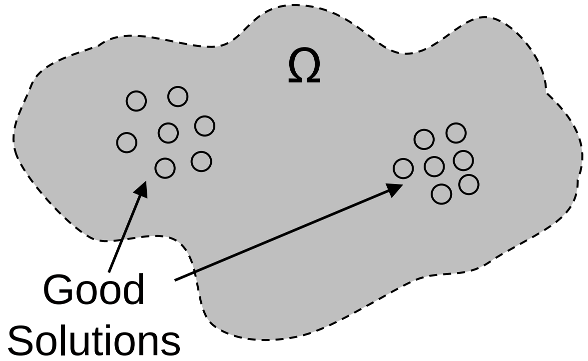
Algorithmic Methods for Mathematical Models (AMMM)

Intensification and Diversification

Luis Velasco
(lvelasco @ ac.upc.edu)
Campus Nord D6-107

Intensification

- Portions of the search space that **seem promising** should be explored more thoroughly to make sure that the best solutions in these areas are indeed found.
- From time to time, one would thus stop the normal searching process to perform an **intensification phase**.
- Intensification is based on some *intermediate-term* memory
  - number of consecutive iterations that various solution components have been present in good solutions.



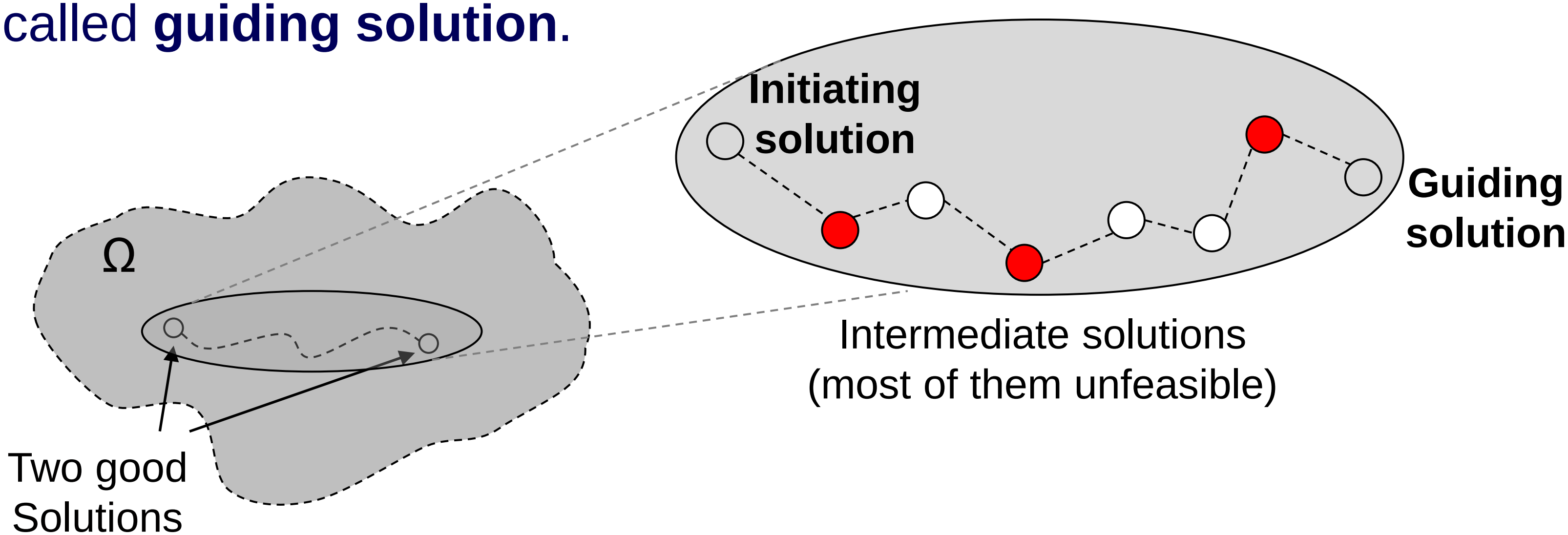
- Intensification is not always necessary.
  - There are many situations where the search performed by the normal searching process is enough.

Diversification

- One of the main problems of all methods based on LS is that they tend to be too "local" (as their name implies).
  - they tend to spend most, if not all, of their time in a restricted portion of the search space.
- Diversification forces the search into previously unexplored areas of the search space.
- It is based on some form of *long-term memory* of the search
  - total number of iterations (since the beginning of the search) that various "solution components" have been present in the current solution or involved in the selected moves.
- Rarely used components can be forced to the current solution (or the best-known solution) and restarting the search from this point.
- Another option is to use frequency as cost such that the components with higher frequency are penalized.

Path Relinking (PR)

- Path relinking (PR) **integrates intensification and diversification** strategies in a search scheme.
- It generates new solutions by exploring trajectories that connect high-quality solutions.
- It starts from one solution, called an **initiating solution**, and generating a path in the neighborhood space that leads toward another solution, called **guiding solution**.



PR algorithm

Given solutions a and b (a: *initiating* and b: *guiding* solution)

$f^* = \min\{f(a), f(b)\}$

$x^* = \text{argmin}\{f(a), f(b)\}$

$x = a$

while $x \neq b$ **do**

    $u^* \leftarrow \text{argmin}\{f(x+u), \forall\, u \in b\backslash x\}$

    $x \leftarrow x \cup \{u^*\}$

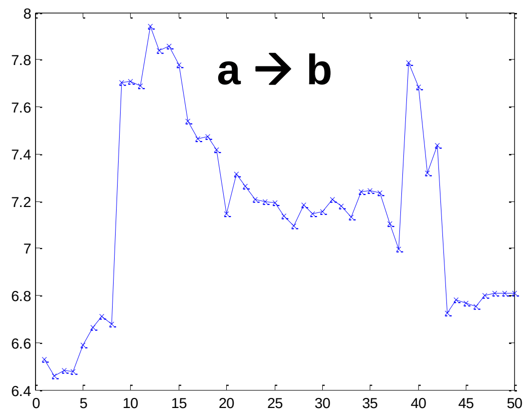
    ($x \leftarrow$ remove element $v \in x\backslash b$ from x, so as to make x feasible)

    if $f(x) < f^*$ **then**

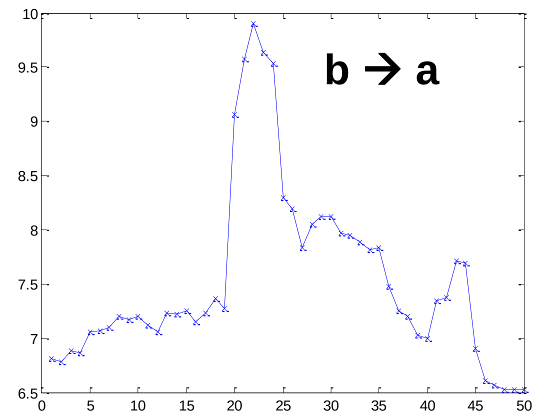
        $f^* \leftarrow f(x)$

        $x^* \leftarrow x$

end



Feasible Solutions



Feasible Solutions

PR: Example

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
init	3	1	3	3	3	1	1	5	1	5
gui	3	1	3	3	1	3	3	1	1	5

Initiating Solution

u	1	2	3	4	5	6	7	8	9	10
a	3	1	3	3	3	1	1	5	1	5

a	1	2	3	4	5
m	R3		R3		R1
U(a)	{2,6,7,9}		{1,3,4,5}		{8,10}
km-cr	1		1		0

Cost=460

u	1	2	3	4	5	6	7	8	9	10
a	3	1	3	3	3	1	3	5	1	5

a	1	2	3	4	5
m	R2		R3		R1
U(a)	{2,6,9}		{1,3,4,5,7}		{8,10}
km-cr	0		0		0

Cost=420

u	1	2	3	4	5	6	7	8	9	10
a	3	1	3	3	1	1	3	5	1	5

a	1	2	3	4	5
m	R3		R2		R1
U(a)	{2,5,6,9}		{1,3,4,7}		{8,10}
km-cr	0		0		0

Cost=420

u	1	2	3	4	5	6	7	8	9	10
a	3	1	3	3	1	3	3	5	1	5

a	1	2	3	4	5
m	R2		R3		R1
U(a)	{2,5,9}		{1,3,4,6,7}		{8,10}
km-cr	0		0		0

Cost=420

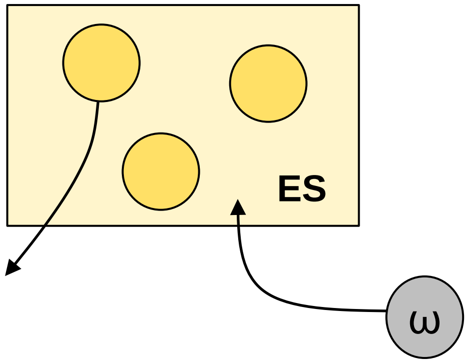
Guiding Solution

u	1	2	3	4	5	6	7	8	9	10
a	3	1	3	3	1	3	3	1	1	5

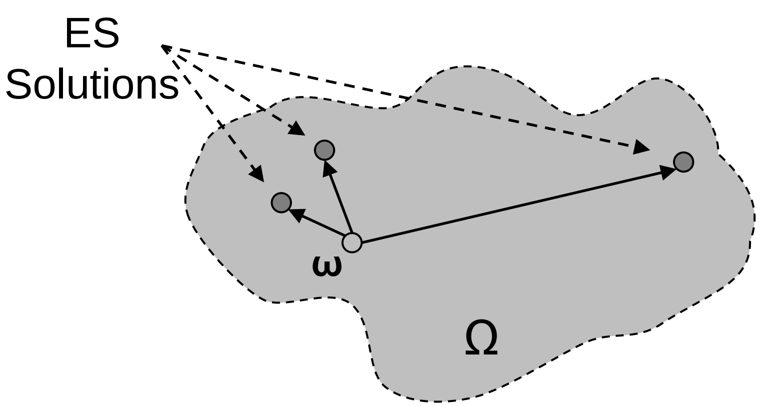
a	1	2	3	4	5
m	R3		R3		R1
U(a)	{2,5,8,9}		{1,3,4,6,7}		{10}
km-cr	0		0		2

Cost=460

Elite Set (ES)

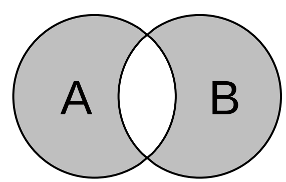


- S enters ES if either:
  - f(ω) < f(bestES) or
  - f(ω) < f(worstES) and d(ω,ES) ≥ d(ES)
- if ω enters ES one solution in ES must leave ES:
  - closest ω’ in ES to ω with f(ω’) ≥ f(ω)



$$d(\omega,ES)=\min \{|\omega \oplus \omega'|, \omega' \in ES\}$$

$$d(ES)=\min \{|\omega' \oplus \omega''|, \omega',\omega'' \in ES\}$$

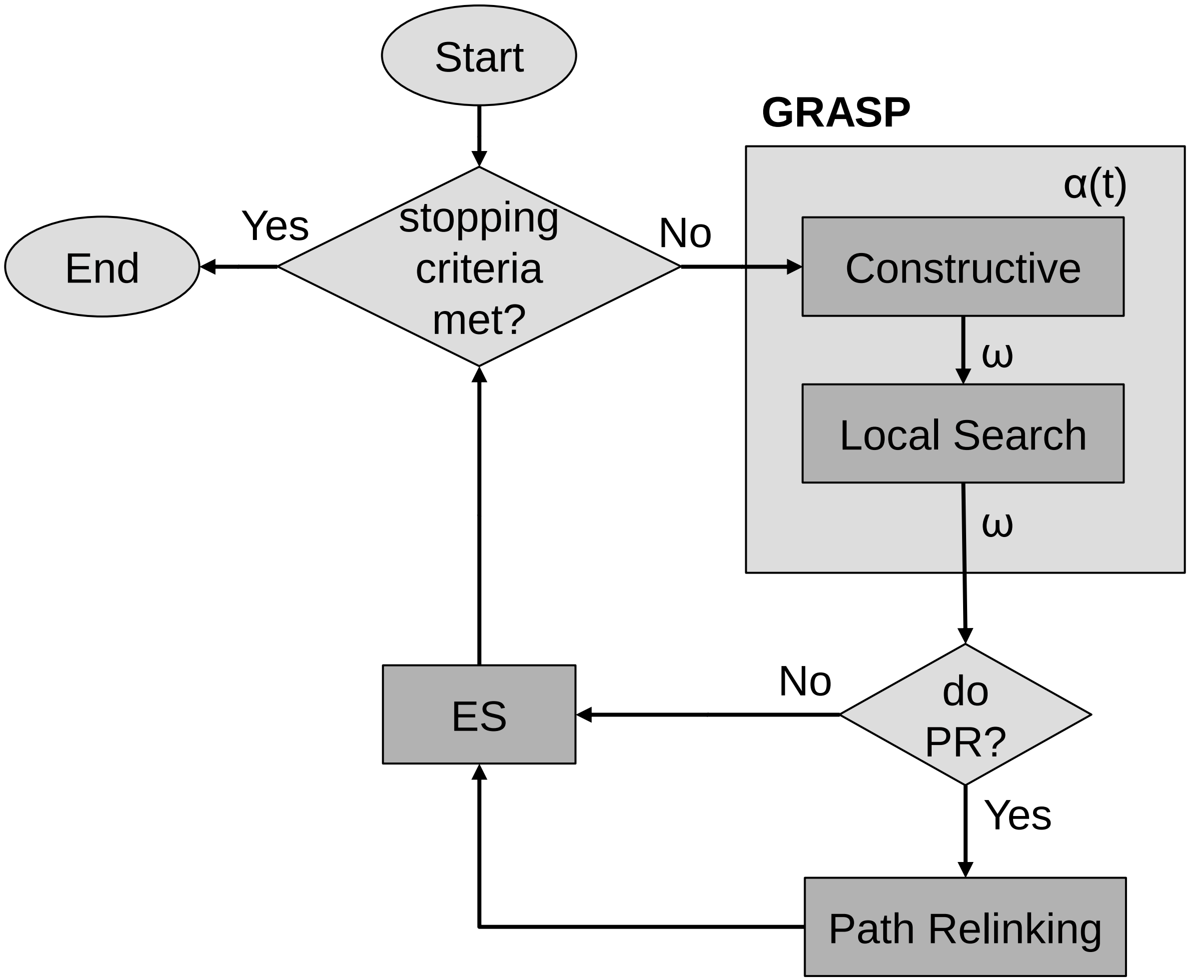


Symmetric difference

$A \oplus B = (A \cup B) \setminus (A \cap B)$

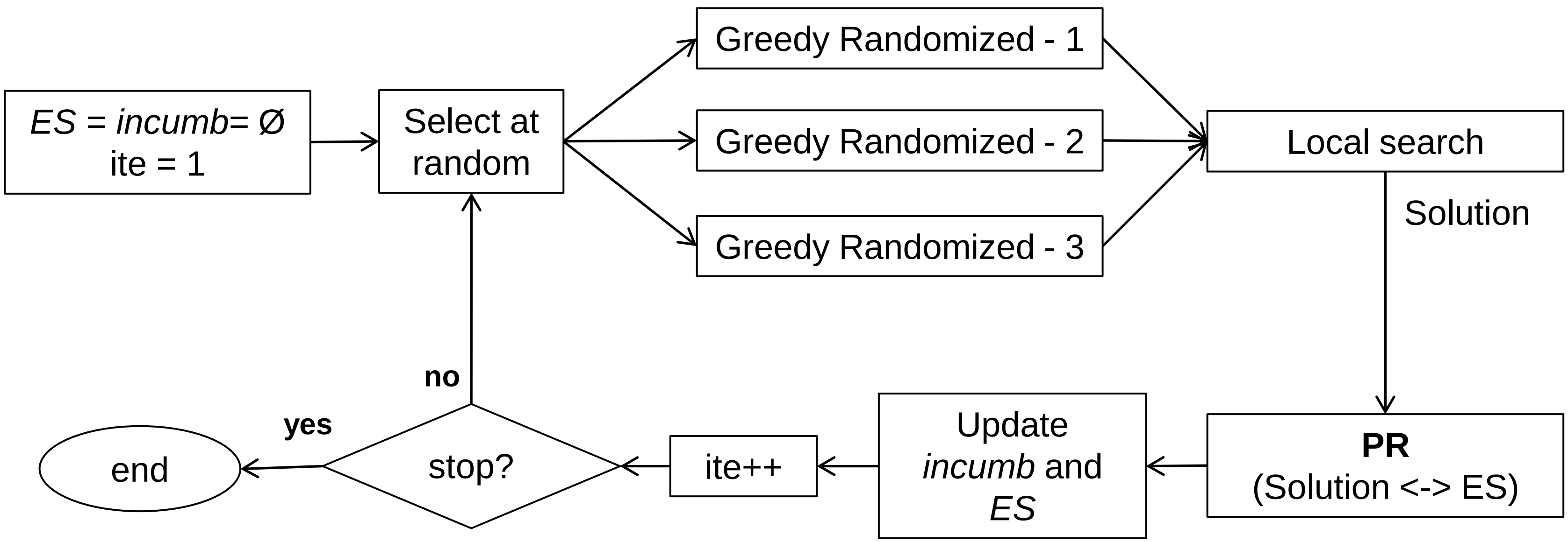
GRASP with PR

- Heuristic hybridization: Combine several techniques.
  - GRASP + PR -> Diversification + Intensification

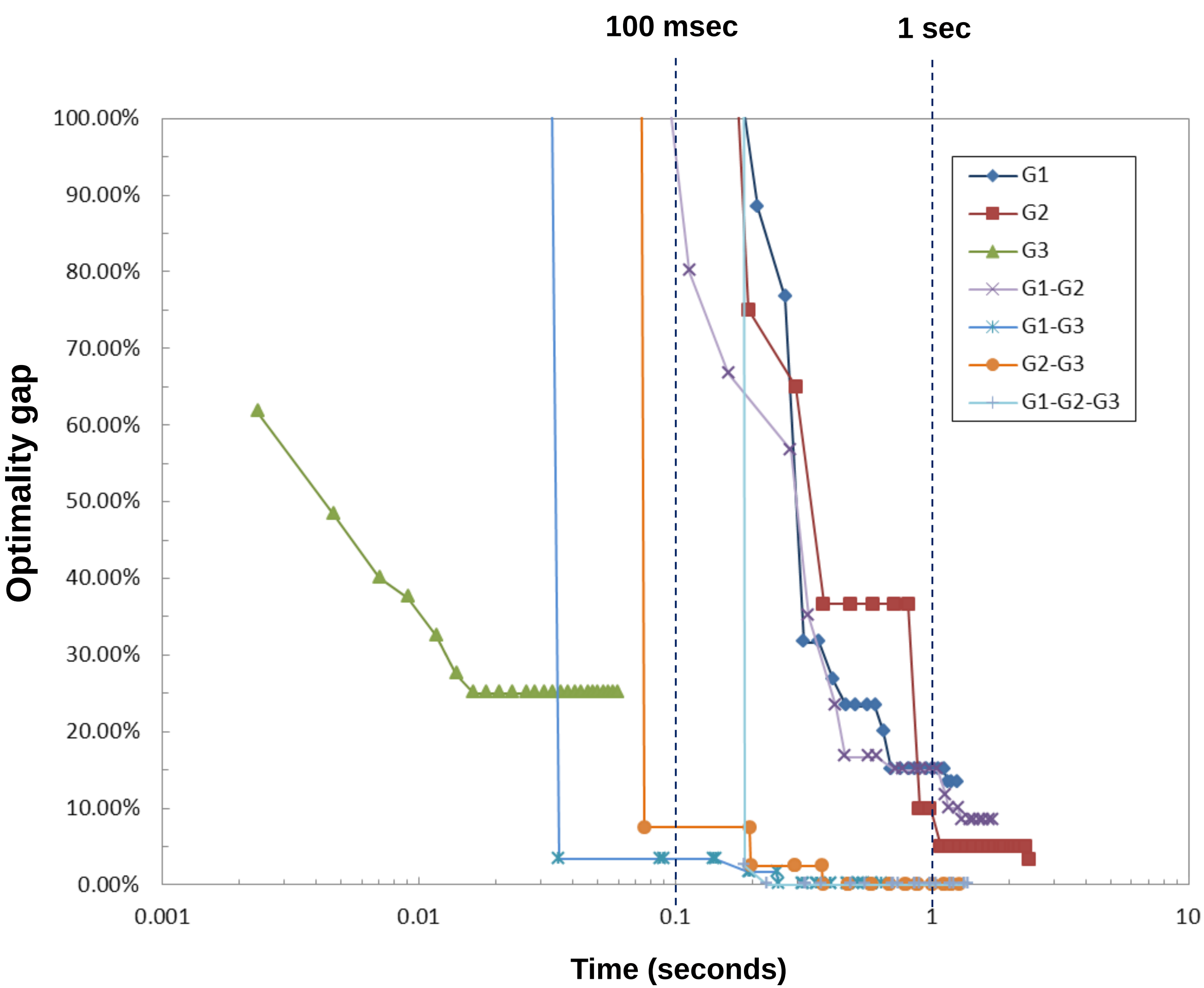


Hybrid meta-heuristics

- Example: multi-greedy + PR
  - **Three different constructive** algorithms to provide diversification.
  - **Path Relinking** finds new solutions in the path connecting two solutions.



Hybrid meta-heuristics: Solving Time



G3 provides feasible solutions in few miliseconds (20-50 msec)

G1 and G2 provide better solutions but at the expense of higher computation time

Multi-start with PR provides the best results:
(G1 + G3)
(G2 + G3)
(G1 + G2 + G3)

UNIVERSITAT POLITÈCNICA DE CATALUNYA

DAC

Algorithmic Methods for Mathematical Models (AMMM)

Intensification and Diversification

Luis Velasco

(lvelasco @ ac.upc.edu)

Campus Nord D6-107