For example, for $c \geq 1$, c is the order of convergence of the sequence $a^{(c^k)}$. We can also see that $1/k$ converges in order 1, although it is not linearly convergent, because $r^{k+1}/r^k \rightarrow 1$. Finally, $(1/k)^k$ converges in order 1, because for any $p > 1$, $r^{k+1}/(r^k)^p \rightarrow \infty$, However, this convergence is super-linear, because $r^{k+1}/r^k \rightarrow 0$.

D.3.3 The Gradient ParTan Algorithm

In this section we present the method of Parallel Tangents, ParTan, developed by Shah, Buehler and Kempthorne (1964) for solving the problem of minimizing an unconstrained convex function. We present a particular case of the General ParTan algorithm, the Gradient ParTan, following the presentation in Luenberger (1983).

The ParTan algorith was developed to solve exactly, after n steps, a general quadratic function $f(x) = x'Ax + b'x + c$. If A is real, symmetric and full rank matrix, it is possible to find the eigenvalue decomposition $V'AV = D = \text{diag}(d)$, see section F.2. If we had the eigen-vector matrix, V, we could consider the coordinate transformation $y = V'x$, $x = Vy$, $f(y) = y'V'AVy + b'Vy = y'Dy + e'y + c$. The coordinate transformation given by (the orthogonal) matrix V can be interpreted as a decoupling operator, see Chap.3, for it transforms an n-vector optimization problem into n independent scalar optimization problems, $y_i \in \arg\min d_i(y_k)^2 + e_i y_i + c$. However, finding the eigenvalue decomposition of A is even harder than solving the original optimization problem. A set of vectors (or directions), w^k is A-conjugate iff, for $k \neq j$, $(w^k)'Aw^j = 0$. A (non-orthogonal) matrix of n A-conjugate vectors, $W = [w^1 \ldots w^n]$ provides an alternative, and much cheaper decoupling operator for the quadratic optimization problem. The Partan algorithm finds, on the fly, a set of n A-conjugate vectors w^k.

To simplify the notation we assume, without loss of generality, a quadratic function that is centered at the origin, $f(x) = x'Ax$. Therefore, $\text{grad}(x) = Ay$, so that $y'Ax = y'\text{grad}(x)$, and vectors x and y are A-conjugate iff y is ortogonal to $\text{grad}(x)$. The Partan algorithm is defined as follows, progressing through points $x^0, x^1, y^1, x^2, \ldots x^{k-1}, y^{k-1}, x^k$, see Figure D.2 (left). The algorithm is initialized by choosing an arbitrary starting point, x^0, by an initial Cauchy step to find y^0, and by taking $x^1 = y^0$.

N-Dimensional (Gradient) ParTan Algorithm:

- Cauchy step: For $k = 0, 1, \ldots n$, find $y^k = x^k + \alpha_k g^k$ in an exact line search along the k-th steepest descent direction, $g^k = -\text{grad} f(x^k)$.

- Acceleration step: For $k = 1, \ldots n-1$, find $x^{k+1} = y^k + \beta_k(y^k - x^{k-1})$ in an exact line search along the k-th acceleration direction, $(y^k - x^{k-1})$.

In order to prove the correctness of the ParTan algorithm, we will prove, by induction, two statements:

(1) The directions $w^k = (x^{k+1} - x^k)$ are A-conjugate.

(2) Although the ParTan never performs the *conjugate direction line search*, $x^{k+1} = x^k + \gamma_k w^k$, this is what implicitly happens, that is, the point x^{k+1}, actually found at the acceleration step, would also solve the (hypothetical) conjugate direction line search.

The basis for the induction, $k = 1$, is trivially true. Let us assume the statements are true up to $k-1$, and prove the induction step for the index k, see Figure D.2 (right).

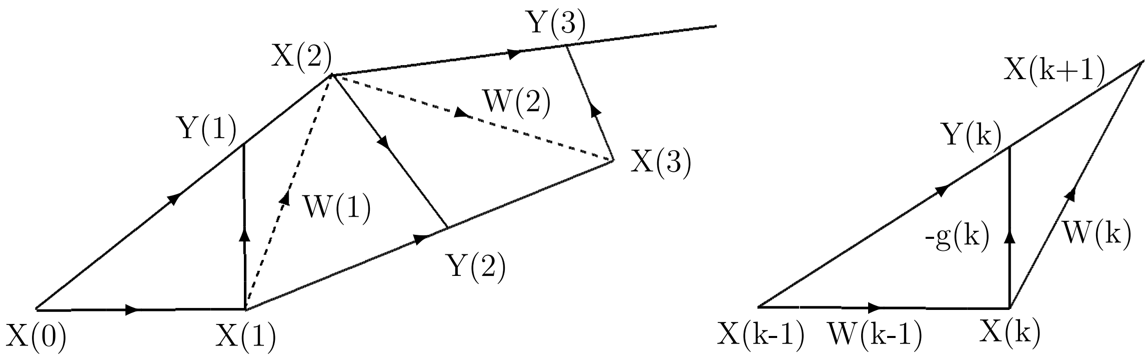

Figure D.2: The Gradient ParTan Algorithm.

By the induction hypothesis, x^k is the minimum of $f(x)$ on the k-dimensional hyperplane through x^0 spanned by all previous conjugate directions, w^j, $j < k$. Hence, $g^k = -\text{grad}f(x^k)$ is orthogonal to all w^j, $j < k$. All previous search directions lie in the same k-hyperplane, hence, g^k is also orthogonal to them. In particular, g^k is orthogonal to $g^{k-1} = -\text{grad}f(x^{k-1})$. Also, from the exact Cauchy step from x^k to y^k, we know that g^k must be orthogonal to $\text{grad}f(y^k)$. Since $\text{grad}f(x)$ is a linear function, it must be orthogonal to g^k at any point in the line search $x^{k+1} = y^k + \beta_k(y^k - x^{k-1})$. Since this line search is exact, $\text{grad}f(x^{k+1})$ is orthogonal to $(y^k - x^{k-1})$. Hence $\text{grad}f(x^{k+1})$ is orthogonal to any linear combination of g^k and $(y^k - x^{k-1})$, including w^k. For all other products $(w^j)'Aw^k$, w^j, $j < k-1$, we only have to write w^k as a linear combination of g^k and w^{k-1} to see that they vanish. This is enough to conclude the induction step of statements (1) and (2). QED.

Since a full rank matrix A can have at most n simultaneous A-conjugate directions, the Gradient ParTan must find the optimal solution of a quadratic function in at most n steps. This fact can be used to show that, if the quadratic model of the objective function is good, the ParTan algorithm converges quadratically. Nevertheless, even if the quadratic model for the objective function is poor, the Cauchy (steepest descent) steps can make good progress. This explains the Gradient ParTan robustness as an optimization algorithm, even if it starts far away from the optimal solution.

The ParTan needs two line searches in order to obtain each conjugate direction. Far away from the optimal solution a Cauchy method would use only one line search. Close to the optimal solution alternative versions of the ParTan algorithm, known as Conjugate Gradient algorithms, achieve quadratic convergence using only one line search per dimension. Nevertheless, in order to use these algorithms one has to devise a monitoring system that keeps track of how well the quadratic model is doing, and use it to decide when to make the transition from the Cauchy to the Conjugate Gradient algorithm. Hence, the Partanization of search directions provides a simple mechanism to upgrade an algorithm based on Cauchy (steepest descent) line search steps, accelerating it to achieve quadratic convergence, while keeping the robustness that is so characteristic of Cauchy methods.

D.3.4 Global Convergence

In this section we give some conditions that assures global convergence for a NLP algorithm. We follow the ideas of Zangwill (1964), similar analyses are presented in Luenberger (1984) and Minoux and Vajda (1986).

We define an Algorithm as an iterative process generating a sequence of points, $x^0, x^1, x^2 \ldots$, that oby a recursion equation of the form $x^{k+1} \in A_k(x^k)$, where the *point-to-set map* $A_k(x^k)$ defines the possible successors of x^k in the sequence.

The idea of using an point-to-set map, instead of a ordinary function or point-to-point map, allows us to study in a unified way a hole class of algorithms, including alternative implementations of several details, approximate or inexact computations, randomized steps, etc. The basic property we look for on the maps defining an algorithm is *closure*, defined as follows.

A point-to-set map from space X to space Y, is *closed* at x if the following condition holds: If a sequence x^k converges to $x \in X$, and the sequence y^k converges to $y \in Y$, where $y^k \in A(x)$, then the also the limit y is in the image $A(x)$, that is,

$$x^k \to x \ , \ y^k \to y \ , \ y^k \in A(x^k) \ \Rightarrow \ y \in A(x) \ .$$

The map is closed in $C \subseteq X$ if it is closed at any point of C. Note that if we replace, in the definition of closed map, the inclusion relation by the equality relation, we get the definition of continuity for point-to-point functions. Therefore, the closure property is a generalization of continuity. Indeed, a continuous function is closed, although the contrary is not necessarily true.

The basic idea of Zangwill's global convergence theorem is to find some characteristic that is continuously "improved" at each iteration of the algorithm. This characteristic is represented by the concept of *descendence function.*

Let A be an algorithm in X for solving the problem P, and let $S \subset X$ be the solution set for P. A function $Z(x)$ é is a descendence function for (X, A, S) if the composition of Z and A is always decreasing outside the solution set, and does not increase inside the solution set, that is,

$$x \notin S \wedge y \in A(x) \Rightarrow Z(y) < Z(x) \quad \text{and} \quad x \in S \wedge y \in A(x) \Rightarrow Z(y) \leq Z(x) \ .$$