Algorithmic Methods for Mathematical Models (AMMM)

Greedy Algorithms
Examples

Luis Velasco

(lvelasco @ ac.upc.edu)

Campus Nord D6-107

Algorithmic Methods for Mathematical Models (AMMM)

Assignments

Luis Velasco

(lvelasco @ ac.upc.edu)

Campus Nord D6-107

Assign tasks to computers (lab session 2)

T Set of tasks, index t.
C Set of computers, index c.
r_t Resources requested by task t.
r_c Available capacity of computer c.

minimize z

subject to:

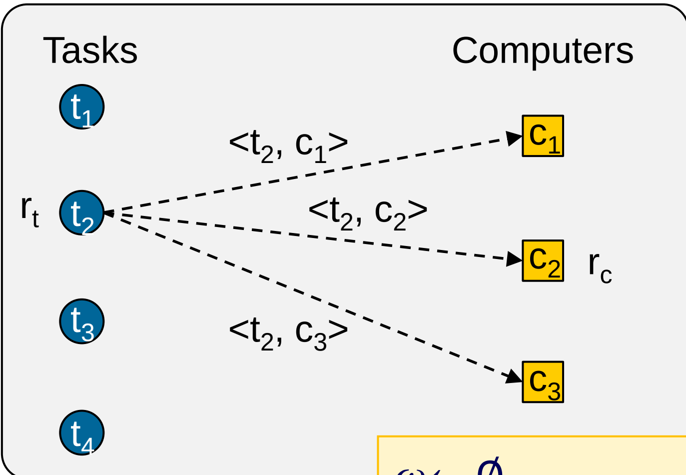
$$\sum_{c \in C} x_{tc} = 1 \quad \forall t \in T$$

$$\sum_{t \in T} r_t \cdot x_{tc} \leq r_c \quad \forall c \in C$$

$$z \geq \frac{1}{r_c} \cdot \sum_{t \in T} r_t \cdot x_{tc} \quad \forall c \in C$$

$$x_{tc} \in \{0,1\} \quad \forall c \in C, t \in T$$

Assign tasks to computers



Load if assignment

$$q(<\mathrm{t},\mathrm{c}>,\omega) = min\left\{ \frac{usedCapaciy(c,\omega)+r_t}{r_c} \quad \frac{usedCapaciy(c',\omega)}{r_{c'}} \mid c' in\ C, c' \neq c \right\}$$

$\omega \leftarrow \emptyset$
$sortedT \leftarrow \mathrm{sort}(T, r_t, DESC)$
for each c in C **do** $usedCapacity_c \leftarrow 0$
for each t in T **do**
    $C(t) \leftarrow \emptyset$
    for each c in C **do**
        if $usedCapacity_c + r_t \leq r_c$ **then** $C(t) \leftarrow C(t) \cup \{c\}$
    if $|C(t)|=0$ **then return** INFEASIBLE
    $c_{best} \leftarrow \mathrm{argmin}\{q(<t,c>, \omega) \mid c \text{ in } C(t)\}$
    $usedCapacity_{cbest} \leftarrow usedCapacity_{cbest} + r_t$
    $\omega \leftarrow \omega \cup \{<t, c_{best}>\}$
return S

Assignment Tasks to computers: Iterative execution

Computers	c1	c2	c3	
rc	505.67	503.68	701.78	
Tasks	t1	t2	t3	t4
rt	261.27	560.89	310.51	105.8

sortedTasks	t2	t3	t1	t4

Computers	c1	c2	c3
residualCap	505.67	503.68	701.78

#1

task: t2	560.89
C(t2)	c3
cbest	c3

Computers	c1	c2	c3
residualCap	505.67	503.68	**140.89**
Load	0	0	**0.799**
Solution	{**<t2,c3>**}		

#2

task: t3	310.51	
C(t2)	c1	c2
Load if assignment		
c1	0.6141	
c2	0.6165	
cbest	c1	

Computers	c1	c2	c3
residualCap	**195.16**	503.68	140.89
load	**0.6141**	0	0.799
Solution	{<t2,c3>,**<t3,c1>**}		

#3

task: t1	261.27
C(t1)	c2
Load if assignment	
c2	0.5187
cbest	c2

Computers	c1	c2	c3
residualCap	195.16	**242.41**	140.89
load	0.6141	**0.5187**	0.799
Solution	{<t2,c3>,<t3,c1>,**<t1,c2>**}		

#4

task: t4	105.8	
C(t4)	c1	c2	c3
Load if assignment			
c1	0.8233		
c2	0.7288		
c3	0.95		
cbest	c2		

Computers	c1	c2	c3
residualCap	195.16	**136.61**	140.89
load	0.6141	**0.7288**	0.799
Solution	{<t2,c3>,<t3,c1>,<t1,c2>,**<t4,c2>**}		

Solution

Solution	**{<t2,c3>,<t3,c1>,<t1,c2>,<t4,c2>}**
f(Solution)	**0.799**

Algorithmic Methods for Mathematical Models (AMMM)

Set Covering

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

Set Covering

- Let M={1, 2, … , m} be the universe of elements to be covered.
- Let $P=\{p_j\}_{j\in N}$, be a family of subsets p_j, N={1, 2, … , n}. Coefficient a_{ij} is 1 if element i is in subset p_j, and 0 otherwise.
- Let c_j be the cost associated with p_j, e.g. its cardinality ($|p_j|$).
- The set covering problem consists on finding the sub-family of elements $\{p_j\}_{j\in N^*}$, $N^* \leq N$, with minimum cost such that $\cup p_j = M$, i.e., covering M.

$$\text{minimize} \quad \sum_{j=1}^{n} c_j x_j$$

subject to:

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1 \quad \forall i \in M$$

$$x_j \in \{0,1\}$$

Set Covering

M/P	p1	p2	p3	p4	p5	p6	p7	p8
1						X		
2			X	X			X	
3	X	X		X	X		X	
4	X			X	X	X		X
5					X	X		
cost	2	1	1	3	3	3	2	1

Optimal solutions (cost 5)
ω ={p6, p7}
ω ={p2, p3, p6}

Other feasible solutions
ω ={p1, p3, p6} (cost=6)
ω ={p4, p6} (cost=6)

Greedy for set covering

Let ω the solution sub-family
Let R the set of covered elements
Greedy function:

$q(p_j,R)=|p_j \cap (M \setminus R)| = |p_j \setminus (R \cap p_j)| \rightarrow$ Number of additional elements of p_j

If every p_j has its own associated cost c_j, the greedy function would be:

$q(p_j,R) = c_j / |p_j \cap (M \setminus R)|$

$\omega=\{\}$
R={}

compute q(pj) $\forall\, p_j \in P \setminus S$	q(p1) = 2	q(p5) = 3
Select the best element: p4	q(p2) = 1	q(p6) = 3
$\omega=\{p4\}$	q(p3) = 1	q(p7) = 2
R={2, 3, 4}	q(p4) = 3	q(p8) = 1

compute q(pj) $\forall\, p_j \in P \setminus S$	q(p1) = 0	q(p5) = 1
Select the best element: p6	q(p2) = 0	q(p6) = 2
$\omega=\{p4,p6\}$	q(p3) = 0	q(p7) = 1
R=M		q(p8) = 0

Cost: 6

UNIVERSITAT POLITÈCNICA
DE CATALUNYA

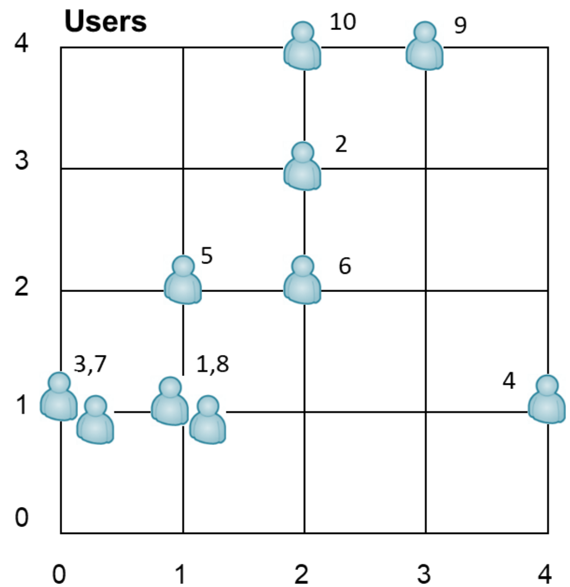
Algorithmic Methods for Mathematical Models (AMMM)

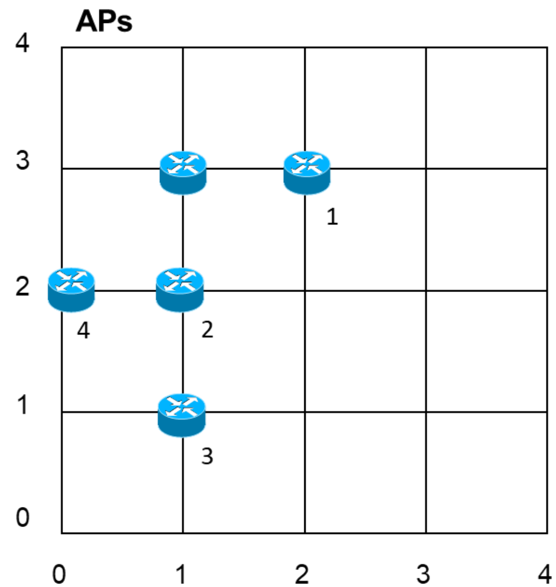
Network planning

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

Network planning

- A set of users *U* needs to be connected to the Internet. For that purpose, we have a set of access point locations *A* where we could install routers (one per access point at the most).
  - For each user *u*, the amount cr_u of capacity units it consumes from the router it is connected to is given.





Network planning

- We have a set *M* of router models.
  - Each model *m* with its fixed cost f_m, capacity k_m, and reach d_m.
  - A router *m* can only connect users that are within a distance d_m from the access point.
- We assume Euclidean distances, so for each user *u* and each access point *a*, we know its Cartesian coordinates (x, y).
- We have to decide:
  - which model of router, if any, should be installed in each access point,
  - which access point each user should be connected to.
  - The goal is to minimize the total cost, computed as the summation of the cost of all the installed routers.

Network planning: Greedy algorithm

$$q(u,\omega) = \min\{q(<u,a>,\omega)\}$$

Infeasible either because of the reach or the load

User u can be served with the router currently installed in location a

$$q(<u,a>,\omega) = \begin{cases} \infty & \boxed{if \quad d(u,a) > \max\{d_m\} \vee cr_u > \left(\max\{k_m\} - \sum_{u\prime \in U(a)} cr_{u\prime}\right)} \\ 0 & \boxed{if \quad d(u,a) \le d_a \wedge cr_u \le \left(k_a - \sum_{u\prime \in U(a)} cr_{u\prime}\right)} \\ f_m - f_a & \boxed{if \quad d(u,a) > d_a \vee cr_u > \left(k_a - \sum_{u\prime \in U(a)} cr_{u\prime}\right), d(u,a) \le d_m \wedge cr_u \le \left(k_m - \sum_{u\prime \in U(a)} cr_{u\prime}\right)} \end{cases}$$

Router currently installed in location a needs to be upgraded because of the reach or the load to serve user u

$\omega \leftarrow \emptyset$, $C \leftarrow U$

Evaluate $q(u, \emptyset)$ $\forall u \in C$

while $C \neq \emptyset$ **do**

$\quad u^{min} \leftarrow \text{argmin } \{q(u, \omega) \mid u \in C\}$

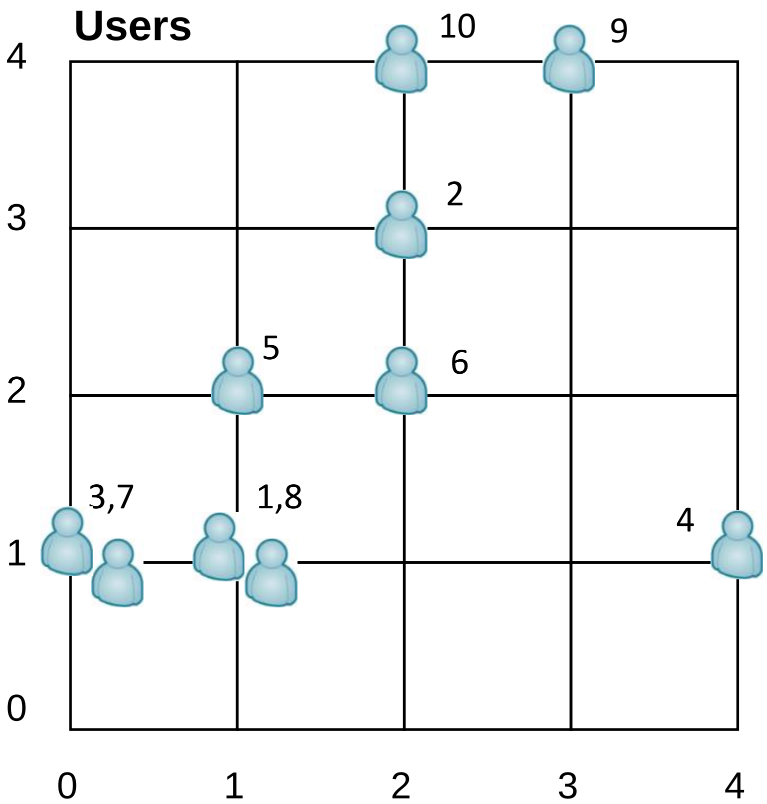
$\quad \omega \leftarrow \omega \cup \{u^{min}\}$

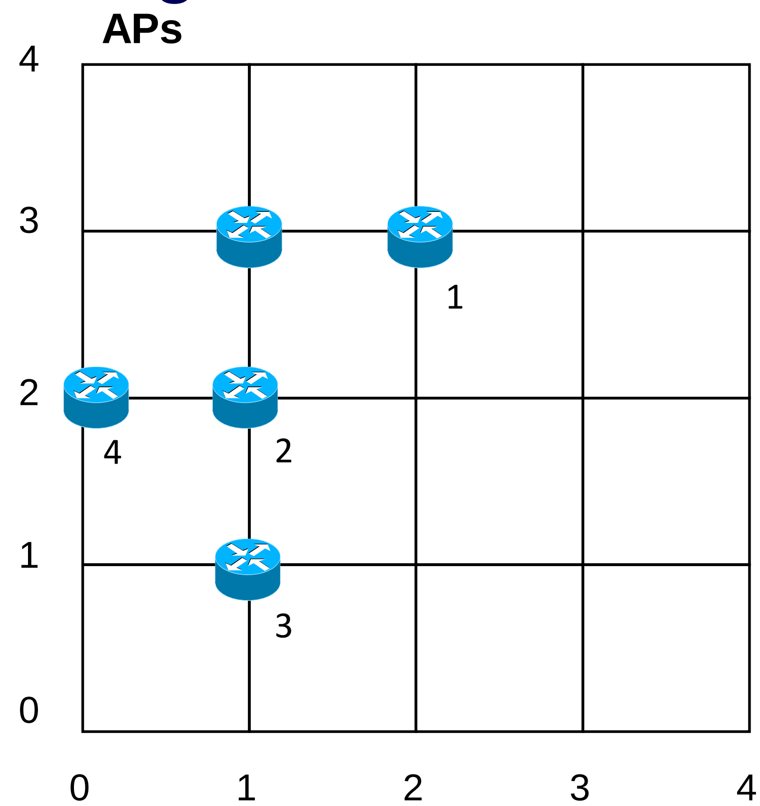
$\quad C \leftarrow C \setminus \{u^{min}\}$

$\quad$ Reevaluate the incremental costs $q(u, \omega)$ $\forall u \in C$

return S

Network planning: Problem Instance





	R1	R2	R3
f	100	140	180
k	6	8	10
d	2	3	4

		u	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
d(u,a)		**x**	1	2	0	4	1	2	0	1	3	2
		y	1	3	1	1	2	2	1	1	4	4
1	2	3	2.2	0.0	2.8	2.8	1.4	1.0	2.8	2.2	1.4	1.0
2	1	2	1.0	1.4	1.4	3.2	0.0	1.0	1.4	1.0	2.8	2.2
3	1	1	0.0	2.2	1.0	3.0	1.0	1.4	1.0	0.0	3.6	3.2
4	0	2	1.4	2.2	1.0	4.1	1.0	2.0	1.0	1.4	3.6	2.8
5	1	3	2.0	1.0	2.2	3.6	1.0	1.4	2.2	2.0	2.2	1.4
a	**x**	**y**										

Network planning: Iterative execution (1/5)

#1

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
q(u)	100	100	100	140	100	100	100	100	100	100
a	3	1	3	1	2	1	3	3	1	1
d(u,a)	0.0	0.0	1.0	2.8	0.0	1.0	1.0	0.0	1.4	1.0

a	1	2	3	4	5
m			R1		
U(a)			{1}		
km-cr			4		

#2

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
q(u)		40	0	40	0	0	0	0	80	80
a	3	3	3	3	3	3	3	3	3	3
d(u,a)	0.0	2.2	1.0	3.0	1.0	1.4	1.0	0.0	3.6	3.2

a	1	2	3	4	5
m			R1		
U(a)			{1,8}		
km-cr			2		

Network planning: Iterative execution (2/5)

#3

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
q(u)		40	40	0	0	0	0		80	80
a	3	3	3	3	3	3	3	3	3	3
d(u,a)	0.0	2.2	1.0	3.0	1.0	1.4	1.0	0.0	3.6	3.2

a	1	2	3	4	5
m			R1		
U(a)			{1,5,8}		
km-cr			0		

#4

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
q(u)		80	80	40		40	40		80	80
a	3	3	3	3	3	3	3	3	3	3
d(u,a)	0.0	2.2	1.0	3.0	1.0	1.4	1.0	0.0	3.6	3.2

a	1	2	3	4	5
m			R2		
U(a)			{1,5,7,8}		
km-cr			1		

Network planning: Iterative execution (3/5)

#5

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
q(u)		100	100	0		40			100	100
a	3	1	4	3	3	3	3	3	1	1
d(u,a)	0.0	0.0	1.0	3.0	1.0	1.4	1.0	0.0	1.4	1.0

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

#6

u	1	2	3	4	5	6	7	8	9	10
cr	2	3	4	1	2	2	1	2	3	4
q(u)		100	100			40			100	100
a	3	1	4	3	3	3	3	3	1	1
d(u,a)	0.0	0.0	1.0	3.0	1.0	1.4	1.0	0.0	1.4	1.0

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

Network planning: Iterative execution (4/5)

#7

u	1	2	3	4	5	6	7	8	9	10
cr	4	3	4	1	2	2	1	2	3	4
q(u)		100	100						100	100
a	3	1	4	3	3	3	3	3	1	1
d(u,a)	0.0	0.0	1.0	3.0	1.0	1.4	1.0	0.0	1.4	1.0

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

#8

u	1	2	3	4	5	6	7	8	9	10
cr	4	3	4	1	2	2	1	2	3	4
q(u)			40						0	40
a	3	1	1	3	3	3	3	3	1	1
d(u,a)	0.0	0.0	2.8	3.0	1.0	1.0	1.0	0.0	1.4	1.0

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

Network planning: Iterative execution (5/5)

#9

u	1	2	3	4	5	6	7	8	9	10
cr	4	3	4	1	2	2	1	2	3	4
q(u)			80							80
a	3	1	1	3	3	3	3	3	1	1
d(u,a)	0.0	0.0	2.8	3.0	1.0	1.0	1.0	0.0	1.4	1.0

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

#10

u	1	2	3	4	5	6	7	8	9	10
cr	4	3	4	1	2	2	1	2	3	4
q(u)			100							
a	3	1	4	3	3	3	3	3	1	1
d(u,a)	0.0	0.0	1.0	3.0	1.0	1.0	1.0	0.0	1.4	1.0

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

Solution Cost=460

Algorithmic Methods for Mathematical Models (AMMM)

Greedy Algorithms

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