Descent Algorithms

- A single iteration of the general descent algorithm consists of two main steps, *viz.*,
  1. determining a good descent direction $\Delta\mathbf{x}^{(k)}$, which is typically forced to have unit norm and
  2. determining the step size using some line search technique.
- If the function f is convex, from the necessary and sufficient condition for convexity restated here for reference:

$$f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)}) + \nabla^T f(\mathbf{x}^{(k)})(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$$

- We require that $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$ and since $t^{(k)} > 0$, we must have

$$\nabla^T f(\mathbf{x}^{(k)})\Delta\mathbf{x}^{(k)} < 0$$

  That is, the descent direction $\Delta\mathbf{x}^{(k)}$ must make (sufficiently) obtuse angle ($\theta \in \left(\frac{\pi}{2}, \frac{3\pi}{2}\right)$) with the gradient vector

- A natural choice of $\Delta\mathbf{x}^{(k)}$ that satisfies the above necessary condition is

Descent Algorithms

- A single iteration of the general descent algorithm consists of two main steps, *viz.*,
  1. determining a good descent direction $\Delta\mathbf{x}^{(k)}$, which is typically forced to have unit norm and
  2. determining the step size using some line search technique.
- If the function f is convex, from the necessary and sufficient condition for convexity restated here for reference:

$$f(\mathbf{x}^{(k+1)}) \geq f(\mathbf{x}^{(k)}) + \nabla^T f(\mathbf{x}^{(k)})(\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)})$$

- We require that $f(\mathbf{x}^{(k+1)}) < f(\mathbf{x}^{(k)})$ and since $t^{(k)} > 0$, we must have

$$\nabla^T f(\mathbf{x}^{(k)})\Delta\mathbf{x}^{(k)} < 0$$

  That is, the descent direction $\Delta\mathbf{x}^{(k)}$ must make (sufficiently) obtuse angle ($\theta \in \left(\frac{\pi}{2}, \frac{3\pi}{2}\right)$) with the gradient vector
- A natural choice of $\Delta\mathbf{x}^{(k)}$ that satisfies the above necessary condition is $\nabla f(\mathbf{x}^{(k)})$ (gradient descent algorithm)

Descent Algorithms (contd.)

```
Find a starting point $\mathbf{x}^{(0)} \in \mathcal{D}$
repeat
    1. Determine $\Delta\mathbf{x}^{(k)}$.
    2. Choose a step size $t^{(k)} > 0$ using ray[a] search.
    3. Obtain $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + t^{(k)}\Delta\mathbf{x}^{(k)}$.
    4. Set $k = k + 1$.
until stopping criterion (such as $||\nabla f(\mathbf{x}^{(k+1)})|| < \epsilon$) is satisfied
```

[a] Many textbooks refer to this as line search, but we prefer to call it ray search, since the step must be positive.

Figure 7: The general descent algorithm.

There are many different empirical techniques for ray search, though it matters much less than the search for the descent direction. These techniques reduce the $n-$dimensional problem to a $1-$dimensional problem, which can be easy to solve by use of plotting and eyeballing or even exact search.

Finding the step size t

- If t is too large, we get diverging updates of x
- If t is too small, we get a very slow descent
- We need to find a t that is *just right*
- We discuss two ways of finding t:
  1. Exact ray search
  2. Backtracking ray search

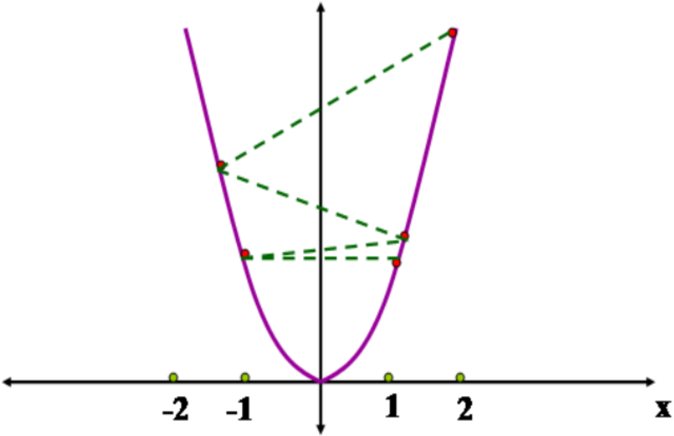
Ray search because we have already firmed up the descent directior
Hence t > 0

Example illustrating importance of line search

- If t is too large, we get diverging updates of x
- If t is too small, we get a very slow descent
- We need to find a t that is *just right*
- Eg: Let $f(x) = x^2$ for $x \in \Re$. Let $x^0 = 2$, $\Delta x^k = (-1)^k$ for all k (since it is a valid descent direction of $x > 0$) and $x^k = (-1)^k(1 + 2^{-k})$. What is the step size t^k implicitly being used?

Example illustrating importance of line search

- If t is too large, we get diverging updates of x
- If t is too small, we get a very slow descent
- We need to find a t that is *just right*
- Eg: Let $f(x) = x^2$ for $x \in \Re$. Let $x^0 = 2$, $\Delta x^k = (-1)^k$ for all k (since it is a valid descent direction of $x > 0$) and $x^k = (-1)^k(1 + 2^{-k})$. What is the step size t^k implicitly being used? The sequence *x^k does not converge.*



$$\{x\} : \{2, -\frac{3}{2}, \frac{5}{4}, -\frac{9}{8}, \ldots\}$$

$$\{f\} : \{4, \frac{9}{4}, \frac{25}{16}, \frac{81}{64}, \ldots\}$$

f certainly descreases (not sufficient though

Example illustrating importance of line search

- If t is too large, we get diverging updates of x
- If t is too small, we get a very slow descent
- We need to find a t that is *just right*
- Eg: Let $f(x) = x^2$ for $x \in \Re$. Let $x^0 = 2$, $\Delta x^k = (-1)^k$ for all k (since it is a valid descent direction of $x > 0$) and $x^k = (-1)^k(1 + 2^{-k})$. What is the step size t^k implicitly being used? The sequence x^k *does not converge*.
- We discussed two ways of determining t:
    1. Exact ray search
    2. Backtracking ray search

Exact ray search

$$t^{k+1} = \operatorname*{argmin}_t f\left(\mathbf{x}^k + t\Delta\mathbf{x}^k)\right)$$

$$= \operatorname*{argmin}_t \phi(t)$$

- This method gives the most optimal step size in the given descent direction $\Delta\mathbf{x}^k$
- It ensures that $f(x^{k+1}) \leq f(x^k)$. Why?

Exact ray search

$$t^{k+1} = \operatorname*{argmin}_t f\left(\mathbf{x}^k + t\Delta\mathbf{x}^k)\right)$$

$$= \operatorname*{argmin}_t \phi(t)$$

- This method gives the most optimal step size in the given descent direction $\Delta\mathbf{x}^k$
- It ensures that $f(x^{k+1}) \leq f(x^k)$. Why? Because $\phi(t^{k+1}) = f(\mathbf{x}^k + t^{k+1}\Delta\mathbf{x}^k) = \min_t \phi(t) = \min_t f\left(\mathbf{x}^k + t\Delta\mathbf{x}^k)\right) \leq \phi(0) = f(x^k)$
- **Homework1**: Consider the function

$$f(\mathbf{x}) = x_1^2 - 4x_1 + 2x_1x_2 + 2x_2^2 + 2x_2 + 14$$

This function has a minimum at $\mathbf{x} = (5, -3)$. Suppose you are at a point $(4, -4)^T$ after few iterations, and $\Delta\mathbf{x} = -\nabla f(\mathbf{x})$ at every $\mathbf{x}$, then using the **exact line search algorithm**, find the point for the next iteration. In how many steps will the algorithm converge?

Ray Search for Descent: Options

1. **Exact ray search:** The exact ray search seeks a scaling factor t that satisfies

$$t = \underset{t>0}{\mathrm{argmin}} f(\mathbf{x} + t\Delta\mathbf{x}) \tag{28}$$

Ray Search for Descent: Options

1. **Exact ray search:** The exact ray search seeks a scaling factor t that satisfies

$$t = \underset{t>0}{\operatorname{argmin}} f(\mathbf{x} + t\Delta\mathbf{x}) \tag{28}$$

2. **Backtracking ray search:** The exact line search may not be feasible or could be expensive to compute for complex non-linear functions. A relatively simpler ray search iterates over values of step size starting from 1 and scaling it down by a factor of $\beta \in (0, \frac{1}{2})$ after every iteration till the following condition, called the *Armijo condition* is satisfied for some $0 < c_1 < 1$.

sufficient decrease condition

$$f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + c_1 t \nabla^T f(\mathbf{x}) \Delta\mathbf{x} \tag{29}$$

Based on first order convexity condition, it can be inferred that when $c_1 = 1$,

this inequality will not hold (at c1 = 1)

Ray Search for Descent: Options

1. **Exact ray search:** The exact ray search seeks a scaling factor t that satisfies

$$t = \underset{t>0}{\operatorname{argmin}} f(\mathbf{x} + t\Delta\mathbf{x}) \tag{28}$$

2. **Backtracking ray search:** The exact line search may not be feasible or could be expensive to compute for complex non-linear functions. A relatively simpler ray search iterates over values of step size starting from 1 and scaling it down by a factor of $\beta \in (0, \frac{1}{2})$ after every iteration till the following condition, called the *Armijo condition* is satisfied for some $0 < c_1 < 1$.

$$f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + c_1 t \nabla^T f(\mathbf{x}) \Delta\mathbf{x} \tag{29}$$

Based on first order convexity condition, it can be inferred that when $c_1 = 1$, the right hand side of (29) gives a lower bound on the value of $f(\mathbf{x} + t\Delta\mathbf{x})$ and hence

Ray Search for Descent: Options

1. **Exact ray search:** The exact ray search seeks a scaling factor t that satisfies

$$t = \underset{t>0}{\text{argmin}} f(\mathbf{x} + t\Delta\mathbf{x}) \tag{28}$$

2. **Backtracking ray search:** The exact line search may not be feasible or could be expensive to compute for complex non-linear functions. A relatively simpler ray search iterates over values of step size starting from 1 and scaling it down by a factor of $\beta \in (0, \frac{1}{2})$ after every iteration till the following condition, called the *Armijo condition* is satisfied for some $0 < c_1 < 1$.

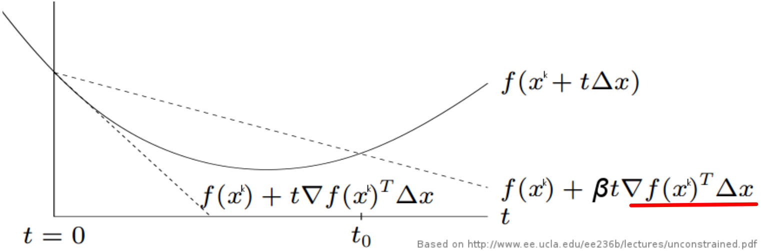
$$f(\mathbf{x} + t\Delta\mathbf{x}) \leq f(\mathbf{x}) + c_1 t \nabla^T f(\mathbf{x})\Delta\mathbf{x} \tag{29}$$

Based on first order convexity condition, it can be inferred that when $c_1 = 1$, the right hand side of (29) gives a lower bound on the value of $f(\mathbf{x} + t\Delta\mathbf{x})$ and hence (29) can never hold. The Armijo condition simply ensures that t decreases f sufficiently.

Backtracking ray search

- The algorithm
  - Choose a $\beta \in (0, 1)$
  - Start with $t = 1$
  - Until $f(\mathbf{x} + t\Delta\mathbf{x}) < f(\mathbf{x}) + c_1 t \nabla^T f(\mathbf{x}) \Delta\mathbf{x}$, do
    - Update $t \leftarrow \beta t$

Interpretation of backtracking line search

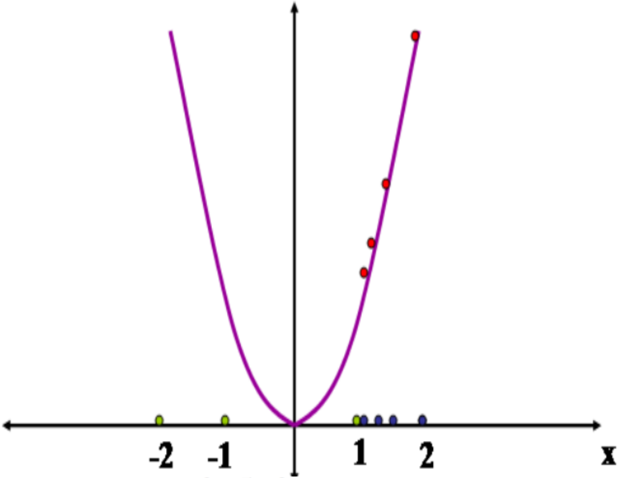


Based on http://www.ee.ucla.edu/ee236b/lectures/unconstrained.pdf

- $\Delta x =$ direction of descent $= -\nabla f(x^k)$ for gradient descent
- A different way of understanding the varying step size with β: Multiplying t by β causes the interpolation to tilt as indicated in the figure

Homework 2: Let $f(x) = x^2$ for $x \in \Re$. Let $x^0 = 2$, $\Delta x^k = -1$ for all k (since it is a valid descent direction of $x > 0$) and $x^k = 1 + 2^{-k}$. What is the step size t^k implicitly being used. While t^k satisifies the Armijo condition (determine a c_1) is this choice of step size ok?

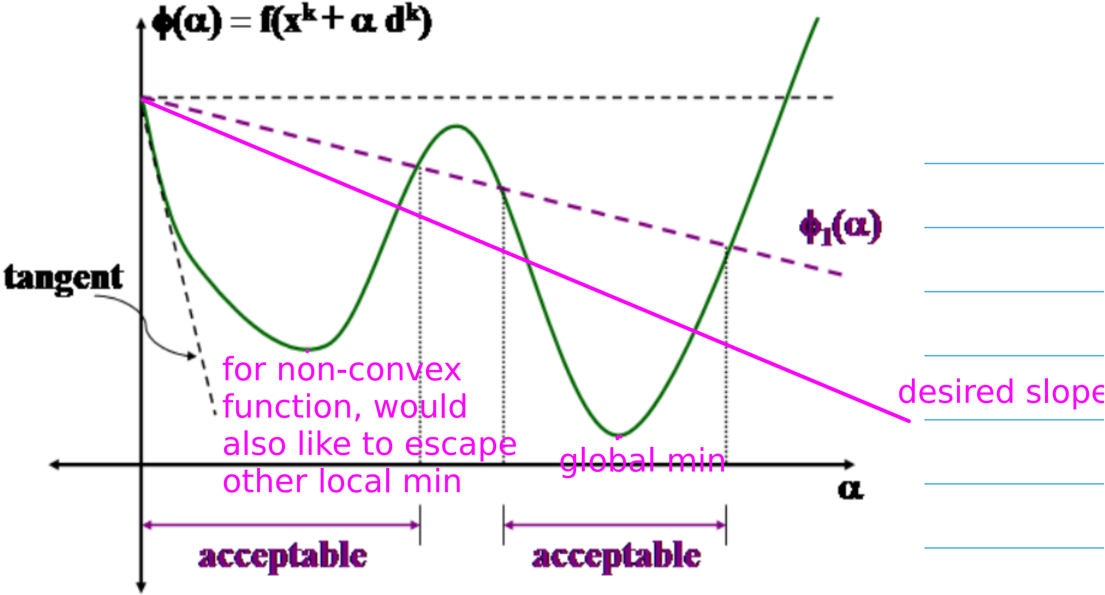
No: Convergence is to 1 (not to 0). Reason: Sufficient decrease is in terms of slope which might diminish



$$\{x\} : \{2, \frac{3}{2}, \frac{5}{4}, \frac{9}{8}, \ldots\}$$

$$\{f\} : \underline{\{4, \frac{9}{4}, \frac{25}{16}, \frac{81}{64}, \ldots\}}$$

tends to 1

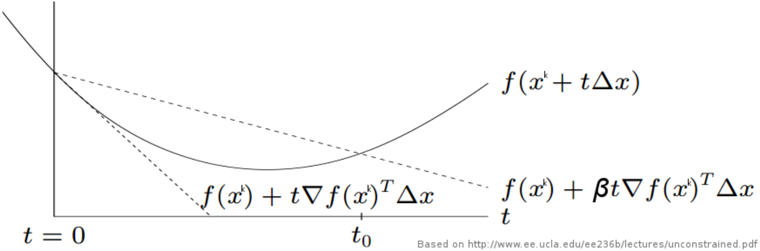


for non-convex function, would also like to escape other local min

global min

desired slope

Interpretation of backtracking line search


$f(x^k + t\Delta x)$
$f(x^k) + \beta t \nabla f(x^k)^T \Delta x$
$f(x^k) + t \nabla f(x^k)^T \Delta x$
t
$t = 0$
t_0

Based on http://www.ee.ucla.edu/ee236b/lectures/unconstrained.pdf

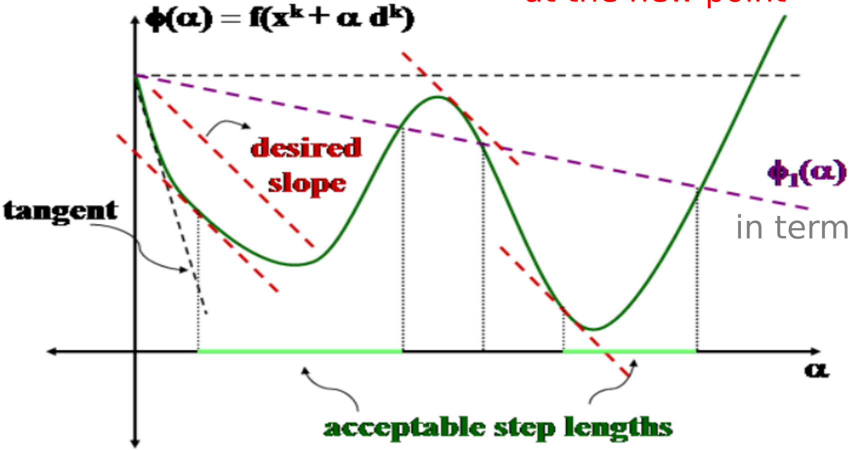
- $\Delta x =$ direction of descent $= -\nabla f(x^k)$ for gradient descent
- A different way of understanding the varying step size with β: Multiplying t by β causes the interpolation to tilt as indicated in the figure

Homework 2: Let $f(x) = x^2$ for $x \in \Re$. Let $x^0 = 2$, $\Delta x^k = -1$ for all k (since it is a valid descent direction of $x > 0$) and $x^k = 1 + 2^{-k}$. What is the step size t^k implicitly being used. While t^k satisifies the Armijo condition (determine a c_1) is this choice of step size ok? **We will motivate a second condition in the following slides.**

Ray Search for First Order Descent: Strong Wolfe Conditions

Wolfe's condition: The function should have a **sufficient rate of decrease**.

at the new point



in terms of c1

Ray Search for First Order Descent: Strong Wolfe Conditions

Wolfe's condition: The function should have a **sufficient rate of decrease**.

$$\underline{\left|\Delta \mathbf{x}^T \nabla f(\mathbf{x} + t\Delta \mathbf{x})\right|} \leq c_2 \left|\Delta \mathbf{x}^T \nabla f(\mathbf{x})\right| \tag{30}$$

where $1 > c_2 > c_1 > 0$. This condition ensures that the slope of the function $f(\mathbf{x} + t\Delta \mathbf{x})$ at t is less than c_2 times that at $t = 0$.

1. The conditions in **(29)** and **(30)** are together called the strong Wolfe conditions. These conditions are particularly very important for non-convex problems.
2. While **(29) ensures guaranteed decrease in $f(\mathbf{x} + \Delta \mathbf{x})$** in terms of the slope, **(30) provides**

Ray Search for First Order Descent: Strong Wolfe Conditions

Wolfe's condition: The function should have a **sufficient rate of decrease**.

$$\left|\Delta \mathbf{x}^T \nabla f(\mathbf{x} + t\Delta \mathbf{x})\right| \leq c_2 \left|\Delta \mathbf{x}^T \nabla f(\mathbf{x})\right| \tag{30}$$

where $1 > c_2 > c_1 > 0$. This condition ensures that the slope of the function $f(\mathbf{x} + t\Delta \mathbf{x})$ at t is less than c_2 times that at $t = 0$.

1. The conditions in **(29)** and **(30)** are together called the strong Wolfe conditions. These conditions are particularly very important for non-convex problems.
2. While **(29) ensures guaranteed decrease in $f(\mathbf{x} + \Delta \mathbf{x})$** in terms of the slope, **(30) provides guaranteed decrease in magnitude of slope and (indirectly) avoids too small steps**. Other conditions such as Goldstein more directly influence step size
3. Claim: If $1 > c_2 > c_1 > 0$ and the function $f(\mathbf{x})$ is convex and differentiable, there exists t such that (29) and (30) are both satisfied for any f. *Hint: Use the Mean Value Theorem*

Convexity $\Rightarrow$ Strong Wolfe Conditions

- Let $\phi(t) = f(\mathbf{x}^k + t\Delta\mathbf{x}^k) \geq f(\mathbf{x}^k) + t\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k$ (where the second inequality is by virtue of convexity). Remember that $\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k < 0$
- Since $0 < c_1 < 1$, the linear approximation $l(t) = f(\mathbf{x}^k) + tc_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k$ is unbounded below and it can be shown to intersect f for some t

Convexity $\Rightarrow$ Strong Wolfe Conditions

- Let $\phi(t) = f(\mathbf{x}^k + t\Delta\mathbf{x}^k) \geq f(\mathbf{x}^k) + t\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k$ (where the second inequality is by virtue of convexity). Remember that $\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k < 0$
- Since $0 < c_1 < 1$, the linear approximation $l(t) = f(\mathbf{x}^k) + tc_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k$ is unbounded below and it can be shown to intersect the graph of ϕ atleast once.
- Let $t' > 0$ be the smallest intersecting value of t, that is:

$$f(\mathbf{x} + t'\Delta\mathbf{x}^k) = f(\mathbf{x}^k) + t'c_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k \tag{31}$$

- For all $t \in [0, t']$,

l(t) must lie above the graph

Convexity ⇒ Strong Wolfe Conditions

- Let $\phi(t) = f(\mathbf{x}^k + t\Delta\mathbf{x}^k) \geq f(\mathbf{x}^k) + t\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k$ (where the second inequality is by virtue of convexity). Remember that $\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k < 0$
- Since $0 < c_1 < 1$, the linear approximation $l(t) = f(\mathbf{x}^k) + tc_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k$ is unbounded below and it can be shown to intersect the graph of ϕ atleast once.
- Let $t' > 0$ be the smallest intersecting value of t, that is:

$$f(\mathbf{x} + t'\Delta\mathbf{x}^k) = f(\mathbf{x}^k) + t'c_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k \tag{31}$$

- For all $t \in [0, t']$,

$$\underline{f(\mathbf{x}^k + t\Delta\mathbf{x}^k) \leq f(\mathbf{x}^k) + tc_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k} \tag{32}$$

  That is, there exists a non-empty set of t such that the first Wolfe condition is met.
- By the mean value theorem, $\exists\, t'' \in (0, t')$ such that

$$f(\mathbf{x}^k + t'\Delta\mathbf{x}^k) - f(\mathbf{x}^k) = t'\nabla^T f(\mathbf{x}^k + t''\Delta\mathbf{x}^k)\Delta\mathbf{x}^k \tag{33}$$

Convexity ⇒ Strong Wolfe Conditions (contd.)

- Combining (31) and (33), and using $c_1 < c_2$,and $\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k < 0$

$$\underline{\nabla^T f(\mathbf{x}^k + t''\Delta\mathbf{x}^k)\Delta\mathbf{x}^k = c_1\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k} > c_2\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k \quad (34)$$

- Again, since $\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k < 0$, we get the $t^k = t''$ satisfying (30)

$$|\nabla^T f(\mathbf{x}^k + t''\Delta\mathbf{x}^k)\Delta\mathbf{x}^k| < c_2|\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k| \quad (35)$$

- In fact, by continuity of $f(.)$, there exists an interval around t'' for which Strong Wolfe conditions hold.

Empirical Observations on Ray Search

- A finding that is borne out of plenty of empirical evidence is that exact ray search does better than empirical ray search in a few cases only. exact search is not often worth it
- Further, the exact choice of the value of c_1 and c_2 seems to have little effect on the convergence of the overall descent method.
- The trend of specific descent methods has been like a parabola - starting with simple steepest descent techniques, then accomodating the curvature hessian matrix through a more sophisticated Newton's method and finally, trying to simplify the Newton's method through approximations to the hessian inverse, culminating in conjugate gradient techniques, that do away with any curvature matrix whatsoever, and form the internal combustion engine of many sophisticated optimization techniques today.
- We start the thread by describing the steepest descent methods.

WE WILL NOW GO BACK TO OPTIONS FOR THE DESCENT DIRECTION

Algorithms: Steepest Descent

- The idea of steepest descent is to determine a descent direction such that for a unit step in that direction, the prediction of decrease in the objective is maximized
- However, consider $\Delta x = \text{argmin}_v \begin{bmatrix} -5 & 10 & 15 \end{bmatrix} v$

  gradient

  $$\implies \Delta x = \begin{bmatrix} \infty \\ -\infty \\ -\infty \end{bmatrix}$$

  which is unacceptable
- Thus, there is a necessity to restrict the norm of v
- The choice of the descent direction can be stated as:

$$\Delta x = \underset{v}{\text{argmin}}\, \nabla^\top f(x) v$$

s.t. $\|v\| = 1$ Let us understand the implication of the choice of the norm on the nature of the steepest descent direction

Algorithms: Steepest Descent

- Let $\mathbf{v} \in \Re^n$ be a unit vector under some norm. By first order convexity condition for convex and differentiable f,

$$f(\mathbf{x}^{(k)}) - f(\mathbf{x}^{(k)} + \mathbf{v}) \leq -\nabla^T f(\mathbf{x}^{(k)})\mathbf{v}$$

- For small $\mathbf{v}$, the inequality turns into approximate equality. The term $-\nabla^T f(\mathbf{x}^{(k)})\mathbf{v}$ can be thought of as (an upper-bound on) the first order prediction of decrease.
- The idea in the steepest descent method is to choose a norm and then determine a descent direction such that for a unit step in that norm, the first order prediction of decrease is maximized. This choice of the descent direction can be stated as

$$\Delta\mathbf{x} = \text{argmin}\left\{\nabla^T f(\mathbf{x})\mathbf{v} \mid ||\mathbf{v}|| = 1\right\}$$

- *Empirical observation:* If the norm chosen is aligned with the gross geometry of the sub-level sets[3], the steepest descent method converges faster to the optimal solution. Else, it often amplifies the effect of oscillations.

[3] The alignment can be determined by fitting, for instance, a quadratic to a sample of the points.

Various choices of the norm result in different solutions for Δx

- For 2-norm, $\Delta x = -\frac{\nabla f(x^{(k)})}{\left\|\nabla f(x^{(k)})\right\|_2}$

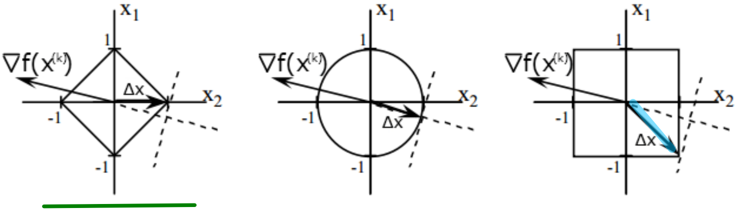
  (gradient descent)

- For 1-norm, $\Delta x = -\operatorname{sign}\left(\frac{\partial f(x^{(k)})}{\partial x_i^{(k)}}\right) e_i$, where e_i is the i^{th} standard basis vector and i is the component $\frac{\partial f(x^{(k)})}{\partial x_i^{(k)}}$ with the maximum magnitude

  ei corresponds to the component of the gradient with the maximum magnitude

  informed version of *(coordinate descent)*

- For ∞-norm, $\Delta x = -\operatorname{sign}(\nabla f(x^{(k)}))$ chooses diagonal direction



SDD in L1-norm

SDD in L2-norm

SDD in L∞-norm

General Algorithm: Steepest Descent (contd)

```
Find a starting point $\mathbf{x}^{(0)} \in \mathcal{D}$.
repeat
    1. Set $\Delta \mathbf{x}^{(k)} = \text{argmin}\left\{\nabla^T f(\mathbf{x}^{(k)})\mathbf{v} \mid ||\mathbf{v}|| = 1\right\}$.
    2. Choose a step size $t^{(k)} > 0$ using exact or backtracking ray search.
    3. Obtain $\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + t^{(k)} \Delta \mathbf{x}^{(k)}$.
    4. Set $k = k + 1$.
until stopping criterion (such as $||\nabla f(\mathbf{x}^{(k+1)})|| \leq \epsilon$) is satisfied
```

Figure 8: The steepest descent algorithm.

Two examples of the steepest descent method are the gradient descent method (for the eucledian or L_2 norm) and the coordinate-descent method (for the L_1 norm). One fact however is that no two norms should give exactly opposite steepest descent directions, though they may point in different directions.

Convergence of Descent Algorithm

- Consider the general descent algorithm ($\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k < 0$ for each k) with each step: $\mathbf{x}^{k+1} = \mathbf{x}^k + t^k\Delta\mathbf{x}^k$.
    - Suppose f is bounded below in $\Re^n$ and
    - is continuously differentiable in an open set $\mathcal{N}$ containing the level set $\{\mathbf{x} | f(\mathbf{x}) \leq f(\mathbf{x}^0)\}$
    - ∇f is Lipschitz continuous.

  Then, $\displaystyle\sum_{k=1}^{\infty} \frac{(\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k)^2}{\|\Delta\mathbf{x}^k\|^2} < \infty$ (that is, it is finite)

i.e. descent direction is close enough
to the gradient

- Thus, $\lim_{k\to\infty} \frac{\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k}{\|\Delta\mathbf{x}^k\|} = 0.$
- If we additionally assume that the descent direction is not orthogonal to the gradient, *i.e.*, $-\frac{\nabla^T f(\mathbf{x}^k)\Delta\mathbf{x}^k}{\|\Delta\mathbf{x}^k\|\|\nabla f(\mathbf{x}^k)\|} \geq \Gamma$ for some $\Gamma > 0$, then, we can show that $\lim_{k\to\infty} \|\nabla f(\mathbf{x}^k)\| = 0$
- Before we try and prove this result, let us discuss Lipschitz continuity (recall from midsem).

Lipschitz Continuity

Recall: Lipschitz Continuity of f

- Formally, $f(x) : \mathcal{D} \subseteq \Re^n \rightarrow \Re$ is Lipschitz continuous if $|f(\mathbf{x}) - f(\mathbf{y})| \leq L\|\mathbf{x} - \mathbf{y}\|$ for all $\mathbf{x}, \mathbf{y} \in \mathcal{D}$.
- A Lipschitz continuous function is limited in how fast it changes: there exists a definite positive real number $L > 0$ such that, for every pair of points on the graph of the function, the absolute value of the slope of the line connecting them is not greater than this real number. This bound is called the function's Lipschitz constant, $L > 0$.
- We can show that if a function $f : \Re \rightarrow \Re$ is convex in (α, β) it is Lipschitz continuous in $[\gamma, \delta]$ where $\alpha < \gamma < \delta < \beta$. We do not assume that f is differentiable.

Convex Function is Lipschitz continuous

$f(x) : \Re \to \Re$ is Lipschitz continuous in $[\gamma, \delta]$ if $|f(x) - f(y)| \leq L|x - y|$ for all $x, y \in [\gamma, \delta]$. We will show that if a function $f : \Re \to \Re$ is convex in (α, β) it is Lipschitz continuous in $[\gamma, \delta]$ where $\alpha < \gamma < \delta < \beta$. Do not assume that f is differentiable. Fill up the three blanks below.

- Let $p, q \in \Re$ such that $\alpha < p < \gamma < \delta < q < \beta$ and let $x_1, x_2 \in [\gamma, \delta]$. Then

$$\underline{\hspace{10cm}}_1 \leq \frac{f(x_2) - f(x_1)}{x_2 - x_1} \leq \underline{\hspace{9cm}}$$

  because of convexity of f.

- Take $L = \underline{\hspace{10cm}}_3$

to prove Lipschitz continuity of f in the interval $[\gamma, \delta]$.