar{\mu} < \bar{\mu}_{\max}$. Let the sequence $\{(x^k, \lambda^k, \mu^k)\}$ generated by Algorithm 2.1 converge to $(\bar{x}, \bar{\lambda}, \bar{\mu})$. Assume finally that for all $k$ large enough, if the point $(\hat{x}^0, \bar{\lambda}^k, \bar{\mu}^k)$ used at Step 1 of the algorithm is close enough to $(\bar{x}, \bar{\lambda}, \bar{\mu})$, then $(\xi^0, \eta^0, \zeta^0)$ satisfies*

$$\|(\xi^0,\, \eta^0,\, \zeta^0)\| \le c\,\mathrm{dist}((\hat{x}^0,\, \bar{\lambda}^k,\, \bar{\mu}^k),\, \{\bar{x}\} \times \mathcal{M}(\bar{x})) \tag{2.25}$$

*for some $c > 0$ independent of $k$.*

*Then all iterations, from some index on, are sSQP iterations, and the rate of convergence of* $\{(x^k, \lambda^k, \mu^k)\}$ *is superlinear.*

**Proof.** By the construction of the algorithm, the sequence $\{r_k\}$ is nonincreasing, and for each $k$ the inequality $r_{k+1} < r_k$ holds if and only if (2.8) is satisfied (as a result of sSQP iteration or of Aug-L iteration). But then, since $\rho(x^k, \lambda^k, \mu^k) \to 0$, the condition (2.8) much be satisfied for infinitely many indices $k$. Note that for each such $k$, we also have that $\sigma_{k+1} = \rho(x^{k+1}, \lambda^{k+1}, \mu^{k+1})$ and $r_{k+1} \geq q\rho(x^{k+1}, \lambda^{k+1}, \mu^{k+1})$.

Moreover, under the assumptions of the theorem, convergence of $\{(\lambda^k, \mu^k)\}$ to $(\bar{\lambda}, \bar{\mu})$ implies that for all $k$ large enough it holds that $\lambda^k \in [\bar{\lambda}_{\min}, \bar{\lambda}_{\max}]$, $\mu^k \in [0, \bar{\mu}_{\max}]$. This means that if for some $k$ large enough holds (2.8), then $\bar{\lambda}^{k+1} = \lambda^{k+1}$, $\bar{\mu}^{k+1} = \mu^{k+1}$.

Therefore, there exists an infinite number of indices $k$ for which on Step 1 Algorithm 2.1 computes $(\xi^0, \eta^0, \zeta^0)$ as a solution of the system (2.10)–(2.12) for $\hat{x}^0 = x^k$, $\bar{\lambda}^k = \lambda^k$, $\bar{\mu}^k = \mu^k$, $\sigma_k = \rho(x^k, \lambda^k, \mu^k)$ and

$$H_0 = \frac{\partial^2 L}{\partial x^2}(x^k, \lambda^k, \mu^k), \tag{2.26}$$

i.e., it computes sSQP direction at the point $(x^k, \lambda^k, \mu^k)$. Also, $r_k \geq q\rho(x^k, \lambda^k, \mu^k)$ holds. Denote by $K \subset \{0, 1, \ldots\}$ the set of all such indices. Taking into account the convergence of $\{(x^k, \lambda^k, \mu^k)\}$ to $(\bar{x}, \bar{\lambda}, \bar{\mu})$, using the result on local convergence of sSQP [12] (with a certain quantitative improvement using [14, Theorem 1 (b)]; see also [31, Chapter 7]) and employing the error bound (1.5), we conclude that for all $k \in K$ large enough it holds that $\bar{\lambda}^k + \eta^0 \in [\bar{\lambda}_{\min}, \bar{\lambda}_{\max}]$, $\bar{\mu}^k + \zeta^0 \in [0, \bar{\mu}_{\max}]$ and

$$\begin{aligned}
\rho(\hat{x}^0 + \xi^0, \bar{\lambda}^k + \eta^0, \bar{\mu}^k + \zeta^0) &= O(\|\hat{x}^0 + \xi^0 - \bar{x}\| + \mathrm{dist}((\bar{\lambda}^k + \eta^0, \bar{\mu}^k + \zeta^0), \mathcal{M}(\bar{x}))) \\
&= o(\|\hat{x}^0 - \bar{x}\| + \mathrm{dist}((\bar{\lambda}^k, \bar{\mu}^k), \mathcal{M}(\bar{x}))) \\
&= o(\rho(\hat{x}^0, \bar{\lambda}^k, \bar{\mu}^k)) \\
&\leq q\rho(\hat{x}^0, \bar{\lambda}^k, \bar{\mu}^k) \\
&= q\rho(x^k, \lambda^k, \mu^k) \\
&\leq r_k,
\end{aligned}$$

i.e., the test (2.2) is satisfied for $j = 0$. In particular, for such $k$ the corresponding iteration is of sSQP type, and from (2.3), (2.4) for $j = 0$, it follows that $k + 1 \in K$.

It then follows that all iterations, from some index on, are of sSQP type. Applying again the result on local convergence of sSQP [12], we obtain that the rate of convergence of $\{(x^k, \lambda^k, \mu^k)\}$ is superlinear. ∎

We emphasize that the localization condition (2.25) in Theorem 2.2, or some condition of this nature, cannot be avoided in convergence rate analyses in the general inequality-constrained case, for any algorithm which solves inequality-constrained subproblems, and does not make strong assumptions implying the uniqueness of subproblems' solutions. We refer to the discussion in [32] and, in particular, Example 5.2 therein which illustrates the issue for the usual SQP. Similar examples can be constructed for sSQP and other methods.

That said, in the simpler equality-constrained case, or when an algorithm reduces the problem locally to an equality-constrained phase, localization conditions can be avoided. This is also so for our algorithm, since for equality constraints the sSQP subproblems have unique solutions under natural assumptions; see [30]. In fact, in the equality-constrained case not only the localization condition (2.25) can be removed but also the SOSC (1.4) in Theorem 2.2 can be weakened to the assumption that the Lagrange multiplier is noncritical (recall (1.6) in Section 1). This is due to the fact that, for the equality-constrained case, noncriticality is sufficient for local superlinear convergence of sSQP [30, Theorem 1] (see also [34, Theorem 4.1] for an improved version of this result, which is also needed in the present context).

Note finally that according to the local convergence theory for sSQP [12], for any point $(x^k,\, \lambda^k,\, \mu^k)$ close enough to a solution $(\bar{x},\, \bar{\lambda},\, \bar{\mu})$ satisfying SOSC there exists a stationary point $(\xi^0,\, \eta^0,\, \zeta^0)$ of the subproblem (2.1) with $\hat{x}^0 = x^k$, $\bar{\lambda}^k = \lambda^k$, $\bar{\mu}^k = \mu^k$, $\sigma_k = \rho(x^k,\, \lambda^k,\, \mu^k)$, and $H_0$ defined according to (2.26), and such that (2.25) holds with some fixed $c > 0$. In particular, this requirement is automatically satisfied for the stationary point $(\xi^0,\, \eta^0,\, \zeta^0)$ with the smallest norm. In practice, one can facilitate satisfying this requirement by using a QP solver which seeks for solutions "not too far" from the current iterate (e.g., using warm-starts). At the same time, we emphasize that (2.25) is merely an ingredient of the analysis rather than something that has to be actually verified in practice. As already commented above, it is simply indispensable for proving superlinear convergence in the general inequality-constrained case and under natural assumptions. As natural assumptions do not imply uniqueness of subproblems' solutions, "far away" solutions must be discarded from any local analysis; this is precisely the role of (2.25).

# 3 On the relations between Algorithm 2.1 and the primal-dual SQP method

In this section we analyze the relation between Algorithm 2.1 above and Algorithm 2.1 in [17], called therein primal-dual SQP (pdSQP). We emphasize that pdSQP was further developed in [18, 19], where it was transformed into the algorithm called pdSQP2, possessing some important new features, like the use of directions of negative curvature, identification of active bounds, and applying sSQP with true Hessian and appropriate control of stabilization parameter to the identified equality-constrained problem. As we use stabilization for the original inequality-constrained problem, in that setting our strategy is clearly different from [18, 19], and a comparison is hardly possible. But in the equality-constrained case, despite using different augmented Lagrangians, the methods actually turn out to be closely related. We show this next.

According to [16, 17], we define the family of primal-dual augmented Lagrangians $M_\sigma \colon \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^l \to \mathbb{R}$ as follows:

$$M_\sigma(x,\, \lambda;\, \lambda_E) = L_\sigma(x,\, \lambda_E) + \frac{\nu\sigma}{2} \left\| \lambda - \left( \lambda_E + \frac{1}{\sigma} h(x) \right) \right\|^2,$$

where $\lambda_E$ is an estimate of the Lagrange multiplier $\bar{\lambda}$, $\sigma > 0$ is the inverse penalty parameter, and $\nu > 0$ is a certain additional fixed parameter. Then pdSQP is the following iterative procedure. At an iteration indexed by $k$, for given $x^k \in \mathbb{R}^n$, $\lambda^k \in \mathbb{R}^l$, $\lambda_E^k \in \mathbb{R}^l$ and $\sigma_k > 0$, a primal-dual search direction $d^k = (\xi^k,\, \eta^k)$ is computed as a stationary point of the subproblem

$$\begin{array}{ll} \text{minimize} & \langle f'(x^k),\, \xi\rangle + \frac{1}{2}\langle H_k\xi,\, \xi\rangle + \frac{\sigma_k}{2}\|\lambda^k + \eta\|^2 \\ \text{subject to} & h(x^k) + h'(x^k)\xi - \sigma_k(\lambda^k + \eta - \lambda_E^k) = 0. \end{array} \tag{3.1}$$

Then, $(x^{k+1},\, \lambda^{k+1}) = (x^k,\, \lambda^k) + \alpha_k d^k$ is obtained, where $\alpha_k$ is computed by so-called flexible linesearch for the family of functions $M_\sigma(\cdot;\, \lambda_E^k)$. After that, $\lambda_E^k$ and $\sigma_k$ are updated (along with some other parameters), and the method proceeds to the next iteration.

The update procedure for $\lambda_E^k$ is as follows. Define the functions $\varphi_V,\, \varphi_O\colon \mathbb{R}^n \times \mathbb{R}^l \to \mathbb{R}$,

$$\varphi_V(x,\, \lambda) = \beta\left\|\frac{\partial L}{\partial x}(x,\, \lambda)\right\| + \|h(x)\|, \quad \varphi_O(x,\, \lambda) = \left\|\frac{\partial L}{\partial x}(x,\, \lambda)\right\| + \beta\|h(x)\|,$$

where $\beta > 0$ is fixed. If $\varphi_V(x^{k+1},\, \lambda^{k+1}) \le \varphi_V^{\max}/2$ or $\varphi_O(x^{k+1},\, \lambda^{k+1}) \le \varphi_O^{\max}/2$, where $\varphi_V^{\max}$ and $\varphi_O^{\max}$ are the current "record values" of the functions $\varphi_V$ and $\varphi_O$ respectively, then $\lambda_E^{k+1} = \lambda^{k+1}$ is taken (in which case the iteration is referred to as V- or O-iteration, respectively). Otherwise, if it holds that

$$\left\|\frac{\partial M_{\sigma_k}}{\partial(x,\, \lambda)}(x^{k+1},\, \lambda^{k+1};\, \lambda_E^k)\right\| \le \varepsilon_k, \tag{3.2}$$

where $\varepsilon_k > 0$ is another parameter (playing the same role of approximate stationarity as in our Algorithm 2.1), then $\lambda_E^{k+1}$ is taken as the projection of $\lambda^{k+1}$ onto $[\bar{\lambda}_{\min},\, \bar{\lambda}_{\max}]$ (M-iteration). If (3.2) is not satisfied, then $\lambda_E^{k+1} = \lambda_E^k$ is taken (F-iteration).

Note that stationary points of the subproblem (3.1) are characterized by the linear system

$$H_k\xi + (h'(x^k))^{\mathrm{T}}\eta = -\frac{\partial L}{\partial x}(x^k,\, \lambda^k), \quad h'(x^k)\xi - \sigma_k\eta = -h(x^k) + \sigma_k(\lambda^k - \lambda_E^k).$$

From the second equation, we obtain that

$$\eta = \frac{1}{\sigma_k}(h(x^k) + h'(x^k)\xi) - (\lambda^k - \lambda_E^k). \tag{3.3}$$

Then using also the first equality is the system above, it follows that

$$\left(H_k + \frac{1}{\sigma_k}(h'(x^k))^{\mathrm{T}}h'(x^k)\right)\xi = -\frac{\partial L_{\sigma_k}}{\partial x}(x^k,\, \lambda_E^k). \tag{3.4}$$

Assuming that the matrix in the left-hand side of (3.4) is nonsingular, from (3.3), (3.4) it follows that the subproblem (3.1) has the unique stationary point. Further, comparing (3.3), (3.4) with (2.14), (2.15), we obtain the following relations between the search directions in Algorithm 2.1 and in pdSQP.

**Proposition 3.1** *Let $(\xi^1,\, \eta^1)$ be the unique solution of the system (2.14), (2.15) for $\hat{x}^j = x^k$, $\bar{\lambda}^k = \lambda_E^k$ and $H_j = H_k$, where the matrix $H_k$ is such that $H_k + (h'(x^k))^{\mathrm{T}}h'(x^k)/\sigma_k$ is nonsingular. Let $(\xi^2,\, \eta^2)$ be the unique solution of the system (3.3), (3.4).*

*Then $\xi^1 = \xi^2$ and $\lambda_E^k + \eta^1 = \lambda^k + \eta^2$.*

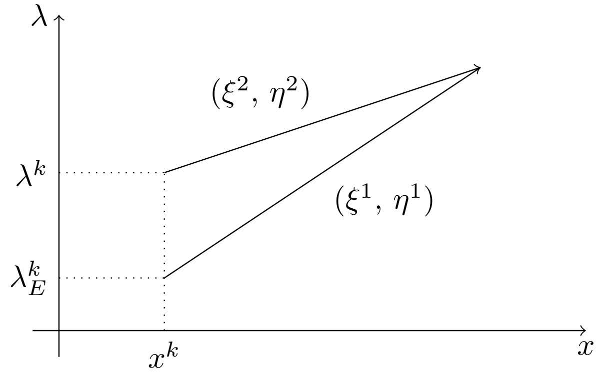



Figure 1: Illustration of Proposition 3.1.

The above implies, in particular, that for fixed $x^k$, $\lambda_E^k$ and $H_k$, the values of $\xi$ and $\lambda^k + \eta$, where $(\xi,\, \eta)$ is given by (3.3), (3.4), do not depend on $\lambda^k$. Therefore, if on some iteration (whether inner or outer) it holds that $\hat{x}^j = x^k$, $\bar{\lambda}^k = \lambda_E^k$ and $H_j = H_k$, then Algorithm 2.1 will attempt a step to the same point as would pdSQP, regardless of current value of $\lambda^k$. The connection between the search directions of the two algorithms is shown in Fig. 1.

The relations exhibited above explain why Algorithm 2.1 and pdSQP are closely related. The multiplier approximations $\lambda_E^k$ in pdSQP play the same role as $\bar{\lambda}^k$ in Algorithm 2.1. Since

$$\frac{\partial M_\sigma}{\partial(x,\, \lambda)}(x,\, \lambda;\, \lambda_E) = \begin{pmatrix} \dfrac{\partial L_\sigma}{\partial x}(x,\, \lambda_E) - \nu(h'(x))^{\mathrm{T}}\left(\lambda - \left(\lambda_E + \dfrac{1}{\sigma}h(x)\right)\right) \\ \nu\sigma\left(\lambda - \left(\lambda_E + \dfrac{1}{\sigma}h(x)\right)\right) \end{pmatrix},$$

the condition

$$\frac{\partial M_{\sigma_k}}{\partial(x,\, \lambda)}(x^{k+1},\, \lambda^{k+1};\, \lambda_E^k) = 0$$

is equivalent to the two equations

$$\frac{\partial L_{\sigma_k}}{\partial x}(x^{k+1},\, \lambda_E^k) = 0, \quad \lambda^{k+1} = \lambda_E^k + \frac{1}{\sigma_k}h(x^{k+1}),$$

which correspond to the pure (exact) iteration of the Aug-L method (for the current multiplier estimate $\lambda_E^k$). Therefore, in both methods global convergence guarantees are based on approximately minimizing augmented Lagrangian for a fixed multiplier estimate. Indeed, this estimate ($\bar{\lambda}^k$ in Algorithm 2.1, $\lambda_E^k$ in pdSQP) can only change in the following cases:

– in Algorithm 2.1, if the record target $r_k$ has been sufficiently improved (giving sSQP iteration); in pdSQP, if the record values $\varphi_V^{\max}$ or $\varphi_O^{\max}$ have been improved (giving V- or O-iteration);

– if a sufficiently good approximation to a stationary point of the augmented Lagrangian is computed, for the current fixed multiplier estimate (Aug-L iteration in Algorithm 2.1, M-iteration in pdSQP).

Otherwise, the multiplier estimates do not change; in particular, F-iteration of pdSQP is essentially an inner iteration in the terminology of Algorithm 2.1.

To summarize, despite of Algorithm 2.1 being based on the usual augmented Lagrangian and pdSQP being based on the more involved primal-dual version, in certain cases (and for the equality-constrained problems) the methods make the same or quite similar steps. In particular, if $\hat{x}^j = x^k$, $\bar{\lambda}^k = \lambda_E^k$ and $H_j = H_k$, then sSQP iteration of Algorithm 2.1 and V- or O-iteration of pdSQP with $\alpha_k = 1$ produce the same point $(x^{k+1},\, \lambda^{k+1})$. Moreover, if we suppose that $\varepsilon_k = 0$, then Aug-L iteration of Algorithm 2.1 and M-iteration of pdSQP also produce the same $(x^{k+1},\, \lambda^{k+1})$. We emphasize, however, that despite these interesting relations, there are also some differences in other aspects and the overall behavior of the algorithms may differ even in the equality-constrained setting. The inequality-constrained case is a whole different story altogether.

# 4 Computational experiments

In this section, we present numerical results for our Algorithm 2.1 and some alternatives. According to the theoretical results above, close to a solution satisfying SOSC, we can expect Algorithm 2.1 to possess a fast convergence rate. That said, the following important issue should be kept in mind. In the case of degenerate problems it is known [29, 24, 35] that, for both sSQP and for Aug-L methods, often there still exist rather large areas of attraction to critical multipliers (thus violating SOSC). Granted, the tendency of attraction to such multipliers is much weaker for sSQP than for the usual SQP and SQP-related methods [28, 29]; see also [31, Chapter 7]. Nevertheless, this attraction can still be observed with certain frequency, and in such cases the convergence rate of sSQP is also usually only linear. Therefore, it would be important to support the theoretical results obtained above by some numerical evidence of reasonable overall behavior of Algorithm 2.1 on degenerate problems, and this is our goal here.

Algorithm 2.1 was implemented in Matlab environment, using the solver quadprog from Matlab Optimization Toolbox with default parameters to solve QPs. Our experiments were performed on the AMPL DEGEN [8] collection (the first version of which is described in [27], and as the name suggests, it consists of various types of degenerate optimization problems, mostly small). We used 98 problems from DEGEN. The following problem instances we excluded: `20205`, `20223`, `30204`, `40207` (they are unbounded though they have degenerate stationary points), `2DD01-50h`, `2DD01-50v`, `2DD01-500h`, `2DD01-500v`, `2DD01-5000h` and `2DD01-5000v` (they are too large with respect to the other problems in the collection, and our simple Matlab implementation of Algorithm 2.1 does not involve any special tools for tackling large-scale problems), and `20201`, `40210`, `40211` (they have only bound constraints). For each problem instance, we performed 20 runs from random starting points $(x^0,\, \lambda^0,\, \mu^0) \in \mathbb{R}^n \times \mathbb{R}^l \times \mathbb{R}^m_+$, such that $\|(x^0,\, \lambda^0,\, \mu^0)\|_1 \leq 100$. (In our experience, the average results stabilize with 20 starting points, so that adding more points does not significantly change the comparisons.) A run was declared a success if for some $k \leq 500$ the stopping test $\rho(x^k,\, \lambda^k,\, \mu^k) \leq 10^{-6}$ has been achieved.

In Algorithm 2.1 the parameters were set in order to minimize, to the extent possible, the number of inner iterations for fixed $k$. In particular, this led us to use $r_0 = 10^4$, $\varepsilon_0 = 10^2$, $q = 0.5$ and $\theta = 0.5$. According to our experience, the weaker the tests are for accepting sSQP and Aug-L iterations, the faster is convergence. Apparently, the reason for this is that on inner iterations the multiplier estimate is not updated, and so only (the primal) part of the information obtained from solving sSQP subproblems is employed.

In the Armijo linesearch rule (2.6), we used $\tau = 0.5$, $\varepsilon = 0.1$. On Aug-L iterations, the parameters are $\delta = 0.5$, $\kappa = 0.1$. The other parameters were $\gamma = 1$, $\bar{\lambda}_{\min} = -10^{10}$, $\bar{\lambda}_{\max} = \bar{\mu}_{\max} = 10^{10}$. In all the runs, $\hat{x}^0 = x^0$, $\bar{\lambda}^0 = \lambda^0$, $\bar{\mu}^0 = \mu^0$, and $\sigma_0 = 10^{-4}$.

We tried two strategies for computing the matrix $H_j$ on Step 4 of Algorithm 2.1 (which happens when the Hessian of the Lagrangian does not provide a satisfactory outcome). The first strategy consisted of sequentially applying the following rule, until problem (2.1) has a solution with the needed properties:

$$\text{replace } H_j \text{ by } H_j + \omega I, \tag{4.1}$$

where $\omega = 10,\ 100,\ \ldots$. The alternative strategy was a direct "convexification" by modifying the matrix $\frac{\partial^2 L}{\partial x^2}(\hat{x}^j,\ \bar{\lambda}^k,\ \bar{\mu}^k) + (h'(\hat{x}^j))^{\mathrm{T}} h'(\hat{x}^j)/\sigma_k$ accordingly in the process of its Cholesky factorization [36, Section 3.4]. We note that this strategy may not be practical for large-scale problems, since the second term is usually a dense matrix even when $h'(\hat{x}^j)$ is sparse. For a more practical "convexification procedure" suitable for matrices of the specified form, see [17, 18, 19].

Finally, in the cases when the test (2.2) or the test (2.8) were satisfied, we used the following slightly modified rule to update $\sigma_k$:

$$\sigma_{k+1} = \min\{\rho(x^{k+1},\ \lambda^{k+1},\ \mu^{k+1}),\ \bar{\sigma}\},$$

where $\bar{\sigma} > 0$ is a fixed number (we used $\bar{\sigma} = 10^{-4}$). Evidently, all theoretical results established in Section 2 remain valid for this version as well.

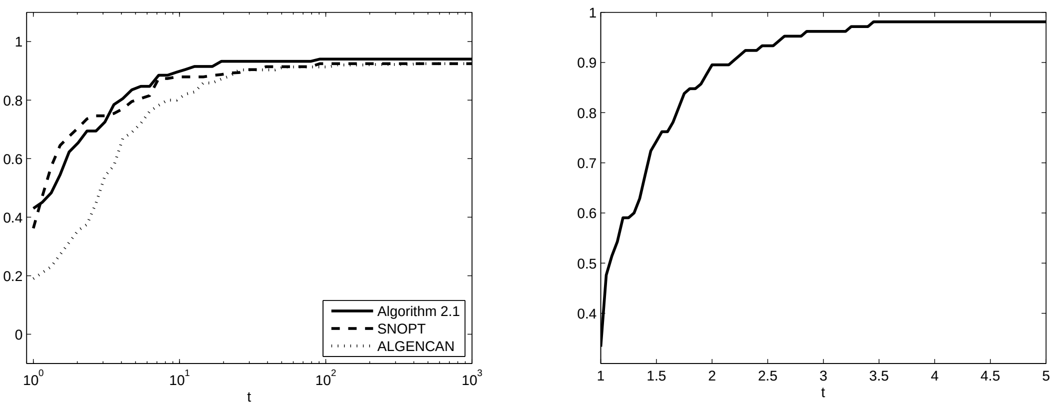
In what follows we compare Algorithm 2.1 with well-established implementations of SQP and Aug-L algorithms, namely, with SNOPT [15] and ALGENCAN [1], respectively. We used ALGENCAN 2.3.7 with AMPL interface, and SNOPT 7.2-8 coming with AMPL, both with the default values of all the parameters.

To present our numerical experience, we adopt a modification of performance profiles in [10], which takes into account that multiple starting points are used for every problem. Specifically, for each algorithm $a$ we plot the function $\pi_a : [1,\ +\infty) \to [0,\ 1]$ defined as follows. Let $k_p^a$ be the average of some measure of efficiency of algorithm $a$ on problem $p$, where the average is taken over the number $s_p^a$ of successful runs of algorithm $a$ on problem $p$. Let $r_p$ be the best (for example, smallest) value of $k_p^a$ among all the algorithms. Then for each $t \in [1,\ +\infty)$ we set

$$\pi_a(t) = \frac{1}{P} \sum_{p \in R_a(t)} s_p^a,$$

where $P$ the number of problems in the test set, and $R_a(t)$ is the subset of problems for which the performance of algorithm $a$ is no more than $t$ times worse than that of the best algorithm:

$$R_a(t) = \{p \in \{1,\ \ldots,\ P\} \mid k_p^a \le t r_p\}.$$




(a) Performance profile

(b) Average number of QPs per iteration

Figure 2: Sequential updating of $H_j$.

In particular, $\pi_a(0)$ corresponds to the portion of problems on which the average performance of the given algorithm over successful runs is the best among all algorithms being tested, while $\pi_a(t)$ for large $t$ corresponds to the portion of successful runs.

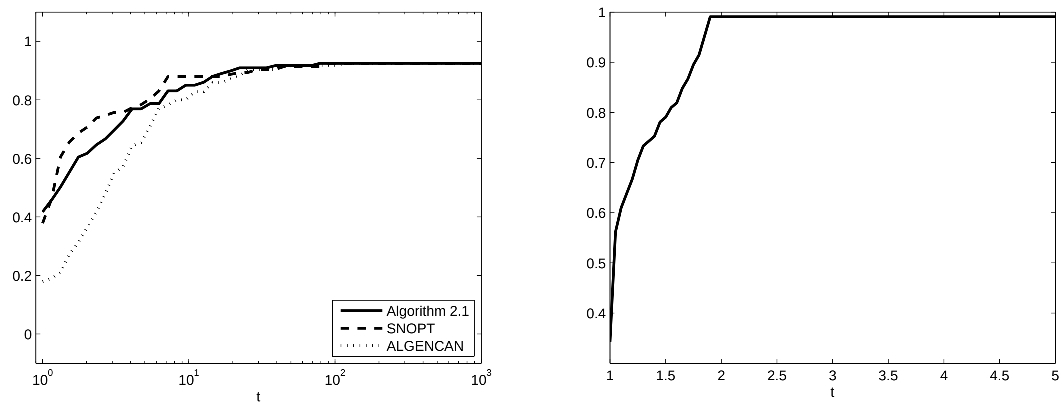
As iterations of the three algorithms are very different by nature and have different costs, we do not compare the algorithms by iteration counts. As a measure of efficiency of all the algorithms (defining the values $k_p^a$ above) we use the number of evaluations of constraints (which is always the same as the number of evaluations of the objective function). Observe also that the default versions of SNOPT and ALGENCAN do not require computation of second derivatives, while our algorithm computes the Hessian of the Lagrangian at each iteration, which certainly restricts its areas of application.

Fig. 2 shows comparisons of the specified algorithms, where Algorithm 2.1 is implemented with the sequential update (4.1) of $H_j$ at Step 4.

Performance profile in Fig. 2a demonstrates that Algorithm 2.1 is slightly more robust on DEGEN than the two other methods, and significantly more efficient than ALGENCAN. In the majority of cases, the improvement is indeed due to the sSQP iterations. Moreover, Algorithm 2.1 is at least not less efficient than SNOPT. These results are in fact quite encouraging, taking into account that both SNOPT and ALGENCAN are well-established professionally implemented solvers, supplied with useful heuristics not present in our simple Matlab implementation of Algorithm 2.1 (like scaling, for instance).

Furthermore, the graph in Fig. 2b demonstrates which portion of problems required solving no more than $t$ QPs per iteration on the average. In particular, for 70% of problems no more than 1.5 QPs had to be solved per iteration on average, and for 90% of problems no more than 2 QPs were needed.

Fig. 3 provides the same kind of information as Fig. 2, but for Algorithm 2.1 computing $H_j$ at Step 4 by the direct "convexification". Behavior of this version of Algorithm 2.1, demonstrated by the performance profile in Fig. 3a, is slightly worse than what is seen in Fig. 2a. However, as it must be the case, the number of QPs solved per iteration is no greater




(a) Performance profile

(b) Average number of QPs per iteration

Figure 3: Direct "convexification".

than 2, which can be seen from Fig. 3b.

Finally, we compare Algorithm 2.1 with the other solvers by the number of failures and the number of cases of "successful" convergence but to nonoptimal points (KKT conditions are satisfied but the point is not the global solution). The objective function value at termination is regarded nonoptimal if its difference with the (known) optimal value exceeds $10^{-2}$.

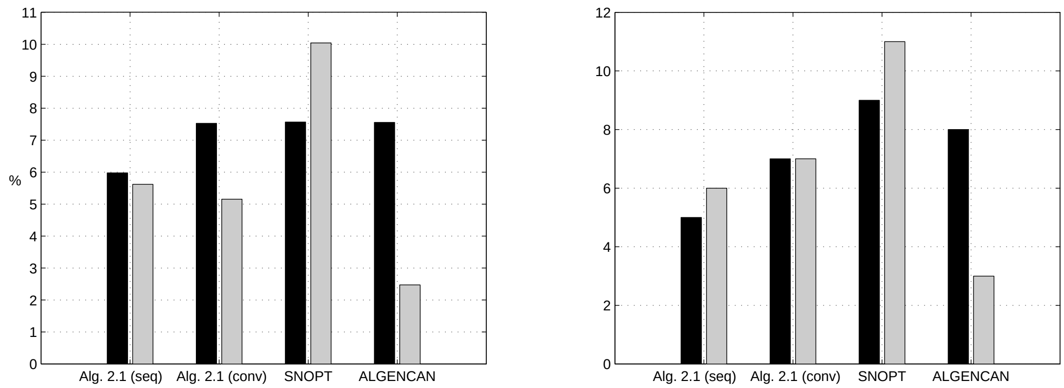
The diagram in Figure 4a reports on the percentage of runs which were failures (black color) or cases of convergence to nonoptimal points (grey color). The variants of Algorithm 2.1 with sequential updating of $H_j$ and with direct "convexification" are marked as "Alg 2.1 (seq)" and "Alg 2.1 (conv)", respectively.

The diagram in Figure 4b reports in a similar way on the number of problems for which more than 40% of runs were failures (black color) or cases of convergence to nonoptimal points (grey color). The threshold of 40% was taken without any special reason; it is just a relatively large portion of runs, exceeding 1/3. However, the relative picture in Figure 4b does not depend significantly on this choice.

The diagrams show that Algorithm 2.1 has somewhat better robustness than both SNOPT and ALGENCAN, and that it has fewer cases of convergence to nonoptimal points than SNOPT, though somewhat more than ALGENCAN.

# 5 Concluding remarks

We proposed an algorithm combining stabilized SQP (sSQP) with the augmented Lagrangian method. Specifically, the primal-dual sSQP step is tried first. If it provides progress for solving the problem, it is accepted. If not, linesearch along the primal direction is used to decrease the augmented Lagrangian, keeping the dual (multiplier) estimate and the penalty parameter fixed. Satisfactory global convergence guarantees are established, as well as fast local convergence under the same (weak) assumptions as those for sSQP.




(a) Percent of such runs

(b) Number of problems with more than 40% of such cases

Figure 4: Failures and cases of convergence to nonoptimal points.


# Acknowledgements

The authors thank the anonymous referees for many insightful comments, which led to a much improved article.


# References


[1] ALGENCAN. http://www.ime.usp.br/~egbirgin/tango/.

[2] R. Andreani, E.G. Birgin, J.M. Martínez, and M.L. Schuverdt. On augmented Lagrangian methods with general lower-level constraints. SIAM J. Optim. 18 (2007), 1286–1309.

[3] R. Andreani, G. Haeser, M.L. Schuverdt, and P.J.S. Silva. A relaxed constant positive linear dependence constraint qualification and applications. Math. Program. 135 (2012), 255–273.

[4] D.P. Bertsekas. Constrained Optimization and Lagrange Multiplier Methods. Academic Press, New York, 1982.

[5] E.G. Birgin and J.M. Martínez. Practical Augmented Lagrangian Methods for Constrained Optimization. SIAM, Philadelphia, 2014.

[6] B.T. Boggs and J.W. Tolle. Sequential quadratic programming. Acta Numerica 4 (1996), 1–51.

[7] A.R. Conn, N. Gould, A. Sartenaer, and P.L. Toint. Convergence properties of an augmented Lagrangian algorithm for optimization with a combination of general equality and linear constraints. SIAM J. Optim. 6 (1996), 674–703.

[8] DEGEN. `http://w3.impa.br/~optim/DEGEN_collection.zip`.

[9] G. Di Pillo and L. Grippo. A new class of augmented Lagrangians in nonlinear programming. SIAM J. Control Optim. 17 (1979), 618–628.

[10] E. Dolan and J. Moré. Benchmarking optimization software with performance profiles Math. Program. 91 (2002), 201–213.

[11] D. Fernández, E.A. Pilotta, and G.A. Torres. An inexact restoration strategy for the globalization of the sSQP method. Comput. Optim. Appl. 54 (2013), 595–617.

[12] D. Fernández and M. Solodov. Stabilized sequential quadratic programming for optimization and a stabilized Newton-type method for variational problems. Math. Program. 125 (2010), 47–73.

[13] D. Fernández and M. Solodov. Local convergence of exact and inexact augmented Lagrangian methods under the second-order sufficient optimality condition. SIAM J. Optim. 22 (2012), 384–407.

[14] A. Fischer. Local behavior of an iterative framework for generalized equations with nonisolated solutions. Math. Program. 94 (2002), 91–124.

[15] P.E. Gill, W. Murray, and M.A. Saunders. SNOPT: An SQP algorithm for large-scale constrained optimization. SIAM J. Optim. 12 (2002), 979–1006.

[16] P.E. Gill and D.P. Robinson. A primal-dual augmented Lagrangian. Comput. Optim. Appl. 51 (2012), 1–25.

[17] P.E. Gill and D.P. Robinson. A globally convergent stabilized SQP method. SIAM J. Optim. 23 (2013), 1983–2010.

[18] P.E. Gill, V. Kungurtsev, and D.P. Robinson. A regularized SQP method with convergence to second-order optimal points. UCSD Center for Computational Mathematics Technical Report CCoM-13-4, October 30, 2013.

[19] P.E. Gill, V. Kungurtsev, and D.P. Robinson. A globally convergent stabilized SQP method: superlinear convergence. UCSD Center for Computational Mathematics Technical Report CCoM-13-4, June 30, 2014.

[20] P.E. Gill and E. Wong. Sequential quadratic programming methods. In J. Lee and S. Leyffer, eds., Mixed Integer Nonlinear Programming, The IMA Volumes in Mathematics and its Applications, Volume 154, Springer-Verlag, pp. 147–224, 2012.

[21] S.T. Glad. Properties of updating methods for the multipliers in augmented Lagrangian. J. Optim. Theory Appl. 28 (1979), 135–156.

[22] W.W. Hager. Stabilized sequential quadratic programming. Comput. Optim. Appl. 12 (1999), 253–273.

[23] W.W. Hager and M.S. Gowda M.S. Stability in the presence of degeneracy and error estimation. Math. Program. 85 (1999), 181–192.

[24] A.F. Izmailov. On the limiting properties of dual trajectories in the Lagrange multipliers method. Comput. Math. Math. Phys. 51 (2011), 1–20.

[25] A.F. Izmailov and A.S. Kurennoy. Abstract Newtonian frameworks and their applications. SIAM J. Optim. 23 (2013), 2369–2396.

[26] A.F. Izmailov, A.S. Kurennoy, and M.V. Solodov. A note on upper Lipschitz stability, error bounds, and critical multipliers for Lipschitz-continuous KKT systems. Math. Progr. 142 (2013), 591–604.

[27] A.F. Izmailov and M.V. Solodov. Examples of dual behaviour of Newton-type methods on optimization problems with degenerate constraints. Comp. Optim. Appl. 42 (2009), 231–264.

[28] A.F. Izmailov and M.V. Solodov. On attraction of Newton-type iterates to multipliers violating second-order sufficiency conditions. Math. Progr. 117 (2009), 271–304.

[29] A.F. Izmailov and M.V. Solodov. On attraction of linearly constrained Lagrangian methods and of stabilized and quasi-Newton SQP methods to critical multipliers. Math. Program. 126 (2011), 231–257.

[30] A.F. Izmailov and M.V. Solodov. Stabilized SQP revisited. Math. Program. 122 (2012), 93–120.

[31] A.F. Izmailov and M.V. Solodov. Newton-Type Methods for Optimization and Variational Problems. Springer Series in Operations Research and Financial Engineering. 2014.

[32] A.F. Izmailov and M.V. Solodov. Newton-type methods: A broader view. J. Optim. Theory and Appl. 164 (2015), 577–620.

[33] A.F. Izmailov, M.V. Solodov, and E.I. Uskov. Global convergence of augmented Lagrangian methods applied to optimization problems with degenerate constraints, including problems with complementarity constraints. SIAM J. Optim. 22 (2012), 1579–1606.

[34] A.F. Izmailov, M.V. Solodov, and E.I. Uskov. Globalizing stabilized SQP by smooth primal-dual exact penalty function. IMPA Preprint A5566, March 2014.

[35] A.F. Izmailov and E.I. Uskov. On the influence of the critical Lagrange multipliers on the convergence rate of the multiplier method. Comput. Math. Math. Phys. 52 (2012), 1504–1519.

[36] J. Nocedal and S.J. Wright. Numerical Optimization. Springer, New York, 2006. Second Edition.

[37] S.J. Wright. Superlinear convergence of a stabilized SQP method to a degenerate solution. Comput. Optim. Appl. 11 (1998), 253–275.

[38] S.J. Wright. An algorithm for degenerate nonlinear programming with rapid local convergence. SIAM J. Optim. 15 (2005), 673–696.