already quite early on in the solution process. The basis for this behaviour is the congestion effects that imply that several OD pairs need more than one route to have a positive volume at the solution; this means that the optimal link volume is not an extreme point, and the solutions to (12.2) will zig-zag between assigning the total volume onto these routes.

12.6 Active set methods

Reduced gradient, Newton, MINOS

Manifold optimization—constrained Newton (Bertsekas)

Leyffer

12.7 *Algorithms given by closed maps

Previously in this chapter we have seen convergence theorems for a sample of algorithms for the optimization problem (12.1). While we did not stress it, their convergence hinges on a special property of the map (or, mapping) that describes the various steps defining a complete iteration of the algorithm; it is this property that we shall here study in detail. In so doing, we will not only provide alternative proofs of some convergence theorems established previously, but we will also provide a framework for proving convergence theorems for new algorithms.

12.7.1 Algorithmic maps

Consider the simplest algorithm from this and the previous chapter: the steepest descent method using an exact line search. Given an iterate $\boldsymbol{x}_k \in \mathbb{R}^n$ we can describe an iteration of this algorithm as follows: Let the search direction be defined by $\boldsymbol{p}_k := -\nabla f(\boldsymbol{x}_k)$; next, let $\alpha_k \geq 0$ be a step length for which $f(\boldsymbol{x}_k + \alpha \boldsymbol{p}_k)$ is minimal over all $\alpha \geq 0$; finally, let $\boldsymbol{x}_{k+1} := \boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k$. This iteration, or algorithmic map, as we shall call it, can then be described as the composition of maps two, namely the construction of the search direction, followed by an exact line search:

$$\boldsymbol{x}_{k+1} := E(D(\boldsymbol{x}_k)),$$

where

$$\begin{aligned} D(\boldsymbol{x}) &:= \{\, \boldsymbol{p} \in \mathbb{R}^n \mid \boldsymbol{p} = -\nabla f(\boldsymbol{x}) \,\}, \\ E(\boldsymbol{x}, \boldsymbol{p}) &:= \left\{ \boldsymbol{y} = \boldsymbol{x} + \alpha \boldsymbol{p} \in \mathbb{R}^n,\ \alpha \geq 0 \mid f(\boldsymbol{y}) = \min_{\ell \geq 0} f(\boldsymbol{x} + \ell \boldsymbol{p}) \right\}. \end{aligned}$$

Composite maps like $E(D)$ above will, for simplicity, be written as ED.

Formally, suppose that $X \subset \mathbb{R}^n$, $Z \subset \mathbb{R}^p$ and $Y \subset \mathbb{R}^q$, and consider the point-to-set maps $B : X \to Z$ and $C : Z \to Y$. The composite map $A = CB$ then is the point-to-set map $A : X \to Y$ with

$$A(\boldsymbol{x}) = \cup\{\, C(\boldsymbol{z}) \mid \boldsymbol{z} \in B(\boldsymbol{x}) \,\}.$$

The map D is simple enough; if $f \in C^1(\mathbb{R}^n)$ then D is a continuous point-to-point map. The case of the exact line search map E is more difficult, because the minimizing step length α need not be unique (unless f is strictly convex, cf. Proposition 4.11). In general, therefore, the line search map E is not a point-to-point map, but a point-to-set map.

A point-to-set map of type D arising in this chapter is the one describing the Frank–Wolfe subproblem, in which one assigns to an iterate $\boldsymbol{x}_k$ an optimal solution to a linear approximation of the original problem through a first-order Taylor expansion of f around $\boldsymbol{x}_k$ (cf. Section 12.2):

$$\begin{aligned} D(\boldsymbol{x}) := \{\, \boldsymbol{p} \in \mathbb{R}^n \mid \boldsymbol{p} = \boldsymbol{y} - \boldsymbol{x},\ \boldsymbol{y} \in X,& \\ \text{and } \nabla f(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{y} \leq \nabla f(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{z}, \forall \boldsymbol{z} \in X \,\}.& \end{aligned}$$

Since linear programs are not always guaranteed to have unique optimal solutions, the Frank–Wolfe algorithm is in fact then described by the composition of two point-to-set maps of type D and E, respectively.

Let us next look at algorithmic maps for a small example problem.

Example 12.6 (algorithmic maps for a small example) Consider the problem of minimizing $f(x) := \frac{1}{2}x^2$ over $\mathbb{R}$. Consider the following three algorithmic maps:

(a) $A_1(x) = \frac{1}{2}x$, $x \in \mathbb{R}$;

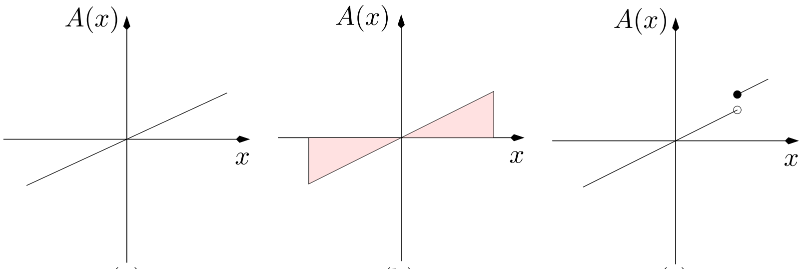
(b) $A_2(x) = \begin{cases} [\frac{1}{2}x, 0], & \text{if } x < 0, \\ [0, \frac{1}{2}x], & \text{if } x \geq 0, \end{cases}$; and

(c) $A_3(x) = \begin{cases} \frac{1}{2}x, & \text{if } x < 2, \\ 1 + \frac{1}{2}x, & \text{if } x \geq 2. \end{cases}$

Figure 12.4 illustrates the three maps.

Each figure shows the *graph* of a map A, namely a set of the form $\{\, (x, y) \in \mathbb{R} \times \mathbb{R} \mid y \in A(x) \,\}$. An algorithm then is constructed thus: choose $x_0 \in \mathbb{R}$, and let $x_{k+1} \in A(x_k)$ for $k = 1, 2, \ldots$.

In all three cases, it holds that $f(x_{k+1}) < f(x_k)$ if $x_k \neq 0$, that is, each algorithm produces sequences of descending objective values. The first two maps induce convergent algorithms, as for any starting



(a) (b) (c)

Figure 12.4: Three algorithmic maps.

point x_0 the sequences generated converge to $x^* = 0$. In case (a), the sequence is given once x_0 is, while in case (b) there is an infinite number of possible sequences starting at x_0 and converging to $x^* = 0$. The third mapping does not induce a convergent algorithm, because for starting points $x_0 > 2$ the sequence converges to 2, which is not an optimal solution. ■

Summarizing the above, we see that the first two algorithms converge, while the algorithm given in (c) does not converge in general; its convergence depends on the starting point. It is also the case that they are all descent algorithms, but as case (c) shows, that property alone does not imply that an algorithm is convergent.

The algorithms are of course constructed especially for the problem instance given. Typically when designing algorithms based on algorihmic maps one is interested in designing maps for which convergence is induced for a larger class of problems and for any starting point. We next turn to the important property of the closedness of a mapping, which is present in cases (a) and (b) above, but not in case (c), thus explaining the latter algorithm's failure.

12.7.2 Closed maps

We recall that the concept of a closed map already appears in Proposition 6.17(b), there described as an intrinsic property of the subgradient mapping of a convex function. We repeat the definition for ease of access.

Definition 12.7 (closed map) *Let $Z \subseteq \mathbb{R}^p$ and $Y \subseteq \mathbb{R}^q$ be non-empty and closed sets. Let $A : Z \to Y$ be a point-to-set map, that is, a map for which $Z(\boldsymbol{z}) \subseteq Y$ holds for every $\boldsymbol{z} \in Z$. Then, we say that the map*

A is closed at $\boldsymbol{z} \in Z$ *if for any sequences* $\{\boldsymbol{z}_k\}$ *and* $\{\boldsymbol{y}_k\}$ *satisfying*

$$\begin{aligned} \boldsymbol{z}_k \in Z, &\qquad \boldsymbol{z}_k \to \boldsymbol{z}, \\ \boldsymbol{y}_k \in A(\boldsymbol{z}_k), &\qquad \boldsymbol{y}_k \to \boldsymbol{y}, \end{aligned}$$

we have that

$$\boldsymbol{y} \in A(\boldsymbol{z}).$$

We say that A is closed on $Z_0 \subseteq Z$ *if it is closed at each point in* Z_0. ■

The meaning of the term *closed* is best understood through an example. Suppose that $Z = [a, b]$ and $Y = [c, d]$ where a, b, c, and d are reals with $a < b$ and $c < d$. Then, for each $z \in Z$ the value of $A(z)$ is a sub-interval of $[b, c]$. Consider then sequences $\{z_k\} \subseteq Z$ and $\{y_k\} \subseteq Y$ such that $y_k \in A(z_k)$ for all k, and limit points z and y, respectively. Figure 12.5 illustrates that a closed map cannot "shrink"; at the point z given, the map A is not continuous, but the set $A(z)$ is larger than sets $A(w)$ for points w near w.

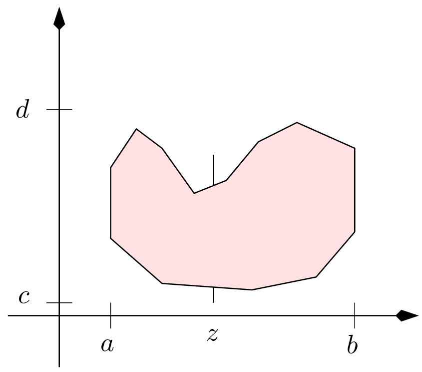


Figure 12.5: The graph of a closed map.

The graph of A is the set $\{\, (z, y) \mid z \in Z,\ y \in A(z) \,\}$. Notice that this set is closed when A is a closed map; this is in fact a characterization of closed maps, and the reason for its name.

Notice then that the map A in case (c) of Example 12.6 is not closed at $x = 2$.

The closedness of a map is a property not only important for the analysis of algorithms. It is also an important stability property of, for example, sets of optimal solutions to parametric optimization problems. The next example illustrates this.

Theorem 12.8 (solution sets of linear programs are closed maps) *Let $X \subseteq \mathbb{R}^n$ be a nonempty polyhedron. Consider a sequence $\{\boldsymbol{c}_k\}$ of cost vectors $\boldsymbol{c}_k \in \mathbb{R}^n$, and let $\boldsymbol{c} \in \mathbb{R}^n$ be a limit point. Correspondingly, let for each k*

$$\boldsymbol{x}_k \in X^*(\boldsymbol{c}_k) := \{\, \boldsymbol{x}^* \in \mathbb{R}^n \mid \boldsymbol{x}^* \text{ minimizes } \boldsymbol{c}_k^{\mathrm{T}}\boldsymbol{x} \text{ over } \boldsymbol{x} \in X \,\}.$$

Then, any limit point of $\{\boldsymbol{x}_k\}$ is in $X^*(\boldsymbol{c})$, that is, any limit of the sequence $\boldsymbol{x}_k$ of optimal solutions solves the limit LP problem. In other words, X^* is a closed mapping on $\mathbb{R}^n$.

Proof. We have that $\boldsymbol{c}_k^{\mathrm{T}}\boldsymbol{x}_k \leq \boldsymbol{c}_k^{\mathrm{T}}\boldsymbol{y}$ for every $\boldsymbol{y} \in X$. Letting $k \to \infty$ then yields that $\boldsymbol{c}^{\mathrm{T}}\boldsymbol{x} \leq \boldsymbol{c}^{\mathrm{T}}\boldsymbol{y}$ for every $\boldsymbol{y} \in X$, that is, $\boldsymbol{x} \in X(\boldsymbol{c})$. ∎

One can in fact prove a stronger result still: there exists a neighbourhood $N(\boldsymbol{c})$ of $\boldsymbol{c}$ such that $X^*(\boldsymbol{d}) \subset X^*(\boldsymbol{c})$ for every $\boldsymbol{d} \in N(\boldsymbol{c})$. As applied to the above setting it implies that $\boldsymbol{x}_k \in X^*(\boldsymbol{c})$ already after a finite integer k, and not just in the limit. Since this result requires more theory about polyhedral sets, notably the concept of *faces*, we provide references to the literature in the Notes section.

In the remainander of the section we establish the convergence of descent algorithms described by closed algorithmic maps

12.7.3 A convergence theorem

Lemma 12.9 (convergence of monotone sequences) *Let $m : X \to \mathbb{R}$ be a continuous function. Suppose there exists a sequence $\{\boldsymbol{x}_k\}$ in X such that*

(1) $m(\boldsymbol{x}_{k+1}) \leq m(\boldsymbol{x}_k)$ for all k, and

(1) $\lim_{k \in \mathcal{K}} \boldsymbol{x}_k = \boldsymbol{x}^\infty$, for some subsequence $\mathcal{K} \subset \mathcal{Z}_+$.

Then

$$\lim_{k\to\infty} m(\boldsymbol{x}_k) = \lim_{k \in \mathcal{K}} m(\boldsymbol{x}_k) = m(\boldsymbol{x}^\infty).$$

Proof. the continuity of m ensures that

$$\lim_{k \in \mathcal{K}} m(\boldsymbol{x}_k) = m(\boldsymbol{x}^\infty). \tag{12.22}$$

From (1) the sequence $\{m(\boldsymbol{x}_k)\}$ is monotone, so for any $l \in \mathcal{Z}_+$ we have that

$$m(\boldsymbol{x}_l) \geq m(\boldsymbol{x}^\infty). \tag{12.23}$$

Using (12.22) and the definition of limit, given $\varepsilon > 0$, there exists a $k_\varepsilon \in \mathcal{K}$ such that for $\mathcal{K} \ni k \geq k_\varepsilon$

$$m(\boldsymbol{x}_k) - m(\boldsymbol{x}^\infty) < \varepsilon. \tag{12.24}$$

For $l \geq k_\varepsilon$, by (a) it holds that

$$m(\boldsymbol{x}_l) \leq m(\boldsymbol{x}_{k_\varepsilon}). \tag{12.25}$$

From (12.23)–(12.25) then follows that

$$|m(\boldsymbol{x}_l) - m(\boldsymbol{x}^\infty)| < \varepsilon$$

for all $l \geq k_\varepsilon$, and we are done. ∎

Before stating and proving the main result we need to introduce one further notion. We call a point $\boldsymbol{x}^*$ a *solution point* if it belongs to the *solution set*, the latter being denoted by Ω. In our context the set Ω can be the set of stationary points, the local or global optimal solutions to the optimization problem at hand, the pairs of primal–dual optimal solutions to a convex program, or the primal–dual pair of KKT vectors in a given problem. The main point is that the above algorithm will be devised and established for the convergence of sequences to this set Ω.

The function m introduced in the above lemma will be the function that monitors convergence; it will often be the objective function f but it could also be chosen as another measure of distance from the solution set, such as the norm of the gradient of f in the case of unconstrained optimization. Clearly, in the convergence result to follow we see that the solution set Ω and the function m are intimately connected.

Theorem 12.10 (convergence of a generic algorithm) *Let the point-to-set map $A : V \to V$ describe an algorithm that given $x_0 \in V$ generates the sequence $\{\boldsymbol{x}_k\}$. Also, let a solution set $\Omega \subset V$ be given. Suppose*

(1) all points $\boldsymbol{x}_k$ lie in a compact subset $X \subset V$;

(2) there exists a continuous function $m : V \to \mathbb{R}$ such that:

- *(i) if $\boldsymbol{x}$ is not a solution, then for any $\boldsymbol{y} \in A(\boldsymbol{x})$ it holds that $m(\boldsymbol{y}) < m(\boldsymbol{x})$;*
- *(ii) if $\boldsymbol{x}$ is a solution then either the algorithm stops or for any $\boldsymbol{y} \in A(\boldsymbol{x})$ it holds that $m(\boldsymbol{y}) \leq m(\boldsymbol{x})$;*

and

(3) the map A is closed at $\boldsymbol{x}$ if $\boldsymbol{x}$ is not a solution.

Then, either the algorithm terminates finitely at a solution, or the limit of any convergent subsequence is a solution. Further, for some $\boldsymbol{x}^* \in \Omega$ *it holds that* $m(\boldsymbol{x}_k) \to m(\boldsymbol{x}^*)$.

Proof. If the algorithm terminates, then it does so at a solution, cf. (2)(ii), so suppose that the sequence generated is infinite.

By (1) there must be a convergent subsequence $\mathcal{K}$ such that $\lim_{k\in\mathcal{K}} \boldsymbol{x}_k = \boldsymbol{x}^\infty$. By (2) we can invoke Lemma 12.9 to state that

$$\lim_{k\to\infty} m(\boldsymbol{x}_k) = m(\boldsymbol{x}^\infty). \tag{12.26}$$

Consider next the subsequence $\{\boldsymbol{x}_{k+1}\}_{k\in\mathcal{K}}$. By (1) there must be a subsequence $\mathcal{K}^1 \subset \mathcal{K}$ such that $\lim_{k\in\mathcal{K}^1} \boldsymbol{x}_{k+1} = \boldsymbol{x}^{\infty+1}$, and again from Lemma 12.9 we must have that

$$\lim_{k\to\infty} m(\boldsymbol{x}_{k+1}) = m(\boldsymbol{x}^{\infty+1}). \tag{12.27}$$

Equations (12.26) and (12.27) then yield that

$$m(\boldsymbol{x}^{\infty+1}) = m(\boldsymbol{x}^\infty). \tag{12.28}$$

To complete the proof we assume that $\boldsymbol{x}^\infty$ is not a solution, and proceed to establish a contradiction. It holds from the construction of the algorithm, and the above, that

$$\begin{aligned} \lim_{k\in\mathcal{K}^1} \boldsymbol{x}_k &= \boldsymbol{x}^\infty, \\ \boldsymbol{x}_{k+1} &\in A(\boldsymbol{x}_k), \qquad k \in \mathcal{K}^1, \end{aligned}$$

and

$$\lim_{k\in\mathcal{K}^1} \boldsymbol{x}_{k+1} = \boldsymbol{x}^{\infty+1}.$$

Then, by (3),

$$\boldsymbol{x}^{\infty+1} \in A(\boldsymbol{x}^\infty).$$

Since we assumed $\boldsymbol{x}^\infty$ is not a solution it follows from (2)(i) that

$$m(\boldsymbol{x}^{\infty+1}) < m(\boldsymbol{x}^\infty). \tag{12.29}$$

But the two statements (12.28) and (12.29) are contradictory, whence our assumption was wrong. We conclude that $\boldsymbol{x}^\infty$ is a solution.

The second part of the theorem follows from the above and (12.26). ∎

Corollary 12.11 (convergence to unique solutions) *Under the assumptions of the theorem, if Ω is the singleton $\boldsymbol{x}^*$ then the sequence $\{\boldsymbol{x}_k\}$ converges to it.*

Proof. Suppose that there exists a subsequence $\mathcal{K}$ such that for some $\varepsilon > 0$, $\|\boldsymbol{x}_k - \boldsymbol{x}^*\| > \varepsilon$ holds for every $k \in \mathcal{K}$. Then take a convergent subsequence $\mathcal{K}^1$ of $\mathcal{K}$, and assume that $\{\boldsymbol{x}_k\}_{k\in\mathcal{K}^1}$ converges to $\boldsymbol{x}^1$. By the first part of the theorem $\boldsymbol{x}^1 \in \Omega$ holds. But $\Omega = \{\boldsymbol{x}^*\}$, so $\boldsymbol{x}^1 = \boldsymbol{x}^*$ must hold, contradicting our first claim. ∎

12.7.4 Additional results on composite maps

We next provide additional results that will enable us to utilize the above theorem in establishing the convergence of concrete algorithms. Recall the examples in Section 12.7.1 showing that a typical algorithm is formed as a composite map, including one that describes the generation of a descent direction, and one that describes a line search. A convergence analysis will be simpler if we can perform it for each map separately. The below results provide examples of analyses for some simple maps, as well as that of a map composed by closed algorithmic maps.

Proposition 12.12 (closed composite maps) *Let $X \subset \mathbb{R}^n$, $Z \subset \mathbb{R}^p$, and $Y \subset \mathbb{R}^q$ be nonempty, closed sets. Let $B : X \to Z$ and $C : Z \to Y$ be point-to-set maps, and consider the composite map $A : X \to Y$ defined by $A = CB$. Suppose that B is closed at x and that C is closed at $B(\boldsymbol{x})$. Moreover, suppose that if $\boldsymbol{x}_k \to \boldsymbol{x}$ and $\boldsymbol{z}_k \in B(\boldsymbol{x}_k)$ then there exists a convergent subsequence of $\{\boldsymbol{z}_k\}$. Then the composite map A is closed at $\boldsymbol{x}$.*

Proof. Let $\boldsymbol{x}_k \to \boldsymbol{x}$, $\boldsymbol{y}_k \in A(\boldsymbol{x}_k)$, and $\boldsymbol{y}_k \to \boldsymbol{y}$. We must show that $\boldsymbol{y} \in A(\boldsymbol{x})$. By the definition of A for each k there exists $\boldsymbol{z}_k \in B(\boldsymbol{x}_k)$ with $\boldsymbol{y}_k \in C(\boldsymbol{z}_k)$. By assumption, there is a convergent subsequence $\{\boldsymbol{z}_k\}_{k\in\mathcal{K}}$ with limit $\boldsymbol{z}$. Since B is closed at $\boldsymbol{x}$, $\boldsymbol{z} \in B(\boldsymbol{x})$. Furthermore, since C is closed on $B(\boldsymbol{x})$, it is closed in particular at $\boldsymbol{z}$, and hence $\boldsymbol{y} \in C(\boldsymbol{z})$. Thus, $\boldsymbol{y} \in C(\boldsymbol{z}) \in CB(\boldsymbol{x}) = A(\boldsymbol{x})$, so A is indeed closed at $\boldsymbol{x}$. ∎

Corollary 12.13 (special cases) *Let $X \subset \mathbb{R}^n$, $Z \subset \mathbb{R}^p$, and $Y \subset \mathbb{R}^q$ be nonempty, closed sets. Let $B : X \to Z$ and $C : Z \to Y$ be point-to-set maps, and consider the composite map $A : X \to Y$ defined by $A = CB$.*

(a) *Suppose that B is closed at $\boldsymbol{x}$, C is closed on $B(\boldsymbol{x})$, and that Z is compact. Then, $A = CB$ is closed at $\boldsymbol{x}$.*

(b) *If B is continuous at $\boldsymbol{x}$ and C is closed on $B(\boldsymbol{x})$ then $A = CB$ is closed at $\boldsymbol{x}$.* ∎

The existence of a convergent subsequence of $\{\boldsymbol{z}_k\}$ (suffient conditions for which are given in the corollary) is essential for the above theorem; without this assumption, the composite map $A = CB$ may not be closed even though both B and C are closed. Exercise 12.10 illustrates this fact.

We can also utilize the above proposition to establish the closedness of simple arithmetic maps; because of their simplicity we may weaken the conditions somewhat compared to the above proposition. The proof is left as an exercise.

Proposition 12.14 (closed arithmetic maps) *Let $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^p$.*

(a) [the sum map] *Let $B : X \to Y$ and $C : X \to Y$ be point-to-set maps. The sum map $A = B + C$ is defined as*

$$A(\boldsymbol{x}) = \{\, \boldsymbol{b} + \boldsymbol{c} \mid \boldsymbol{b} \in B(\boldsymbol{x}),\ \boldsymbol{c} \in C(\boldsymbol{x}) \,\}.$$

Suppose that both B and C are closed at $\boldsymbol{x}$. Then the sum map A is closed at $\boldsymbol{x}$ under any of the following additional conditions:

(1) *Y is compact;*

(2) *either B or C is continuous at $\boldsymbol{x}$;*

(3) *If $\boldsymbol{x}_k \to \boldsymbol{x}$ and $\boldsymbol{b}_k \in B(\boldsymbol{x}_k)$, $k \in \mathcal{K}$, then there exists $\mathcal{K}^1 \subset \mathcal{K}$ such that $\lim_{k \in \mathcal{K}^1} \boldsymbol{b}_k = \boldsymbol{b}$.*

(b) [the inner-product map] *Let $B : X \to Y$ and $C : X \to Y$ be point-to-set maps. The inner-product map $A = B^{\mathrm{T}}C$ is defined as*

$$A(\boldsymbol{x}) = \{\, \boldsymbol{b}^{\mathrm{T}}\boldsymbol{c} \mid \boldsymbol{b} \in B(\boldsymbol{x}),\ \boldsymbol{c} \in C(\boldsymbol{x}) \,\}.$$

Suppose that that B and C are closed at $\boldsymbol{x}$. Then the inner-product map A is closed at $\boldsymbol{x}$ under any of the following additional conditions:

(1) *Y is compact;*

(2) *both B and C are continuous at $\boldsymbol{x}$;*

(3) *If $\boldsymbol{x}_k \to \boldsymbol{x}$, $\boldsymbol{b}_k \in B(\boldsymbol{x}_k)$, and $\boldsymbol{c}_k \in C(\boldsymbol{x}_k)$, $k \in \mathcal{K}$, then there exists $\mathcal{K}^1 \subset \mathcal{K}$ such that $\lim_{k \in \mathcal{K}^1} \boldsymbol{b}_k = \boldsymbol{b}$ and $\lim_{k \in \mathcal{K}^1} \boldsymbol{c}_k = \boldsymbol{c}$ hold.*

(c) [the scalar-vector product map] *Let $B : X \to V \subset \mathbb{R}$ and $C : X \to Y$ be point-to-set maps. The scalar-vector product map $A = B\dot{C}$ is defined as*

$$A(\boldsymbol{x}) = \{\, b\boldsymbol{c} \mid b \in B(\boldsymbol{x}),\ \boldsymbol{c} \in C(\boldsymbol{x}) \,\}.$$

Suppose that both B and C are closed at $\boldsymbol{x}$. Then the scalar-vector product map A is closed at $\boldsymbol{x}$ under any of the following additional conditions:

(1) *both V and Y are compact;*

(2i) *both B and C are continuous at $\boldsymbol{x}$;*

(2ii) *either B is continuous and Y is compact or C is continuous and V is compact.*

(3i) *If* $\boldsymbol{x}_k \to \boldsymbol{x}$ *and* $b_k \in B(\boldsymbol{x}_k)$*,* $k \in \mathcal{K}$*, then there exists* $\mathcal{K}^1 \subset \mathcal{K}$ *such that* $\lim_{k \in \mathcal{K}^1} b_k = b \neq 0$.

(3ii) *If* $\boldsymbol{x}_k \to \boldsymbol{x}$ *and* $\boldsymbol{c}_k \in C(\boldsymbol{x}_k)$*,* $k \in \mathcal{K}$*, then there exists* $\mathcal{K}^1 \subset \mathcal{K}$ *such that* $\lim_{k \in \mathcal{K}^1} \boldsymbol{c}_k = \boldsymbol{c} \neq \boldsymbol{0}^q$.

The above simple proposition helps us to analyze the convergence of an algorithmic map, through the analysis of it in terms of the maps composing it.

Several practical algorithms are devised such that two different algorithmic maps are used: one map, say C, is perhaps devised based on experience and is such that the merit function m at least does not increase but may otherwise involve quite arbitrary operations, while the other map, B, is closed and satisfies the convergence requirements of Theorem 12.10. While the overall (composite) map $A = CB$ may not be closed (and thus we cannot apply Theorem 12.10 directly) we establish next that the map A does converge. The implication of this proposition is that it is possible to establish the convergence of complex algorithms that only occasionally (but enough often) are defined by a closed descent map. In algorithms of this type the map A is referred to as the *spacer step.*

Proposition 12.15 (convergence of algorithms based on spacer steps) *Let* $X \subset \mathbb{R}^n$ *be a closed set, and let a solution set* $\Omega \subset X$ *be given. Let* $m : \mathbb{R}^n \to \mathbb{R}$ *be a continuous function, and consider the following two maps.*

(1) $C : X \to X$ *is a point-to-set map that satisfies* $m(\boldsymbol{y}) \leq m(\boldsymbol{x})$ *for every* $\boldsymbol{x}$ *and* $\boldsymbol{y} \in C(\boldsymbol{x})$.

(2) $B : X \to X$ *is a point-to-set map that is closed over* $X \setminus \Omega$ *and satisfies* $m(\boldsymbol{y}) < m(\boldsymbol{x})$ *for every* $\boldsymbol{x} \in X \setminus \Omega$ *and* $\boldsymbol{y} \in B(\boldsymbol{x})$.

Consider the algorithm defined by the composite map $A = CB$*, with arbitrary starting point* $\boldsymbol{x}_0 \in X$*, and with* $\boldsymbol{x}_{k+1} \in A(\boldsymbol{x}_k)$*, unless* $\boldsymbol{x}_k \in \Omega$ *whence the algorithm stops. Suppose that the set* $M = \{ \boldsymbol{x} \in X \mid m(\boldsymbol{x}) \leq m(\boldsymbol{x}_0) \}$ *is bounded. Then, either the algorithm terminates finitely at a solution, or the limit of any convergent subsequence is a solution.*

Proof. If for any k $\boldsymbol{x}_k \in \Omega$ holds, then the algorithm stops. Suppose therefore that an infinite sequence is generated, and let $\{x_k\}_{k \in \mathcal{K}}$ denote a convergent subsequence with limit $\boldsymbol{x}$. Lemma 12.9 and the descent property of the map A imply that not only does $m(\boldsymbol{x}_k) \to m(\boldsymbol{x})$ hold in $\mathcal{K}$ but

$$\lim_{k \to \infty} m(\boldsymbol{x}_k) = m(\boldsymbol{x}). \tag{12.30}$$

We want to show that $\boldsymbol{x} \in \Omega$. By contradiction, suppose that $\boldsymbol{x} \notin \Omega$ and consider the sequence $\{\boldsymbol{x}_{k+1}\}_{k\in\mathcal{K}}$. By the definition of A we have that $\boldsymbol{x}_{k+1} \in C(\boldsymbol{y}_k)$ where $\boldsymbol{y}_k \in B(\boldsymbol{x}_k)$. Note that $\boldsymbol{y}_k \in M$ and $\boldsymbol{x}_{k+1} \in M$ holds. Since M is compact there is a convergent subsequence $\mathcal{K}^1 \subset \mathcal{K}$ with $\boldsymbol{y}_k \to \boldsymbol{y}$ and $\boldsymbol{x}_{k+1} \to \boldsymbol{x}^1$ in $\mathcal{K}^1$. Since B is closed at $\boldsymbol{x} \notin \Omega$, $\boldsymbol{y} \in B(\boldsymbol{x})$ and $m(\boldsymbol{y}) < m(\boldsymbol{x})$. Further, since $\boldsymbol{x}_{k+1} \in C(\boldsymbol{y}_k)$, by assumption we have that $m(\boldsymbol{x}_{k+1}) \leq m(\boldsymbol{y}_k)$ in $\mathcal{K}^1$; in the limit then $m(\boldsymbol{x}^1) \leq m(\boldsymbol{y})$, whence from $m(\boldsymbol{y}) < m(\boldsymbol{x})$ we obtain that $m(\boldsymbol{x}^1) < m(\boldsymbol{x})$. Since $m(\boldsymbol{x}_{k+1}) \to m(\boldsymbol{x}^1)$ in $\mathcal{K}^1$, this strict inequality contradicts (12.30). Therefore, $\boldsymbol{x} \in \Omega$. ∎

Since line searches are so often involved in the composition of an algorithmic map, we finally analyze the closedness of such operations. The following result applies to the exact line search and to the inexact line search rule due to Armijo [Arm66]. The below version of the Armijo rule is more general than the one provided in (11.11) in that an entire interval is specified; the rule given by (11.11) is in this sense a discrete version of the below rule (12.32) which at the same time produces a finite time procedure for generating an acceptable step length. Further note that normally $l = 0$ holds.

Proposition 12.16 (closedness of two line search maps) *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuous function. Also, let $I := [l, u]$ for some reals l and u with $l < u$.*

(a) *Let the exact line search map $E : \mathbb{R}^{2n} \to \mathbb{R}^n$ be given as follows: for given vectors $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{p} \in \mathbb{R}^n$,*

$$E(\boldsymbol{x}, \boldsymbol{p}) := \left\{ \boldsymbol{y} = \boldsymbol{x} + \alpha \boldsymbol{p} \in \mathbb{R}^n,\ \alpha \in I \mid f(\boldsymbol{y}) = \min_{\ell \in I} f(\boldsymbol{x} + \ell \boldsymbol{p}) \right\}. \tag{12.31}$$

Then, the map E is closed on $\mathbb{R}^{2n}$.

(b) *Let $f \in C^1$. Let the Armijo rule map $E_A : \mathbb{R}^{2n} \to \mathbb{R}^n$ be given as follows: assuming $\mu \in (0, 1)$, for given vectors $\boldsymbol{x} \in \mathbb{R}^n$ and $\boldsymbol{p} \in \mathbb{R}^n$,*

$$E(\boldsymbol{x}, \boldsymbol{p}) := \left\{ \boldsymbol{y} = \boldsymbol{x} + \alpha \boldsymbol{p} \in \mathbb{R}^n,\ \alpha \in I \mid f(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \mu\alpha \nabla f(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{p} \right\}. \tag{12.32}$$

Then, the map E_A is closed on $\mathbb{R}^{2n}$.

Proof. (a) Let $(\boldsymbol{x}_k, \boldsymbol{p}_k) \to (\boldsymbol{x}, \boldsymbol{p})$ in $\mathbb{R}^{2n}$ for some subsequence $\mathcal{K}$, and let further $\boldsymbol{y}_k \in E(\boldsymbol{x}_k, \boldsymbol{p}_k)$ for all k. Our task is to show that any limit $\boldsymbol{y}$ of $\{\boldsymbol{y}_k\}_{k\in\mathcal{K}}$ satisfies $\boldsymbol{y} \in E(\boldsymbol{x}, \boldsymbol{p})$. Because $\alpha_k \in I$ for all k and I is compact, $\{\alpha_k\}$ is bounded and has a limit point α^∞ in a convergent subsequence $\mathcal{K}^1 \subset \mathcal{K}$. It follows that $\alpha^\infty \in I$.

For any $\alpha \in I$, by the definition of $\boldsymbol{y}_k$ we have that $f(\boldsymbol{y}_k) \leq f(\boldsymbol{x}_k + \alpha_k \boldsymbol{p}_k)$. Due to the continuity of f, taking limits leads to the inequality $f(\boldsymbol{y}) \leq f(\boldsymbol{x} + \alpha \boldsymbol{p})$. Since this inequality holds for any $\alpha \in I$ it follows that for any $\boldsymbol{y}^\infty \in E(\boldsymbol{x}, \boldsymbol{p})$, $f(\boldsymbol{y}) \leq f(\boldsymbol{y}^\infty)$ holds.

On the other hand, because $\boldsymbol{y}^\infty \in E(\boldsymbol{x}, \boldsymbol{p})$ and $\boldsymbol{y} = \boldsymbol{x} + \alpha^\infty \boldsymbol{p}$ where $\alpha^\infty \in I$, $f(\boldsymbol{y}) \geq f(\boldsymbol{y}^\infty)$ also holds.

Hence, $\boldsymbol{y} \in E(\boldsymbol{x}, \boldsymbol{p})$, and we are done.

(b) Let $(\boldsymbol{x}_k, \boldsymbol{p}_k) \to (\boldsymbol{x}, \boldsymbol{p})$ in $\mathbb{R}^{2n}$ for some subsequence $\mathcal{K}$, and let further $\boldsymbol{y}_k \in E_A(\boldsymbol{x}_k, \boldsymbol{p}_k)$ for all k. Our task is to show that any limit $\boldsymbol{y}$ of $\{\boldsymbol{y}_k\}_{k \in \mathcal{K}}$ satisfies $\boldsymbol{y} \in E(\boldsymbol{x}, \boldsymbol{p})$. Because $\alpha_k \in I$ for all k and I is compact, $\{\alpha_k\}$ is bounded and has a limit point α^∞ in a convergent subsequence $\mathcal{K}^1 \subset \mathcal{K}$. It follows that $\alpha^\infty \in I$.

By the continuity assumptions on f and its gradient, it follows from taking the limit of $\mathcal{K}^1$ of the inequality (12.32) that $f(\boldsymbol{y}) \leq f(\boldsymbol{x}) + \mu \alpha^\infty \nabla f(\boldsymbol{x})^{\mathrm{T}} \boldsymbol{p}$. It follows that $\boldsymbol{y} \in E_A(\boldsymbol{x}, \boldsymbol{p})$. ■

12.7.5 Convergence of some algorithms defined by closed descent maps

We apply the above results to some of the algorithms of this and the last chapter.

Theorem 12.17 (convergence of the steepest descent algorithm) *Consider the problem to minimize $f(\boldsymbol{x})$ over $\boldsymbol{x} \in \mathbb{R}^n$, where $f : \mathbb{R}^n \to \mathbb{R}$ is in C^1. Suppose that we apply the steepest descent algorithm to this problem, where the line search is performed either exactly or according to the Armijo rule* (11.11). *Suppose further that the starting point x_0 is such that the level set $\operatorname{lev}_f(f(\boldsymbol{x}_0)) := \{\, \boldsymbol{x} \in \mathbb{R}^n \mid f(\boldsymbol{x}) \leq f(\boldsymbol{x}_0) \,\}$ is bounded. Then, the sequence $\{\boldsymbol{x}_k\}$ is bounded, the sequence $\{f(\boldsymbol{x}_k)\}$ is descending and lower bounded and therefore has a limit, and every limit point of $\{\boldsymbol{x}_k\}$ is stationary.*

Proof. With $\Omega = \{\, \boldsymbol{x} \in \mathbb{R}^n \mid \nabla f(\boldsymbol{x}) = \boldsymbol{0}^n \,\}$ and $m = f$ we wish to apply Theorem 12.10. That $\{\boldsymbol{x}_k\}$ lie in a compact set follows from the boundedness of $\operatorname{lev}_f(f(\boldsymbol{x}_0))$ and the continuity of f. The steepest descent algorithm is moreover a descent algorithm, under either one of the two step length rules used. It remains to establish that the algorithmic map is closed at every point $\boldsymbol{x} \notin \Omega$. The algorithmic map has the form $A = ED$ where $D(\boldsymbol{x}) = -\nabla f(\boldsymbol{x})$ is single-valued and continuous with respect to $\boldsymbol{x}$, and E is closed on $\operatorname{lev}_f(f(\boldsymbol{x}_0))$ according to Proposition 12.16. The composite map A is therefore also closed, thanks to the result of Corollary 12.13(b). ■

Theorem 12.18 (convergence of the Frank–Wolfe algorithm) *Consider the problem to minimize $f(\boldsymbol{x})$ over $\boldsymbol{x} \in X$, where $f : \mathbb{R}^n \to \mathbb{R}$ is in C^1 and $X \subset \mathbb{R}^n$ is a nonempty and bounded polyhedron. Suppose that we apply the Frank–Wolfe algorithm to this problem, according to its description in Section 12.2, where the line search is performed either exactly or according to the Armijo rule with maximum step length one. Then, the sequence $\{\boldsymbol{x}_k\}$ is bounded, the sequence $\{f(\boldsymbol{x}_k)\}$ is descending and lower bounded and therefore has a limit, and every limit point of $\{\boldsymbol{x}_k\}$ is stationary.*

Proof. The proof is left as Exercise 12.13. ∎

Theorem 12.19 (convergence of the gradient projection algorithm) *Consider the problem to minimize $f(\boldsymbol{x})$ over $\boldsymbol{x} \in X$, where $f : \mathbb{R}^n \to \mathbb{R}$ is in C^1 and $X \subset \mathbb{R}^n$ is a nonempty and bounded polyhedron. Suppose that we apply the gradient projection algorithm to this problem, according to its description in Section 12.4, where the line search is performed either exactly or according to the boundary Armijo rule presented therein. Then, the sequence $\{\boldsymbol{x}_k\}$ is bounded, the sequence $\{f(\boldsymbol{x}_k)\}$ is descending and lower bounded and therefore has a limit, and every limit point of $\{\boldsymbol{x}_k\}$ is stationary.*

Proof. The proof is left as Exercise 12.14. ∎

12.8 Notes and further reading

Algorithms for linearly constrained optimization problems are disappearing from modern text books on optimization. It is perhaps a sign of maturity, as we are now better at solving optimization problem with general constraints, and therefore do no longer have to especially consider the class of linearly constrained optimization problems. Nevertheless we feel that it provides a link between linear programming and nonlinear optimization problems with general constraints, being a subclass of nonlinear optimization problems for which primal feasibility can be retained throughout the procedure.

The Frank–Wolfe method was developed for QP problems in [FrW56], and later for more general problems, including non-polyhedral sets, in [Gil66] and [PsD78, Section III.3], among others. The latter source includes several convergence results for the method under different step length rules, assuming that ∇f is Lipschitz continuous, for example a Newton-type step length rule. The convergence Theorem 12.1 for the

Frank–Wolfe algorithm was taken from [Pat98, Theorem 5.8]. The convergence result for convex problems given in Theorem 12.2 is due to Dunn and Harshbarger [DuH78]. The version of the Frank–Wolfe algorithm produced by the selection $\alpha_k := 1/k$ is known as the *method of successive averages* (MSA).

The simplicial decomposition algorithm was developed in [vHo77]. Restricted simplicial decomposition methods have been developed in [HLV87, Pat98]. Note the strong relationships between the development of simplicial decomposition and that of the column generation principle in Section 10.6. Both are based on the generation of information from a relaxation to the original problem, followed by the solution of a restriction of the original problem which is iteratively enriched with the information generated from the relaxed problem. The subproblems are relaxations of the outer representation of the original problem; while simplicial decomposition utilizes a linearization as relaxation, column generation is based on Lagrangian relaxation. The restrictions are, in both approaches, based on the inner representation of the original formulation through the application of the Representation Theorem 8.9.

The gradient projection method presented here was first given in [Gol64, LeP66]; see also the textbook [Ber99]. Theorem 12.4 is due to [Ius03], while Lemma 12.5 is due to [BGIS95].

The traffic equilibrium models of Section 12.5 are described and analyzed more fully in [She85, Pat94].

The theory of closed maps underlying the development in Section 12.7 stems largely from Berge [Ber63]. Pioneering work on its application to nonlinear programming appears in Zangwill [Zan69], which is also the main source of inspiration for the text that appears here; some additional results stem from [BSS93]. Proposition 12.15 on spacer steps stems from [Lue84, p. 231], and several of its applications within nonlinear programming appear in [LPS96, Pat98].

Apart from the algorithms developed here, there are other classical algorithms for linearly constrained problems, including the reduced gradient method, Rosen's gradient projection method, active set methods, and other sub-manifold methods. They are not treated here, as some of them have fallen out of popularity. Reduced gradient methods still constitute the main building block of some commercial software, however.

12.9 Exercises

Exercise 12.1 (extensions of the Frank–Wolfe algorithm to unbounded sets) Develop an extension to the Frank–Wolfe algorithm applicable to cases where X is unbounded. Which steps need to be changed? What can go wrong?

(a) Show that the problem is convex.

(b) Apply one step of the Frank–Wolfe algorithm, starting at the origin. Provide an interval where f^* lies.

Exercise 12.4 (numerical example of the Frank–Wolfe algorithm) Consider the problem to

$$\begin{aligned} \text{maximize} \ & f(\boldsymbol{x}) := -x_1^2 - 4x_2^2 + 16x_1 + 24x_2, \\ \text{subject to} \ & x_1 + x_2 \leq 6, \\ & x_1 - x_2 \leq 3, \\ & x_1, x_2 \geq 0. \end{aligned}$$

(a) Show that the problem is convex.

(b) Solve the problem by using the Frank–Wolfe algorithm, starting at the origin.

Exercise 12.5 (numerical example of the Frank–Wolfe algorithm) Consider the problem to

$$\begin{aligned} \text{minimize} \ & f(\boldsymbol{x}) := \frac{1}{2}\left(x_1 - \frac{1}{2}\right)^2 + \frac{1}{2}x_2^2, \\ \text{subject to} \ & x_1 \leq 1, \\ & x_2 \leq 1, \\ & x_1, x_2 \geq 0. \end{aligned}$$

Apply two iterations of the Frank–Wolfe algorithm, starting at $\boldsymbol{x}_0 := (1,1)^{\mathrm{T}}$. Give upper and lower bounds on the optimal value.

Exercise 12.6 (convergence of a gradient projection algorithm) Establish Theorem 12.3.

Exercise 12.7 (numerical example of the simplicial decomposition algorithm) Solve the problem in Exercise 12.3 by using the simplicial decomposition algorithm.

Exercise 12.8 (numerical example of the simplicial decomposition algorithm) Solve the problem in Exercise 12.4 by using the simplicial decomposition algorithm.

Exercise 12.9 (numerical example of the simplicial decomposition algorithm) On the problem in Exercise 12.5 apply two iterations of the simplicial decomposition algorithm. Is $\boldsymbol{x}_2$ optimal? Why/why not?

Exercise 12.10 (non-closedness of composite maps) Consider the maps $B : \mathbb{R} \to \mathbb{R}$ and $C : \mathbb{R} \to \mathbb{R}$ given by

$$B(x) = \begin{cases} 1/x, & \text{if } x \neq 0, \\ 0, & \text{otherwise}, \end{cases}$$

and $C(z) = \{ y \in \mathbb{R} \mid |y| \leq |z| \}$. Show that the maps B and C both are closed on $\mathbb{R}$ but the composite map $A = CB$ is not, in particular not at $x = 0$.

Exercise 12.11 (closedness of maps) Let $X \subset \mathbb{R}^p$ and $Y \subset \mathbb{R}^q$, and let $A : X \to Y$ be a function. Suppose that if we were to interpret the function A as a map (where the image set then is a singleton) then it is closed at some $\boldsymbol{x} \in X$. Prove that if Y is closed and bounded then A is actually continuous at $\boldsymbol{x}$.

Exercise 12.12 (closedness of maps) Consider the LP problem to

$$\begin{aligned} \text{minimize } & \boldsymbol{c}^{\mathrm{T}}\boldsymbol{z} \\ \text{subject to } & \boldsymbol{A}\boldsymbol{z} = \boldsymbol{x}, \\ & \boldsymbol{z} \geq \boldsymbol{0}^n, \end{aligned}$$

where $\boldsymbol{A}$ is an $m \times n$ matrix, $\boldsymbol{c} \in \mathbb{R}^n$, and $\boldsymbol{x} \in \mathbb{R}^m$ is a parameter. Let $X^*(\boldsymbol{x})$ denote the set of optimal solutions to this LP problem for the given value of $\boldsymbol{x}$. Show that the mapping X^* is closed at $\boldsymbol{x}$ if the feasible set of the corresponding LP problem is nonempty and bounded.

Exercise 12.13 (convergence of the Frank–Wolfe algorithm) Establish Theorem 12.18.

Exercise 12.14 (convergence of the gradient projection algorithm) Establish Theorem 12.19.