

# A note on feasibility in Benders Decomposition\*

A. Grothey<sup>†</sup>, S. Leyffer<sup>‡</sup> and K.I.M. McKinnon<sup>§</sup>

January 15, 1999

Revised March 24, 1999, June 1, 2000

## Abstract

A convex nonlinear programming problem is presented which causes a simplistic application of Benders Decomposition to converge to a non-stationary point. The failure of Benders Decomposition to converge is due to the way in which infeasible subproblems are handled. A new feasibility restoration is proposed and shown to resolve this problem. The new feasibility restoration is also relevant to bundle-based decomposition schemes.

**Keywords:** Nonlinear Programming, Benders Decomposition, Variable Decomposition, Bundle-based Decomposition.

## 1 Introduction

This note considers feasibility issues arising in Benders Decomposition (e.g. Geoffrion [7] or Flippo and Rinnoy Kan [5]). We are interested in Benders Decomposition as a mechanism for decomposing and solving large scale Nonlinear Programming (NLP) problems.

An important and often difficult part in solving large NLP problems is to attain feasible points. This task is harder in Benders Decomposition as points must be obtained which are feasible in the generated subproblems, whereas SQP methods for instance only require feasible QP approximations.

In the remainder of this section Benders Decomposition is briefly reviewed (see [7] and [5] for a more detailed description). Consider the NLP problem

$$(P) \begin{cases} \underset{x,y}{\text{minimize}} & f(\mathbf{x}, \mathbf{y}) \\ \text{subject to} & \mathbf{c}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & (\mathbf{x}, \mathbf{y}) \in X \times Y, \end{cases}$$

---

\*Numerical Analysis Report NA/188, Department of Mathematics, University of Dundee

<sup>†</sup>Department of Mathematics & Statistics, University of Edinburgh, [agr@maths.ed.ac.uk](mailto:agr@maths.ed.ac.uk)

<sup>‡</sup>Department of Mathematics, University of Dundee, DD1 4HN, U.K.  
[sleyffer@maths.dundee.ac.uk](mailto:sleyffer@maths.dundee.ac.uk)

<sup>§</sup>Department of Mathematics & Statistics, University of Edinburgh, [ken@maths.ed.ac.uk](mailto:ken@maths.ed.ac.uk)

where the variables  $\mathbf{y}$  are regarded as the complicating variables. In the context of decomposition,  $\mathbf{y}$  are the linking variables and when  $\mathbf{y}$  is fixed ( $P$ ) decomposes into a number of smaller independent NLP problems.

Throughout the paper the following assumptions are made.

**A1**  $f$  and  $\mathbf{c}$  are twice continuously differentiable convex functions.

**A2**  $X$  and  $Y$  are compact polyhedral sets.

**A3** There exists  $(\mathbf{x}, \mathbf{y}) \in X \times Y : \mathbf{c}(\mathbf{x}, \mathbf{y}) < \mathbf{0}$ .

Assumption **A3** is effectively Slater's constraint qualification (e.g. [1, p. 190]) which together with **A1** ensures the existence of Lagrange multipliers. These assumptions are standard assumptions in nonlinear programming except for the convexity assumption in **A1**. In practice heuristics can be used which do not require  $f$  and  $\mathbf{c}$  to be convex. These heuristics usually involve the shifting down of cuts in the master problem. The following analysis can be readily extended to this case.

Fixing the complicating variables  $\mathbf{y} = \hat{\mathbf{y}}$  a (simpler) NLP subproblem is obtained.

$$(S(\hat{\mathbf{y}})) \left\{ \begin{array}{ll} v(\hat{\mathbf{y}}) := & \underset{x,y}{\text{minimize}} \quad f(\mathbf{x}, \mathbf{y}) \\ & \text{subject to} \quad \mathbf{c}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \mathbf{y} = \hat{\mathbf{y}} \\ & (\mathbf{x}, \mathbf{y}) \in X \times Y, \end{array} \right.$$

where  $v(\hat{\mathbf{y}})$  is the value function of  $(S(\hat{\mathbf{y}}))$ . It can be shown that the multipliers of the constraint  $\mathbf{y} = \hat{\mathbf{y}}$  are subgradients of  $v(\hat{\mathbf{y}})$ .

In general, not all choices of  $\hat{\mathbf{y}} \in Y$  give rise to feasible NLP subproblems. Therefore, in [7] and [5] a feasibility subproblem is solved, if the subproblem is infeasible. A general form of this feasibility problem is

$$(F(\hat{\mathbf{y}})) \left\{ \begin{array}{ll} v_0(\hat{\mathbf{y}}) := & \underset{x,y}{\text{minimize}} \quad \sum_{j \in J} w_j c_j^+(\mathbf{x}, \mathbf{y}) \\ & \text{subject to} \quad c_j(\mathbf{x}, \mathbf{y}) \leq 0, \forall j \in J^\perp \\ & \mathbf{y} = \hat{\mathbf{y}} \\ & (\mathbf{x}, \mathbf{y}) \in X \times Y, \end{array} \right.$$

where  $a^+ = \max(0, a)$  and the weights  $w_j$  are nonnegative and not all zero. The constraints have been divided into two disjoint sets:  $J$  contains the constraints with positive residual and  $J^\perp$  is its complement (i.e. the constraints in  $J^\perp$  are satisfied at  $\hat{\mathbf{y}}$ ). This form of feasibility problem is used in SQP methods (e.g. [3]) and is general enough to include more common forms such as the least  $l_1$  or  $l_\infty$  norm feasibility problem. As above,  $v_0(\hat{\mathbf{y}})$  is the value function of  $(F(\hat{\mathbf{y}}))$ .

In practice, it is not necessary to distinguish between  $(S(\hat{\mathbf{y}}))$  and  $(F(\hat{\mathbf{y}}))$  as most solvers will return a solution to some  $(F(\hat{\mathbf{y}}))$  if  $(S(\hat{\mathbf{y}}))$  is not feasible.

Next note that  $(P)$  is equivalent to

$$(M) \begin{cases} \underset{\eta, \mathbf{y}}{\text{minimize}} & \eta \\ \text{subject to} & \eta \geq v(\mathbf{y}) \\ & 0 \geq v_0(\mathbf{y}) \\ & \mathbf{y} \in Y. \end{cases}$$

And then define the Benders master program which is solved at iteration  $k$  as the following relaxation of this problem.

$$(M^k) \begin{cases} \underset{\eta, \mathbf{y}}{\text{minimize}} & \eta \\ \text{subject to} & \eta \geq v(\mathbf{y}^{(j)}) + \mathbf{w}^{(j)T}(\mathbf{y} - \mathbf{y}^{(j)}), \forall j \in \mathcal{S}^k \\ & 0 \geq v_0(\mathbf{y}^{(i)}) + \mathbf{w}_0^{(i)T}(\mathbf{y} - \mathbf{y}^{(i)}), \forall i \in \mathcal{F}^k \\ & \mathbf{y} \in Y. \end{cases}$$

where

$$\mathcal{S}^k := \{j \leq k : S(\mathbf{y}^{(j)}) \text{ feasible}\} \quad \text{and} \quad \mathcal{F}^k := \{i \leq k : S(\mathbf{y}^{(i)}) \text{ infeasible}\}$$

and  $\mathbf{w}^{(j)} \in \partial v(\mathbf{y}^{(j)})$  and  $\mathbf{w}_0^{(i)} \in \partial v_0(\mathbf{y}^{(i)})$ .

Benders Decomposition solves an alternating sequence of NLP subproblems ( $S(\mathbf{y}^{(k)})$  or  $F(\mathbf{y}^{(k)})$  respectively) and (LP) master problems  $(M^k)$ . The optimal values of  $(M^k)$  form a non-decreasing sequence of lower bounds on the optimum of  $(P)$  and the values  $v(\mathbf{y}^{(j)})$  (for feasible  $\mathbf{y}^{(j)}$ ) are upper bounds. The algorithm terminates when the gap between these bounds is small.

Convergence of Benders Decomposition is established in [7] under the assumption that either  $Y$  is finite (which corresponds to the integer programming case) or that  $(S(\hat{\mathbf{y}}))$  is feasible for all  $\hat{\mathbf{y}} \in Y$ . In [5] convergence is established under the assumption that either  $Y$  is finite or that the set of dual multipliers is finite. Neither of these assumptions apply if  $(P)$  is a general NLP problem and this appears to rule Benders Decomposition out for decomposing large NLP problems.

In the next section we present a convex NLP example for which Benders Decomposition fails to converge to a stationary point. The failure to converge is due to the fact that the feasibility cuts derived from  $F(\mathbf{y})$  do not provide subgradient information for  $v(\mathbf{y})$ . In Section 3 a new feasibility restoration is proposed which is shown to avoid these difficulties.

## 2 Convergence to non-stationary points

Consider the convex NLP

$$(P_1) \begin{cases} \underset{x_1, x_2, y}{\text{minimize}} & y^2 - x_2 \\ \text{subject to} & (x_1 - 1)^2 + x_2^2 \leq \ln(y) \\ & (x_1 + 1)^2 + x_2^2 \leq \ln(y) \\ & y \geq 1. \end{cases}$$

This problem is feasible with optimal solution  $(x_1, x_2, y) \simeq (0.0, 0.0337568, 2.721381)$ . To decompose this problem  $y$  is taken as the complicating variable and  $x_1$  and  $x_2$  are the subproblem variables. By fixing  $y = \hat{y}$  the following subproblem is obtained

$$(S_1(\hat{y})) \begin{cases} \underset{x_1, x_2, y}{\text{minimize}} & y^2 - x_2 \\ \text{subject to} & (x_1 - 1)^2 + x_2^2 \leq \ln(y) \\ & (x_1 + 1)^2 + x_2^2 \leq \ln(y) \\ & y = \hat{y}. \end{cases}$$

The feasible region of  $(S_1(\hat{y}))$  is the intersection of two circles in the  $(x_1, x_2)$  plane. This is illustrated in Figure 1 which shows the feasible region as the shaded area. The feasible region is controlled by  $\sqrt{\ln(\hat{y})}$  the radius of the two circles. For  $\hat{y} < e$  the feasible region is empty.

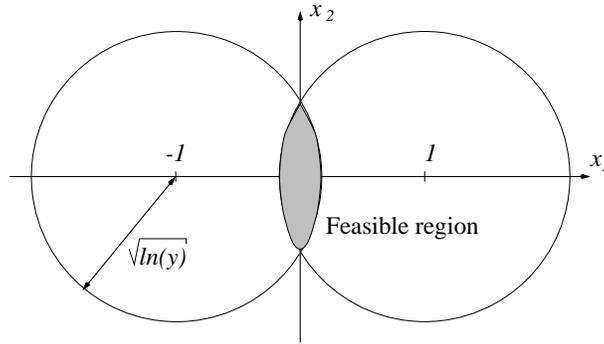


Figure 1: Feasible region of example NLP

Starting with  $y = e^2$  the subproblem is feasible and the optimal solution is  $x_1 = 0, x_2 = 1$ . A Benders cut is derived from dual information of the subproblem and the master problem is given by

$$(M_1) \begin{cases} \underset{\eta, y}{\text{minimize}} & \eta \\ \text{subject to} & \eta \geq (e^2 - 1) + (2e^2 - \frac{1}{2e^2})(y - e^2) \\ & y \geq 1. \end{cases}$$

The solution of  $(M_1)$  is  $y = 1$ . Note that the Benders cut “pushes”  $y$  to the left end of its range, as the slope  $2e^2 - \frac{1}{2e^2}$  is positive.

The next subproblem is infeasible (as  $y = 1 < e$ ) and an optimal solution to the corresponding feasibility problem is given by  $x_1 = x_2 = 0$  ( $J^\perp = \emptyset$ ). In general, if  $y^{(k)} < e$ , the following feasibility cut is generated and added to the master problem

$$0 \geq (2 - 2 \ln(y^{(k)})) + (-\frac{2}{y^{(k)}})(y - y^{(k)}) \Leftrightarrow y \geq (2 - \ln(y^{(k)}))y^{(k)}.$$

The first Benders cut will always ensure that the new  $y$  is chosen as small as possible. Thus the solution of the next master problem will be  $(2 - \ln(y^{(k)}))y^{(k)}$ . This is formalized in the following lemma.

**Lemma 2.1** If  $1 \leq y^{(k)} < e$ , then the next iterate of Benders Decomposition is  $y^{(k+1)} = (2 - \ln(y^{(k)}))y^{(k)} < e$ .

**Proof:** It is clear that provided no new optimality cuts are generated, the solution to the successive master problem relaxations is  $y^{(k+1)} = (2 - \ln(y^{(k)}))y^{(k)}$ . It only remains to show that  $y < e$  implies  $(2 - \ln(y))y < e$ . Now write  $y = e\gamma$  with  $0 < \gamma < 1$ . Then it follows that  $(2 - \ln(y))y = (1 - \ln(\gamma))\gamma e$ .

Thus it suffices to show that  $(1 - \ln(\gamma))\gamma < 1$ ,  $\forall 0 < \gamma < 1$ . Next examine the behaviour of  $(1 - \ln(\gamma))\gamma$  over the open interval  $(0, 1)$ . It is easy to show that

$$\lim_{\gamma \rightarrow 1} (1 - \ln(\gamma))\gamma = 1 \quad \text{and} \quad \lim_{\gamma \rightarrow 0} (1 - \ln(\gamma))\gamma = 0$$

Differentiating with respect to  $\gamma$  gives

$$\frac{d}{d\gamma}(1 - \ln(\gamma))\gamma = -\ln(\gamma) > 0 \quad \forall 0 < \gamma < 1.$$

Thus  $(1 - \ln(\gamma))\gamma$  is monotonic over  $0 < \gamma < 1$  and attains its supremum at  $\gamma = 1$ . This supremum is 1, which completes the proof.  $\square$

A consequence of this lemma is that Benders Decomposition will never generate a feasible  $y$  but converge to

$$\lim_{k \rightarrow \infty} y^{(k)} = e$$

a feasible but *nonstationary* point.

## 2.1 Significance to convergence of Benders Decomposition

At a first glance, the example seems to indicate that there are errors in the various convergence results in the literature. However, this is not the case as the assumptions usually exclude our example. In the case of Geoffrion's analysis [7], the example does not satisfy the assumption in Theorem 2.5 that all points  $y \in Y$  generate feasible subproblems. So the important issue of feasibility is really avoided. In their convergence Theorem 3.3, Flippo and Rinnoy Kan [5] make the assumption that either the primal or the dual feasible set are finite. Neither of the conditions for these papers hold for the example  $(P_1)$ .

As a consequence, Benders Decomposition of [7] and [5] is not applicable to the decomposition of general NLP problems. In the next section a general method for ensuring convergence to optimal solutions is presented which avoids the pitfalls of earlier work.

### 3 Feasibility Restoration & Benders Decomposition

The failure of Benders Decomposition to converge is due to the fact that the Benders cuts only approach feasibility in the limit and never collect subgradient information from the objective of  $(P)$ .

In this section a new version of Benders Decomposition is presented that resolves the problem highlighted in the previous section. The key idea is to solve an NLP problem with enlarged feasible region whenever an infeasible subproblem is encountered. The subgradient information from this second NLP is then used in Benders Decomposition. In this way it is possible to glean objective information from infeasible subproblems.

Let  $\mathbf{y}^{(k)}$  be a set of complicating variables for which the subproblem  $(S(\mathbf{y}^{(k)}))$  is infeasible. The NLP solver will detect infeasibility by generating a point  $\hat{\mathbf{x}}^{(k)}$  which solves a feasibility problem  $(F(\mathbf{y}^{(k)}))$ . Let  $\hat{\mathbf{c}}^{(k)} = \mathbf{c}(\hat{\mathbf{x}}^{(k)}, \mathbf{y}^{(k)})$  denote the corresponding constraint values and let  $\hat{\mathbf{c}}^{(k)+} = \max(\mathbf{0}, \hat{\mathbf{c}}^{(k)})$  be the vector of infeasibilities (here the max is taken componentwise). Let  $m_1 > 1$  be a constant and define a relaxed NLP subproblem as

$$(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+})) \left\{ \begin{array}{ll} \underset{x,y}{\text{minimize}} & f(\mathbf{x}, \mathbf{y}) \\ \text{subject to} & \mathbf{c}(\mathbf{x}, \mathbf{y}) \leq m_1 \hat{\mathbf{c}}^{(k)+} \\ & \mathbf{y} = \mathbf{y}^{(k)} \\ & (\mathbf{x}, \mathbf{y}) \in X \times Y. \end{array} \right.$$

Clearly,  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  is feasible by construction and its solution,  $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)})$  say, provides objective information that can be used in Benders Decomposition. The Benders cut derived from this auxiliary problem has the form (see proof of Lemma 3.4)

$$\eta \geq f^{(k)} + \boldsymbol{\lambda}^{(k)T} \mathbf{c}^{(k)} + \hat{\mathbf{w}}^{(k)T} (\mathbf{y} - \mathbf{y}^{(k)}),$$

where  $f^{(k)} = f(\mathbf{x}^{(k)}, \mathbf{y}^{(k)})$ ,  $\mathbf{c}^{(k)} = \mathbf{c}(\mathbf{x}^{(k)}, \mathbf{y}^{(k)})$ ,  $\hat{\mathbf{w}}^{(k)} \in \partial v(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  and  $\boldsymbol{\lambda}^{(k)}$  are the multipliers of  $\mathbf{c}$  in  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$ . This relaxed optimality cut is added to the master problem whenever  $(S(\mathbf{y}^{(k)}))$  is infeasible. The choice  $m_1 > 1$  implies that  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  satisfies a Slater type constraint qualification.

It is possible that for a fixed  $\mathbf{y} \in Y$ , there exist no strictly interior points of the nonlinear constraints. In this case, multipliers are either finite or they do not exist (e.g. example  $(P_1)$  with  $y = e$ ). Thus the optimality cut degenerates to a feasibility cut. In this case it is important to collect an approximate optimality cut. This can be achieved by setting  $\hat{\mathbf{c}}^{(k)+} = \epsilon \mathbf{e}$ , where  $\mathbf{e} = (1, \dots, 1)^T$ ,  $\epsilon > 0$  and then solving the relaxed problem  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$ . Since  $m_1 > 1$ , this problem satisfies Slater's constraint qualification which ensures the existence of bounded multipliers.

The master problem that is solved at every iteration of the algorithm is now given by

$$(M^k) \begin{cases} \underset{\eta, \mathbf{y}}{\text{minimize}} & \eta \\ \text{subject to} & \eta \geq v(\mathbf{y}^{(j)}) + \mathbf{w}^{(j)T}(\mathbf{y} - \mathbf{y}^{(j)}), \forall j \in \mathcal{S}^k \\ & 0 \geq v_0(\mathbf{y}^{(i)}) + \mathbf{w}_0^{(i)T}(\mathbf{y} - \mathbf{y}^{(i)}), \forall i \in \mathcal{F}^k \\ & \eta \geq f^{(i)} + \boldsymbol{\lambda}^{(i)T} \mathbf{c}^{(i)} + \hat{\mathbf{w}}^{(i)T}(\mathbf{y} - \mathbf{y}^{(i)}), \forall i \in \mathcal{F}^k \\ & \mathbf{y} \in Y. \end{cases}$$

This gives rise to the following algorithm which only differs from the algorithms presented in [7] and [5] in the way in which infeasible subproblems are treated. Let  $\epsilon_1, \epsilon_2 > 0$ .

### Benders Decomposition with Feasibility Restoration

*Initialization:* Given  $y^0$ , set  $k = 0$ , upper bound  $U^0 = \infty$  lower bound  $L^0 = -\infty$ .

#### **REPEAT**

Attempt to solve  $(S(\mathbf{y}^{(k)}))$ .

**IF**  $((S(\mathbf{y}^{(k)}))$  feasible with bounded multipliers) **THEN**

Add an optimality cut to the master problem, set  $\mathcal{S}^{k+1} = \mathcal{S}^k \cup \{k\}$ ,  $\mathcal{F}^{k+1} = \mathcal{F}^k$ .

Update the upper bound  $U^k = \min(U^k, f^{(k)})$  and the current best point.

**ELSE**

**IF**  $((S(\mathbf{y}^{(k)}))$  infeasible) **THEN**

Solve  $(F(\mathbf{y}^{(k)}))$  and determine  $\hat{\mathbf{c}}^{(k)+}$ .

**ENDIF**

Set  $\hat{\mathbf{c}}^{(k)+} = \max(\epsilon_1 \mathbf{e}, \hat{\mathbf{c}}^{(k)})$  and solve  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$ .

Derive a feasibility cut from  $(F(\mathbf{y}^{(k)}))$  and an optimality cut from  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$ , adding both to the master problem.

Set  $\mathcal{F}^{k+1} = \mathcal{F}^k \cup \{k\}$  and  $\mathcal{S}^{k+1} = \mathcal{S}^k$

**ENDIF**

Solve the master program relaxation  $(M^k)$ , obtain a new  $\mathbf{y}^{(k+1)}$

and a new lower bound  $L^k$ . Increase  $k = k + 1$ .

**UNTIL**  $(U^{k-1} - L^{k-1} < \epsilon_2)$

Note that NLP codes usually provide a solution to  $(F(\mathbf{y}^{(k)}))$  at no additional cost, if infeasibility is detected. The additional NLP problem  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  that has to be solved can be warm started from the solution of  $(F(\mathbf{y}^{(k)}))$ , reducing the amount of additional work.

An obvious alternative to the use of feasibility problems is to reformulate  $(P)$  using an exact penalty function. However, this approach has some serious disadvantages. Firstly, the penalty parameter is not known a priori and choosing a suitable penalty parameter may not be easy. Secondly, some exact penalty algorithm require the solution of  $l_1$ QP problems. This is a nontrivial task and to our knowledge, there exist no  $l_1$ QP solvers.

Next some Lemmas are presented which are useful in establishing convergence for the new feasibility restoration. Since the algorithm only differs in the feasibility restoration, it suffices to show that the cuts generated in that part are underestimators and that they ensure that the algorithm does not get trapped at a nonstationary point. Throughout it is assumed that assumptions **A1** - **A3** hold.

**Lemma 3.1** The subgradient  $\mathbf{w}^{(j)T}$  and  $\hat{\mathbf{w}}^{(i)T}$  are bounded.

**Proof:** If  $(S(\mathbf{y}^{(k)}))$  is feasible with bounded multipliers, then the corresponding subgradient is also bounded, as it is the multiplier of the  $\mathbf{y} = \mathbf{y}^{(k)}$  constraint. If  $(S(\mathbf{y}^{(k)}))$  has no strict interior feasible region or is infeasible, then the feasible region is enlarged by setting  $\hat{\mathbf{c}}^{(k)+} = \max(\epsilon_1 \mathbf{e}, \hat{\mathbf{c}}^{(k)})$ . This ensures that  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  satisfies Slater's constraint qualification and therefore has bounded multipliers [6].  $\square$

**Remark 3.1** In our example of the previous section, the multipliers are bounded for any *fixed*  $\epsilon_1 > 0$ . However, as the feasible region (i.e.  $\epsilon_1$ ) becomes smaller, the multipliers and hence the cuts approach infinity.

I.e. if  $m > 0$  is the bound of Lemma 3.1, then it could happen that  $m \simeq \mathcal{O}(\epsilon_1^{-1})$ , which will turn out to be too large.

**Remark 3.2** The subgradients  $\mathbf{w}_0^{(j)T}$  of the feasibility cuts are well defined and bounded.

**Lemma 3.2** All converging subsequence  $\mathbf{y}^{(k)} \rightarrow \mathbf{y}^*$  have a feasible limit point  $\mathbf{y}^*$ .

**Proof:** The existence of a converging subsequence follows from the compactness of  $Y$  (**A2**). Now assume that the limit  $\mathbf{y}^*$  is *not* feasible and seek a contradiction. There exists a (possibly different) subsequence  $\mathbf{y}^{(k_i)} \rightarrow \mathbf{y}^*$  with corresponding sequence  $\mathbf{x}^{(k_i)} \rightarrow \mathbf{x}^*$ . Since  $\mathbf{y}^*$  is not feasible there exists at least one component  $c_i^* > 0$  of  $\mathbf{c}^*$  which is not feasible. The continuity of  $\mathbf{c}(\mathbf{x}, \mathbf{y})$  implies that there exists a subsequence  $\mathbf{y}^{(k_i)} \rightarrow \mathbf{y}^*$  such that  $\mathbf{y}^{(k_i)}$  are not feasible.

Benders Decomposition adds a feasibility cut at each iteration of this subsequence which has the form

$$0 \geq v_0(\mathbf{y}^{(k_i)}) + \mathbf{w}_0^{(k_i)}(\mathbf{y} - \mathbf{y}^{(k_i)}).$$

The limit  $\mathbf{y}^*$  must be feasible in these cuts for all  $k$ . Now consider each of the terms in turn:  $v_0(\mathbf{y}^{(k_i)})$  converges to  $v_0(\mathbf{y}^*) > 0$  while  $\mathbf{w}_0^{(k_i)}$  remain bounded by Lemma 3.1 and  $\mathbf{y}^* - \mathbf{y}^{(k_i)} \rightarrow 0$ . Thus in the limit,  $0 \geq v_0(\mathbf{y}^*) > 0$  which is a contradiction.  $\square$

**Lemma 3.3** The value functions  $v(\mathbf{y})$  and  $v_0(\mathbf{y})$  are convex.

**Proof:** This follows directly from the convexity of  $f$  and  $\mathbf{c}$ .  $\square$

**Lemma 3.4** The cuts derived from  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  are underestimators of the value function  $v(\mathbf{y})$ .

**Proof:** Assume that  $\|\hat{\mathbf{c}}^{(k)+}\| > 0$ , otherwise the result follows from the fact that Benders cuts underestimate the value function. Let  $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)})$  be the solution to  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$ . The convexity of  $f$ ,  $\mathbf{c}$  implies that

$$\eta \geq f^{(k)} + \nabla f^{(k)T} \begin{pmatrix} \mathbf{x} - \mathbf{x}^{(k)} \\ \mathbf{y} - \mathbf{y}^{(k)} \end{pmatrix} \quad (1)$$

$$0 \geq \mathbf{c}^{(k)} + \nabla \mathbf{c}^{(k)T} \begin{pmatrix} \mathbf{x} - \mathbf{x}^{(k)} \\ \mathbf{y} - \mathbf{y}^{(k)} \end{pmatrix} \quad (2)$$

are outer approximations of the feasible region and underestimate the objective. Let  $\boldsymbol{\lambda}^{(k)}$  be the multipliers of the “ $\mathbf{c}$  constraints” of  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  and let  $\boldsymbol{\mu}^{(k)}$  be an element of the normal cone to  $X \times Y$  at  $(\mathbf{x}^{(k)}, \mathbf{y}^{(k)})$ . Summing the outer approximations (1) and (2) weighted with  $(1, \boldsymbol{\lambda}^{(k)})$  and the simple constraints  $X \times Y$  weighted with  $\boldsymbol{\mu}^{(k)}$  gives the valid inequality

$$\eta \geq f^{(k)} + \boldsymbol{\lambda}^{(k)T} \mathbf{c}^{(k)} + [\nabla f^{(k)} + \nabla \mathbf{c}^{(k)} \boldsymbol{\lambda}^{(k)} + \boldsymbol{\mu}^{(k)}] \begin{pmatrix} \mathbf{x} - \mathbf{x}^{(k)} \\ \mathbf{y} - \mathbf{y}^{(k)} \end{pmatrix}$$

The Kuhn-Tucker conditions of  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  imply that

$$\nabla_x f^{(k)} + \nabla_x \mathbf{c}^{(k)} \boldsymbol{\lambda}^{(k)} + \boldsymbol{\mu}_x^{(k)} = \mathbf{0}.$$

Hence

$$\eta \geq f^{(k)} + \boldsymbol{\lambda}^{(k)T} \mathbf{c}^{(k)} + [\nabla_y f^{(k)} + \nabla_y \mathbf{c}^{(k)} \boldsymbol{\lambda}^{(k)} + \boldsymbol{\mu}_y^{(k)}]^T (\mathbf{y} - \mathbf{y}^{(k)})$$

is a Benders cut which underestimates  $v(\mathbf{y})$ . The Kuhn-Tucker conditions of  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  imply that the subgradient in this last cut is equal to the multipliers of the constraint which fixes  $\mathbf{y} = \mathbf{y}^{(k)}$ .  $\square$

Now we can prove the main convergence result.

**Theorem 3.1** Benders Decomposition with feasibility restoration converges to an optimal solution.

**Proof:** Lemma 3.2 implies that a limit point exists and is feasible.

Let  $m < \infty$  be an upper bound on  $\|\boldsymbol{\lambda}\|_1$  and the size of the subgradients  $\|\mathbf{w}^{(k)}\|_1, \|\hat{\mathbf{w}}^{(k)}\|_1$ . This bound exists by Lemma 3.1 and is finite.

Lemma 3.4 ensures that any Benders cut derived from  $(R(\mathbf{y}^{(k)}, m_1 \hat{\mathbf{c}}^{(k)+}))$  underestimates the value function  $v(\mathbf{y})$ . Thus the value of  $(M^k)$  always underestimates that of  $(P)$ .

Now assume that there exists a converging subsequence  $\mathbf{y}^{(k)}$  which converges to a nonstationary point  $\hat{\mathbf{y}}$  and seek a contradiction. From Lemma 3.2 it follows that  $\hat{\mathbf{y}}$  is feasible. Let  $\hat{f} = v(\hat{\mathbf{y}})$  be the corresponding objective value. Then it follows that  $\hat{f} > f^*$ , the optimal value. Now assume that the two objective values are sufficiently different, i.e. that  $\hat{f} > f^* + \epsilon_1 m$  (the algorithm does not discriminate between objective values which are closer).

The Benders cut added at each iteration has the form

$$\eta \geq f^{(k)} + \boldsymbol{\lambda}^{(k)T} \mathbf{c}^{(k)} + \hat{\mathbf{w}}^{(k)T} (\mathbf{y} - \mathbf{y}^{(k)}) \geq f^{(k)} - m \|\mathbf{c}^{+(k)}\|_1 - m \|\mathbf{y} - \mathbf{y}^{(k)}\|_1$$

if  $\mathbf{y}^{(k)}$  is infeasible and

$$\eta \geq f^{(k)} + \mathbf{w}^{(k)T} (\mathbf{y} - \mathbf{y}^{(k)}) \geq f^{(k)} - m \|\mathbf{y} - \mathbf{y}^{(k)}\|_1$$

if it is feasible. Now take the limit as  $k \rightarrow \infty$  and observe that the infeasibility  $\mathbf{c}^{+(k)} \rightarrow \epsilon_1$  as the auxiliary problem approaches feasibility. Hence in the limit

$$\eta \geq \lim_{k \rightarrow \infty} f^{(k)} - m\epsilon_1 = \hat{f} - m\epsilon_1 > f^*$$

by continuity of the value function. But this contradicts the fact that the master programs  $(M^k)$  are underestimators of the optimal value  $f^*$ .  $\square$

**Remark 3.3** Note that the proof only shows that  $\hat{f} > f^* + \epsilon_1 m$  cannot occur. Unfortunately, as noted in Remark 3.1, it is possible for  $\epsilon_1 m$  to be of order 1.

### 3.1 Relevance to bundle-based decomposition

Bundle-based decomposition methods build up a bundle of  $\epsilon$ -subgradients of  $v(\mathbf{y})$  and  $v_0(\mathbf{y})$ . This bundle defines a local piecewise linear model of the nonsmooth master problem  $(M)$ . This model is then minimized inside a trust-region. The step from the model is either accepted in which case the algorithm moves to a new point, discarding (part of) the bundle or the step is rejected in which case a new cut may be added to the bundle or the trust-region is reduced.

There exist various bundle schemes (e.g. [9], [4], [8] and references therein) which could be adopted to solve  $(P)$  by a decomposition scheme. Since bundle methods use the same cuts as Benders Decomposition, the example of Section 2 and the feasibility restoration introduced above are clearly relevant in this situation. Our development shows that it is necessary to include a feasibility restoration in bundle-based decomposition schemes and indicates how this can be done.

## 3.2 Further discussion

### 3.2.1 Bundle Methods

Bundle based decomposition schemes are *not* directly applicable to the example in Section 2 since the subdifferential at  $y = e$  is *not* bounded. Moreover, bundle-methods for nonsmooth optimization usually assume that a subgradient to *both*  $v(\mathbf{y})$  and  $v_0(\mathbf{y})$  can be computed at every iterate. This is clearly not the case, if these functions represent value functions of NLPs which may be infeasible.

### 3.2.2 Penalty function approach

We could write problem  $(P)$  in terms of an exact penalty function. Assuming a sufficiently large penalty parameter  $\mu > 0$  is known,  $(P)$  is equivalent to

$$(P_1) \begin{cases} \underset{x,y}{\text{minimize}} & f(\mathbf{x}, \mathbf{y}) + \mu \|\mathbf{c}^+(\mathbf{x}, \mathbf{y})\|_1 \\ \text{subject to} & (\mathbf{x}, \mathbf{y}) \in X \times Y. \end{cases}$$

This problem avoids the pitfalls of the Section 2. Moreover, assumption **A3** implies that  $\mu$  is bounded ( $\mu = 10$  is fine for our example). We could easily formulate a Benders approach following for example [2] and iterating in the following way.

Choose  $\mu > 0$  as initial penalty parameter.

**REPEAT**

Solve problem  $(P_1)$  by Benders decomposition.

**IF**  $(\mathbf{x}^*, \mathbf{y}^*)$  is infeasible **THEN**

Increase the penalty parameter  $\mu = 10\mu$

**ELSE**

Current solution is optimal.

**ENDIF**

**UNTIL** optimal

1. This is Roger's approach to finding a suitable penalty parameter. In our case it is guaranteed to work (as  $\mu$  is bounded by assumption **A3**) in a finite number of outer iterations.
2. Note that you cannot "estimate" the penalty parameter from solving the subproblems. In our example,  $y = e$  gives "infinite" multipliers. So following the usual penalty function theory we would deduce that the penalty parameter  $\mu = \infty$  which is wrong! On the other hand, if we used the above iteration to deduce a penalty parameter for every subproblem, then again, near  $y = e$  we may end up increasing  $\mu$  to infinity (and beyond).

3. I am not sure that adding a constraint like

$$\mathbf{c}(\mathbf{x}, \mathbf{y}) \leq \epsilon_1 \mathbf{e}$$

to the penalty problem ( $P_1$ ) helps us much. This problem will again have multipliers which can potentially tend to infinity as  $\epsilon_1$  is reduced. So no useful information can be gleaned from here.

4. There seems to be a paradox here. On the one hand, the overall problem has perfectly bounded multipliers and a bounded penalty parameter. On the other hand, some subproblems have unbounded multipliers. Is this maybe an argument against decomposition (at least at the NLP level)?
5. Andreas' augmented Lagrangian approach is another alternative. If you have an augmented Lagrangian term, then you do not enforce  $\mathbf{y} = \hat{\mathbf{y}}$  in the subproblems and all subproblems are then feasible again by assumption **A3**.

## References

- [1] Bazaraa, M.S., Sherali, H.D. and Shetty, C.M. *Nonlinear Programming: Theory and Algorithms*. John Wiley & Sons, Inc., New York, 1993.
- [2] Fletcher, R. and Leyffer, S. Solving mixed integer nonlinear programs by outer approximation. *Mathematical Programming*, 66:327–349, 1994.
- [3] Fletcher, R. and Leyffer, S. Nonlinear programming without a penalty function. Numerical Analysis Report NA/171, Department of Mathematics, University of Dundee, September 1997. (Submitted to *Mathematical Programming*).
- [4] Fletcher, R. and Leyffer, S. A bundle filter method for nonsmooth nonlinear optimization,. Numerical Analysis Report NA/195, Dundee University, April 1999.
- [5] Flippo, O.E. and Rinnoy Kan, A.H.G. Decomposition in general mathematical programming. *Mathematical Programming*, 60:361–382, 1993.
- [6] J Gauvin. A necessary and sufficient regularity condition to have bounded multipliers in nonconvex programming. *Mathematical Programming*, 12:136–138, 1977.
- [7] Geoffrion, A.M. Generalized Benders decomposition. *Journal of Optimization Theory and Applications*, 10:237–260, 1972.

- [8] Grothey, A. and McKinnon, K.I.M. A superlinearly convergent trust region bundle method. Technical report MS-98-015, Department of Mathematics & Statistics, Edinburgh University, December 1998.
- [9] K.C. Kiwiel. A constraint linearization method for nondifferentiable convex minimization. *Numerische Mathematik*, 51:395–414, 1987.