IMA workshop – *Optimization in simulation based models*

Generalized Pattern Search Algorithms : unconstrained and constrained cases

Mark A. Abramson

Air Force Institute of Technology

Charles Audet

École Polytechnique de Montréal

John Dennis

Rice University

www.gerad.ca/Charles.Audet/

Presentation outline

Generalized Pattern Search (GPS)

- Unconstrained problem
  $$\min_{x \in \mathbb{R}^n} f(x)$$
- Bound and linear constraints
  $$\min_{x \in X} f(x) \text{ where } X = \{x \in \mathbb{R}^n : Ax \leq b\} \cap [\ell, u]$$
- General constraints
  $$\min_{x \in X \cap \Omega} f(x) \text{ where } \Omega = \{x : c_i(x) \leq 0, i = 1, 2, \ldots, m\}$$

Unconstrained optimization

$$\min_{x \in \mathbb{R}^n} \quad f(x)$$

where $f : \mathbb{R}^n \to \mathbb{R} \cup \{\infty\}$ may be discontinuous or infinite valued, and:

- f is usually given as a black box (typically a computer code),
- f is expensive and have few correct digits,
- $f(x)$ may fail expensively and unexpectedly.

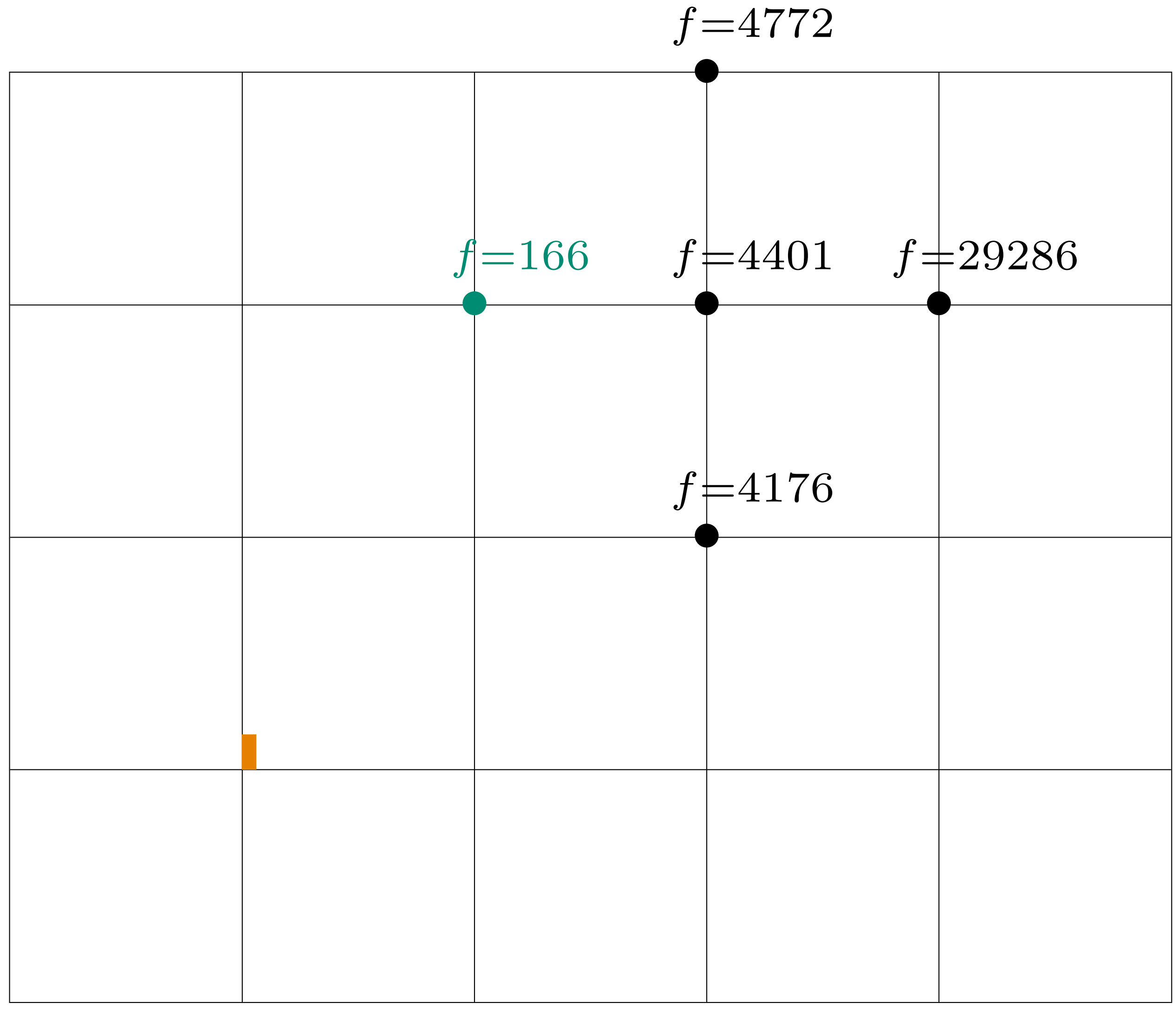
Ancestor of GPS : Coordinate search

- INITIALIZATION:
  x_0 : initial point in $\mathbb{R}^n$
  $\Delta_0 > 0$: initial step size.

- POLL STEP: For $k = 0, 1, \ldots$
  If $f(t) < f(x_k)$ for some $t \in P_k := \{x_k \pm \Delta_k e_i : i \in N\}$,
  set $x_{k+1} = t$
  and $\Delta_{k+1} = \Delta_k$;
  otherwise x_k is a minimizer over the set P_k,
  set $x_{k+1} = x_k$
  and $\Delta_{k+1} = \frac{\Delta_k}{2}$.

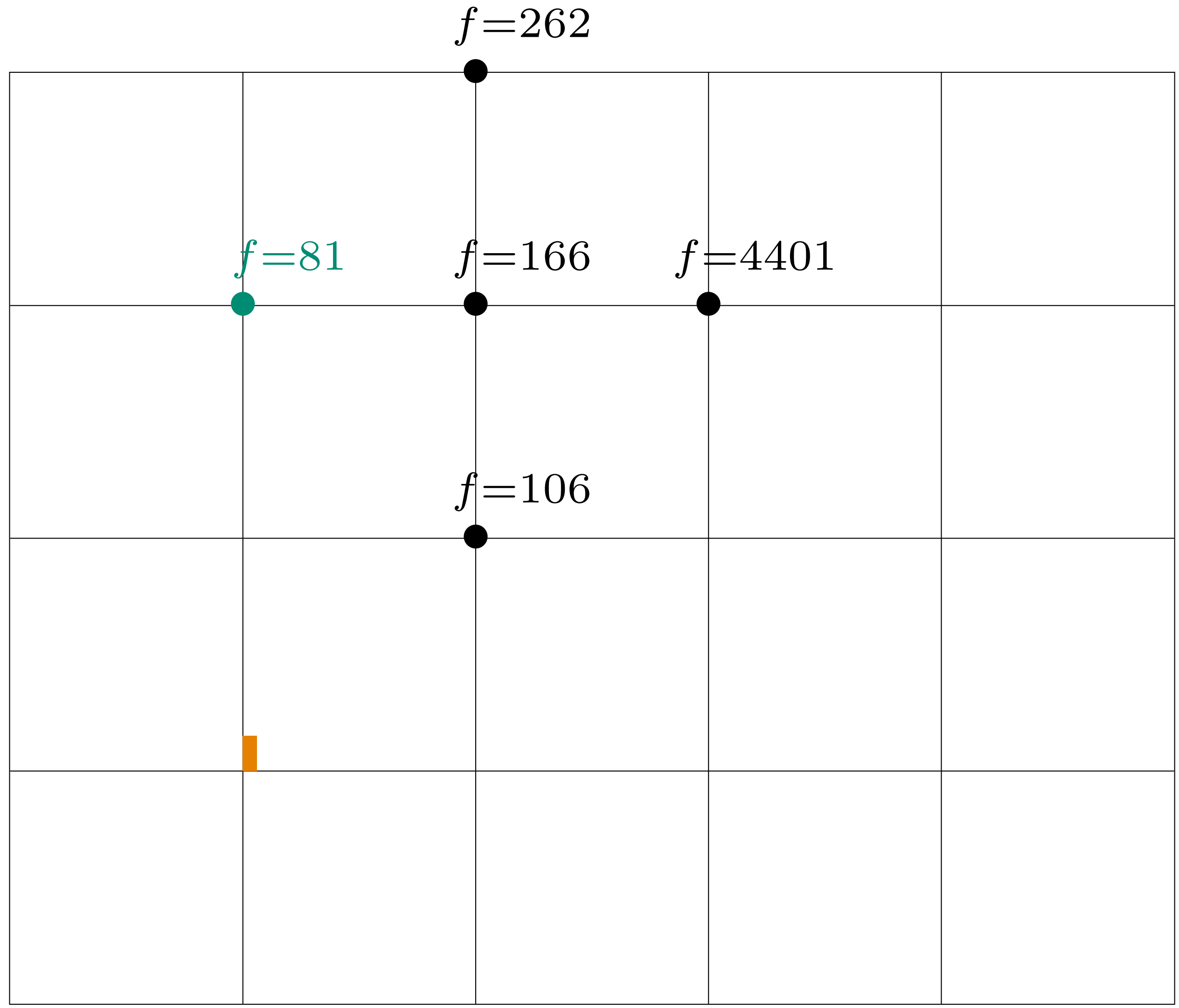
Coordinate search run

$x_0 = (2,2)^T, \Delta_0 = 1$

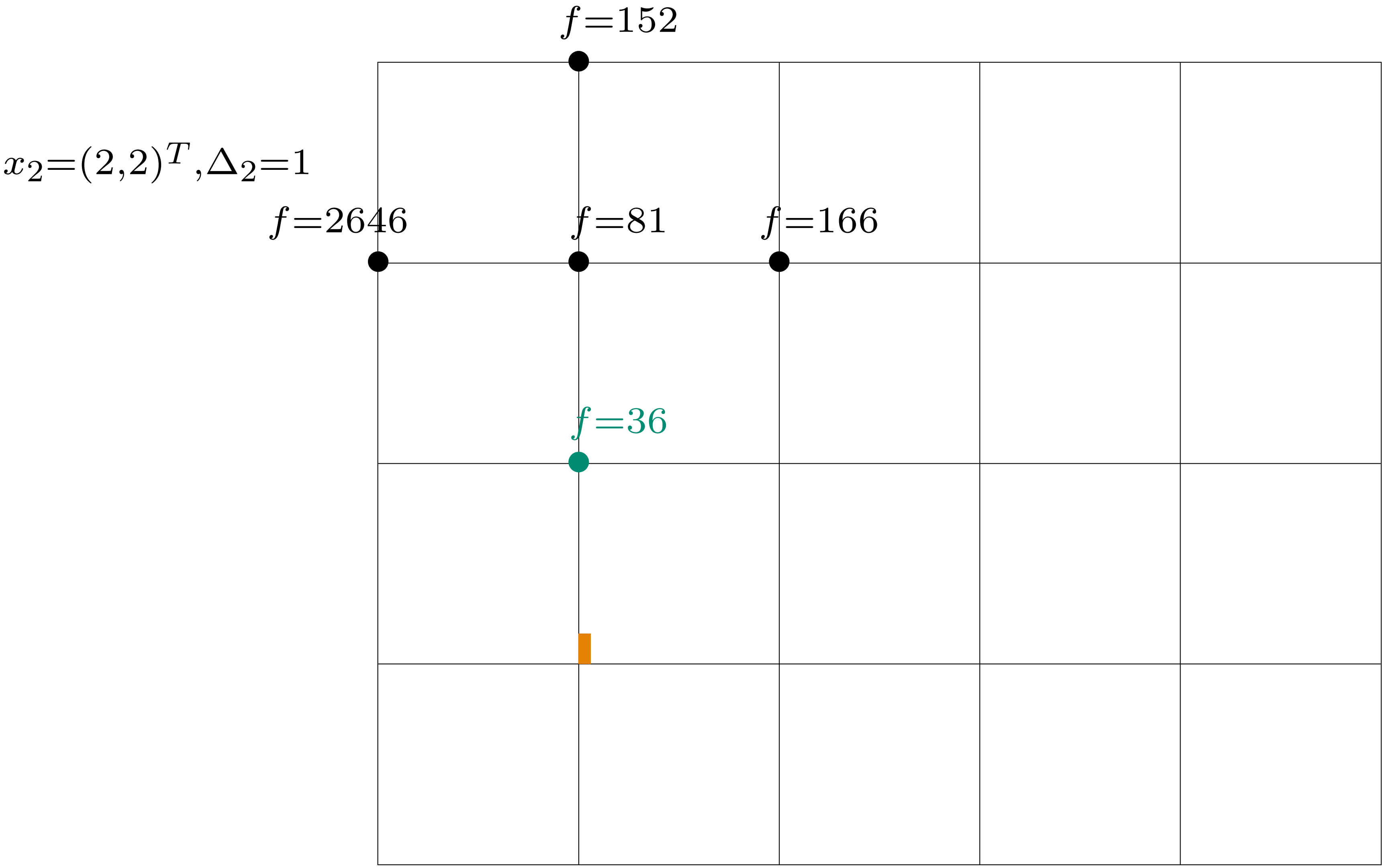


Coordinate search run

$x_1 = (1,2)^T, \Delta_1 = 1$

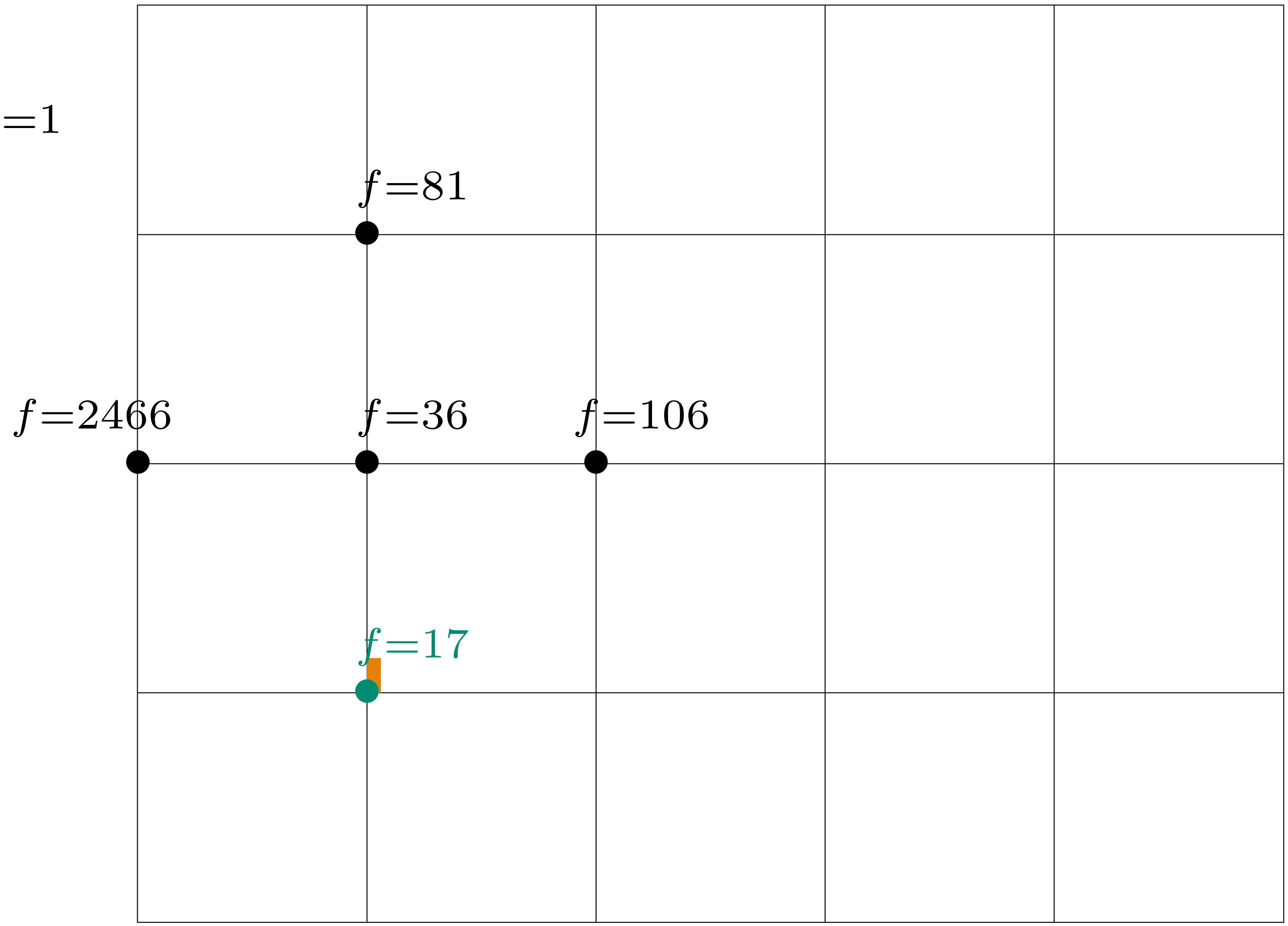


Coordinate search run



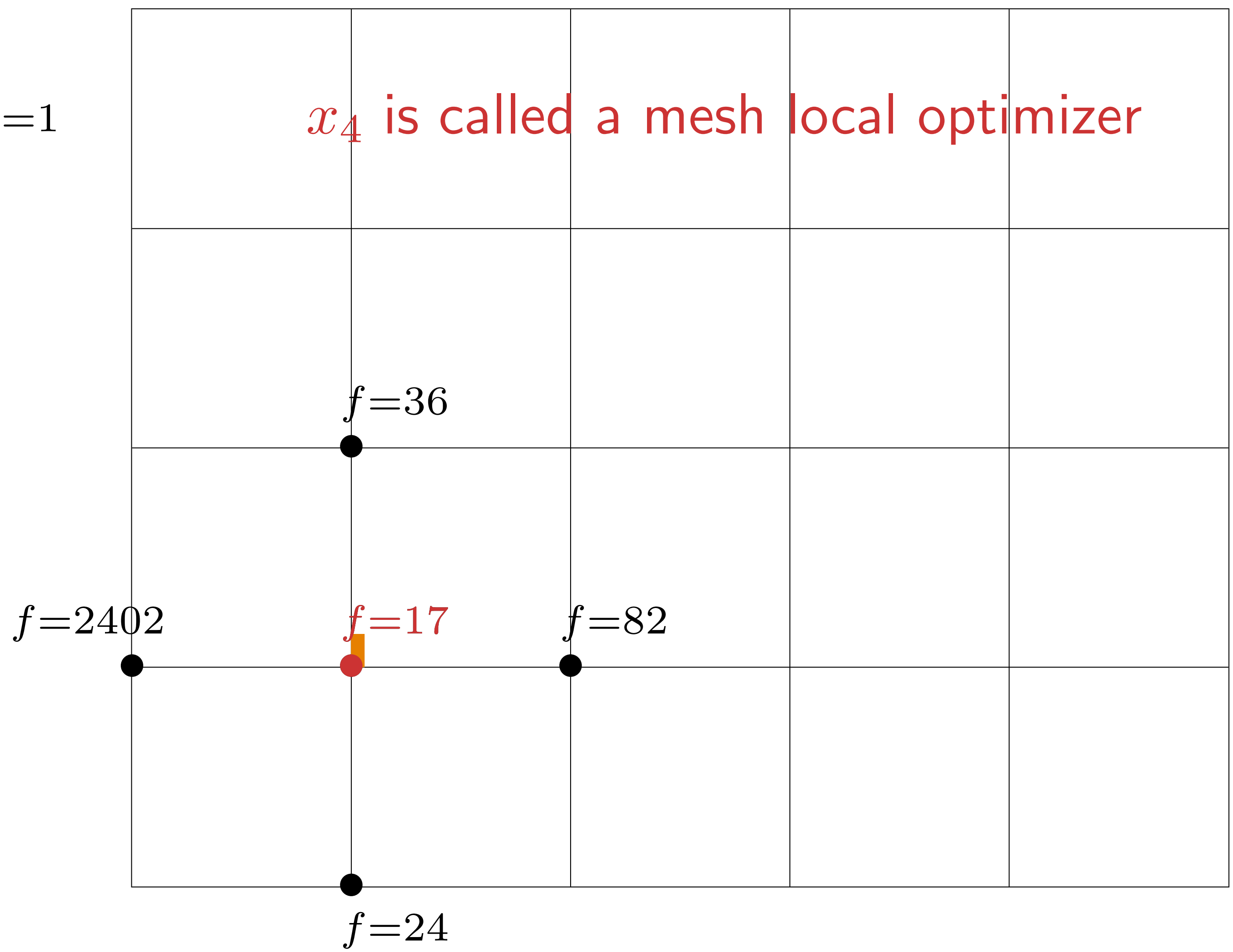
Coordinate search run

$x_3 = (0, 1)^T, \Delta_3 = 1$



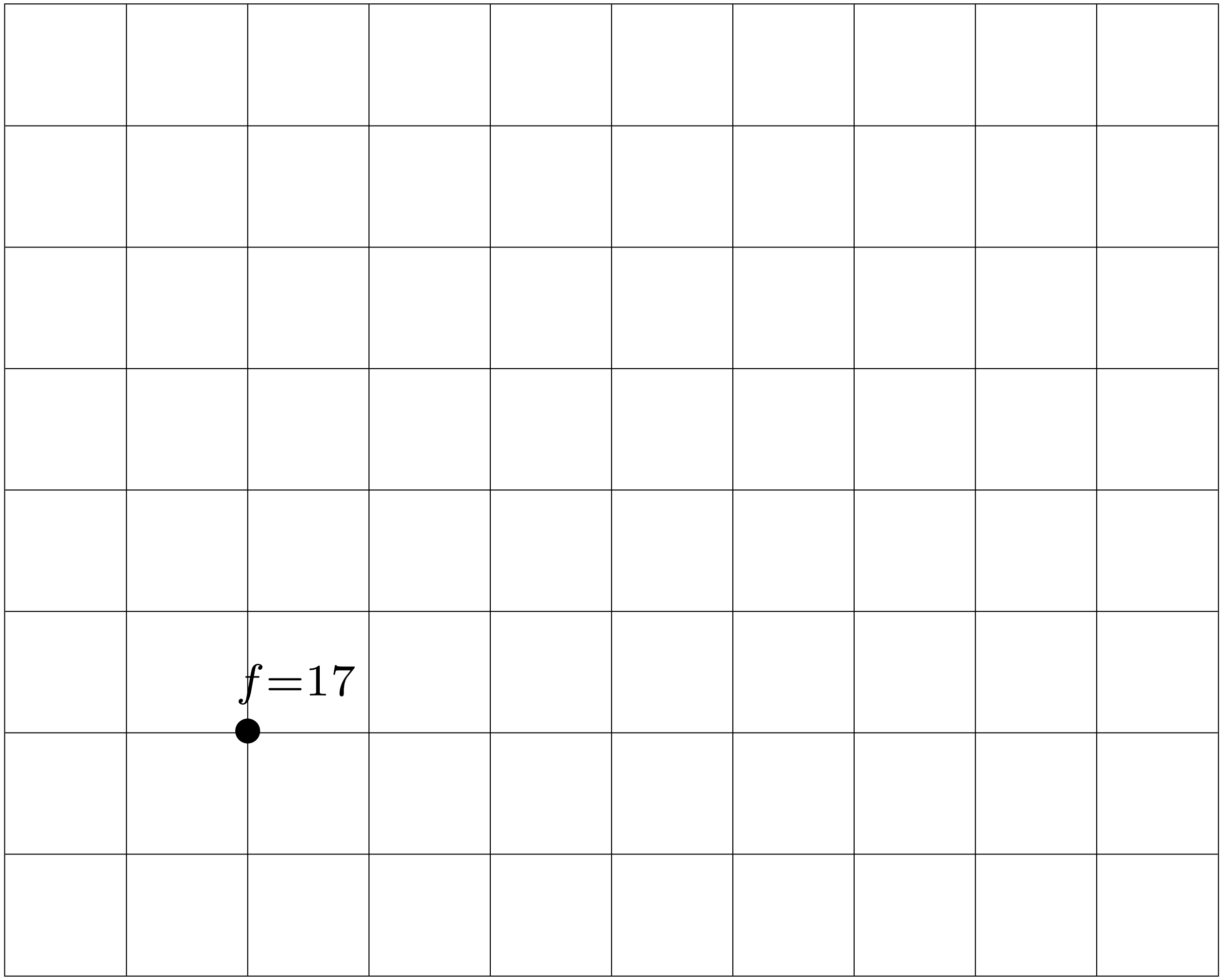
Coordinate search run

$x_4=(0,0)^T, \Delta_4=1$



Coordinate search run

$x_5 = x_4 = (0,0)^T, \Delta_4 = \frac{1}{2}$



Coordinate search run

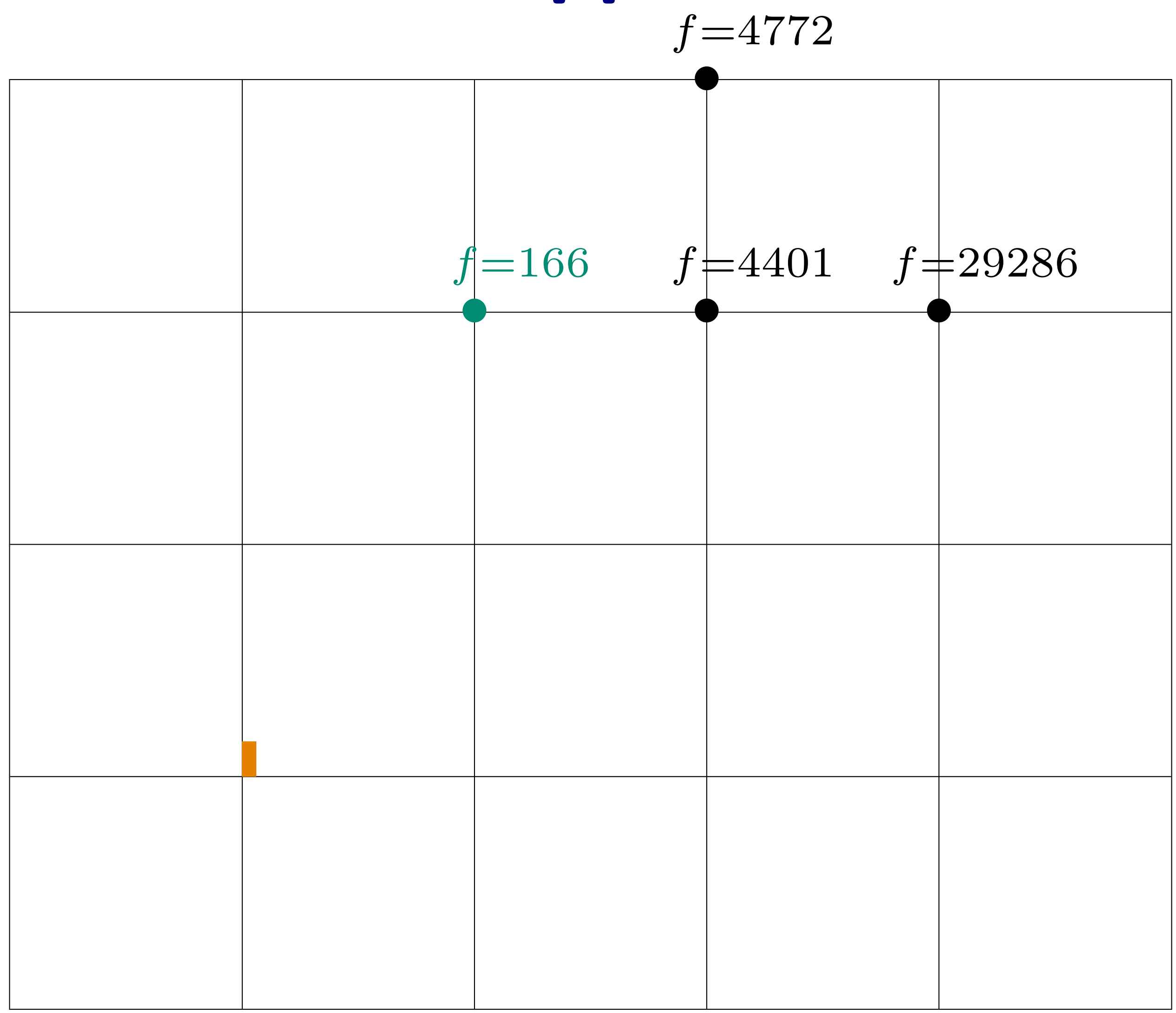
A budget of 20 function evaluations produces

$$x = (0,0)^T \text{ with } f(x) = 17.$$

Can we do better with the coordinate search ?

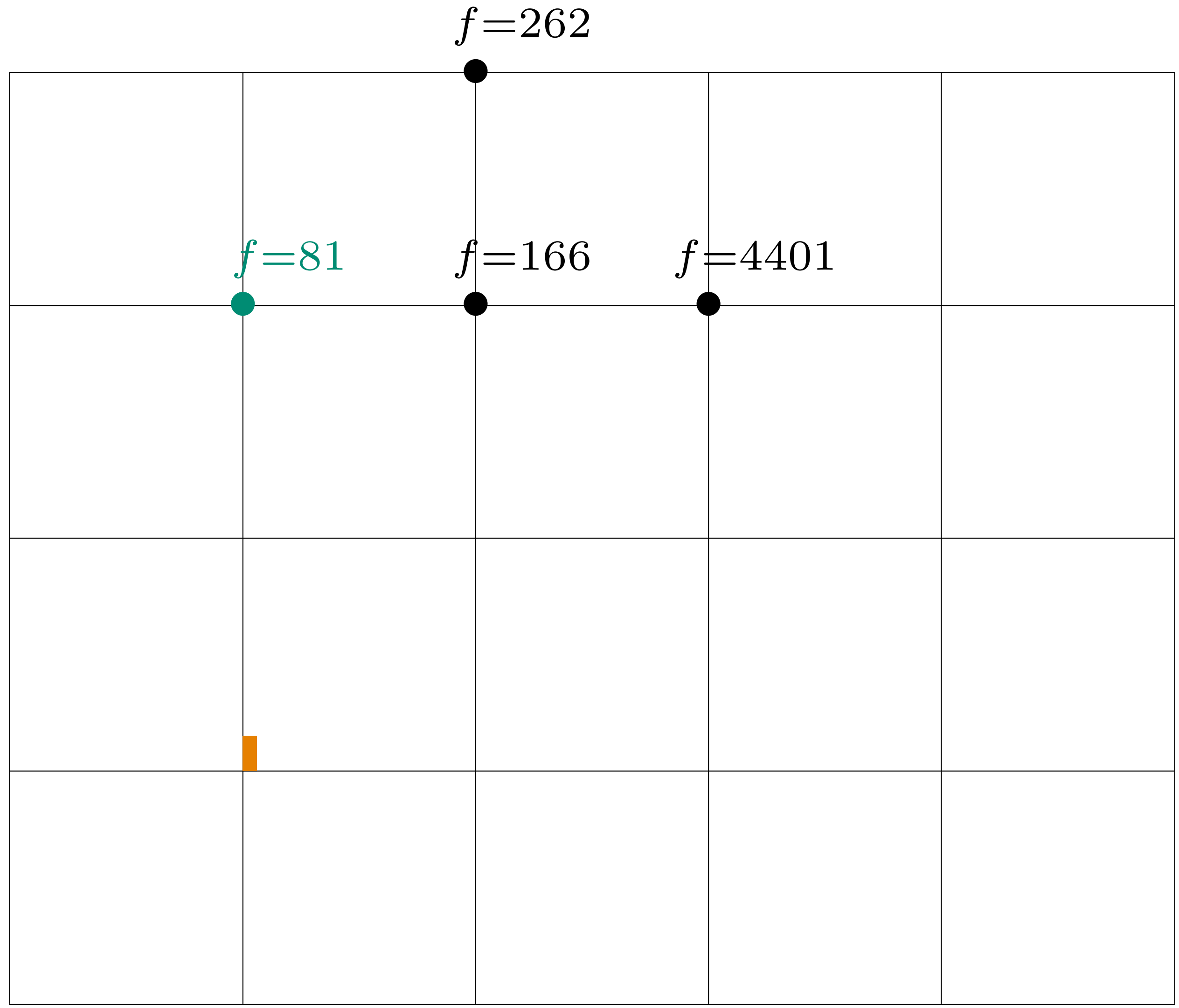
Coordinate search : opportunistic run

$x_0 = (2,2)^T, \Delta_0 = 1$

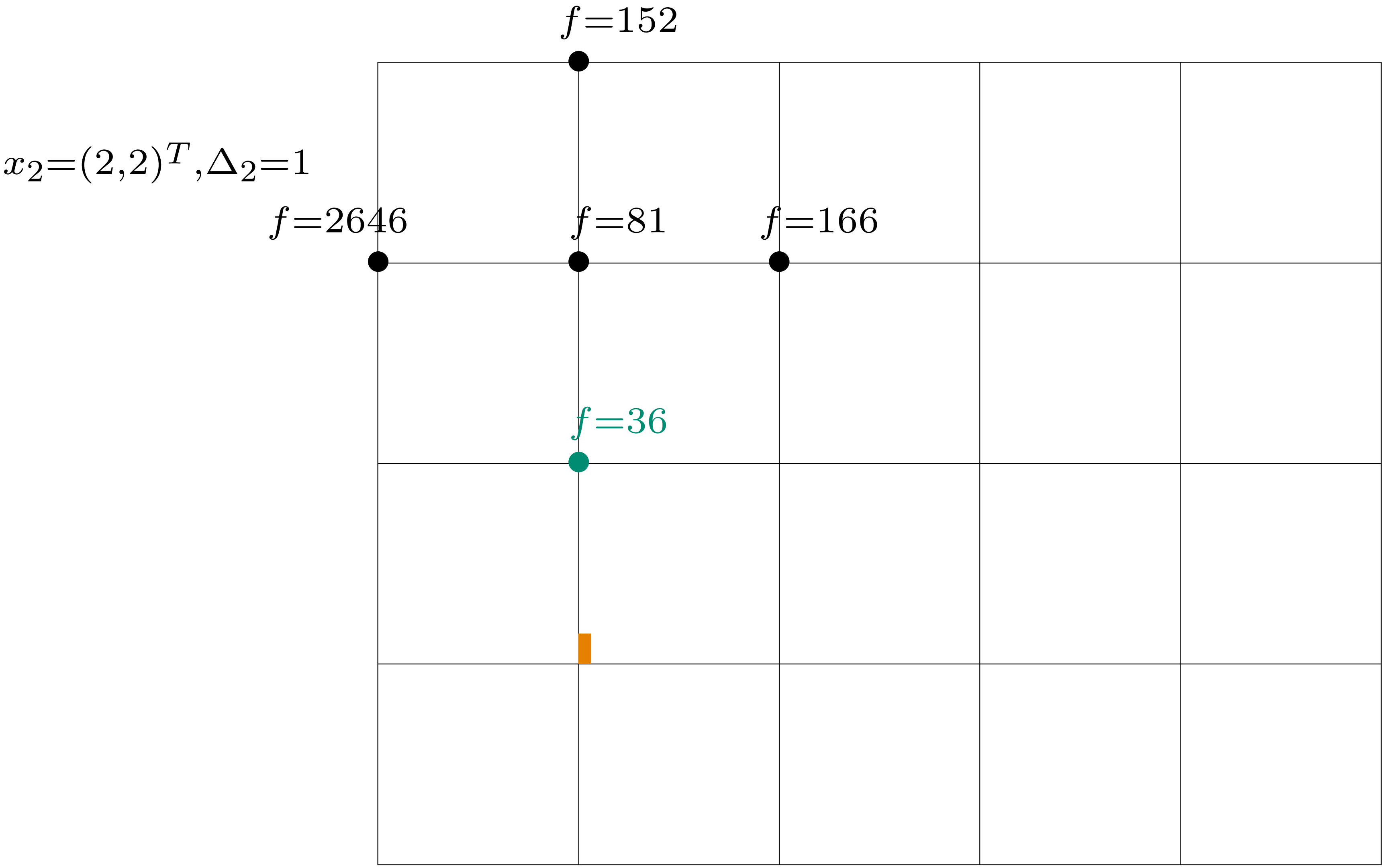


Coordinate search : opportunistic run

$x_1 = (1,2)^T, \Delta_1 = 1$

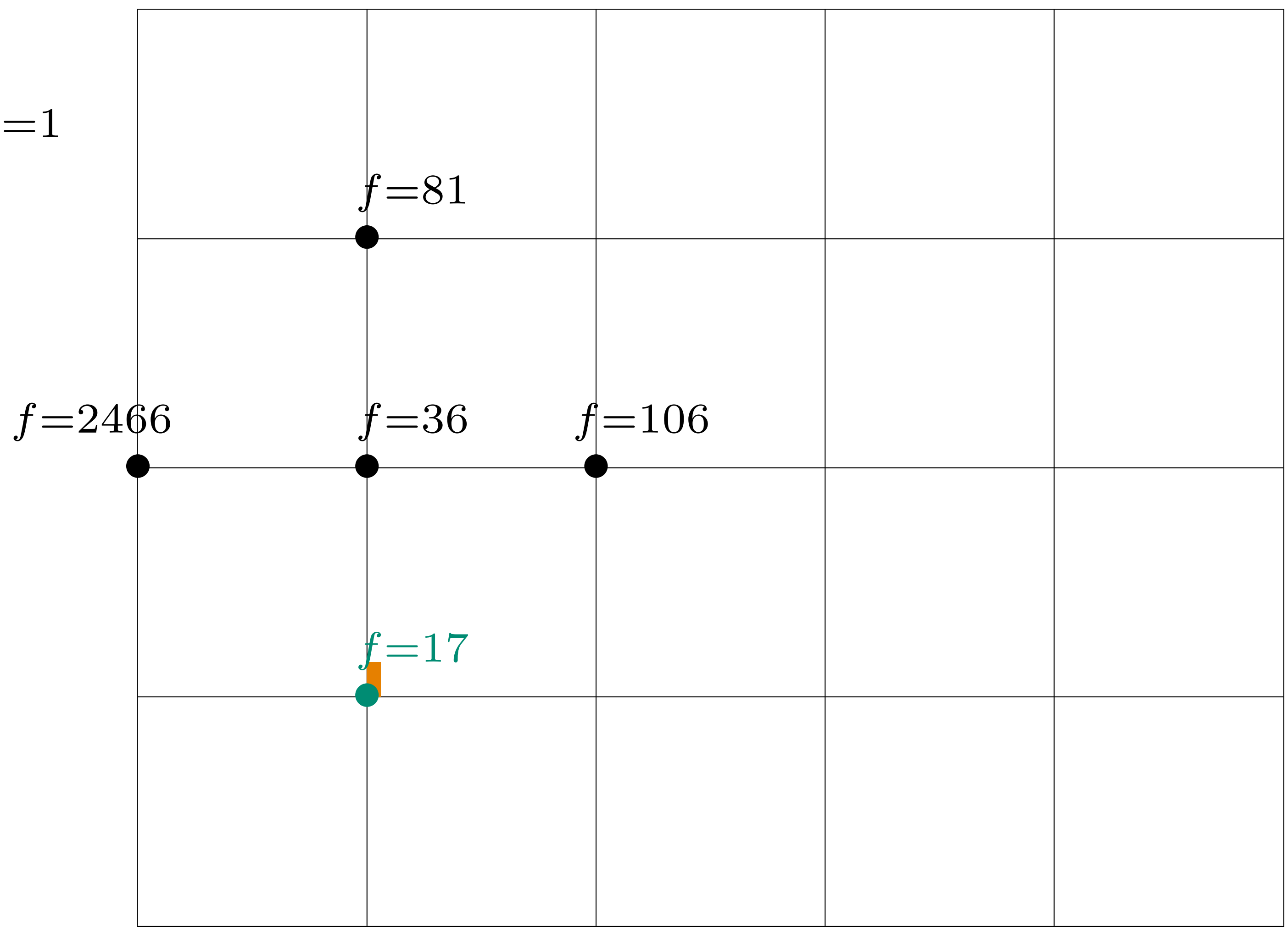


Coordinate search : opportunistic run



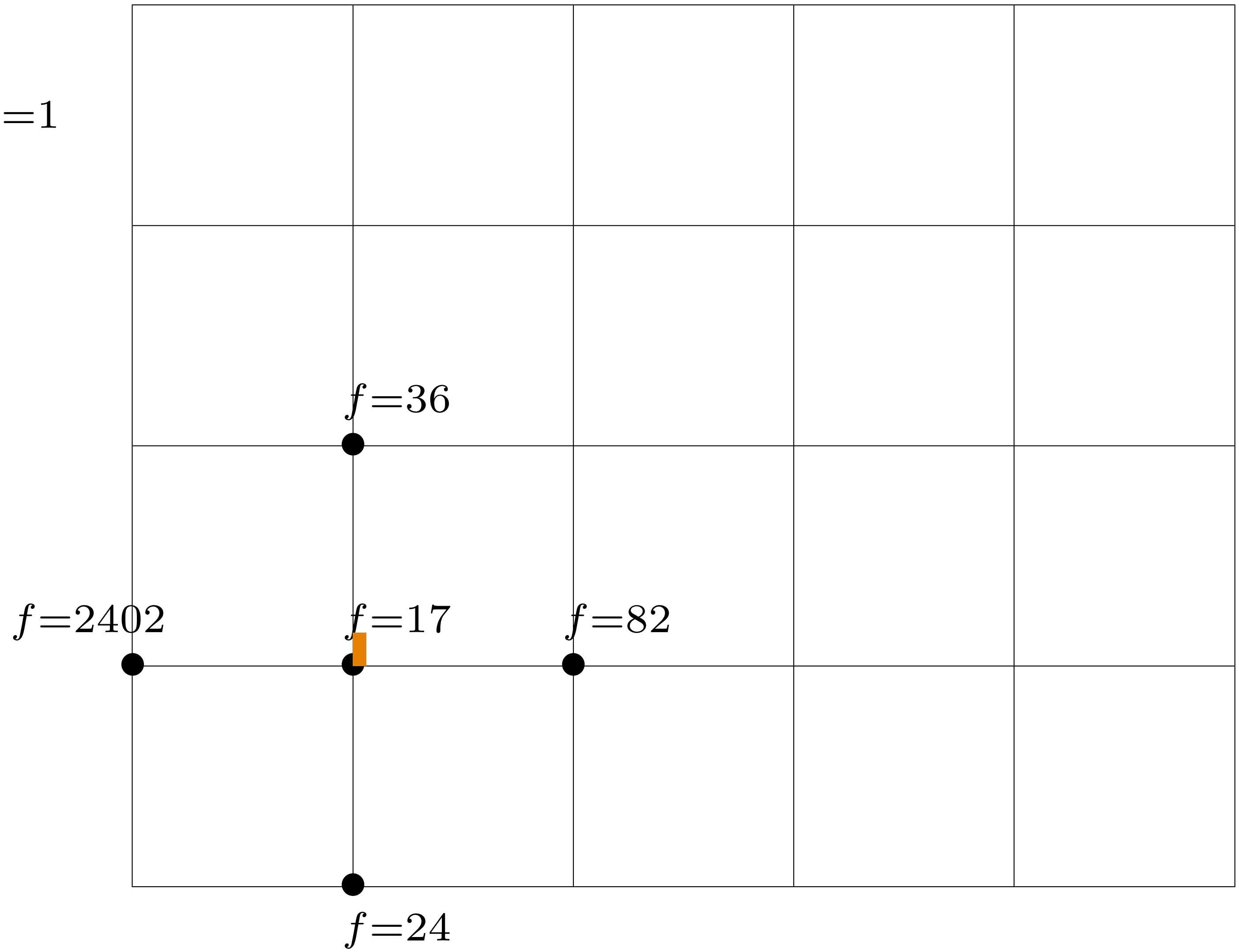
Coordinate search : opportunistic run

$x_3 = (0,1)^T, \Delta_3 = 1$



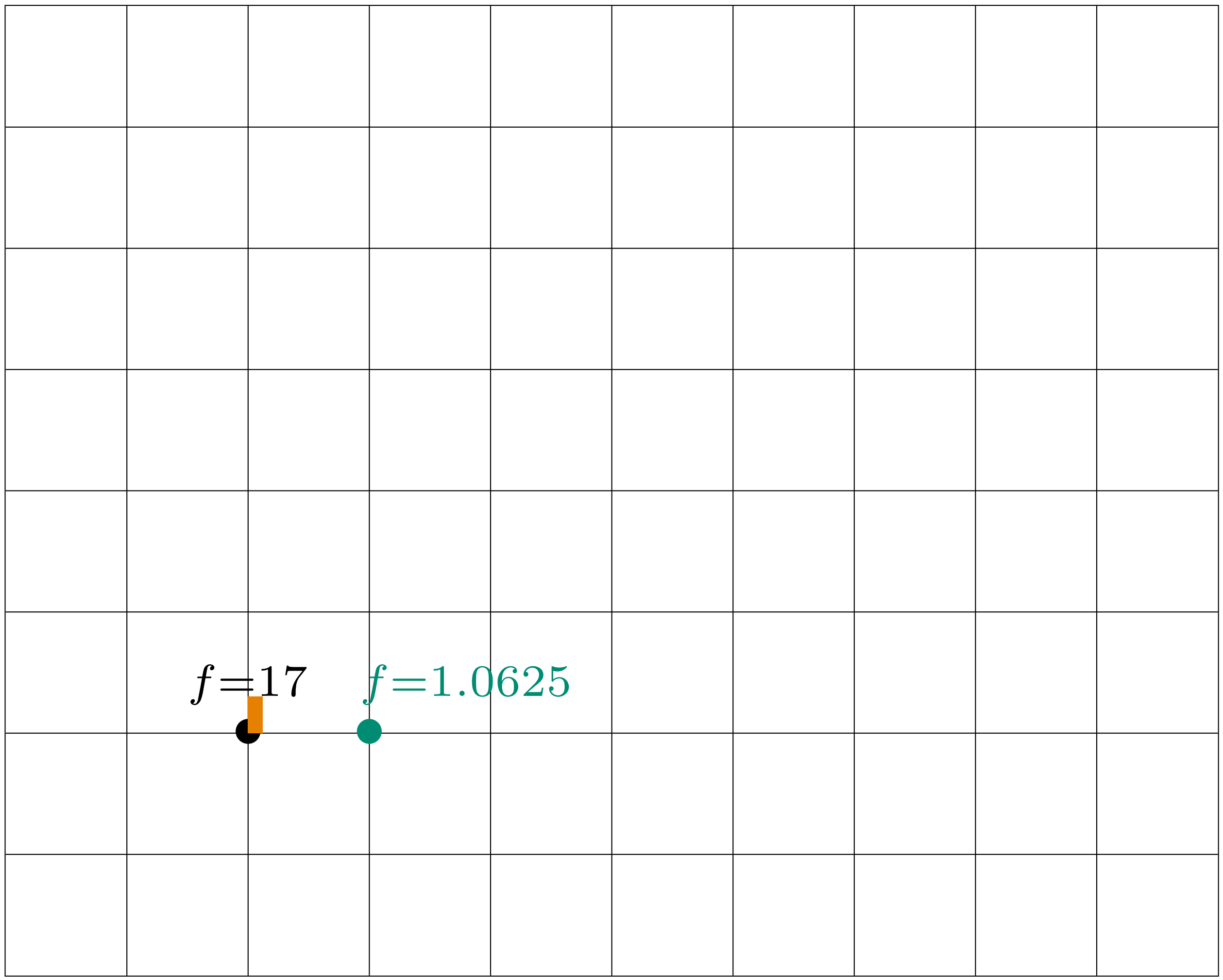
Coordinate search : opportunistic run

$x_4 = (0,0)^T, \Delta_4 = 1$

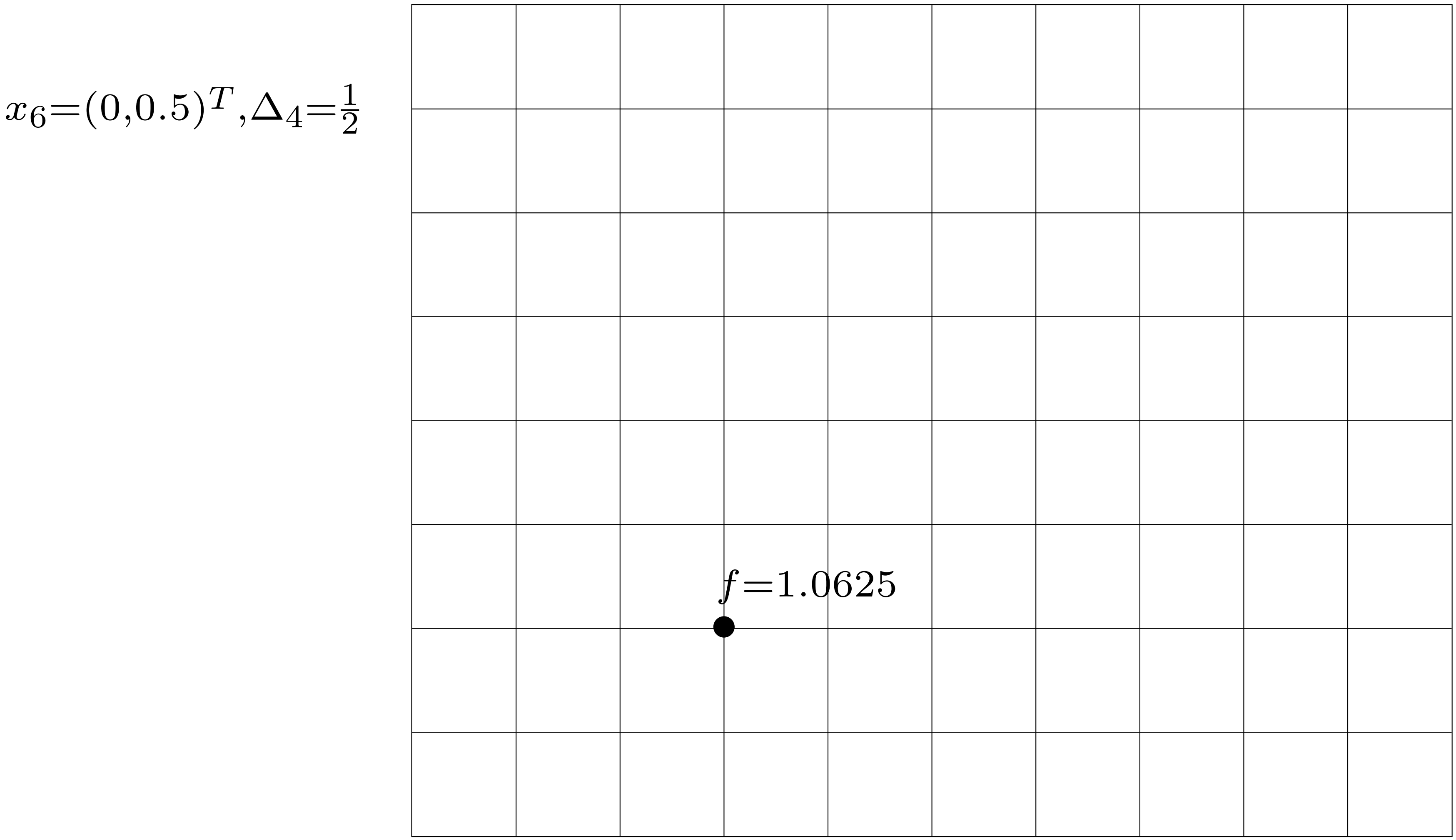


Coordinate search : opportunistic run

$x_5 = x_4 = (0,0)^T, \Delta_4 = \frac{1}{2}$



Coordinate search : opportunistic run



Coordinate search : opportunistic run

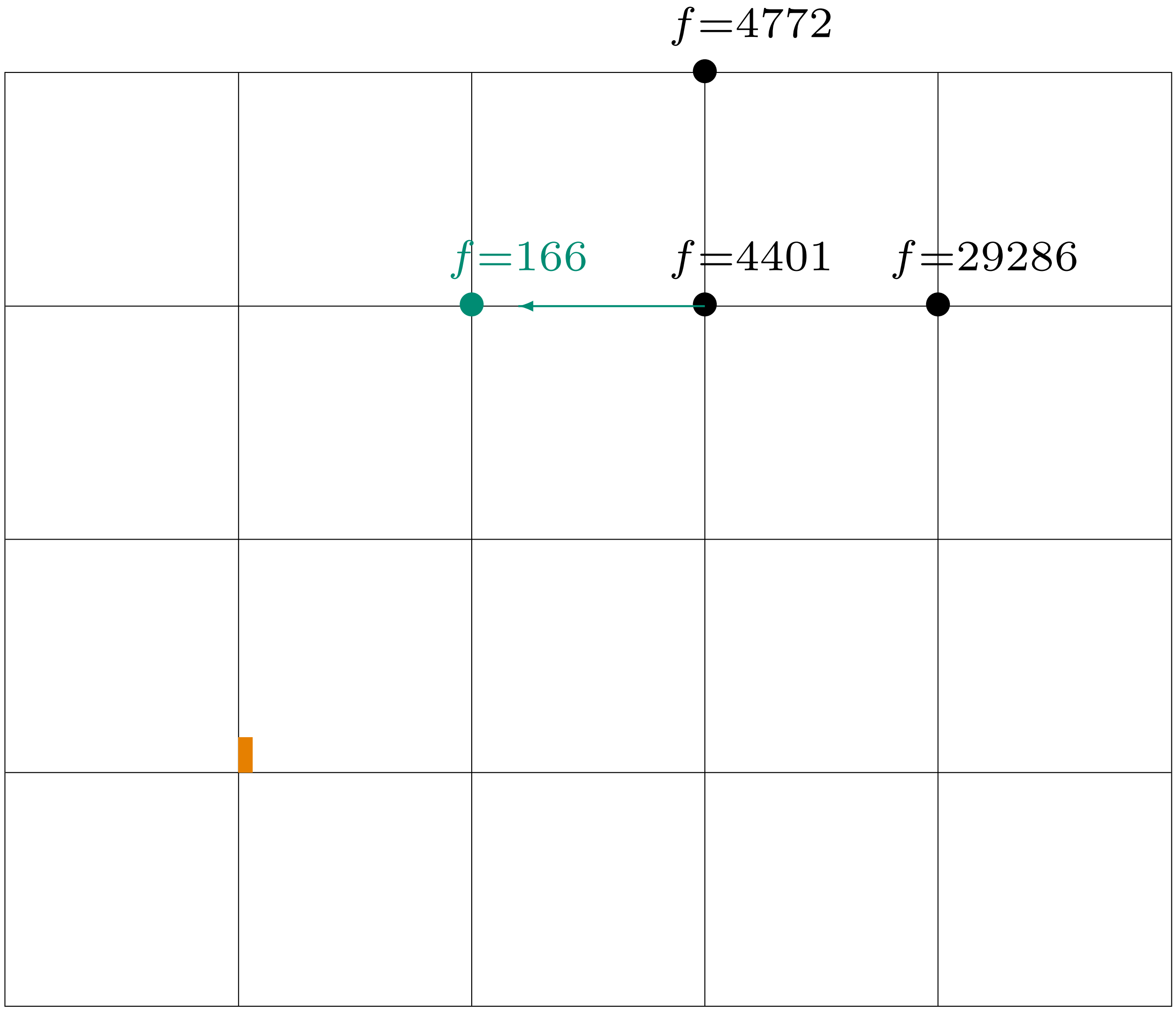
A budget of 20 function evaluations produces

$$x = (0.5, 0)^T \text{ with } f(x) = 1.0625.$$

Can we do better with the coordinate search ?

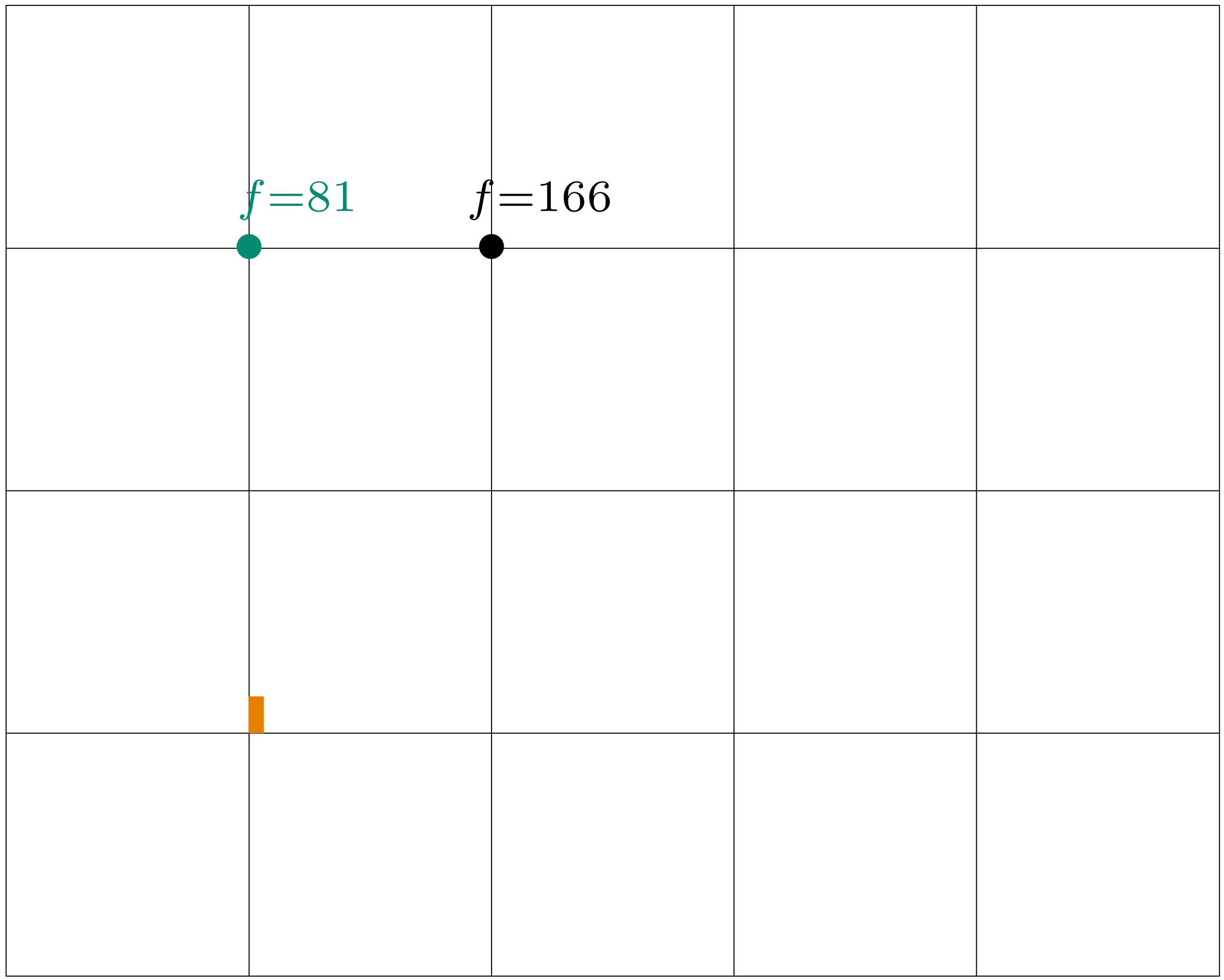
Coordinate search : dynamic run

$x_0 = (2,2)^T, \Delta_0 = 1$

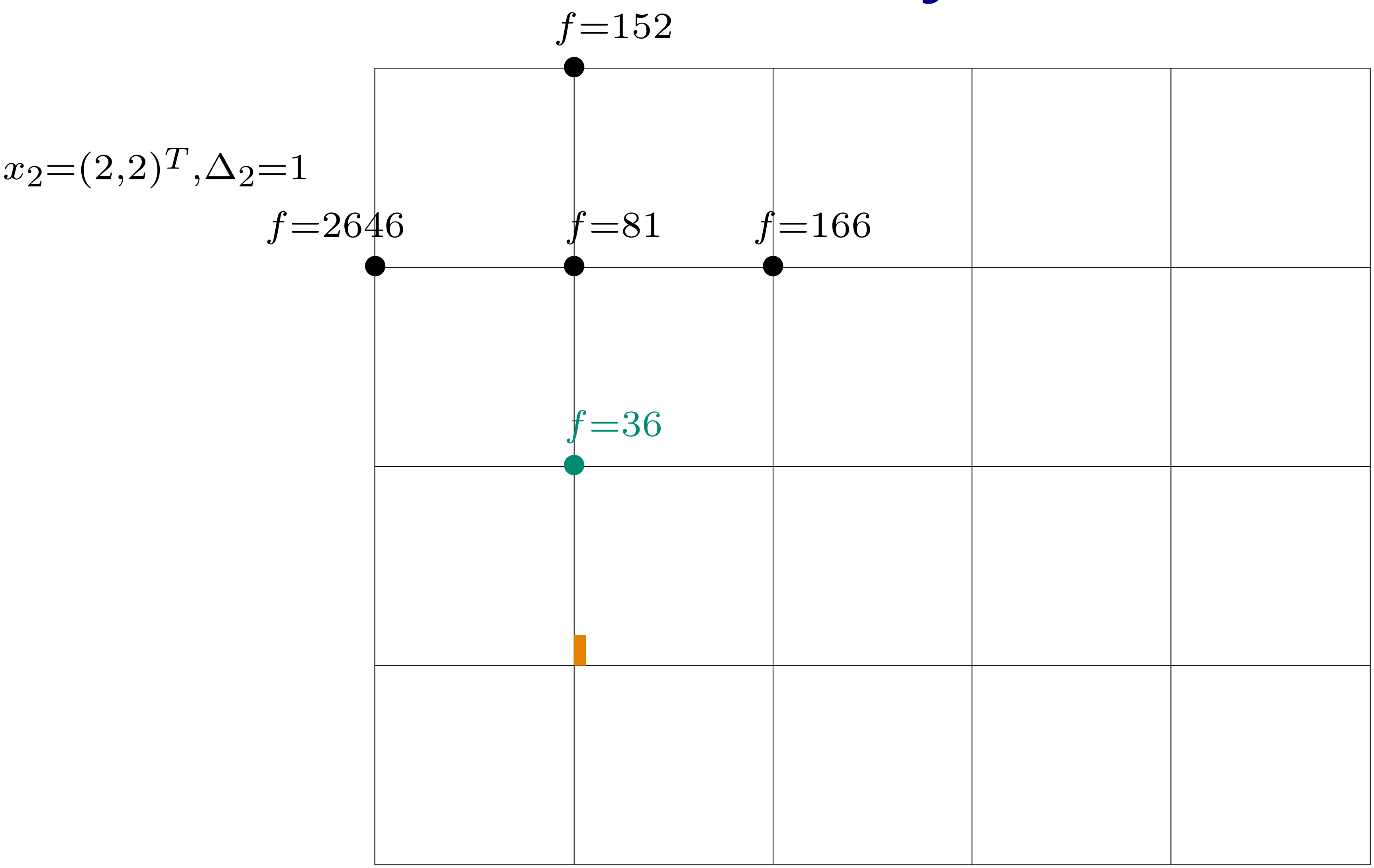


Coordinate search : dynamic run

$x_1 = (1,2)^T, \Delta_1 = 1$

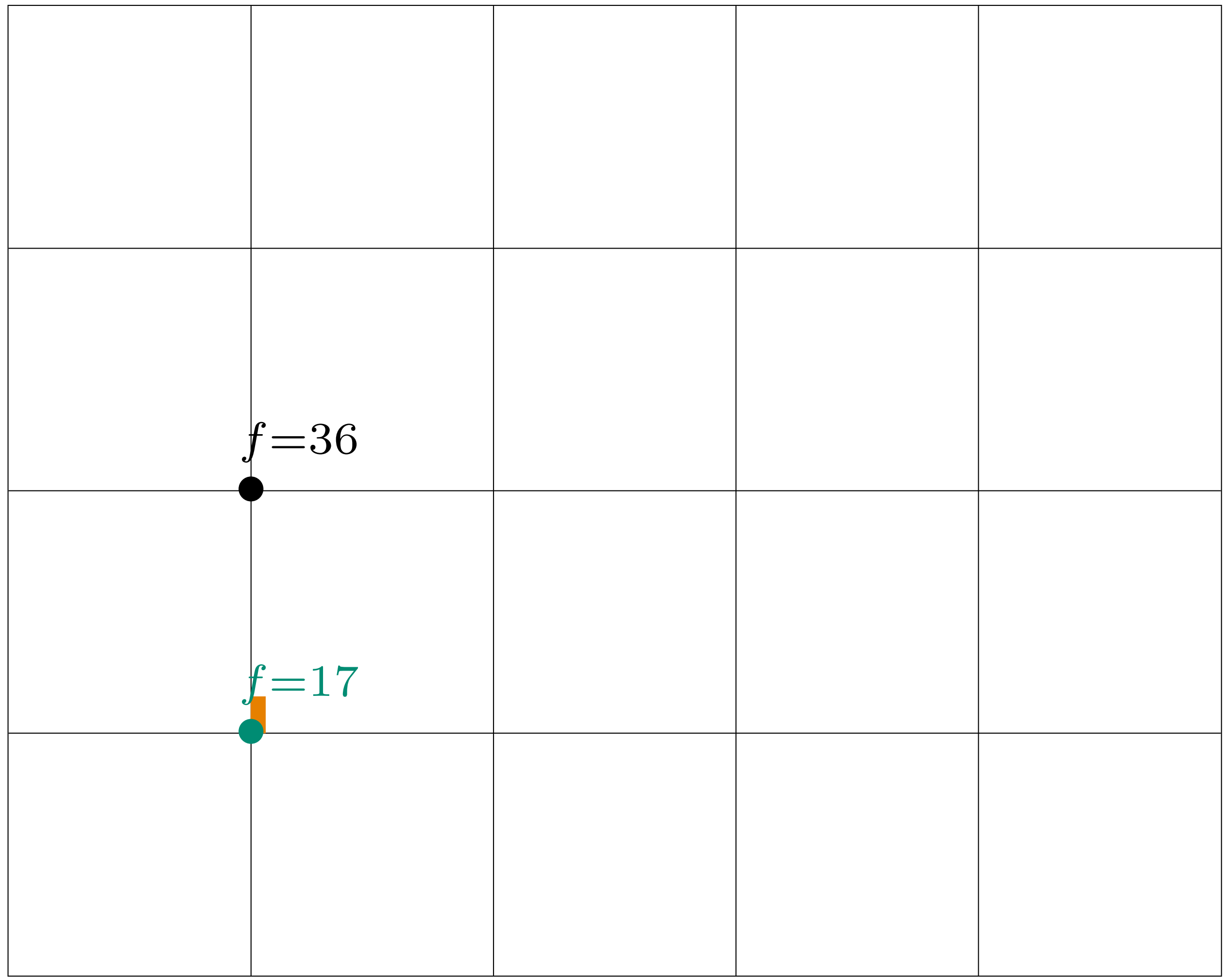


Coordinate search : dynamic run



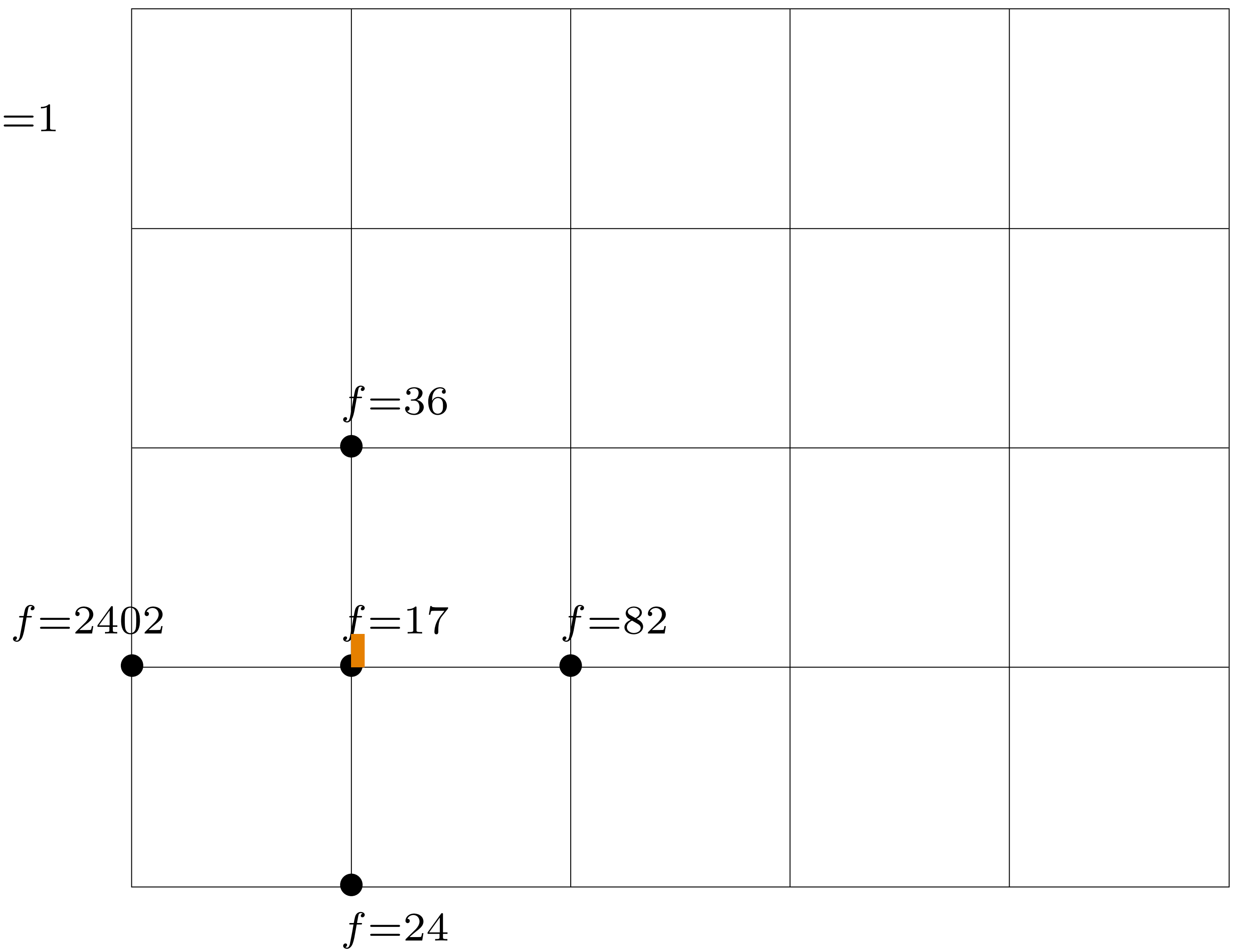
Coordinate search : dynamic run

$x_3 = (0,1)^T, \Delta_3 = 1$



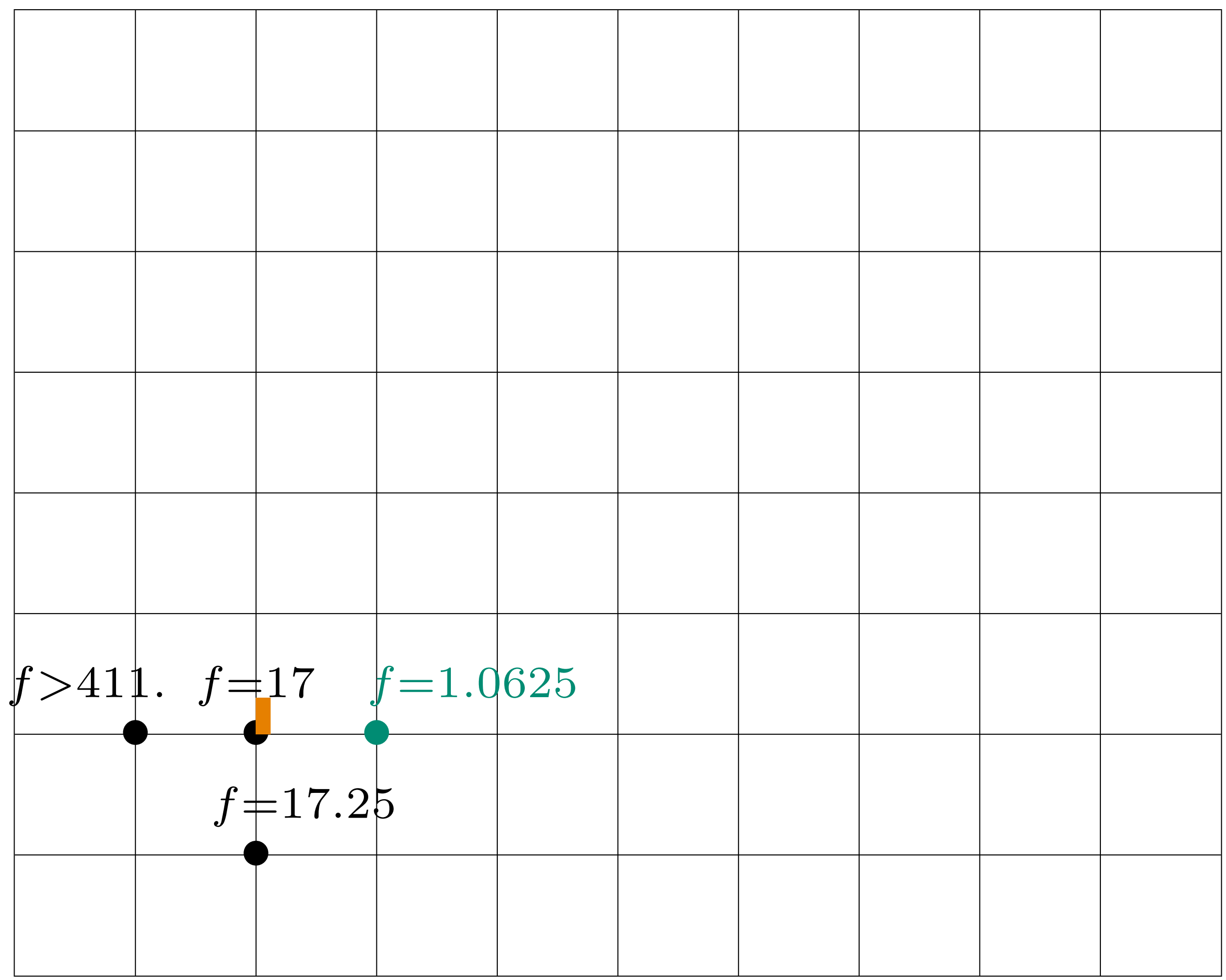
Coordinate search : dynamic run

$x_4 = (0,0)^T, \Delta_4 = 1$



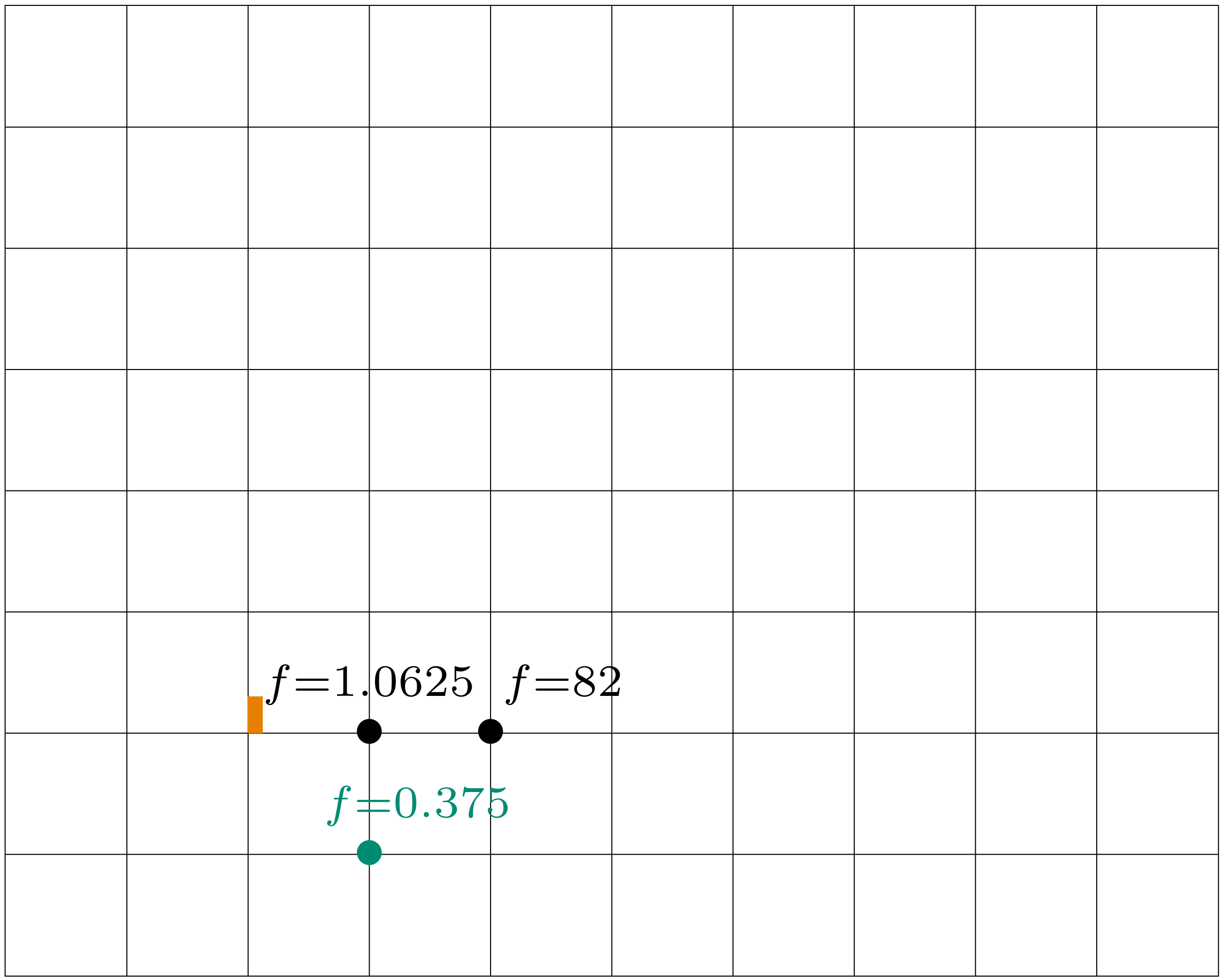
Coordinate search : dynamic run

$x_5 = x_4 = (0,0)^T, \Delta_4 = \frac{1}{2}$



Coordinate search : dynamic run

$x_6=(0,0.5)^T, \Delta_4=\frac{1}{2}$



Coordinate search : 3 strategies

Complete		Fixed order		Dynamic order	
x^T	$f(x)$	x^T	$f(x)$	x^T	$f(x)$
After 20 function evaluations					
$(0, 0)$	17	$(.5, 0)$	1.0625	$(.5, -.5)$	0.375
After 50 function evaluations					
$(.375, -.375)$	1.8e-2	$(.375, -.312)$	5.7e-3	$(.375, -.344)$	3.1e-3

Convergence of coordinate searches

if the sequence of iterates $\{x_k\}$ belongs to a compact set

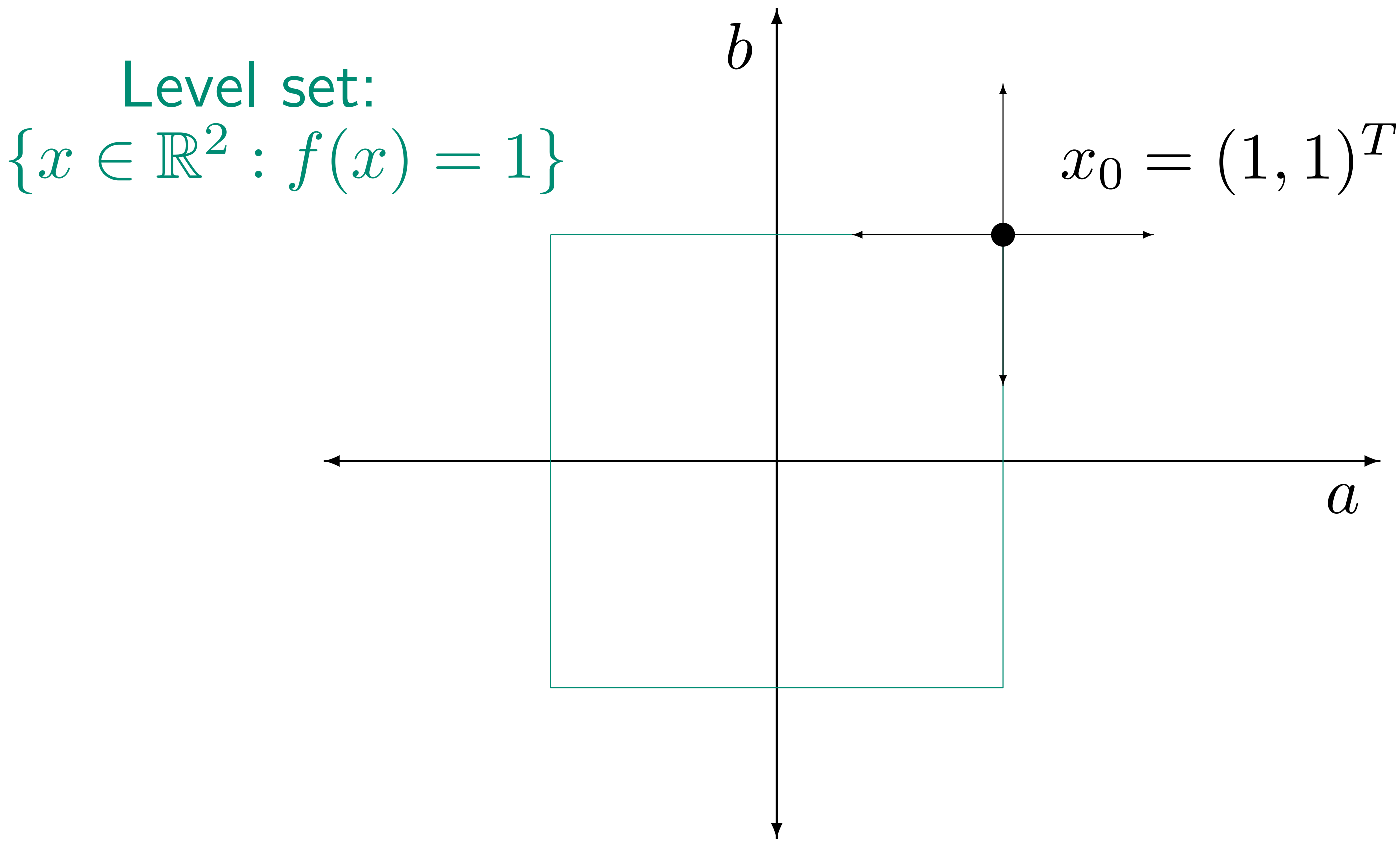
- $\lim_k \Delta_k = 0$
- there is an $\hat{x}$ which is the limit of a sequence of mesh local optimizers
- If f is continuously differentiable at $\hat{x}$, then $\nabla f(\hat{x}) = 0$.

... but we do not know anything about f.

We need to work with less restrictive differentiability assumptions.

Coordinate search on a non-differentiable function

$f(x) = \|x\|_\infty$ with $x_0 = (1,1)^T$.



A directional algorithm

Clarke Calculus

If f is Lipschitz[1] near $\bar{x} \in \mathbb{R}^n$, then Clarke's generalized derivative at $\bar{x}$ in the direction $v \in \mathbb{R}^n$ is

$$f^\circ(\bar{x}; v) \;=\; \limsup_{y \to \bar{x},\; t \downarrow 0} \frac{f(y+tv) - f(y)}{t}.$$

[1] there exists a nonnegative scalar K such that

$$|f(x) - f(x')| \leq K \|x - x'\|$$

for all x, x' in some open neighborhood of $\bar{x}$.

Facts on Clarke calculus

- The *generalized gradient* of f at x is the set

$$\partial f(x) \ := \ \{\zeta \in \mathbb{R}^n : f^\circ(x; v) \geq v^T\zeta \text{ for all } v \in \mathbb{R}^n\}.$$

- Let f be Lipschitz near x, then

$$\partial f(x) = \text{co}\{\lim \nabla f(x_i) \ : \ x_i \to x \text{ and } \nabla f(x_i) \text{ exists } \}.$$

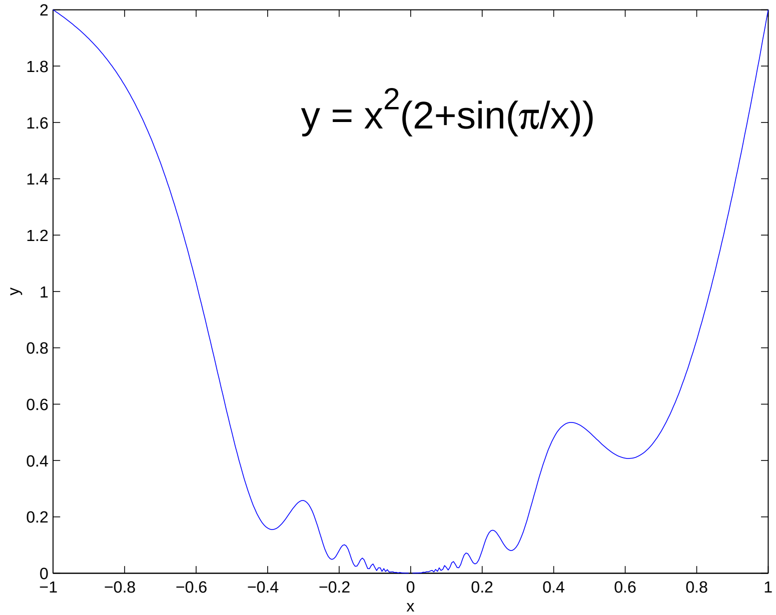
- Generalized derivative can be obtained from the generalized gradient : $f^\circ(x; v) = \max\{v^T\zeta : \zeta \in \partial f(x)\}$.
- If x is a minimizer of f, and f is Lipschitz near x, then $0 \in \partial f(x)$. *Generalizes the* $1st$ *order necessary condition for continuously differentiable* f*:* $0 = \nabla f(x)$.

- If f is differentiable (Hadamard, Gâteaux, or Fréchet) at x, then the derivative of f at x is in the generalized gradient $\partial f(x)$.
- When f is convex, $\partial f(x) =$ subdifferential.
- f is *regular* at x if for all $v \in \mathbb{R}^n$, the one-sided directional derivative $f'(x; v)$ exists and equals $f^\circ(x; v)$.
- f is *strictly differentiable* at x if for all $v \in \mathbb{R}^n$,

$$\lim_{y \to x, t \downarrow 0} \frac{f(y + tv) - f(y)}{t} = \nabla f(x)^T v.$$

- If f is Lipschitz near x and $\partial f(x)$ reduces to a singleton $\{\zeta\}$, then f is strictly differentiable at x and $\nabla f(x) = \zeta$.

Differentiable, not strictly differentiable

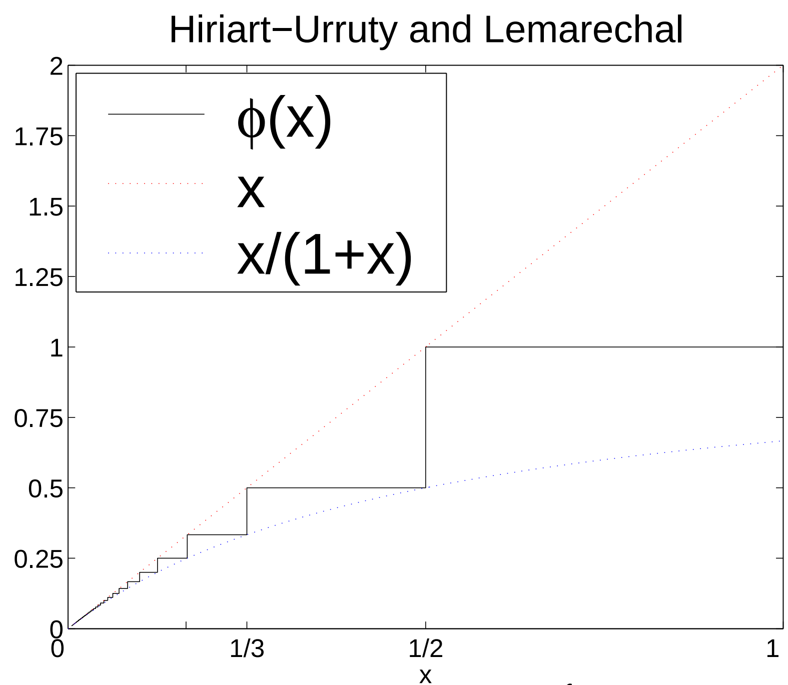


f is Lipschitz and differentiable, near 0:

$$y'(0) = 0 \text{ and } y' = 2x(2 + \sin(\frac{\pi}{x})) - \pi\cos(\frac{\pi}{x})$$

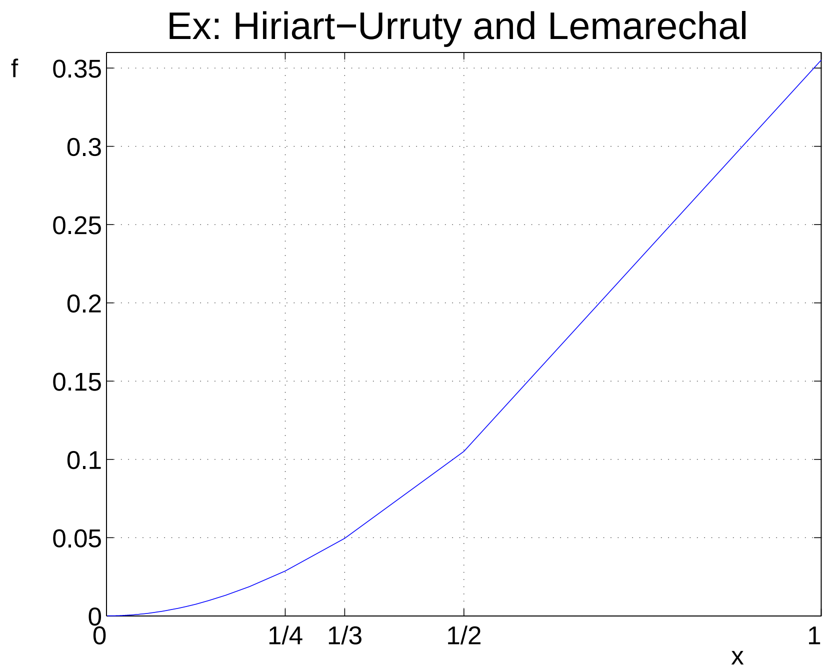
- The derivative is not continuous at 0: $\quad y'(\frac{1}{2k}) = \frac{2}{k} - \pi$
- f is not strictly differentiable : $\quad \partial f(0) = [-\pi, \pi]$
- f is not regular : $\quad f^\circ(0, \pm 1) = \pi \neq f'(0, \pm 1) = 0$

Strictly, not continuously differentiable



$$f(x) = \int_0^x \varphi(u)du \quad \text{where} \quad \varphi(u) = \begin{cases} u & \text{if } u \le 0 \\ \frac{1}{1+\kappa} & \text{if } \kappa + 1 > \frac{1}{u} \ge k \end{cases}$$

Strictly, not continuously differentiable



f is Lipschitz near $\hat{x} = 0$, has kinks at $\frac{1}{\kappa}$ with $\partial f(\frac{1}{\kappa}) = [\frac{1}{\kappa+1}, \frac{1}{\kappa}]$

- f is not strictly differentiable, nor continuously differentiable, in any neighborhood of $\hat{x} = 0$.
- $\partial f(0) = \{0\}$ therefore f is strictly differentiable at $\hat{x} = 0$.

Convergence of coordinate searches

if the sequence of iterates $\{x_k\}$ belongs to a compact set.

- $\lim_k \Delta_k = 0$
- there is an $\hat{x}$ which is the limit of a sequence $\{x_k\}_{k \in K}$ of mesh local optimizers ($f(x_k \pm \Delta_k e_i) \geq f(x_k)$ for all e_i)
- If f is Lipschitz near $\hat{x}$, then $f^\circ(\hat{x}; \pm e_i) \geq 0$ for every e_i

$$\text{Proof:} f^\circ(\hat{x}; e_i) := \limsup_{y \to \hat{x},\ t \downarrow 0} \frac{f(y + te_i) - f(y)}{t}$$

$$\geq \limsup_{k \in K} \frac{f(x_k + \Delta_k e_i) - f(x_k)}{\Delta_k} \geq 0$$

Convergence of coordinate searches

if the sequence of iterates $\{x_k\}$ belongs to a compact set

- $\lim_k \Delta_k = 0$
- there is an $\hat{x}$ which is the limit of a sequence $\{x_k\}_{k \in K}$ of mesh local optimizers
- If f is Lipschitz near $\hat{x}$, then $f^\circ(\hat{x}; \pm e_i) \geq 0$ for every e_i
- If f is regular at $\hat{x}$, then $f'(\hat{x}; \pm e_i) \geq 0$ for every e_i
- If f is strictly differentiable at $\hat{x}$, then $\nabla f(\hat{x}) = 0$

The two phases of Pattern search algorithms

GLOBAL SEARCH on the mesh

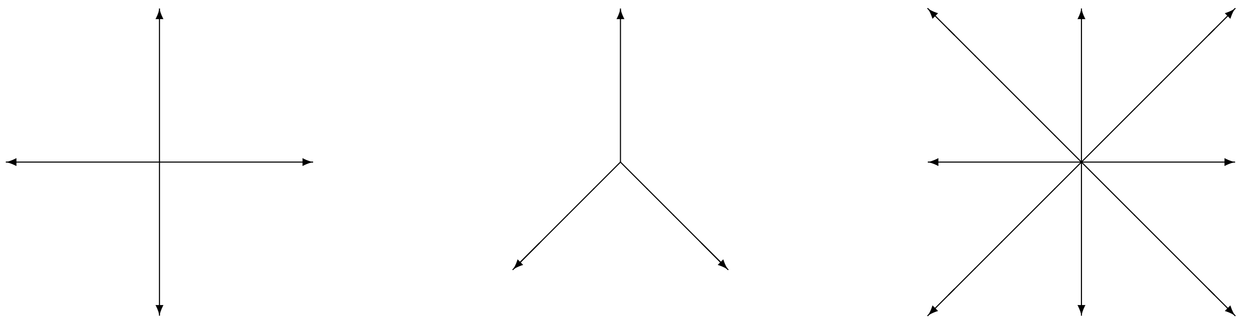
Flexibility,
User heuristics,
Knowledge of the model,
Surrogate functions.

LOCAL POLL around the incumbent solution

More rigidly defined.
Ensures appropriate first order optimality conditions.

Positive spanning sets

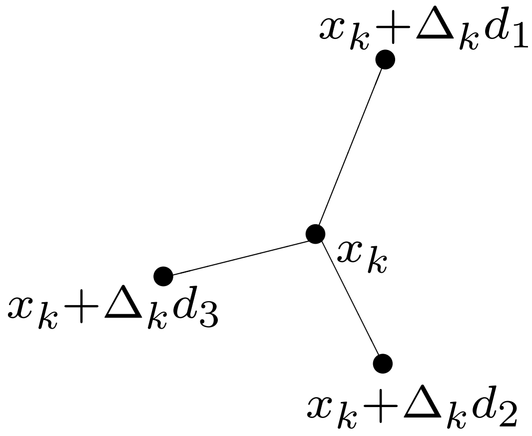
$D \subset \mathbb{R}^n$ is a positive spanning set if non-negative linear combinations of the elements of D span $\mathbb{R}^n$.



D contains at least $n+1$ directions

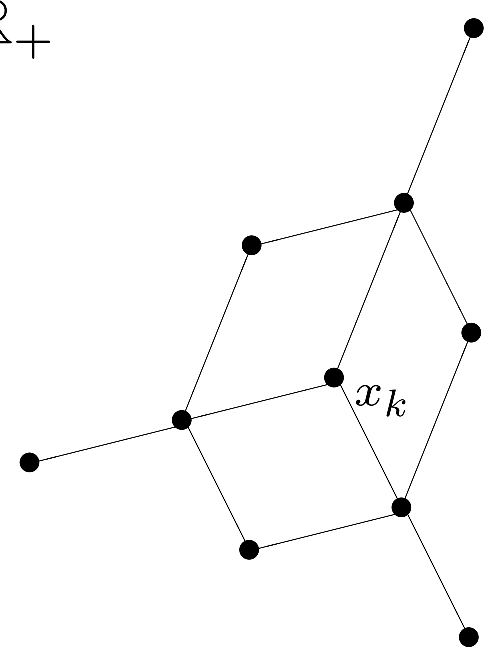
Basic pattern search algorithm for unconstrained optimization

Positive spanning directions: $D_k = \{d_1, d_2, \ldots, d_p\} \subseteq D$
Current best point: $x_k \in \mathbb{R}^n$
Current mesh size parameter: $\Delta_k \in \mathbb{R}_+$



Basic pattern search algorithm for unconstrained optimization

Positive spanning directions: $D_k = \{d_1, d_2, \ldots, d_p\} \subseteq D$
Current best point: $x_k \in \mathbb{R}^n$
Current mesh size parameter: $\Delta_k \in \mathbb{R}_+$



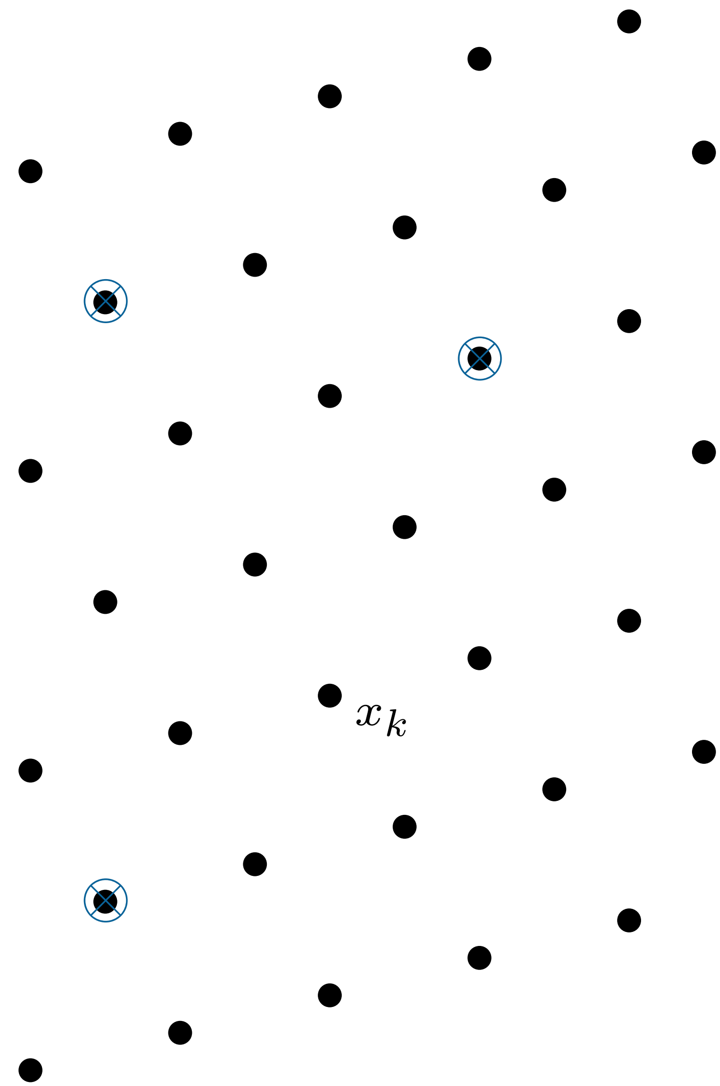
Search step (global)

Given Δ_k, x_k:

SEARCH anywhere on M_k.

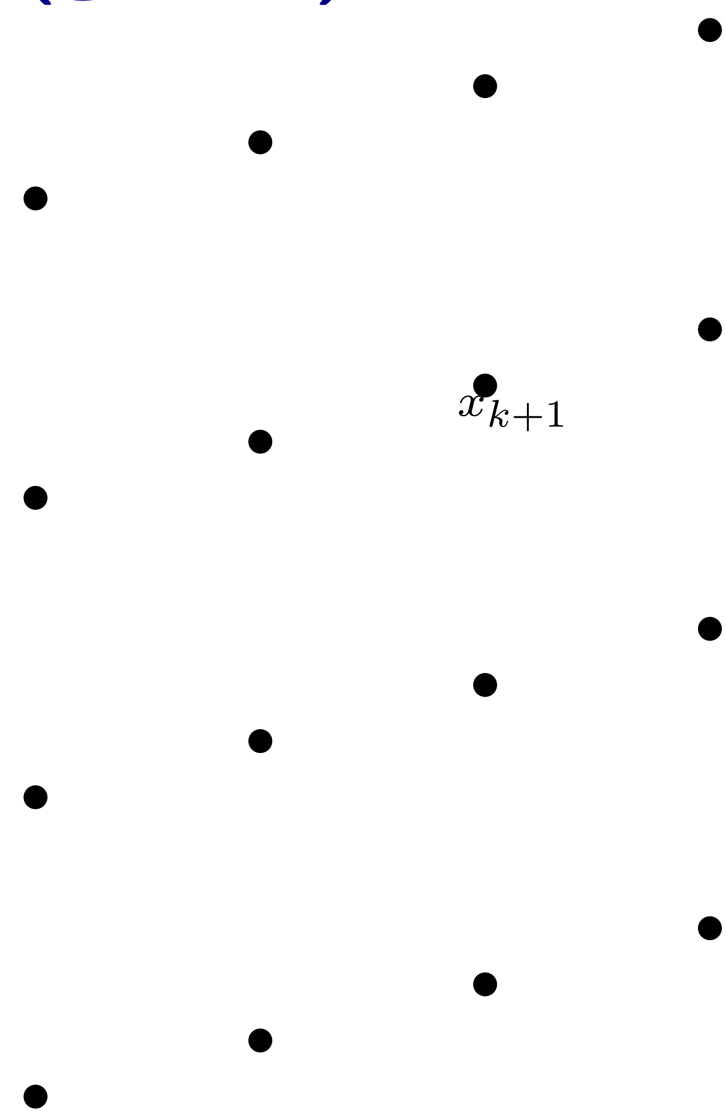
If an IMPROVED MESH POINT is found *i.e.* $f(x_{k+1}) < f(x_k)$

then set $\Delta_{k+1} \geq \Delta_k$,



Search step (global)

Given Δ_k, x_k:



SEARCH anywhere on M_k.
If an IMPROVED MESH POINT is found *i.e.* $f(x_{k+1}) < f(x_k)$

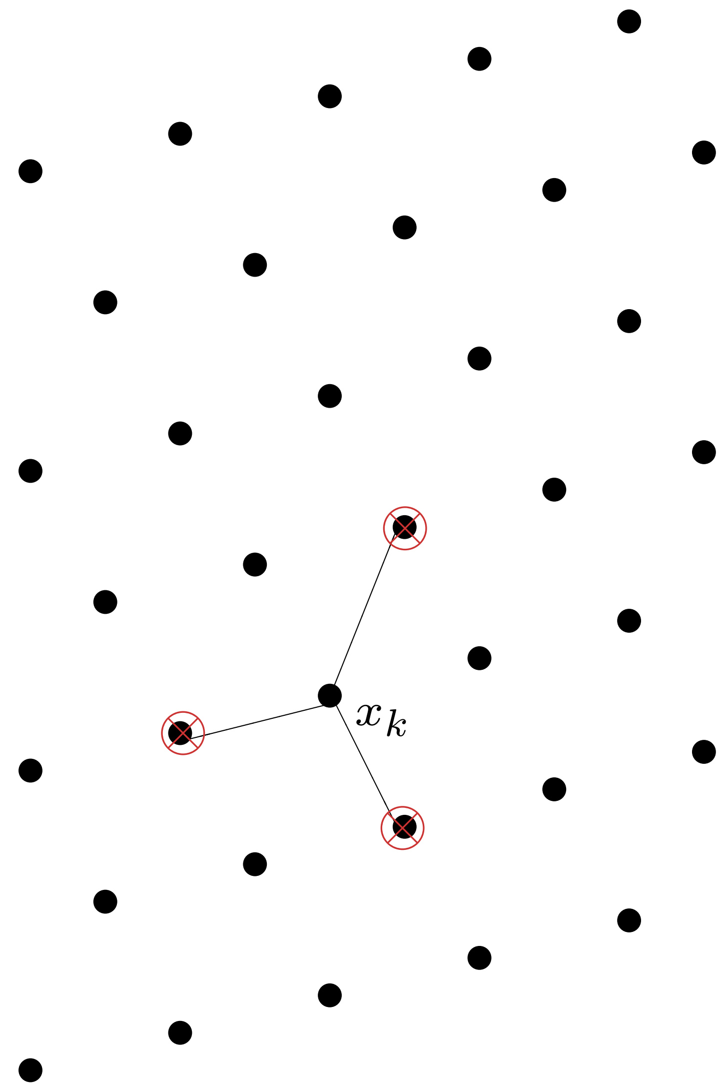
then set $\Delta_{k+1} \geq \Delta_k$,

and restart the SEARCH from this improved point.

Poll step (local)

If SEARCH fails,
POLL at mesh neighbors:

If an IMPROVED MESH POINT is found *i.e.* $f(x_{k+1}) < f(x_k)$, set $\Delta_{k+1} \geq \Delta_k$, and restart SEARCH from improved point.

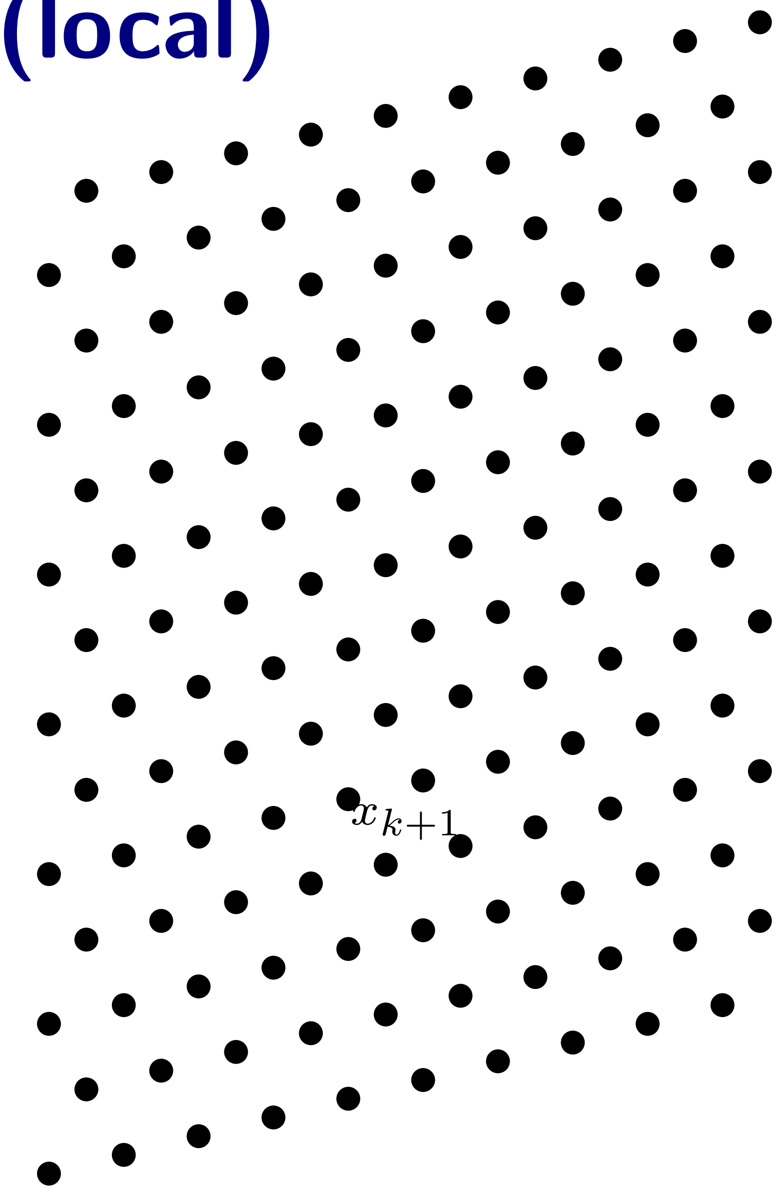


Poll step (local)

If SEARCH fails,
POLL at mesh neighbors:

If an IMPROVED MESH POINT is found *i.e.* $f(x_{k+1}) < f(x_k)$, set $\Delta_{k+1} \geq \Delta_k$, and restart SEARCH from improved point.

Else $x_{k+1} = x_k$ is a MESH LOCAL OPTIMIZER. Set $\Delta_{k+1} < \Delta_k$, and restart SEARCH from this point.



Convergence results

If all iterates are in a compact set then

- $\liminf_k \Delta_k = 0$ (the proof is non trivial since Δ_{k+1} may increase)
- For every limit point $\hat{x}$ of any subsequence $\{x_k\}_{k \in K}$ of mesh local optimizers where $\{\Delta_k\}_{k \in K} \to 0$, and for the set $\hat{D}$ of POLL directions used infinitely many times in this subsequence
- If f is Lipschitz near $\hat{x}$, then $f^{\circ}(\hat{x}; d) \geq 0$ for every $d \in \hat{D}$
- If f is regular at $\hat{x}$, then $f'(\hat{x}; d) \geq 0$ for every $d \in \hat{D}$.
- If f is strictly differentiable at $\hat{x}$, then $\nabla f(\hat{x}) = 0$.

The convergence results do not state that

if the sequence of iterates $\{x_k\}$ belongs to a compact set

- $\lim_k \Delta_k = 0$

- If f is continuously differentiable everywhere then $\nabla f(\hat{x}) = 0$ for any limit point $\hat{x}$ of the sequence of iterates.

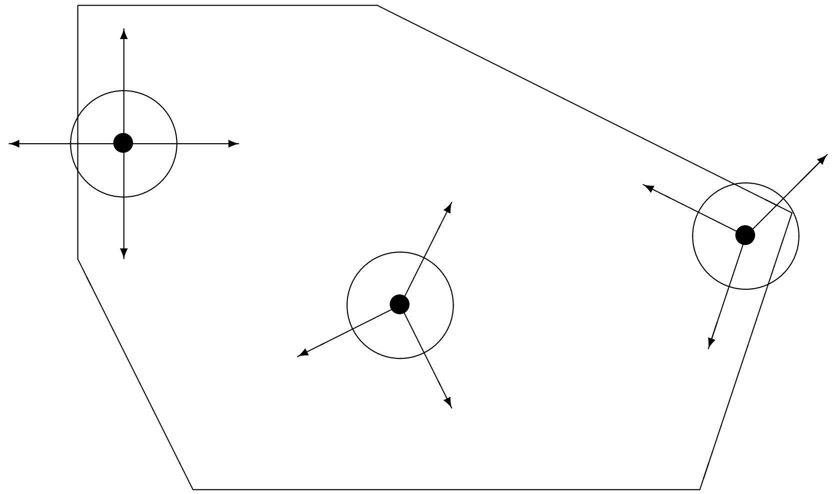
- If the entire sequence of iterates converges (at $\hat{x}$ say), and if f is differentiable then $\nabla f(\hat{x}) = 0$.

- If the entire sequence of iterates converges (at $\hat{x}$ say), and if f is Lipschitz near $\hat{x}$ then $0 \in \partial f(\hat{x})$. Thus, the method does not necessarily produce a stationary point in the Clarke sense.

Basic pattern search algorithm for linearly constrained optimization

- Infeasible trial points are pruned (the objective function is not evaluated and set to infinity).

- When x_k is close to the boundary of the feasible region, the POLL directions must conform to the boundary.



Convergence results – linearly constraints

If all iterates are in a compact set then

- $\liminf_{k} \Delta_k = 0$
- For every limit point $\hat{x}$ of any subsequence $\{x_k\}_{k \in K}$ of mesh local optimizers where $\{\Delta_k\}_{k \in K} \to 0$, and for the set $\hat{D}$ of POLL directions used infinitely many times in this subsequence
- If f is Lipschitz near $\hat{x}$, then $f^\circ(\hat{x}; d) \geq 0$ for every $d \in \hat{D} \cap T_X(\hat{x})$
- If f is regular at $\hat{x}$, then $f'(\hat{x}; d) \geq 0$ for every $d \in \hat{D} \cap T_X(\hat{x})$.
- If f is s.d. at $\hat{x}$, then $\nabla f(\hat{x})^T v \geq 0$ for every $v \in T_X(\hat{x})$

Positive spanning sets

- A richer set of D increases the number of directions in $\hat{D}$, and thus the directions for which $f^\circ(\hat{x}; d) \geq 0$.
- User's domain specific knowledge can help choose $D_k \subset D$.
- Theory limited to a finite number of directions in D, so the *barrier* approach (reject infeasible points) does not apply to general constraints because a finite number can not conform to the boundary of Ω.

General nonlinear constrained optimization

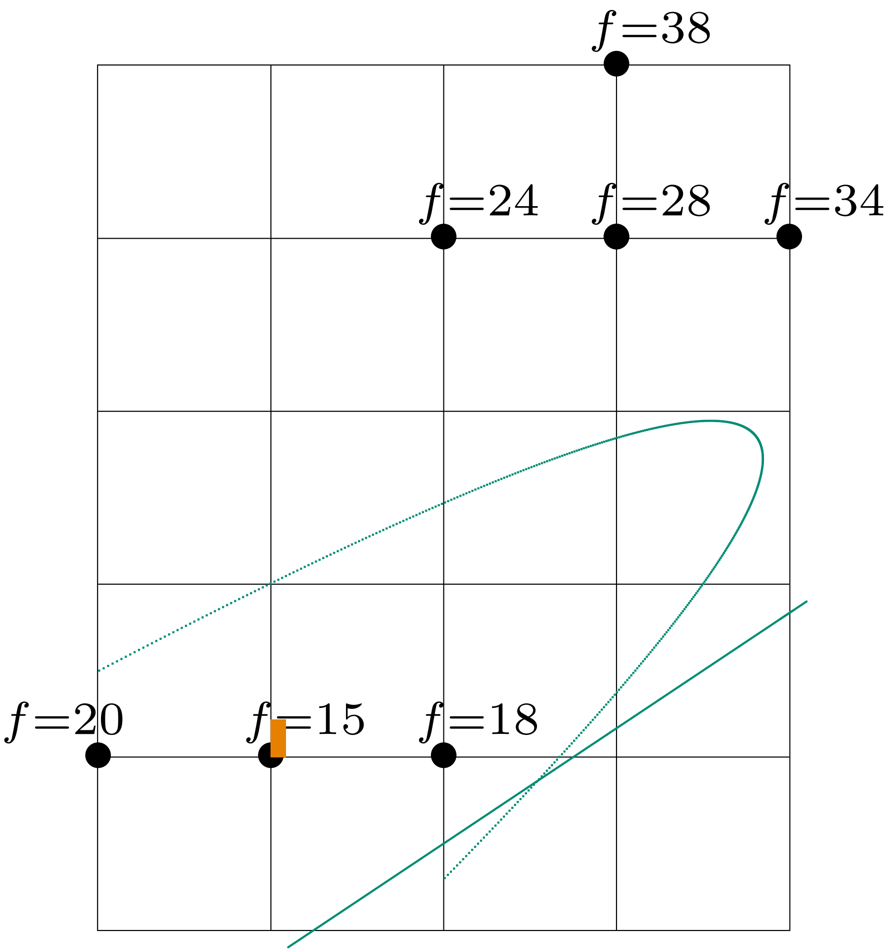
$$\min_{x \in X} \quad f(x)$$
$$\text{s.t.} \quad x \in \Omega \equiv \{x \mid C(x) \leq 0\}, \quad \text{where } C = (c_1, c_2, \ldots, c_m)^T$$

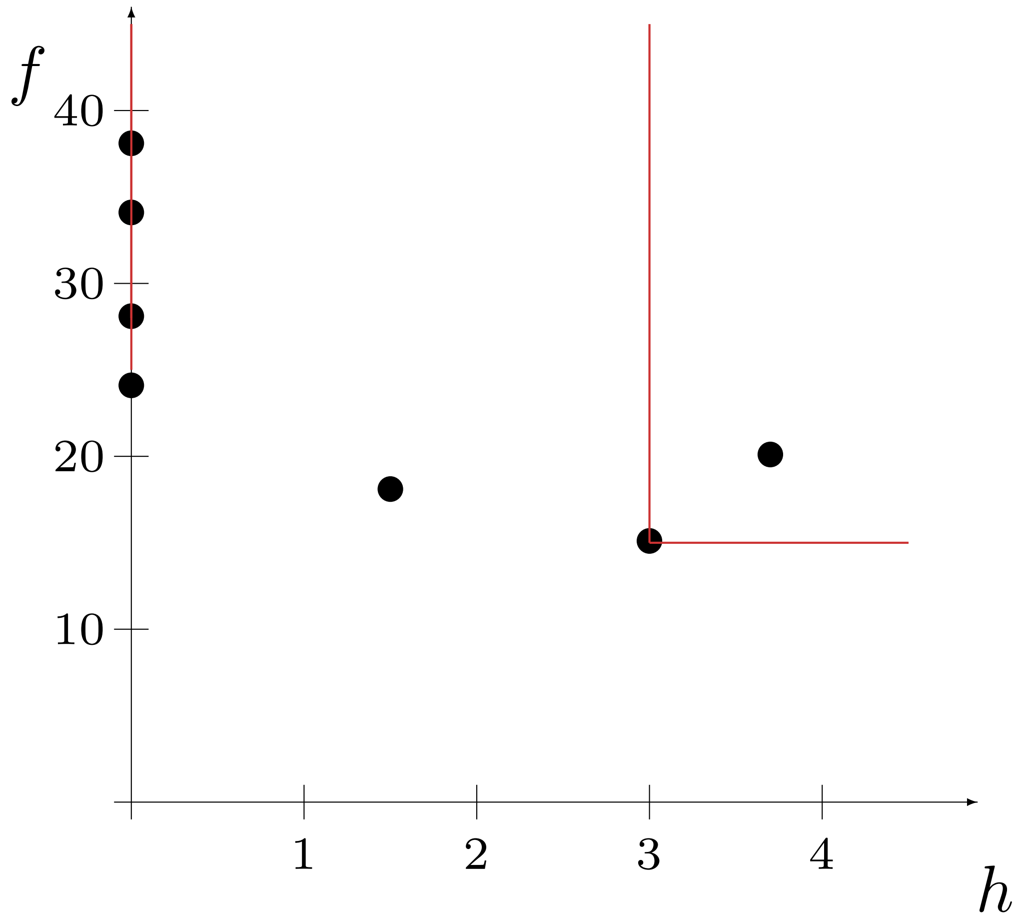
X is defined by bound and linear constraints.
Define the nonnegative constraint violation function $h : \mathbb{R}^n \to \mathbb{R}$

$$h(x) = \sum_j \max(0, c_j(x))^2.$$

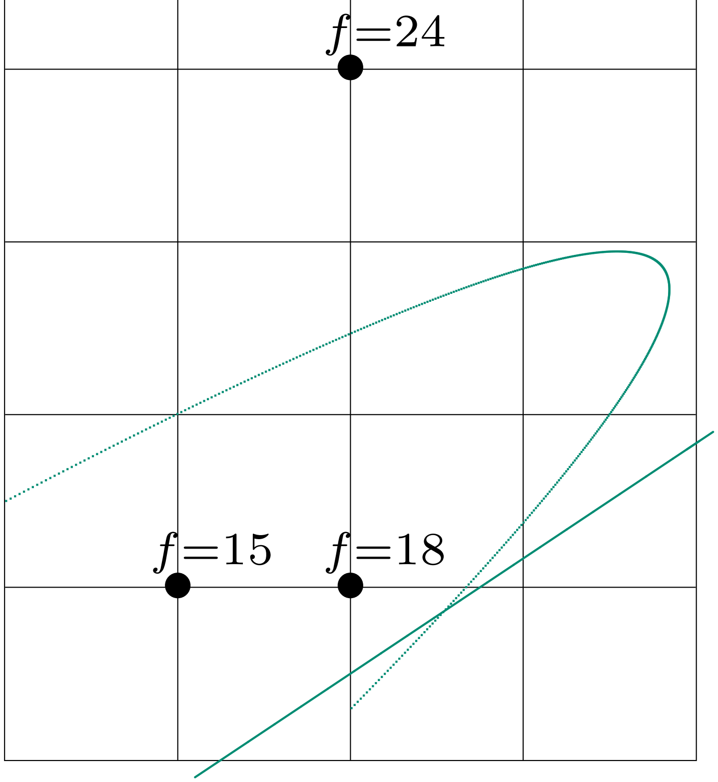
Note that $h(x) = 0$ iff $x \in \Omega$, and h inherits smoothness from C.
We look at the biobjective optimization problem where a priority is given to the minimization of h over the minimization of f.

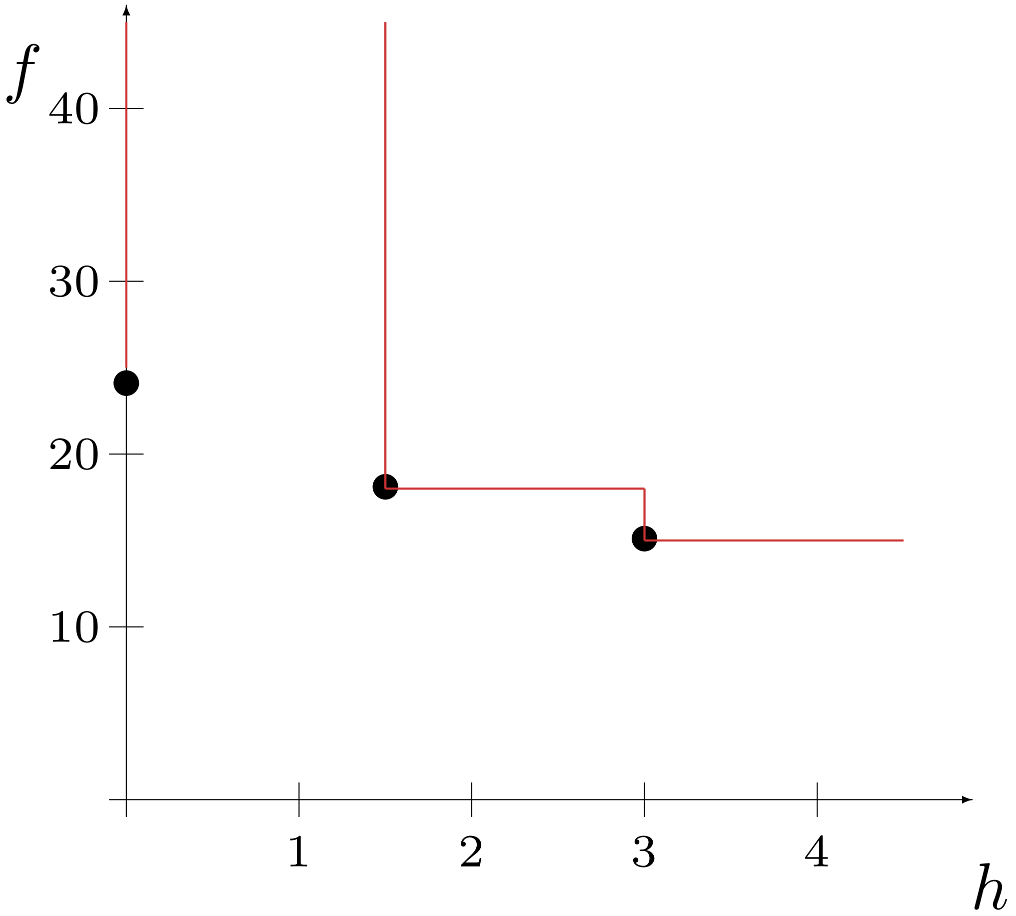
Pattern Search with filter



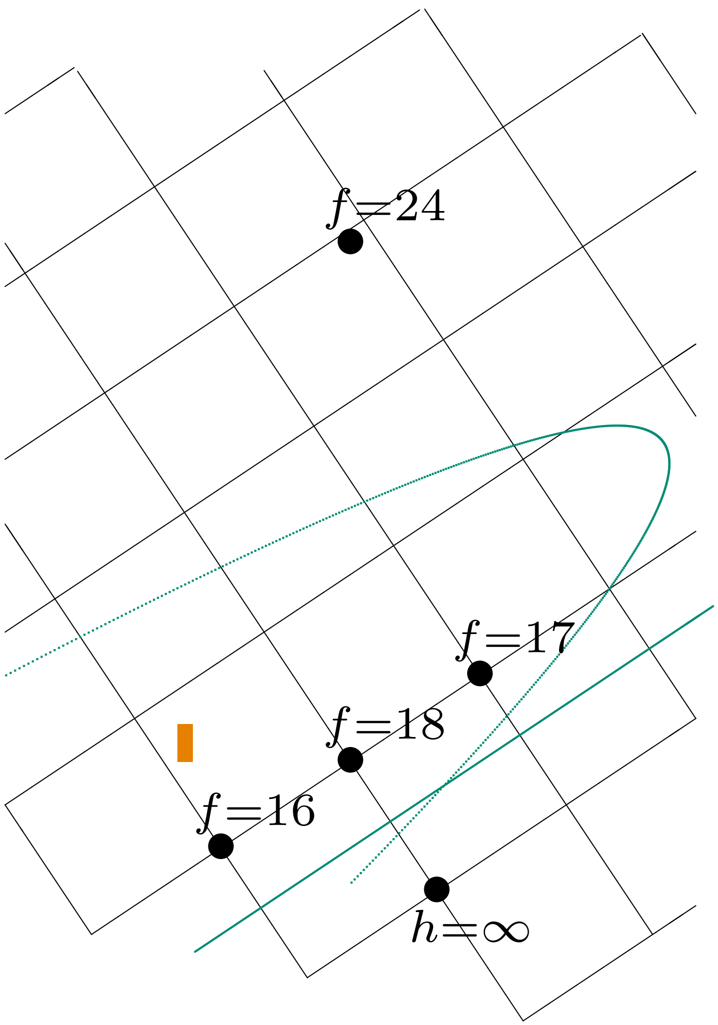


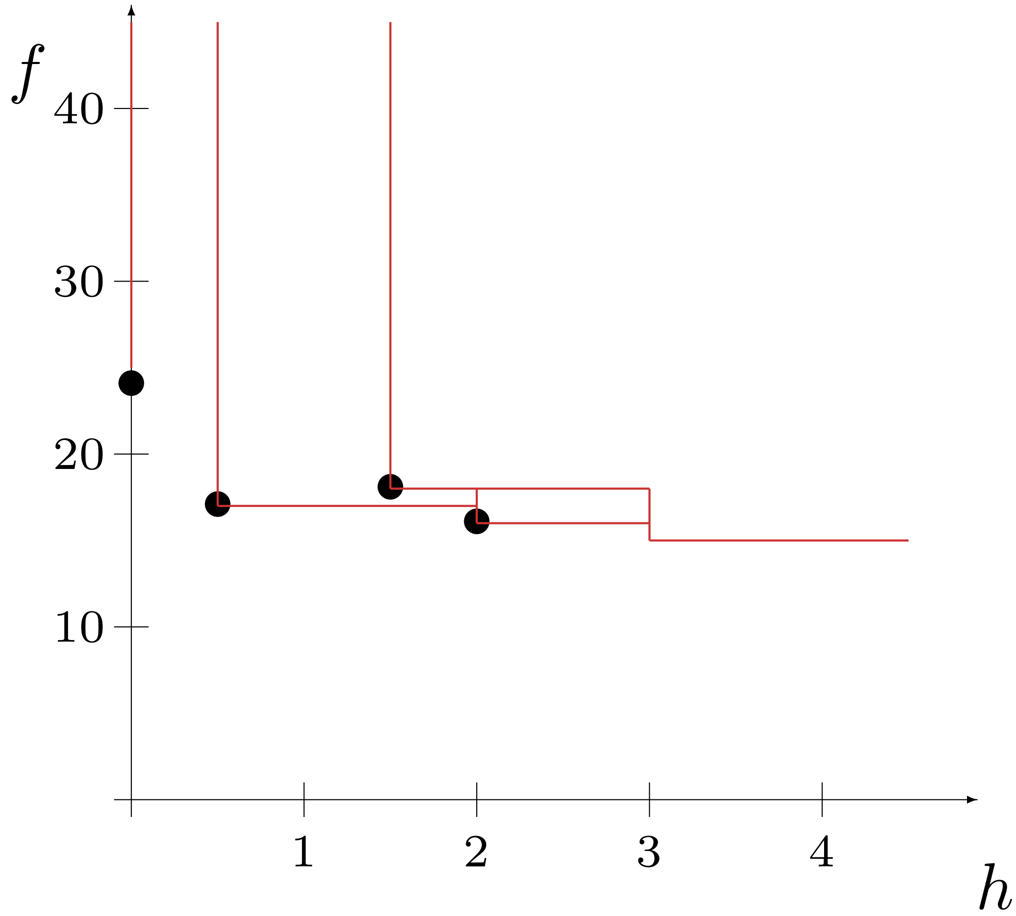
Pattern Search with filter



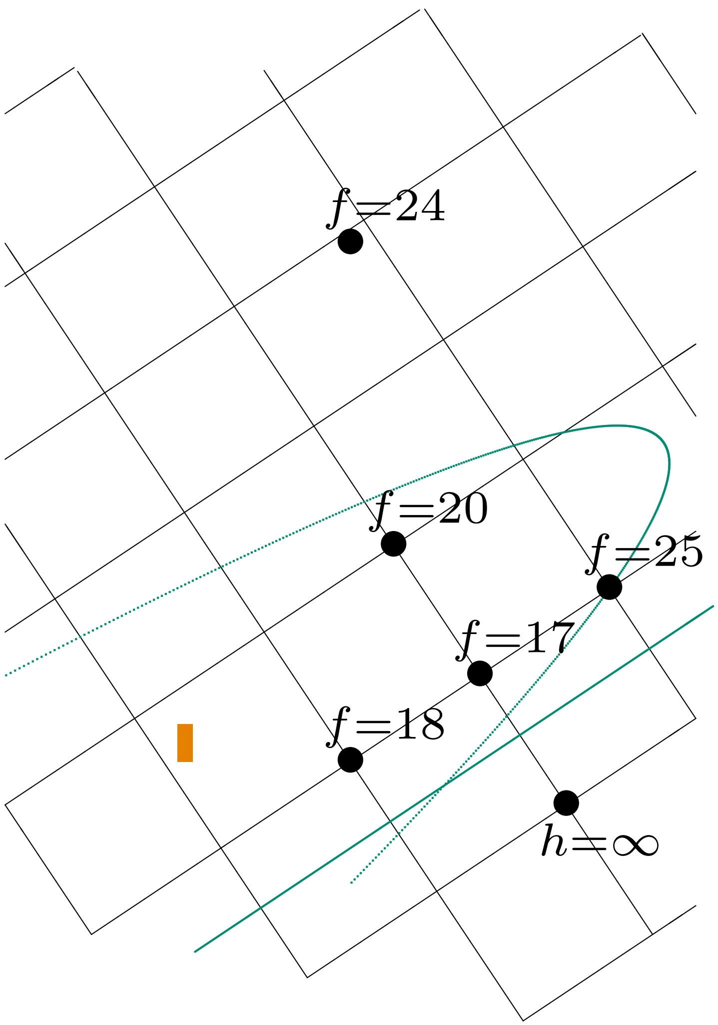


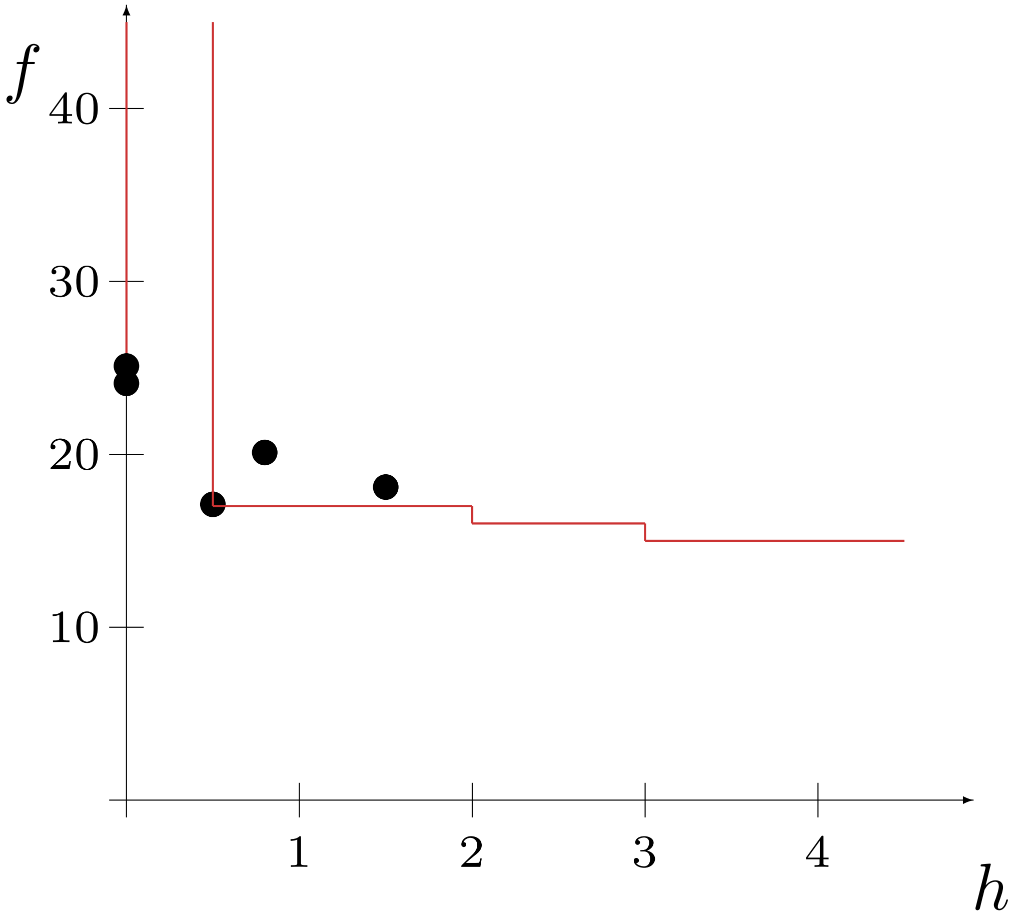
Pattern Search with filter





Pattern Search with filter





Mesh isolated filter point

Convergence results – general constraints

Polling around least infeasible point gives priority to the minimization of h versus the minimization of f.

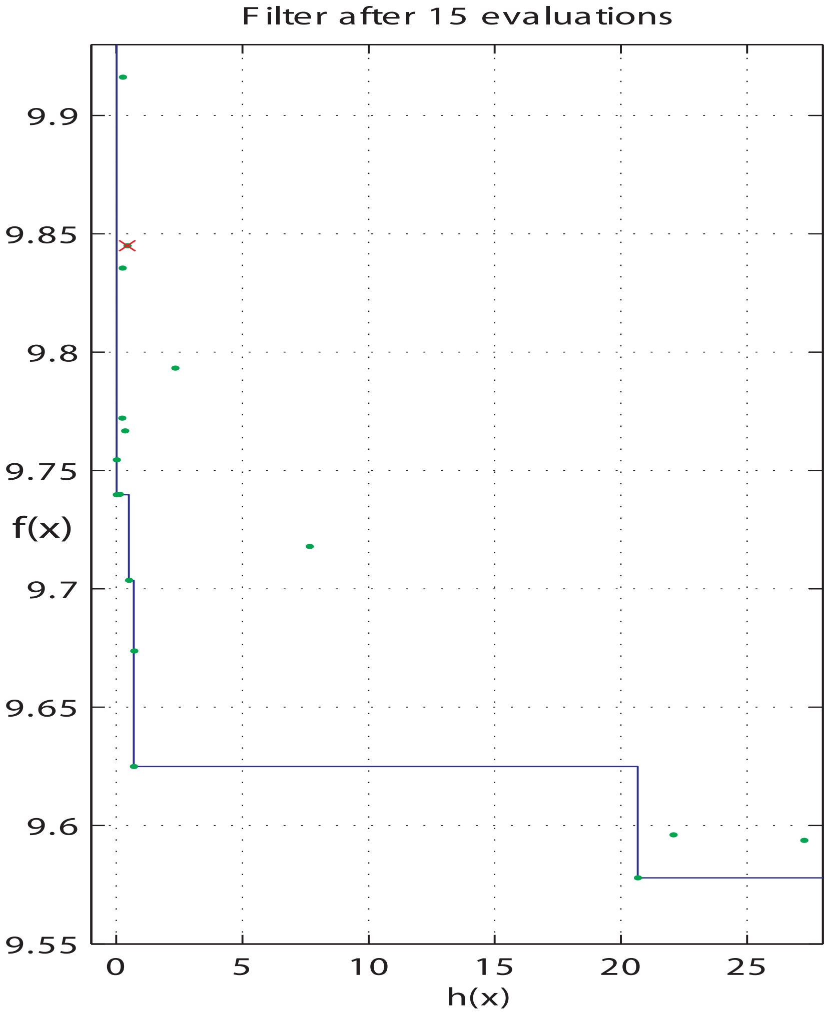
If all iterates are in a compact set then

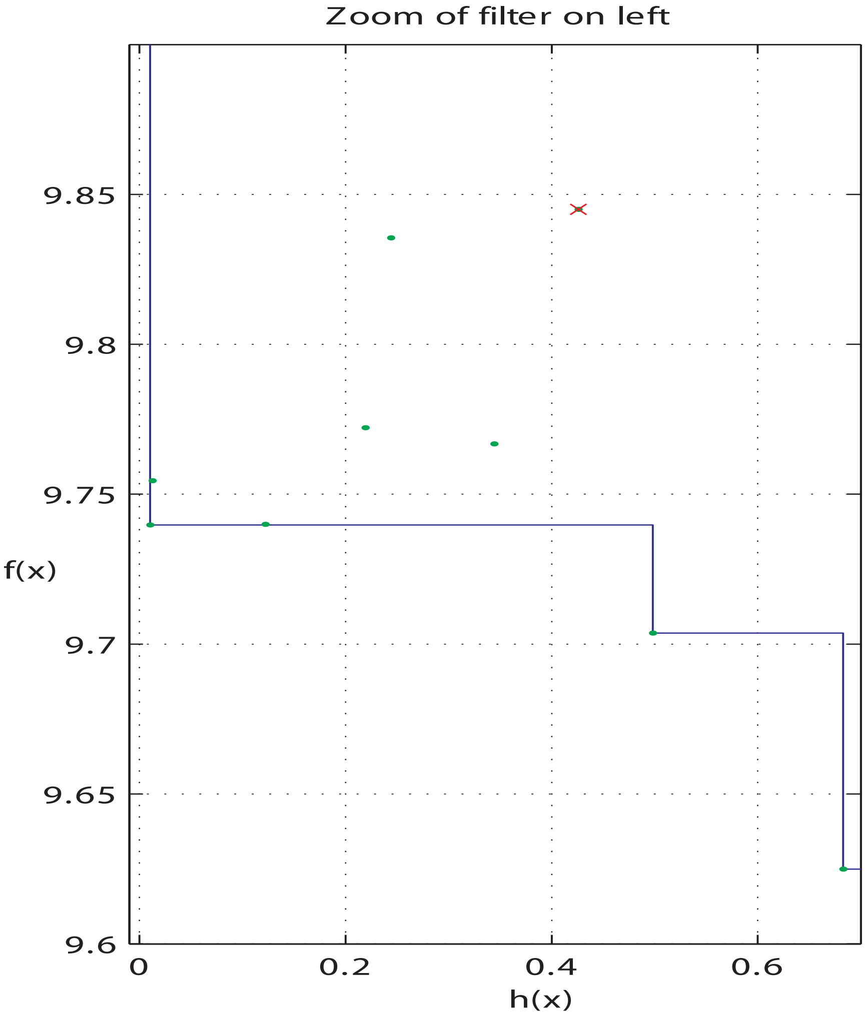
- $\liminf_k \Delta_k = 0$
- For every limit point $\hat{p}$ of any subsequence $\{p_k\}_{k \in K}$ of mesh isolated poll centers where $\{\Delta_k\}_{k \in K} \to 0$, and for the set $\hat{D}$ of POLL directions used infinitely many times in this subsequence
- If h is Lipschitz near $\hat{p}$, and $\hat{p}$ is feasible then $h^\circ(\hat{x}; v) \geq 0$ for every $v \in T_X(\hat{p})$
- If h is Lipschitz near $\hat{p}$, then $h^\circ(\hat{p}; d) \geq 0$ for every $d \in \hat{D} \cap T_X(\hat{p})$

- If h is regular at $\hat{p}$, then $h'(\hat{x}; d) \geq 0$ for every $d \in \hat{D} \cap T_X(\hat{p})$
- If h is s.d. at $\hat{p}$, then $\nabla h(\hat{x})^T v \geq 0$ for every $v \in T_X(\hat{x})$

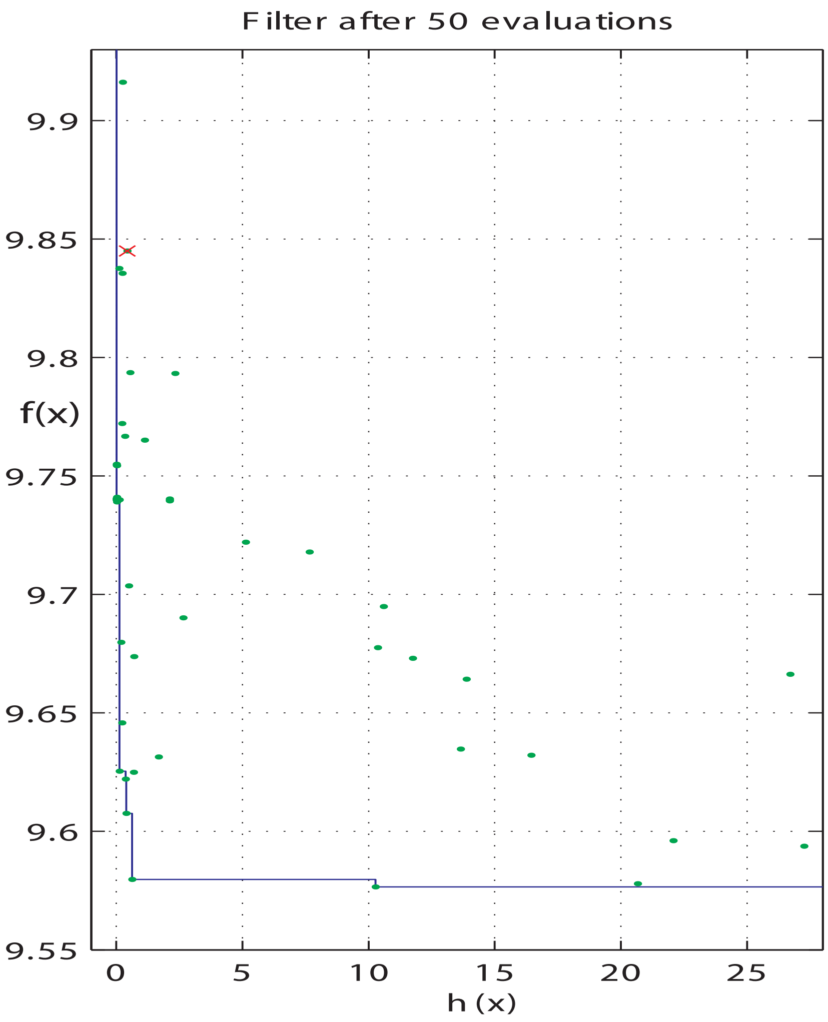
Note: The same convergence results hold for f, with an additional requirement: $\hat{p}$ must be strictly feasible with respect to the general constraints.

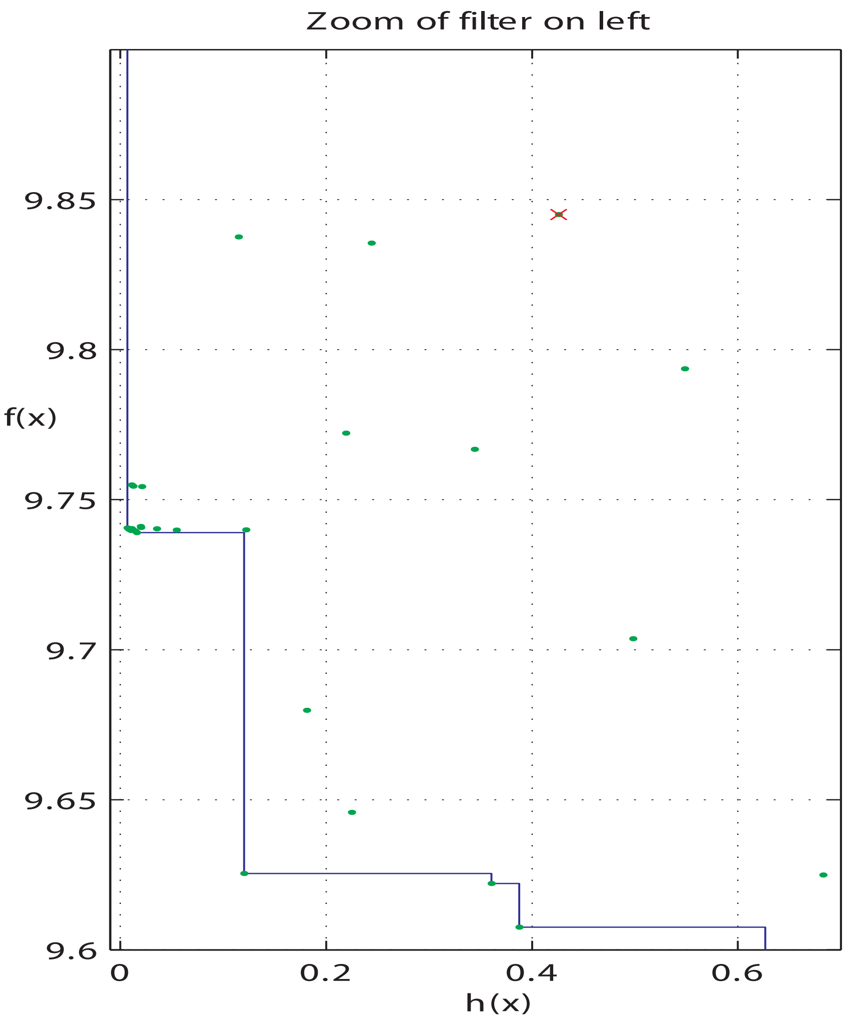
A planform filter : 17 vars, 13 ctrs, kriging



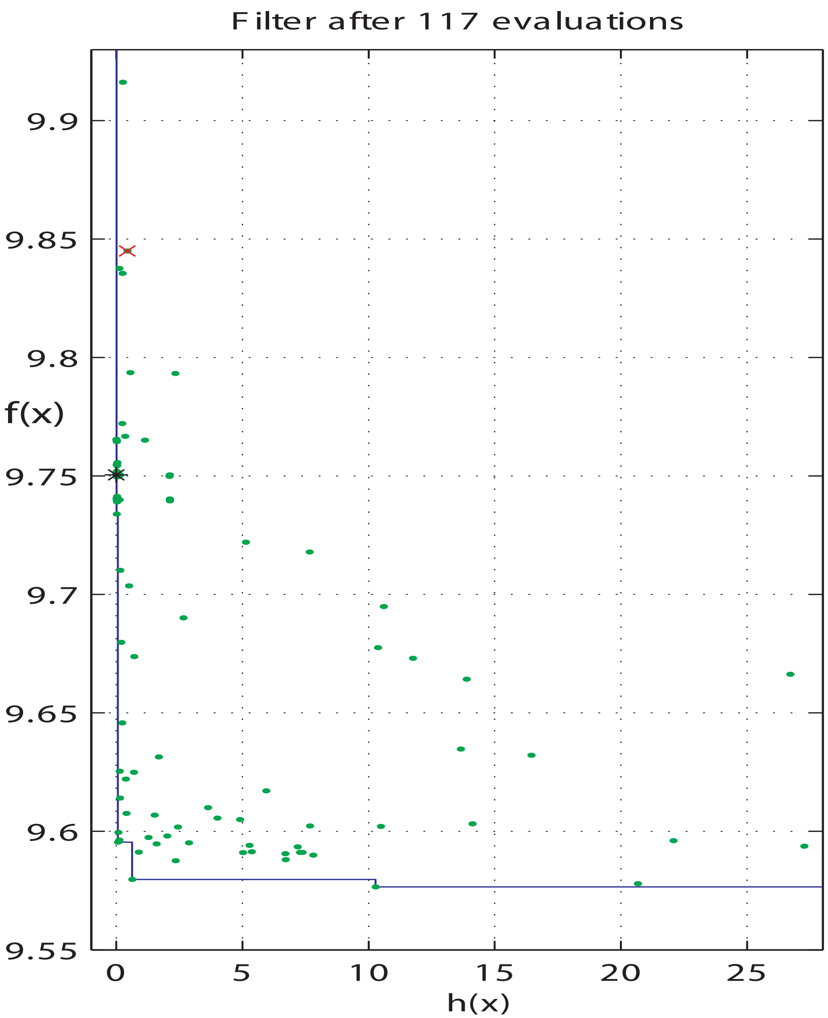


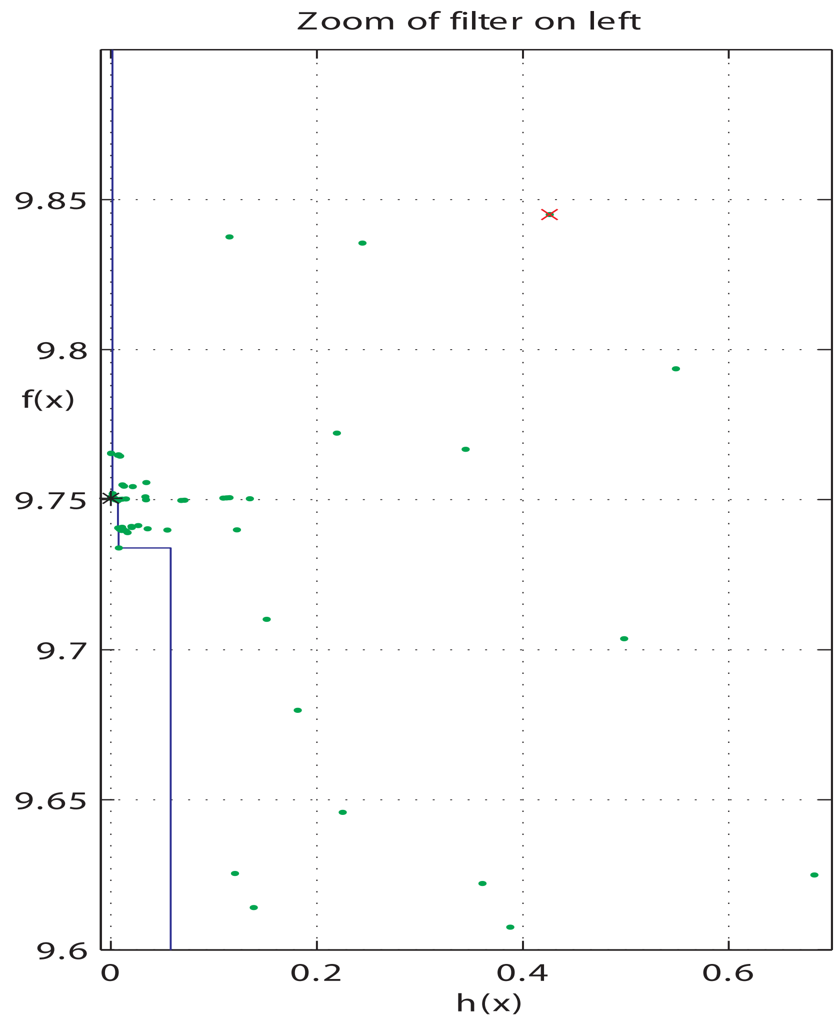
A planform filter : 17 vars, 13 ctrs, kriging





A planform filter : 17 vars, 13 ctrs, kriging





Discussion

- GPS algorithms are directional algorithms:
  Analysis is easy with Clarke calculus ▮
- Optimality results depend on local differentiability
  The analysis can be extended by considering other directions ▮
- Linear and bound constraints are treated by appropriate polling directions ▮
- General constraints are treated by the filter ▮
- GPS for problems with categorical variables
  Tomorrow's talk by Mark Abramsom