

WUCSS 2016
April 5, 2016

Page 1



จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University
Pillar of the Kingdom

Reviving the simplex algorithm

**Sponsored by Walailak University (WU)
Interdisciplinary Center of Scientific Computing (IWR)
National Electronics and Computer Technology Center
(NECTEC)**

**The Development for the Promotion of Science and
Technology Talents Project (DPST)**

**Assistant Professor Krung Sinapiromsaran, Ph. D.
Department of Mathematics and Computer Science
Faculty of Science, Chulalongkorn University**

Applied Mathematics and Computational Sciences



Outline

- **Linear programming history**
- **Linear programming problems and models**
- **The standard simplex algorithm from Dantzig**
- **Improvement of the simplex algorithm**
 - **Cosine angle or minimal angle concept**
 - **Artificial-free simplex algorithm**
 - **Simplex algorithm with an unrestricted variable problem**





จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University
Pillar of the Kingdom

Page 3

<http://pioneer.netserv.chula.ac.th/~skrung>

**Chulalongkorn University | Faculty of Science | Department of Mathematics and
Computer Science**



Krung Sinapiromsaran

Assistant Professor
Department of Mathematics and Computer Science,
Faculty of Science
Chulalongkorn University
Phryathai Road, Wangmai, Patumwon,
Bangkok 10330, Thailand
Office: Mahawachirunahit 1208/16
Telephone: +662-218-5150
Fax: +662-255-2287
Email: Krung.S@chula.ac.th
<http://pioneer.netserv.chula.ac.th/~skrung>
<http://www.infolyst.net/>
ICCCO
KDDMINI
LPICCC

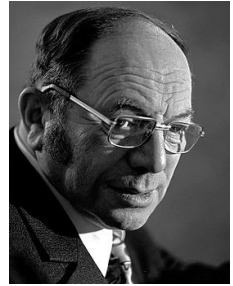


Linear programming history



Joseph Fourier

• 1827: Fourier-Motzkin elimination algorithm was proposed to reduce the number of variables in a system of linear inequalities. Complexity $O(m^{2^k})$ where k is the number of successive step and m is the number of inequalities.



Leonid Kantorovich

• 1939: Leonid Kantorovich formulated and solved a planning for expenditures and returns during World War II

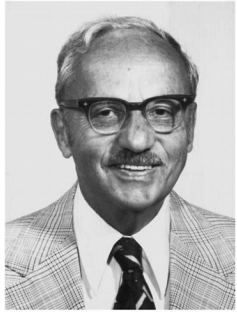


Tjalling Koopmans

• 1975: Nobel prize in economics was award to Leonid Kantorovich together with Tjalling Charles Koopmans



Linear programming history



George B. Dantzig

• 1947: George B. Dantzig published the simplex method (the simplex algorithm) to efficiently find the solution of a linear programming problem for US Air Force.



Leonid Khachiyan

• 1979: Leonid Khachiyan published the ellipsoid algorithm for solving a linear programming model in polynomial time. However, his algorithm is not practical for the real world problems.



Narendra Karmarkar

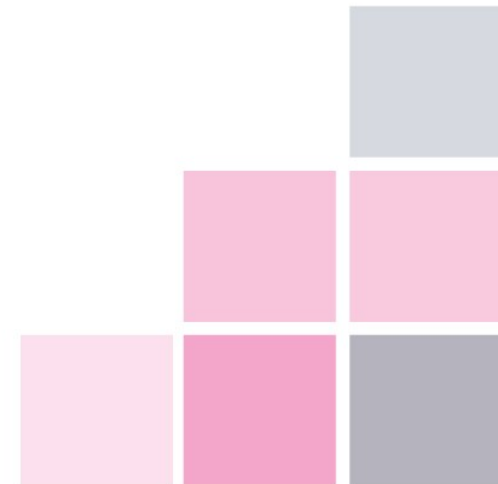
• 1984: A new interior-point method was introduced by Narendra Karmarkar while he was working for Bell Laboratories.



จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University
Pillar of the Kingdom

Page 6

Linear programming problems and models





- *Transshipment* (Transshipment) is the shipment of goods to an intermediate station then to the destination.
- If there is no intermediate station, then this is called the *transportation*.

Minimize $\sum_{i=1}^m \sum_{j=1}^n c_{i,j} x_{i,j}$

$c_{i,j}$ cost of shipping

a_i goods available at node i

m origins

subject to $\sum_{s=1}^{m+n} x_{i,s} - \sum_{r=1}^{m+n} x_{r,i} = a_i \quad \forall i=1, \dots, m$

n destination

$$\sum_{r=1}^{m+n} x_{r,m+j} - \sum_{s=1}^{m+n} x_{m+j,s} = b_{m+j} \quad \forall j=1, \dots, n$$

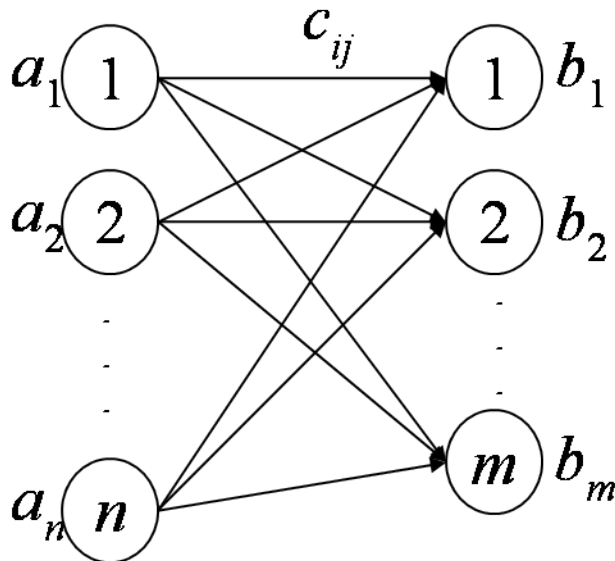
b_{m+j} demand for the goods at node $m+j$

Total available =
Total demand = $\sum_{i=1}^m a_i = \sum_{j=1}^n b_{m+j}$



- *Transportation* of n origins and m destinations.

a_i = goods available at origin i , b_j = goods demand at destination j .



b_{m+j} demand for the goods at node $m+j$

Minimize

$$\sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

c_{ij} cost of shipping

m origins

subject to

$$\sum_{j=1}^m x_{ij} \leq a_i$$

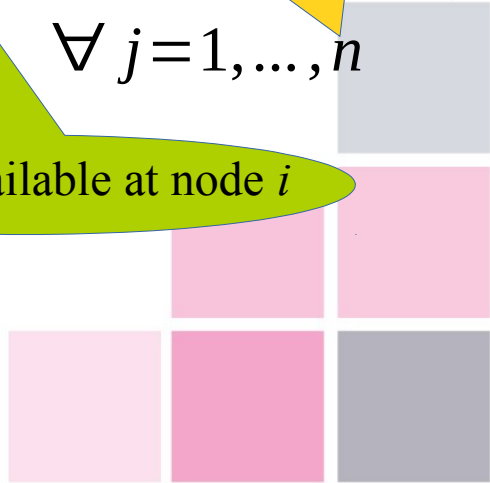
$$\forall i=1, \dots, m$$

n destination

$$\sum_{i=1}^n x_{ij} \geq b_j$$

$$\forall j=1, \dots, n$$

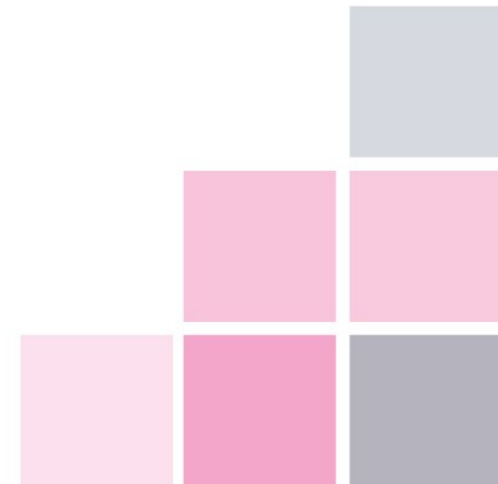
a_i goods available at node i



Total available =
Total demand



The standard simplex algorithm from Dantzig



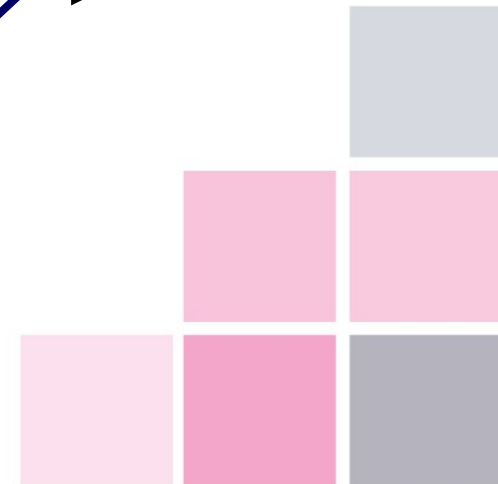
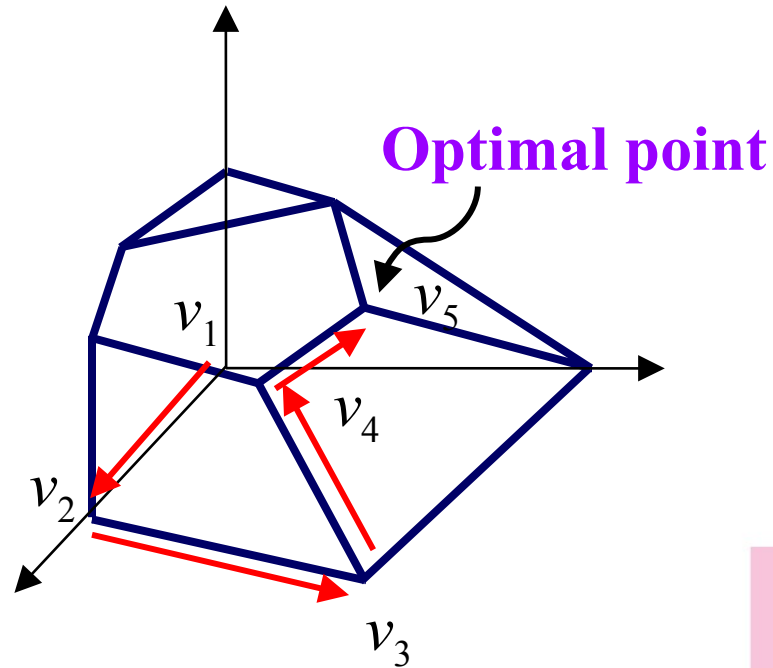


Canonical form of the LP model

Maximize $c^T x$ Easily translate to standard form if $b \geq 0$
 subject to $Ax \leq b$
 $x \geq 0$



Maximize $c^T x + 0^T s$
 subject to $Ax + Is = b$
 $x \geq 0, s \geq 0$





Maximize $\mathbf{c}^T \mathbf{x}$
 subject to $\mathbf{A} \mathbf{x} = \mathbf{b}$
 $\mathbf{x} \geq \mathbf{0}$

← Multiply by -1 if $b_i < 0$

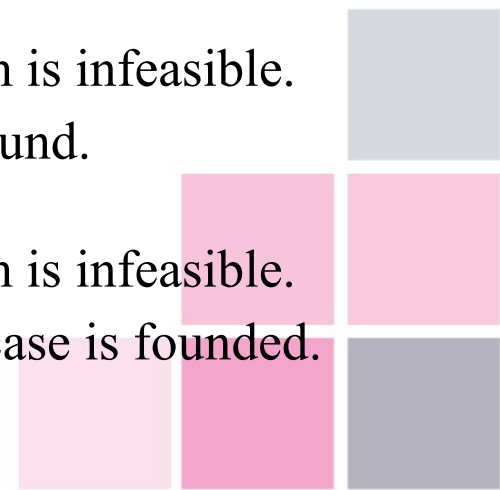


M is a very large positive number.

Maximize $\mathbf{c}^T \mathbf{x} - M \mathbf{1}^T \mathbf{x}_a$
 subject to $\mathbf{A} \mathbf{x} + \mathbf{I} \mathbf{x}_a = \mathbf{b}$
 $\mathbf{x} \geq \mathbf{0}, \mathbf{x}_a \geq \mathbf{0}$

Stop at the optimal,
 If $\mathbf{1}^T \mathbf{x}_a > 0$, then the problem is infeasible.
 Otherwise, the optimal is found.

Stop with unbounded case,
 If $\mathbf{1}^T \mathbf{x}_a > 0$, then the problem is infeasible.
 Otherwise, the unbounded case is founded.





Maximize $\mathbf{c}^T \mathbf{x}$
 subject to $\mathbf{A} \mathbf{x} = \mathbf{b}$
 $\mathbf{x} \geq \mathbf{0}$

← Multiply by -1 if $b_i < 0$

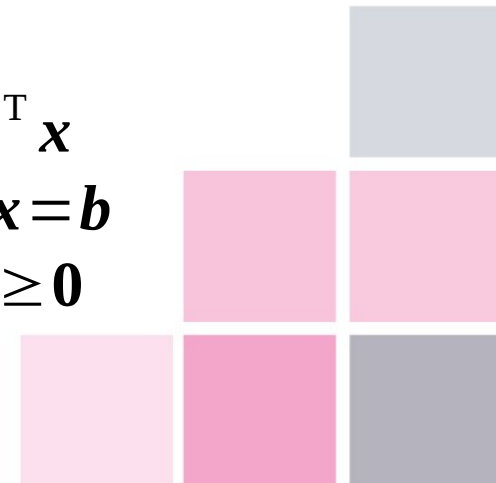


If the optimal is zero,
 Then eliminate \mathbf{x}_a , continue with this new BFS.

Minimize $\mathbf{1}^T \mathbf{x}_a$
 subject to $\mathbf{A} \mathbf{x} + \mathbf{I} \mathbf{x}_a = \mathbf{b}$
 $\mathbf{x} \geq \mathbf{0}, \mathbf{x}_a \geq \mathbf{0}$

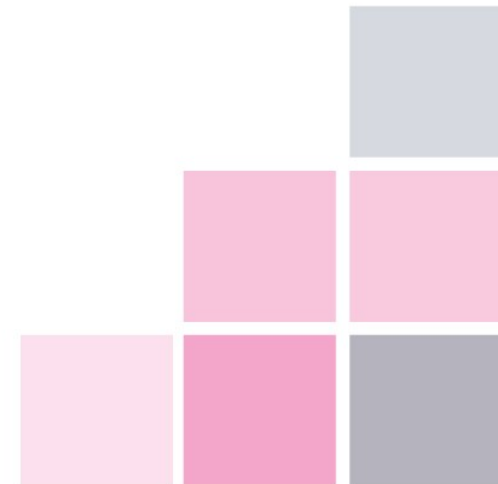
→

Maximize $\mathbf{c}^T \mathbf{x}$
 subject to $\mathbf{A} \mathbf{x} = \mathbf{b}$
 $\mathbf{x} \geq \mathbf{0}$



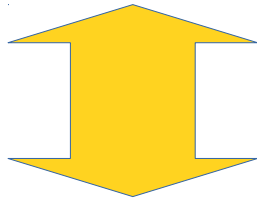


Cosine angle and minimal angle concept (Small dimensional problems)



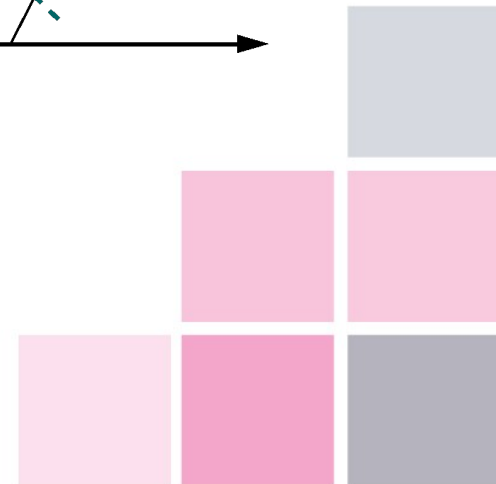
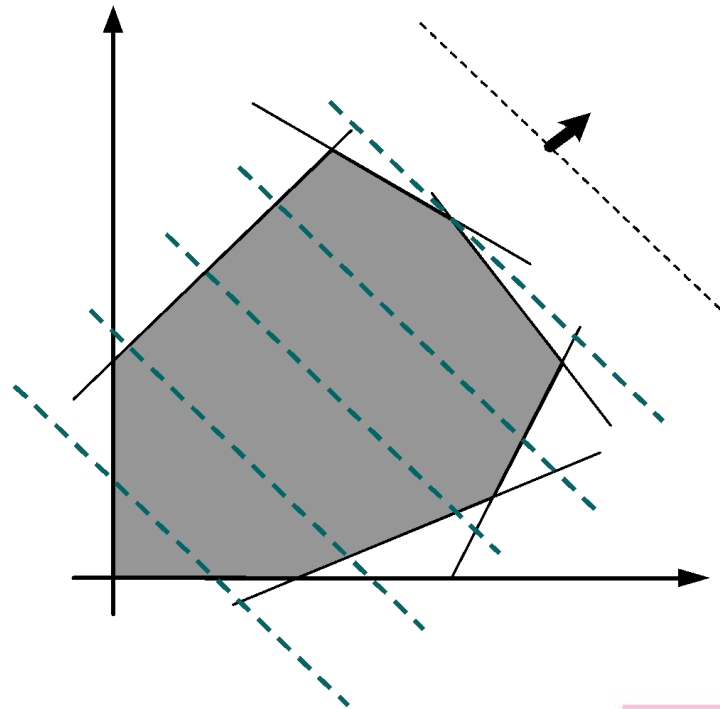


Minimize $\mathbf{c}^T \mathbf{x}$
subject to $\mathbf{A} \mathbf{x} \leq \mathbf{b}$



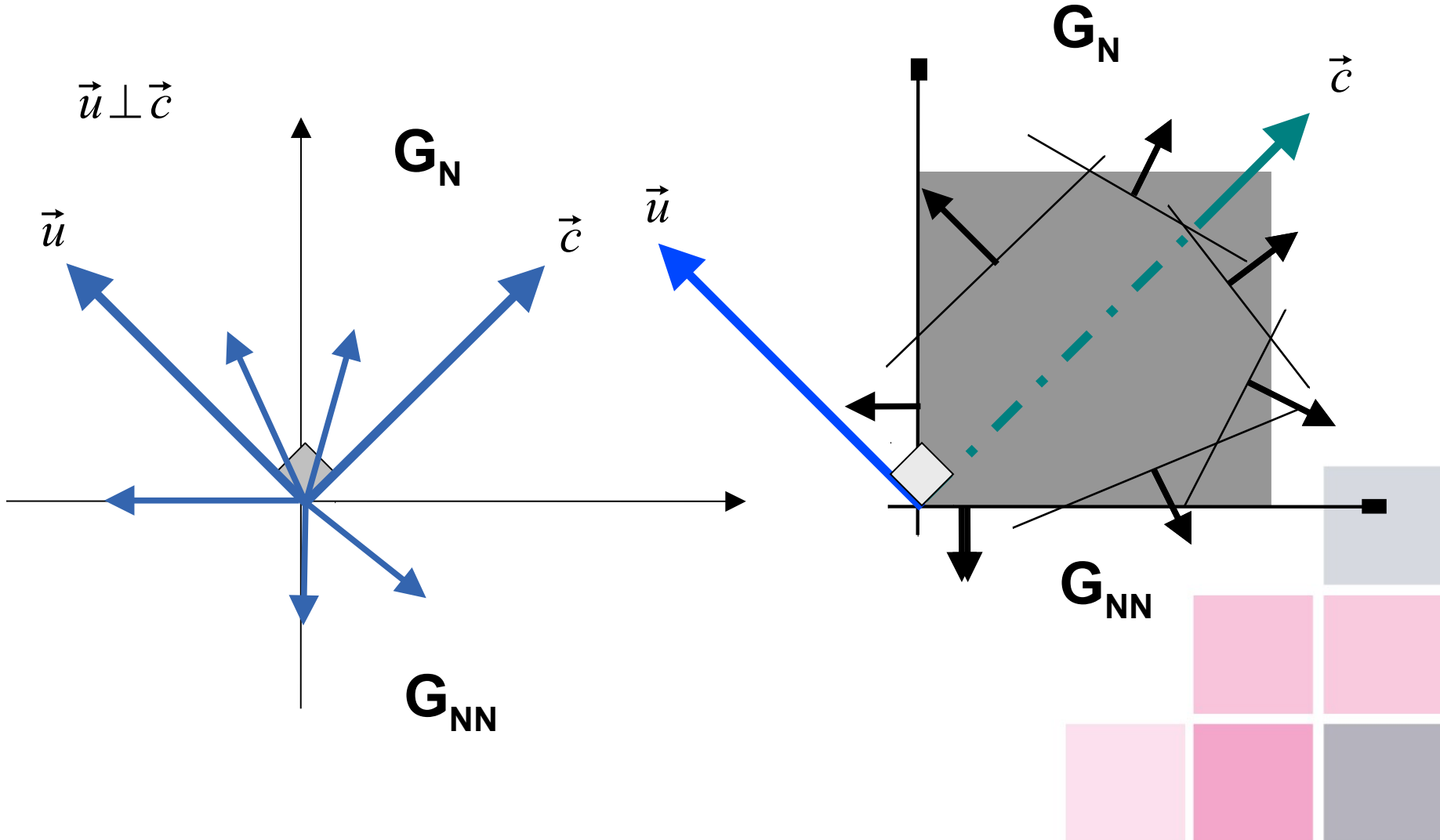
Minimize $c_1 x_1 + c_2 x_2$
subject to $a_{11} x_1 + a_{12} x_2 \leq b_1$
 $a_{21} x_1 + a_{22} x_2 \leq b_2$
...
 $a_{m1} x_1 + a_{m2} x_2 \leq b_m$

Gradient vector of objective function





Minimal angle in 2D (K. Narong, K. Sinapiromsaran 2004)





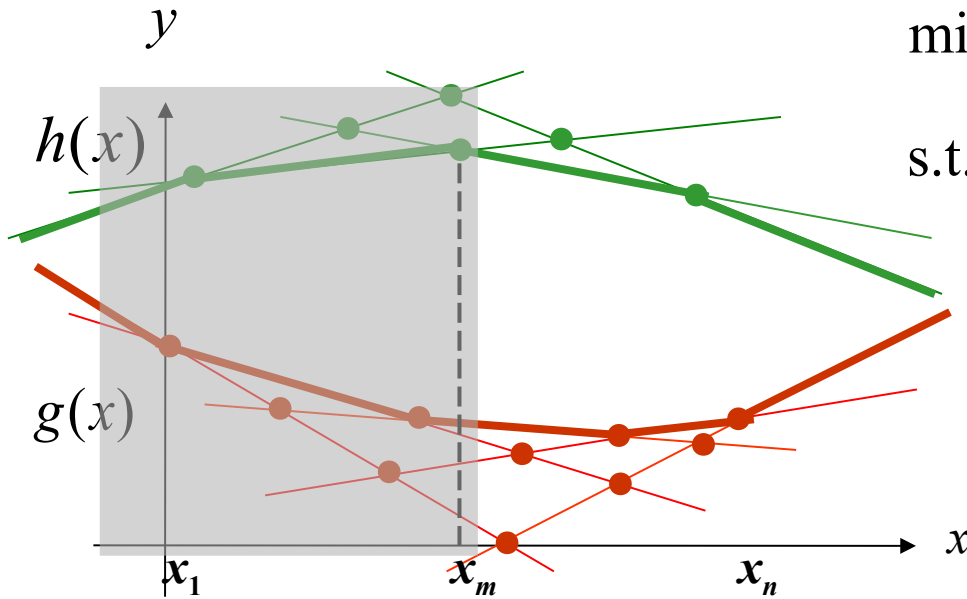
minimize
 x_1, x_2

$$c_1x_1 + c_2x_2$$

s.t.

$$a_{i1}x_1 + a_{i2}x_2 \geq \beta_i \quad (i = 1, \dots, n)$$

$$c_1 \neq 0, x_1 = \frac{y - c_2x_2}{c_1}$$



minimize
 x, y

y

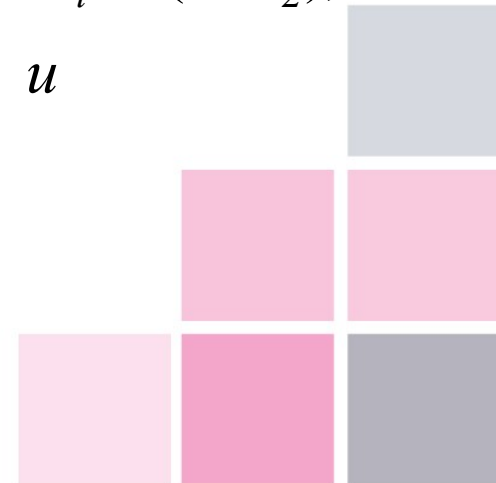
s.t.

$$y \geq a_i x + b_i \quad (i \in I_1),$$

$$y \leq a_i x + b_i \quad (i \in I_2),$$

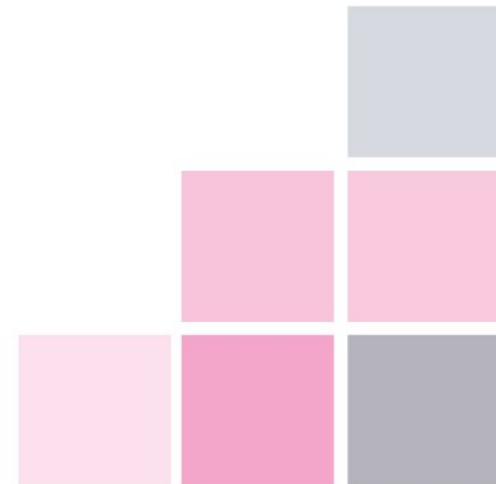
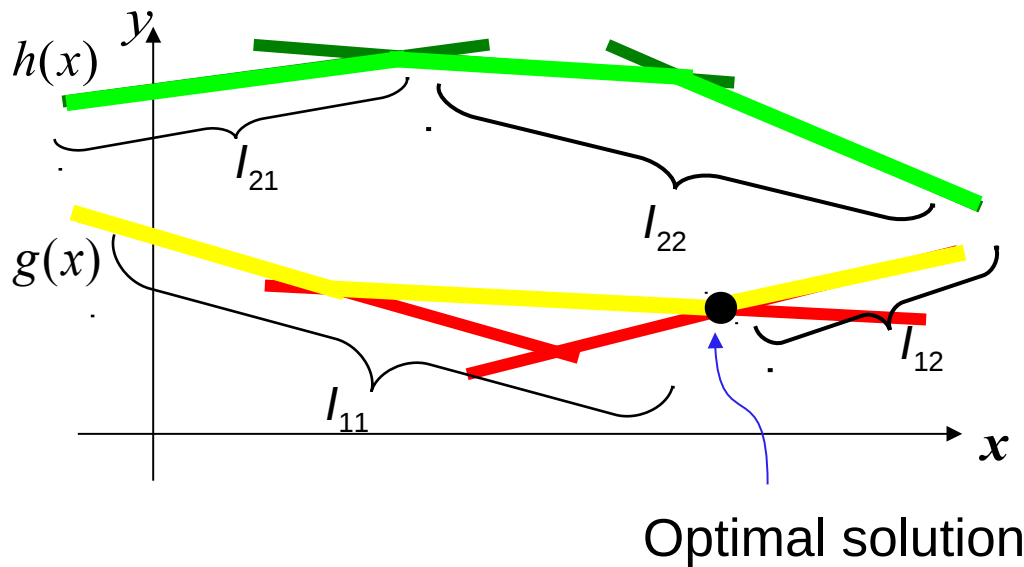
$$l \leq x \leq u$$

Time $\rightarrow O(n^2)$





Only for non-redundant constrained linear programming problems





The closest angle

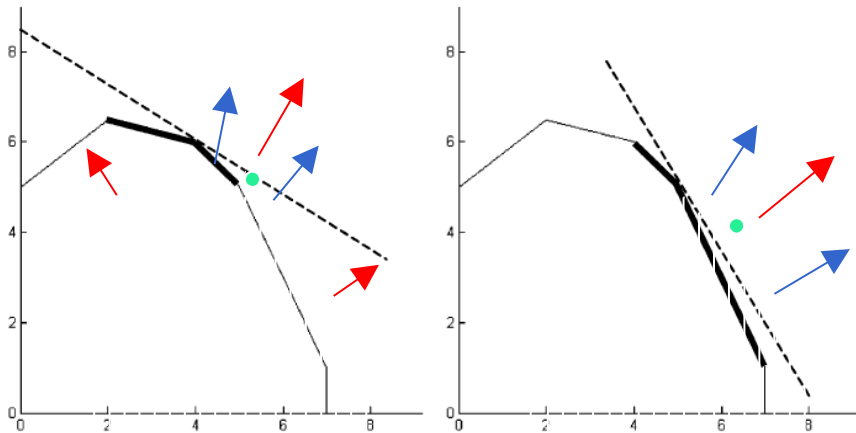
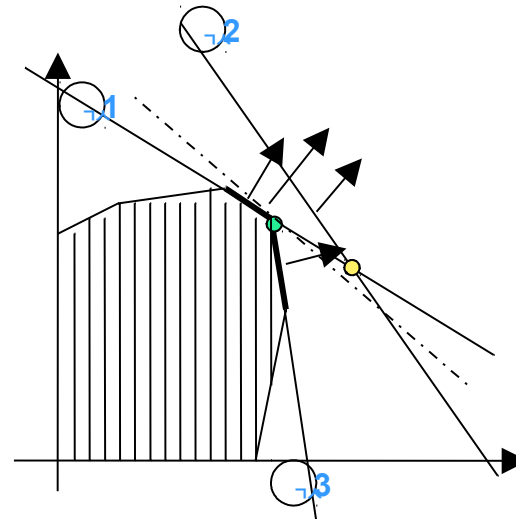


Fig. 1.

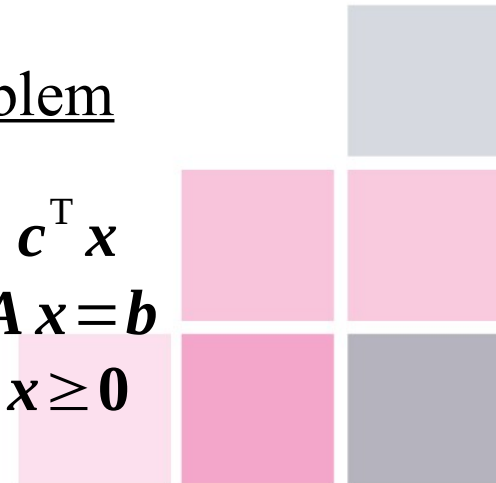


Primal problem

$$\begin{aligned} &\text{Maximize} && \mathbf{b}^T \boldsymbol{\omega} \\ &\text{subject to} && \mathbf{A}^T \boldsymbol{\omega} \leq \mathbf{c} \end{aligned}$$

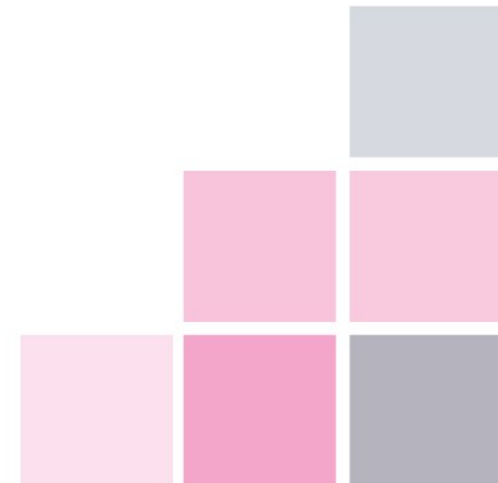
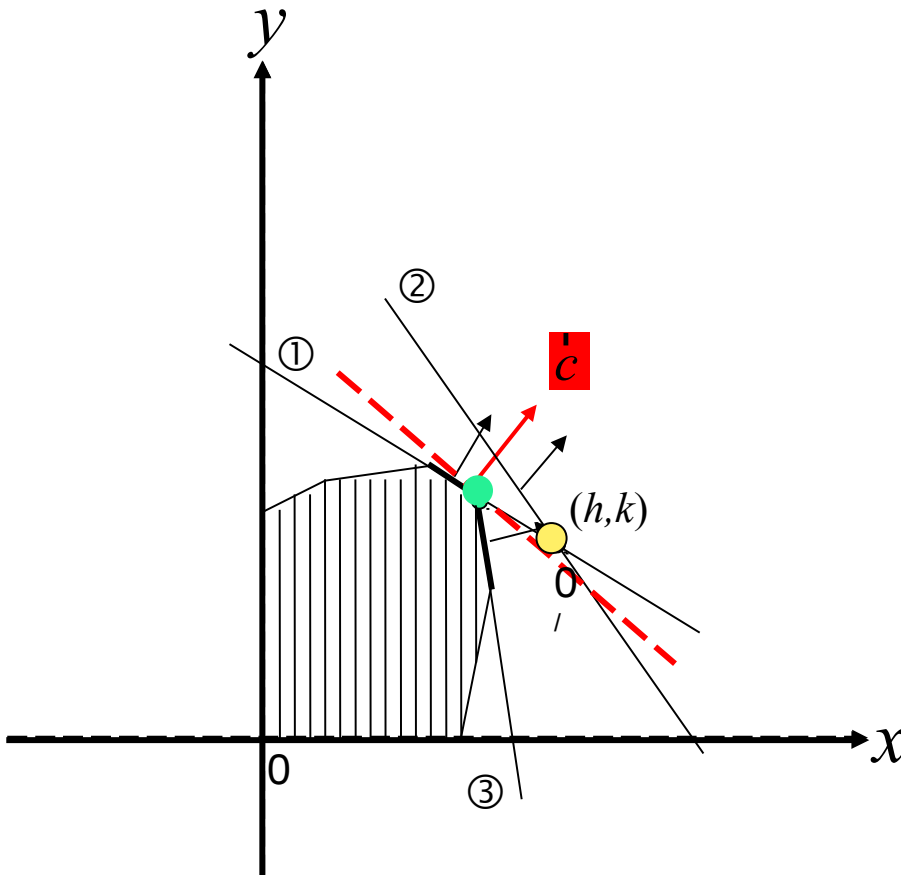
Dual problem

$$\begin{aligned} &\text{Minimize} && \mathbf{c}^T \mathbf{x} \\ &\text{subject to} && \mathbf{A} \mathbf{x} = \mathbf{b} \\ &&& \mathbf{x} \geq \mathbf{0} \end{aligned}$$



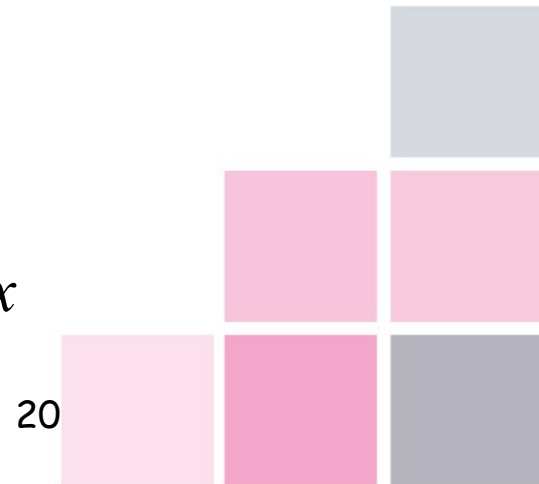
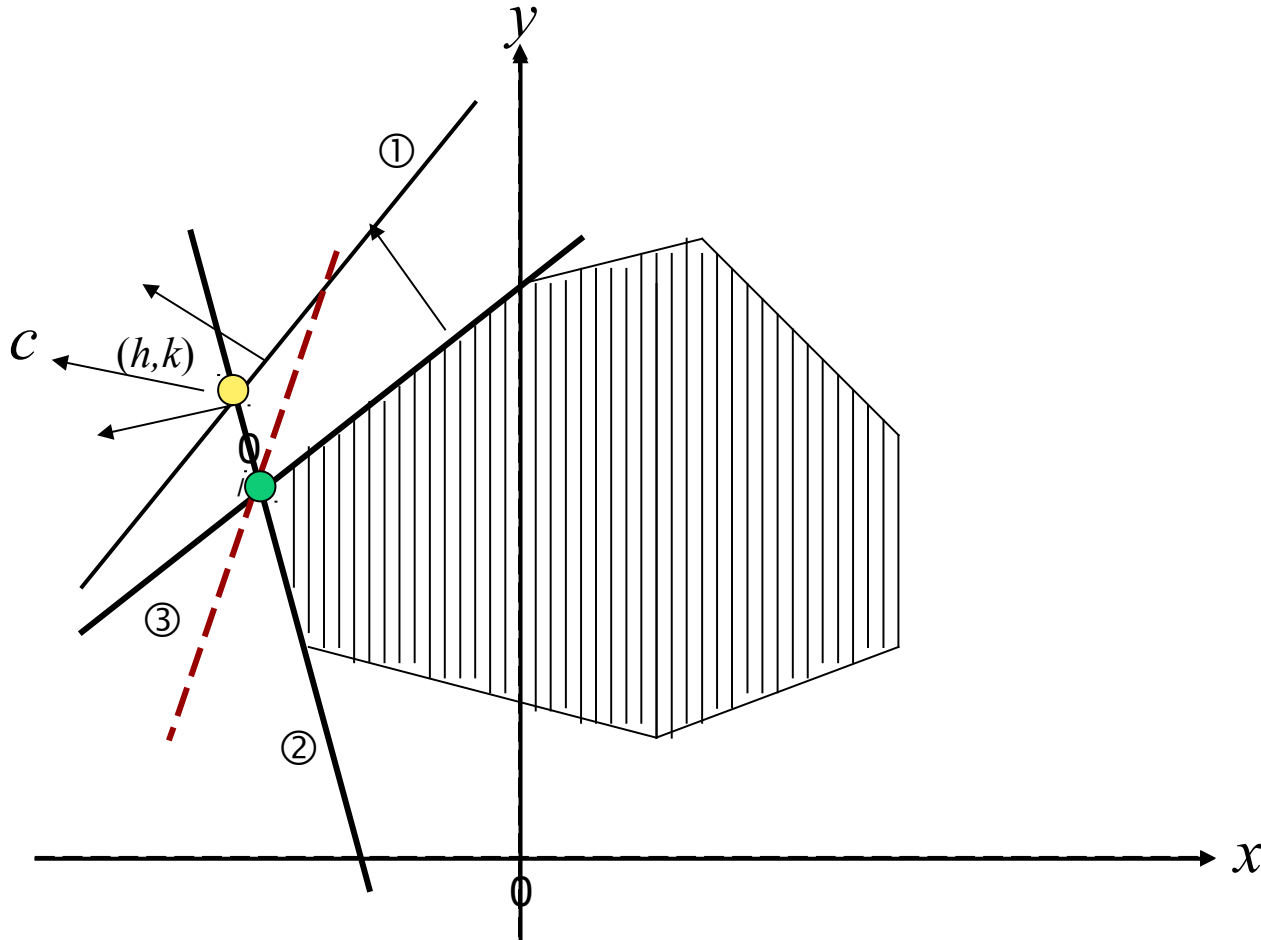


The new origin point for starting Simplex Algorithm (A. Boonperm & K. Sinapiromsaran, 2012)





The new origin point for starting Simplex Algorithm
(A. Boonperm & K. Sinapiromsaran, 2012)





Minimize
subject to

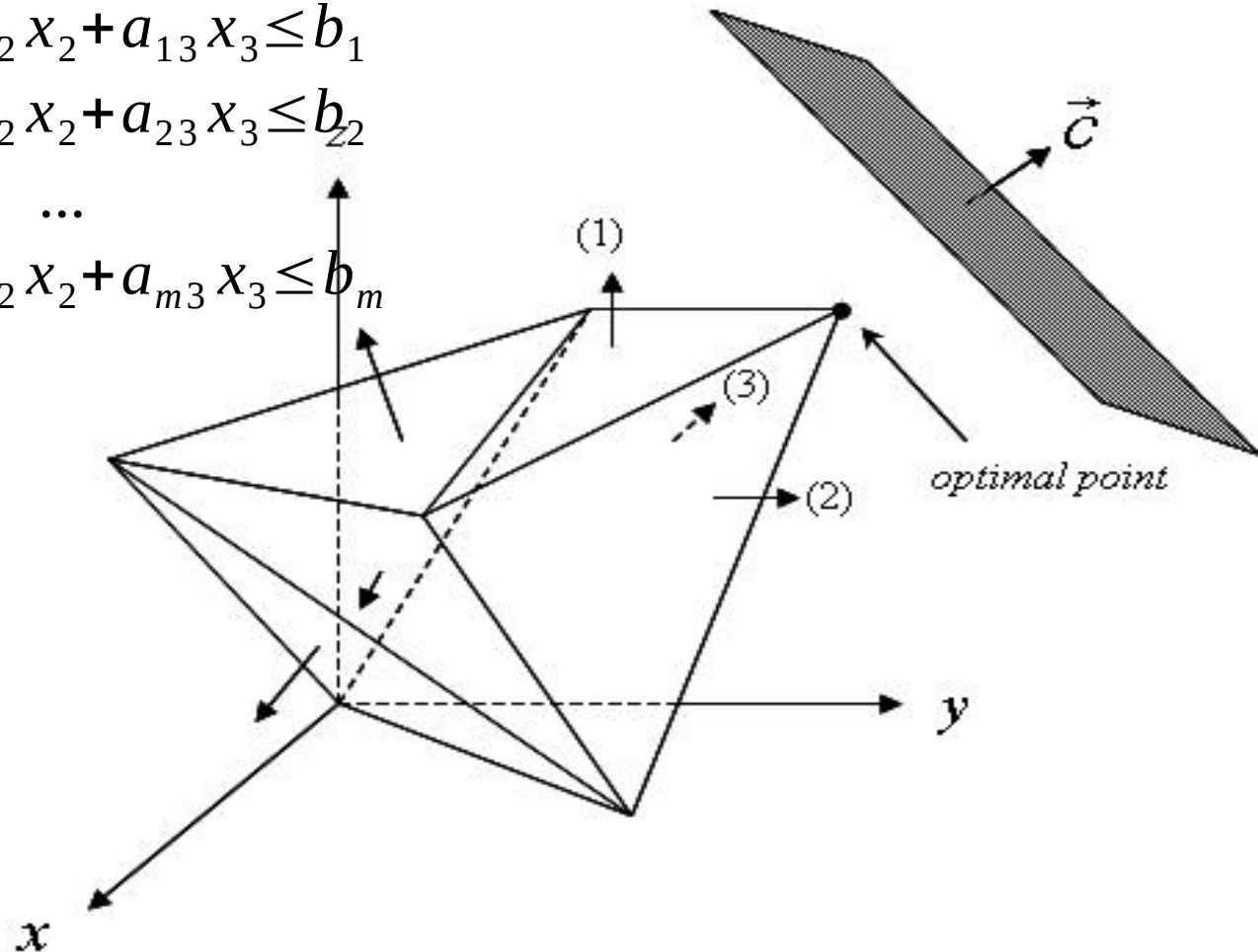
$$c_1 x_1 + c_2 x_2 + c_3 x_3$$

$$a_{11} x_1 + a_{12} x_2 + a_{13} x_3 \leq b_1$$

$$a_{21} x_1 + a_{22} x_2 + a_{23} x_3 \leq b_2$$

...

$$a_{m1} x_1 + a_{m2} x_2 + a_{m3} x_3 \leq b_m$$





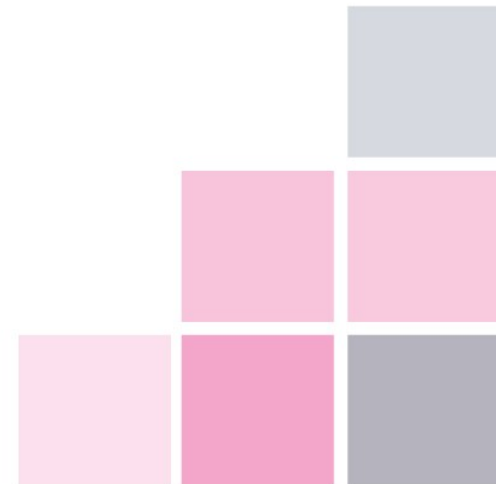
Projection to 2D LP Iterative vs. KKT choice

- Select the binding constraint

$$a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 = b_i$$

- Define $x_3 = (b_i - a_{i1}x_1 - a_{i2}x_2)/a_{i3}$
- Replace it to the remaining constraints and the objective function and solve this 2-dimensional LP.
- Solve it using the minimal angled algorithm.

$$\begin{aligned} \mathbf{A} \mathbf{x}^* &\leq \mathbf{b} \\ \mathbf{w}^{*T} \mathbf{A} &= \mathbf{c}^T, \mathbf{w}^* \geq \mathbf{0} \\ \mathbf{w}^{*T} (\mathbf{A} \mathbf{x}^* - \mathbf{b}) &= 0, (\mathbf{c}^T - \mathbf{w}^{*T} \mathbf{A}) \mathbf{x}^* = 0 \end{aligned}$$





time (ms)

0.40
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00

4 5 6 7 8 9 10 11 12 13 14 15

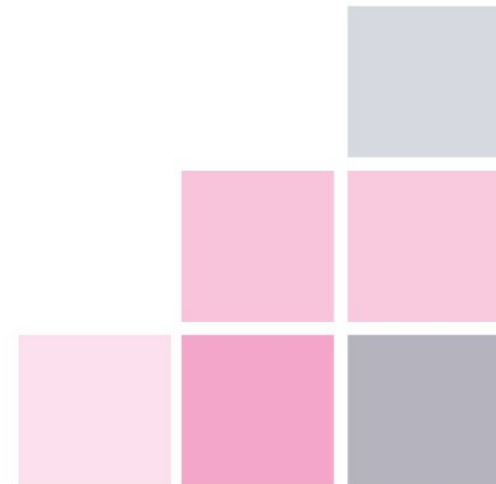
number of constraints

- ◆ Simplex Method
- Naïve Method
- ▲ KKT-MAP





Artificial-free simplex algorithm (How to avoid artificial variables)





The Partitioned Problem:

$$\begin{aligned} &\text{Maximize} && \mathbf{c}_1^T \mathbf{x}_1 + \mathbf{c}_2^T \mathbf{x}_2 \\ &\text{subject to} && \mathbf{A}_{11} \mathbf{x}_1 + \mathbf{A}_{12} \mathbf{x}_2 \leq -\mathbf{b}_1 \\ &&& \mathbf{A}_{21} \mathbf{x}_1 + \mathbf{A}_{22} \mathbf{x}_2 \leq \mathbf{b}_2 \\ &&& \mathbf{x}_1 \geq \mathbf{0}, \mathbf{x}_2 \geq \mathbf{0} \end{aligned}$$

The algorithm performs alternating during primal and dual iterations.

Ineffective in practice.





The primal perturbation simplex method (Pan, 2000)

- At the origin point, if the problem was neither primal nor dual feasible,
 - Perturbed cost of objective function in primal to guarantee the dual feasibility. Then starts the dual simplex method.
- The the original cost of objective function was restored and the primal simplex will be performed.

The computational results were shown to be superior for low dimensional linear programming problems.



Start with:

$$\begin{aligned} &\text{Maximize} && \mathbf{c}^T \mathbf{x} \\ &\text{subject to} && \mathbf{A} \mathbf{x} \leq \mathbf{a} \\ & && \mathbf{B} \mathbf{x} \leq \mathbf{b} \\ & && \mathbf{x} \geq \mathbf{0} \end{aligned}$$

where $\mathbf{b} < \mathbf{0}$, $\mathbf{a} \geq \mathbf{0}$.

Relaxed problem:

$$\begin{aligned} &\text{Maximize} && \mathbf{c}^T \mathbf{x} \\ &\text{subject to} && \mathbf{A} \mathbf{x} \leq \mathbf{a} \\ & && \mathbf{x} \geq \mathbf{0} \end{aligned}$$



$$\begin{aligned} &\text{Maximize} && \mathbf{c}^T \mathbf{x} \\ &\text{subject to} && \mathbf{A} \mathbf{x} + \mathbf{I} \mathbf{s} = \mathbf{a} \\ & && \mathbf{x} \geq \mathbf{0}, \mathbf{s} \geq \mathbf{0} \end{aligned}$$

Only small examples were shown.

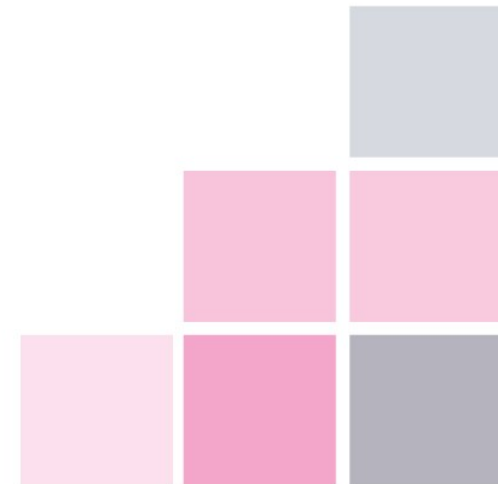




จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University
Pillar of the Kingdom

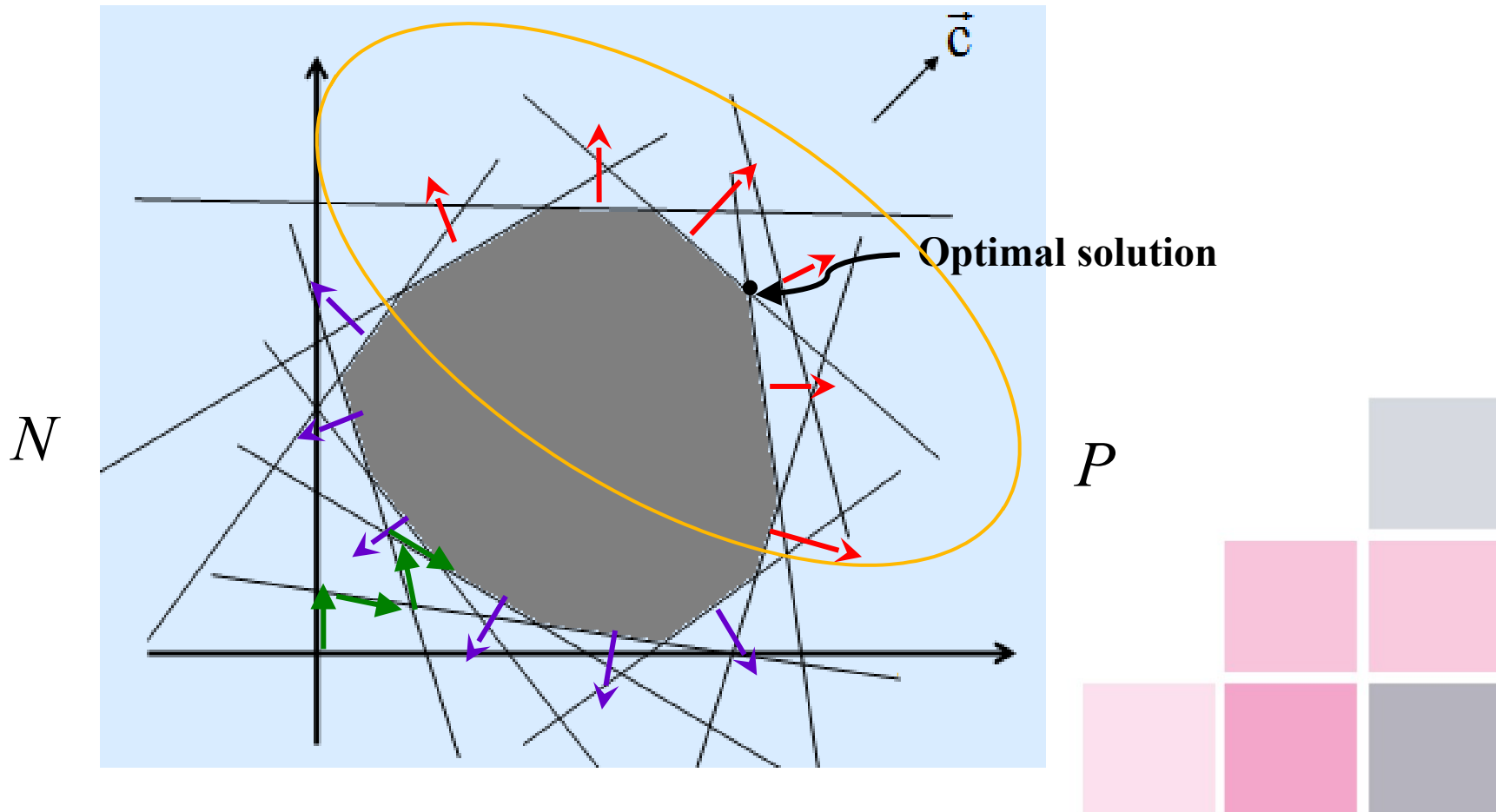
Page 28

Simplex implementation Without artificial variables (Our methods)



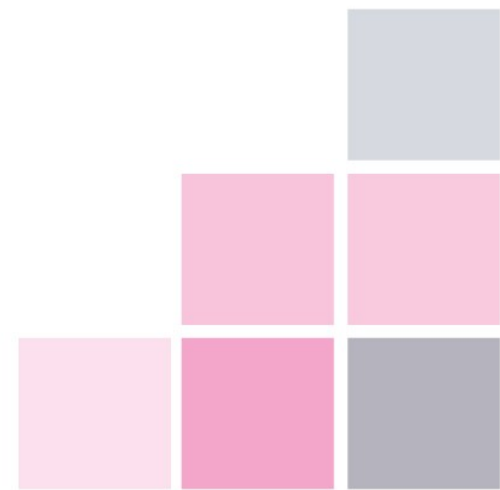
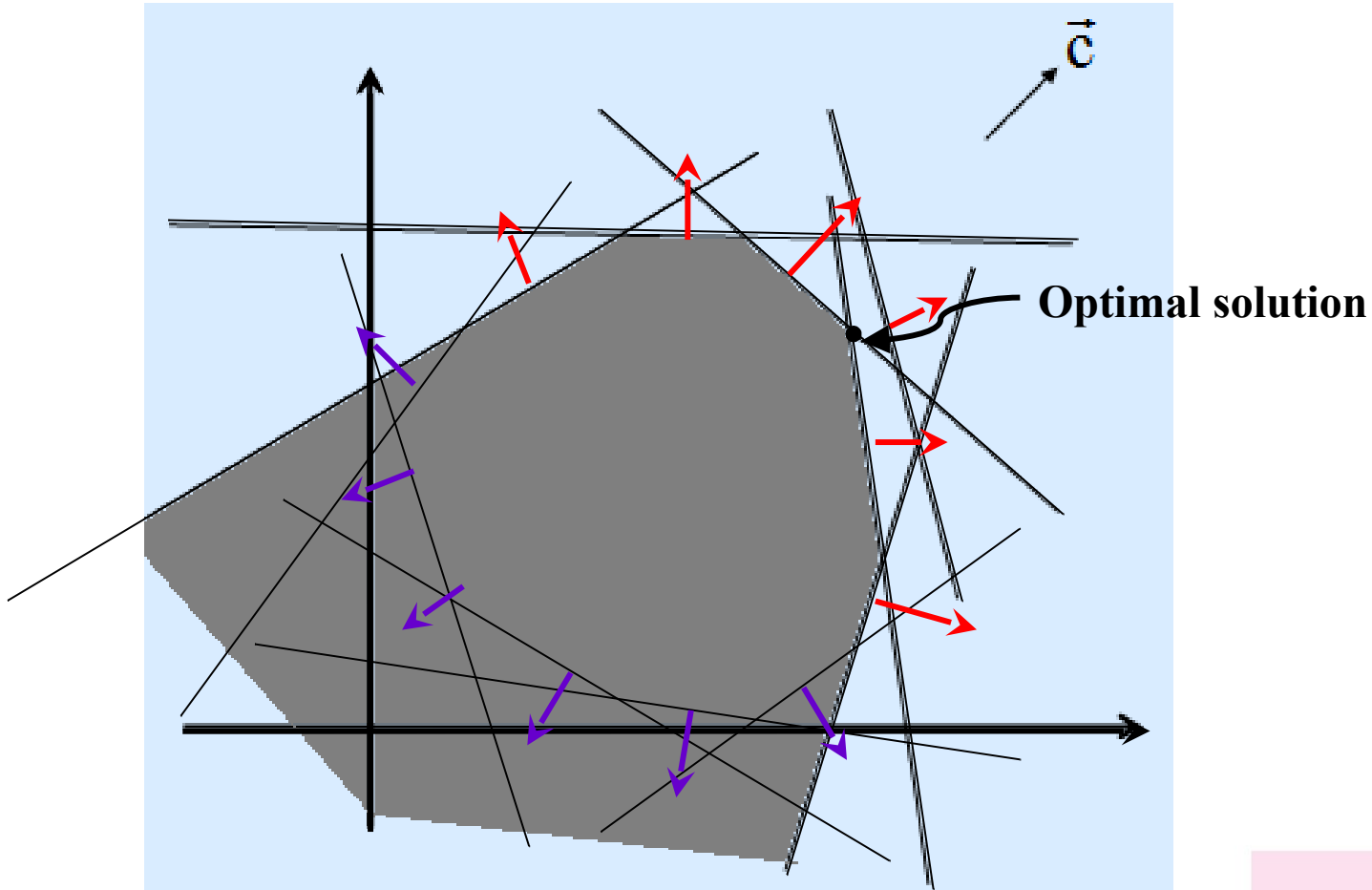


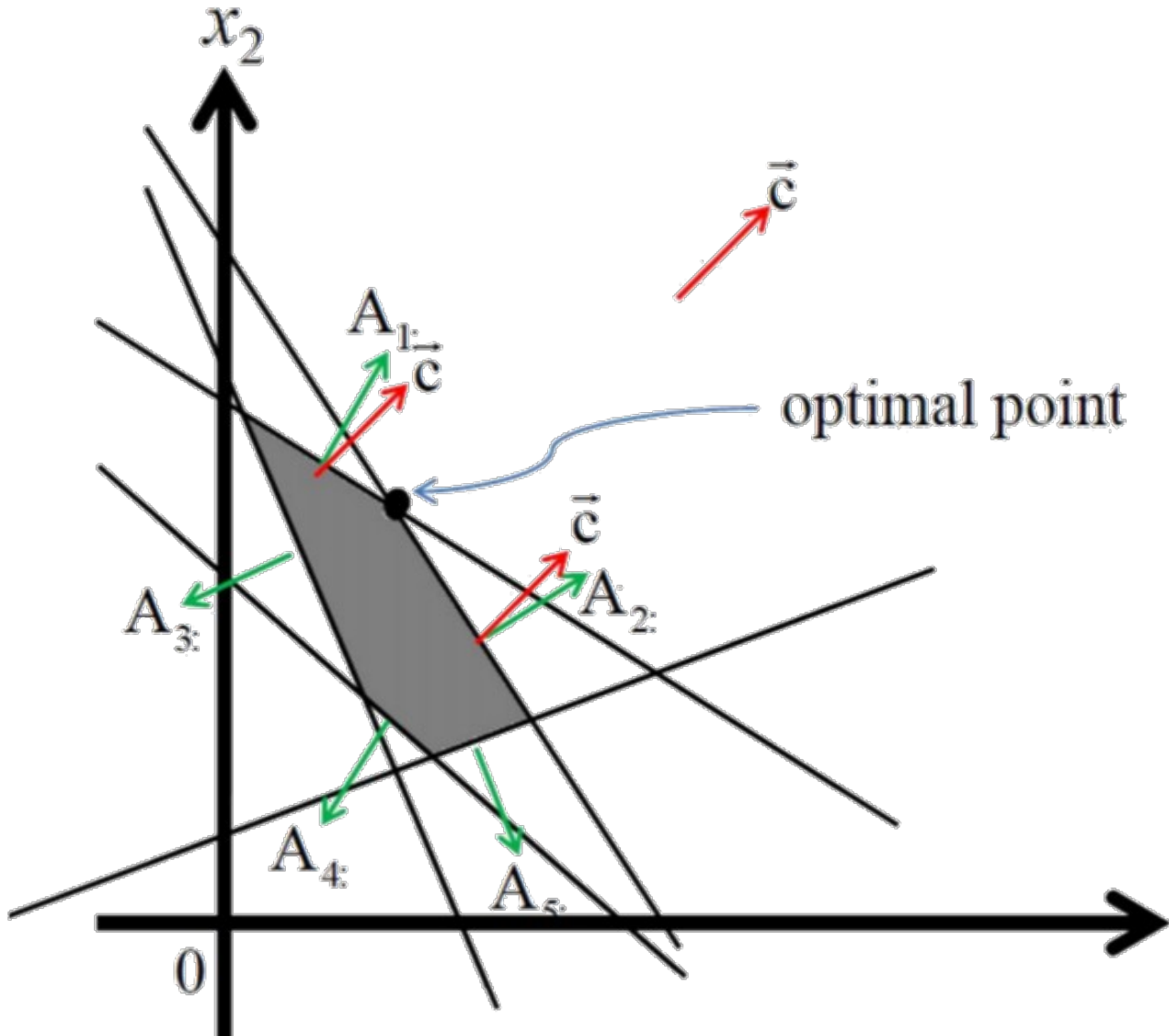
$$\begin{aligned} & \text{Maximize} && \mathbf{c}^T \mathbf{x} \\ & \text{subject to} && \mathbf{A} \mathbf{x} \leq \mathbf{b} \end{aligned}$$

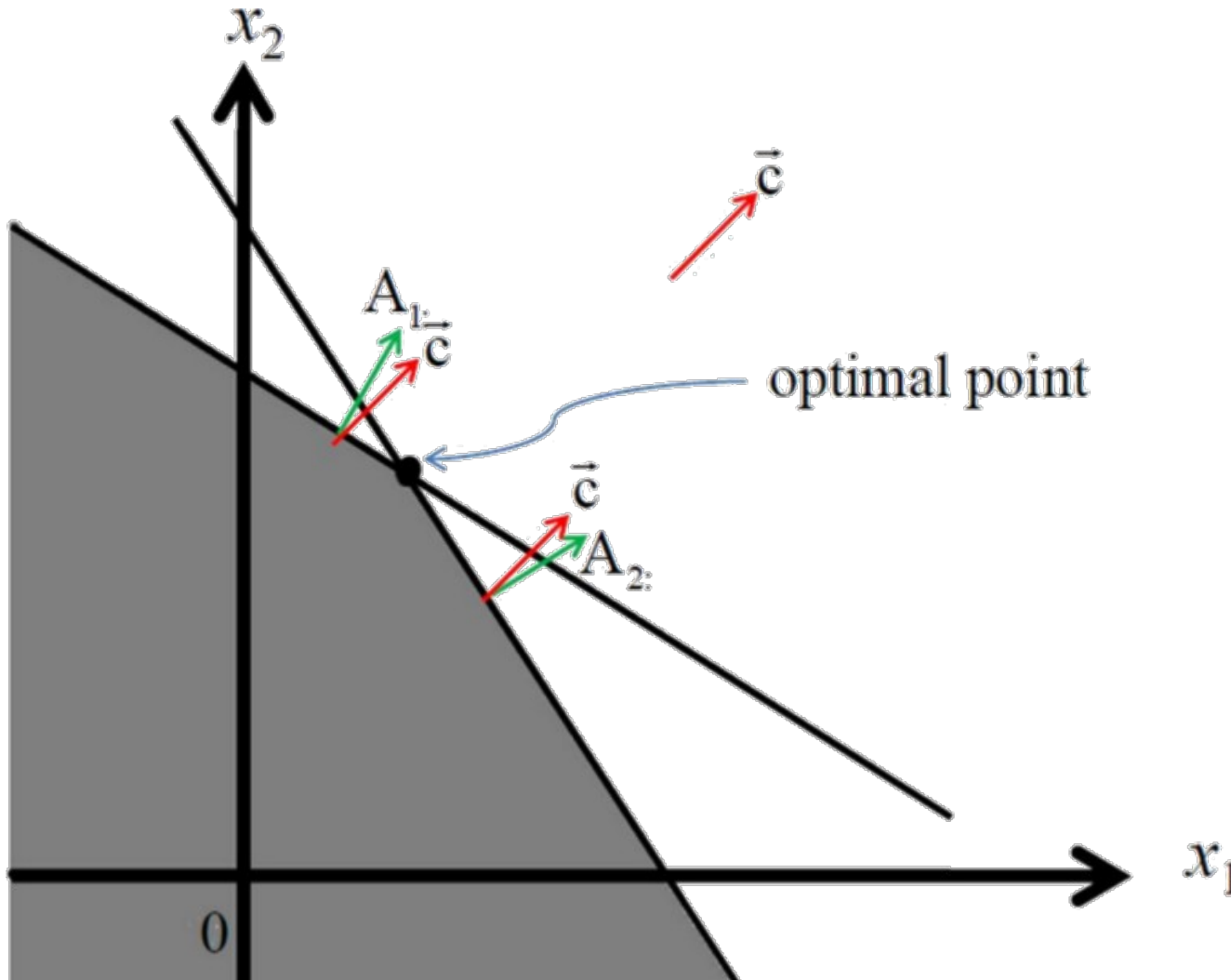


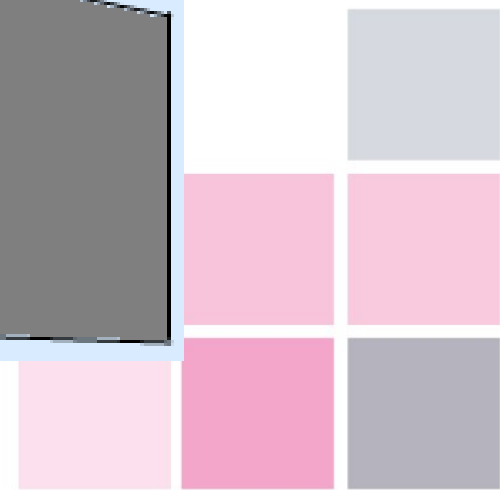
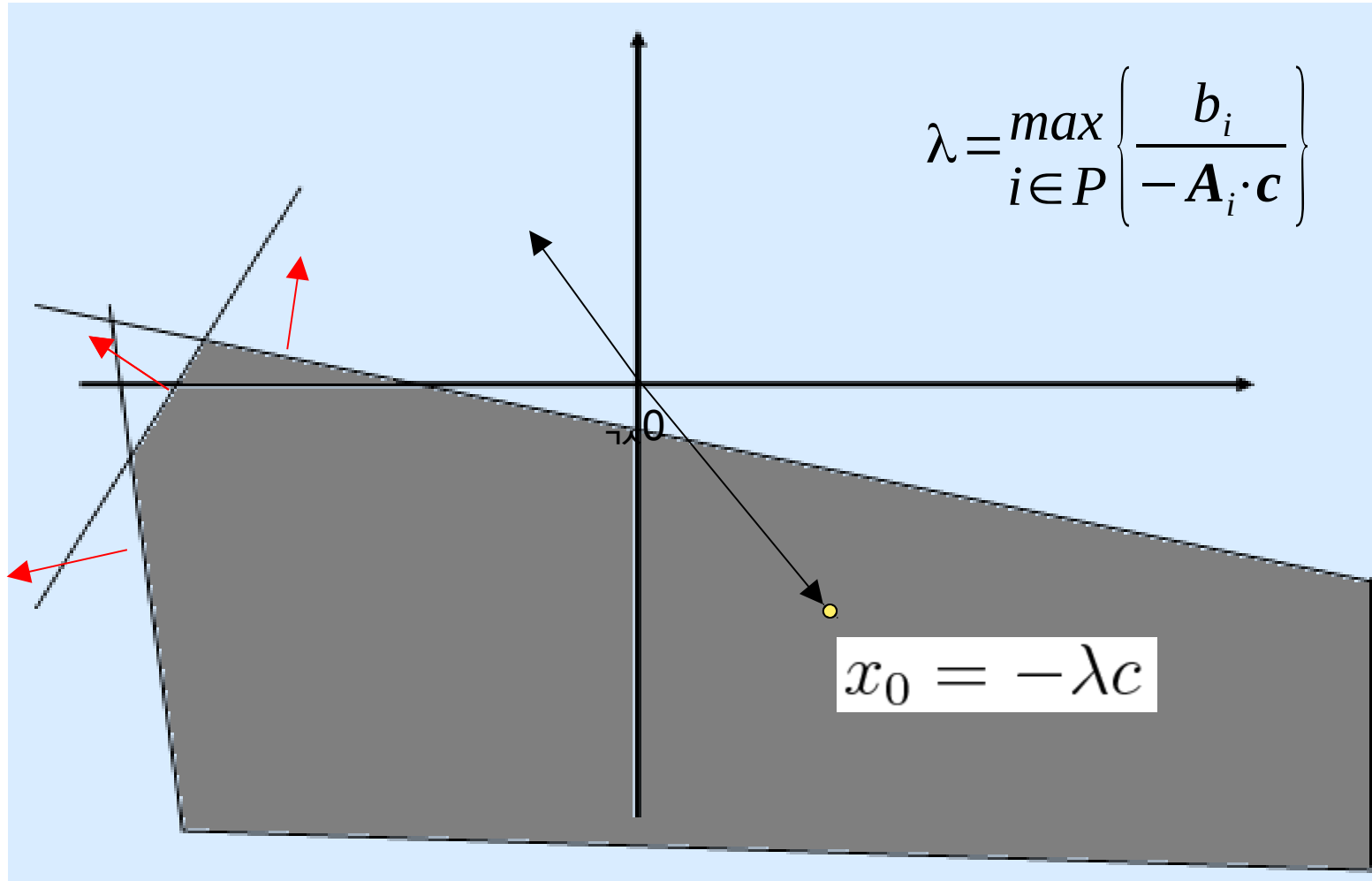


Non-acute constraint relaxation











$$\begin{aligned} \text{Max} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s. t.} \quad & \mathbf{A} \mathbf{x} \leq \mathbf{b} \end{aligned}$$

P is the collection of acute constraints.
N is the collection of non-acute constraints.



Excluded initially

$$\begin{aligned} \text{Max} \quad & \mathbf{c}^T \mathbf{x} \\ \text{s. t.} \quad & \mathbf{A}_P \mathbf{x} \leq \mathbf{b}_P, \mathbf{b}_P \geq \mathbf{0} \end{aligned}$$

$$\text{s. t.} \quad \mathbf{A}_N \mathbf{x} \leq \mathbf{b}_N, \mathbf{b}_N < \mathbf{0}$$

$$\mathbf{x}_0 = -\lambda \mathbf{c}, \lambda = \max_{i \in P} \left\{ \frac{b_i}{-\mathbf{A}_i \cdot \mathbf{c}} \right\}$$





Theorem 1

Let $P^- = \{i \in P | b_i < 0\}$ and $P^+ = \{i \in P | b_i \geq 0\}$. If $P^- \neq \emptyset$ and $\lambda = \max_{i \in P^-} \left\{ \frac{b_i}{-A_{i:\cdot}c} \right\}$. Then $\mathbf{x}_0 = -\lambda \mathbf{c}$ is a feasible point of the NAR.

Corollary 2

If $P \neq \emptyset$ then the NAR is always feasible.





NAR (Non-Acute constraint Relaxation)

$$\begin{aligned} \text{Max} \quad & \mathbf{c}^T \mathbf{x}' + \mathbf{c}^T \mathbf{x}_0 \\ \text{s. t.} \quad & \mathbf{A}_P \mathbf{x}' \leq \mathbf{b}_P - \mathbf{A}_P \mathbf{x}_0 \end{aligned}$$



RHS is positive.

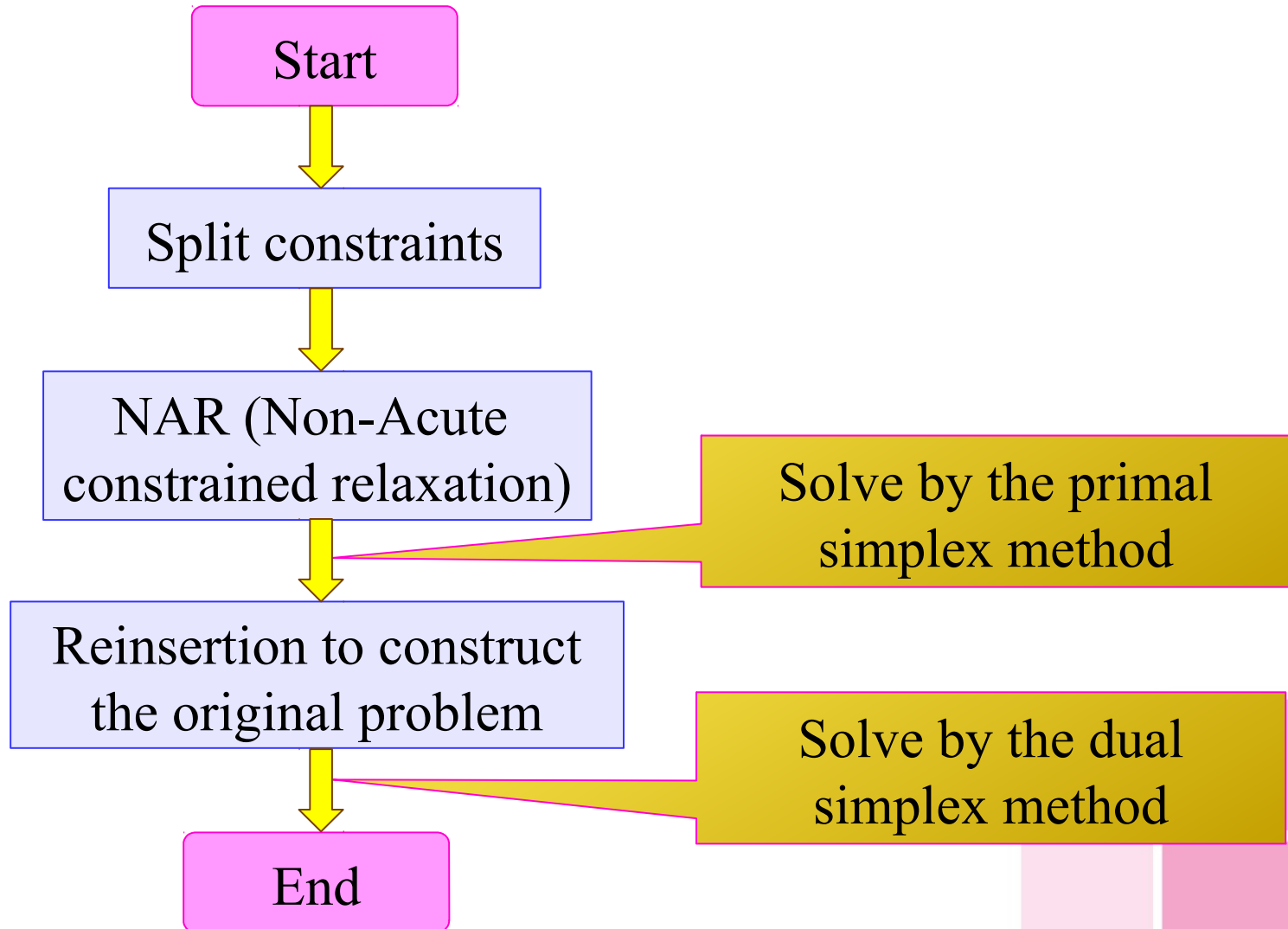
$$\begin{aligned} \text{Max} \quad & \mathbf{c}^T \mathbf{x}' + \mathbf{c}^T \mathbf{x}_0 \\ \text{s. t.} \quad & \mathbf{A}_P \mathbf{x}' + \mathbf{I} \mathbf{s} = \mathbf{b}_P - \mathbf{A}_P \mathbf{x}_0, \mathbf{s} \geq \mathbf{0} \end{aligned}$$

$$\begin{aligned} \text{Max} \quad & \mathbf{c}^T \mathbf{x}^+ - \mathbf{c}^T \mathbf{x}^- + \mathbf{c}^T \mathbf{x}_0 \\ \text{s. t.} \quad & \mathbf{A}_P \mathbf{x}^+ - \mathbf{A}_P \mathbf{x}^- + \mathbf{I} \mathbf{s} = \mathbf{b}_P - \mathbf{A}_P \mathbf{x}_0 \\ & \mathbf{s} \geq \mathbf{0}, \mathbf{x}^+ \geq \mathbf{0}, \mathbf{x}^- \geq \mathbf{0} \end{aligned}$$





Simplex method based on the Non-Acute constraint Relaxation





Simplex method based on the Non-Acute constraint Relaxation

$$\begin{aligned}
 &\text{Max} && x_1 & + & 2x_2 \\
 &\text{s.t.} && -2x_1 & - & x_2 & \leq & -4 \\
 &&& -3x_1 & - & 3x_2 & \leq & -9 \\
 &&& -x_1 & - & 2x_2 & \leq & -4 \\
 &&& -3x_1 & + & x_2 & \leq & 6 \\
 &&& x_1 & - & 3x_2 & \leq & 6 \\
 &&& 2x_1 & - & 3x_2 & \leq & 12 \\
 &&& 3x_1 & + & 5x_2 & \leq & 30 \\
 &&& & & x_2 & \leq & 5 \\
 &&& -x_1 & - & x_2 & \leq & -2 \\
 &&& -4x_1 & - & x_2 & \leq & -4
 \end{aligned}$$

No. (i)	A_i	$A_i \cdot c$
1	[-2, -1]	-4
2	[-3, -3]	-9
3	[-1, -2]	-5
4	[-3, 1]	-1
5	[1, -3]	-5
6	[2, -3]	-4
7	[3, 5]	13
8	[0, 1]	2
9	[-1, -1]	-3
10	[-4, -1]	6



Simplex method based on the Non-Acute constraint Relaxation

$$\begin{aligned}
 \text{Max} \quad & x_1 + 2x_2 \\
 \text{s.t.} \quad & 3x_1 + 5x_2 \leq 30 \\
 & x_2 \leq 5
 \end{aligned}$$

$x_0 = \mathbf{0}$ is the feasible point.

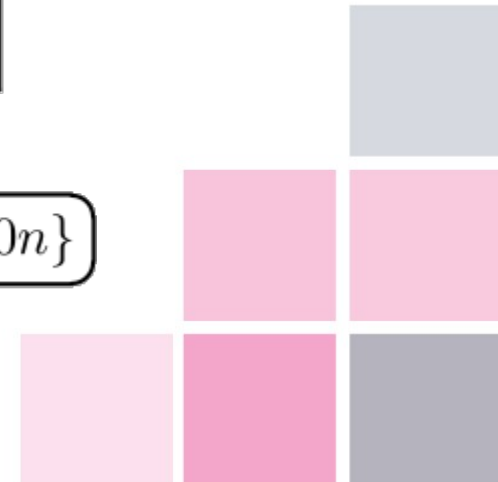
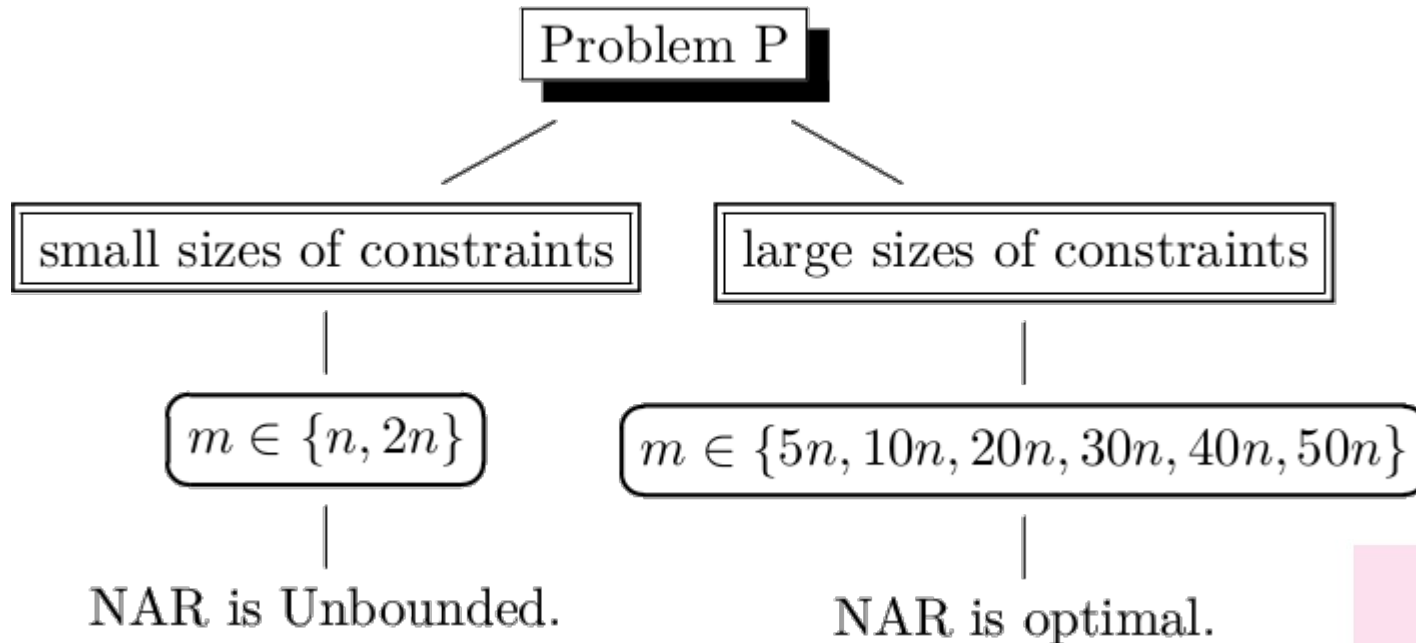
The optimal solution is $x_1 = 5/3$ and $x_2 = 5$.

This point satisfies the rest of the constraints. Therefore, it is the optimal solution for the original problem.

No. (i)	A_i	$A_i \cdot c$
1	[-2, -1]	-4
2	[-3, -3]	-9
3	[-1, -2]	-5
4	[-3, 1]	-1
5	[1, -3]	-5
6	[2, -3]	-4
7	[3, 5]	13
8	[0, 1]	2
9	[-1, -1]	-3
10	[-4, -1]	6



- # variables (n) \rightarrow 5, 10, 20, 50, 100
- # constraints (m) \rightarrow $1n$, $2n$, $5n$, $10n$, $20n$, $30n$, $40n$, $50n$
- For each problem size, 100 different problems were simulated and solved.





The average number of iterations for a small number of constraints

<i>m</i>	<i>n</i>	SNAR				Two-Phase Method				Arsham's Method			
		NAR	AN	NAR+AN	<i>SD</i> ₁	PhaseI	PhaseII	PhaseI+II	<i>SD</i> ₂	RP	≥	RP+≥	<i>SD</i> ₃
5	5	1.91	2.45	4.36	2.52	5.20	2.42	7.62	3.35	1.20	3.78	4.98	2.95
10	5	4.14	3.60	7.74	3.82	10.39	3.51	13.90	4.50	2.18	8.53	10.71	5.63
20	20	8.71	14.10	22.81	10.43	28.11	8.11	36.22	10.67	5.58	28.30	33.88	11.81
40	20	20.68	66.94	87.62	34.69	61.93	18.48	80.41	18.80	10.91	110.34	121.25	46.07
100	100	50.66	222.17	272.83	70.61	177.05	29.16	206.21	42.48	37.57	318.88	356.45	94.60
200	100	127.69	1145.84	1273.53	306.97	411.36	101.59	512.95	80.72	77.22	1884.85	1962.07	448.18

The average number of iterations for a large number of constraints

<i>m</i>	<i>n</i>	SNAR				Two-Phase Method				Arsham's Method			
		NAR	AN	NAR+AN	<i>SD</i> ₁	PhaseI	PhaseII	PhaseI+II	<i>SD</i> ₂	RP	≥	RP+≥	<i>SD</i> ₃
25	5	8.67	1.93	10.60	2.81	20.28	3.97	24.25	4.46	4.90	14.03	18.93	6.25
250	5	17.76	1.38	19.14	3.94	149.18	6.17	155.35	14.31	9.40	25.96	35.36	15.13
100	20	51.23	20.27	71.50	24.97	120.54	27.07	147.61	17.72	22.72	228.51	251.23	55.30
1000	20	112.34	11.12	123.46	12.97	772.61	44.10	816.71	67.38	49.96	300.93	350.89	126.99
500	100	475.54	142.33	617.88	51.44	1077.71	293.92	1371.63	78.42	159.83	6640.71	6800.54	1215.04
5000	100	1000.56	126.56	1127.11	61.58	6109.11	504.44	6613.56	343.01	302.33	7105.11	7407.44	1712.84



Small # of constraints
 $n = 100, 200$ variables
(Iterations)

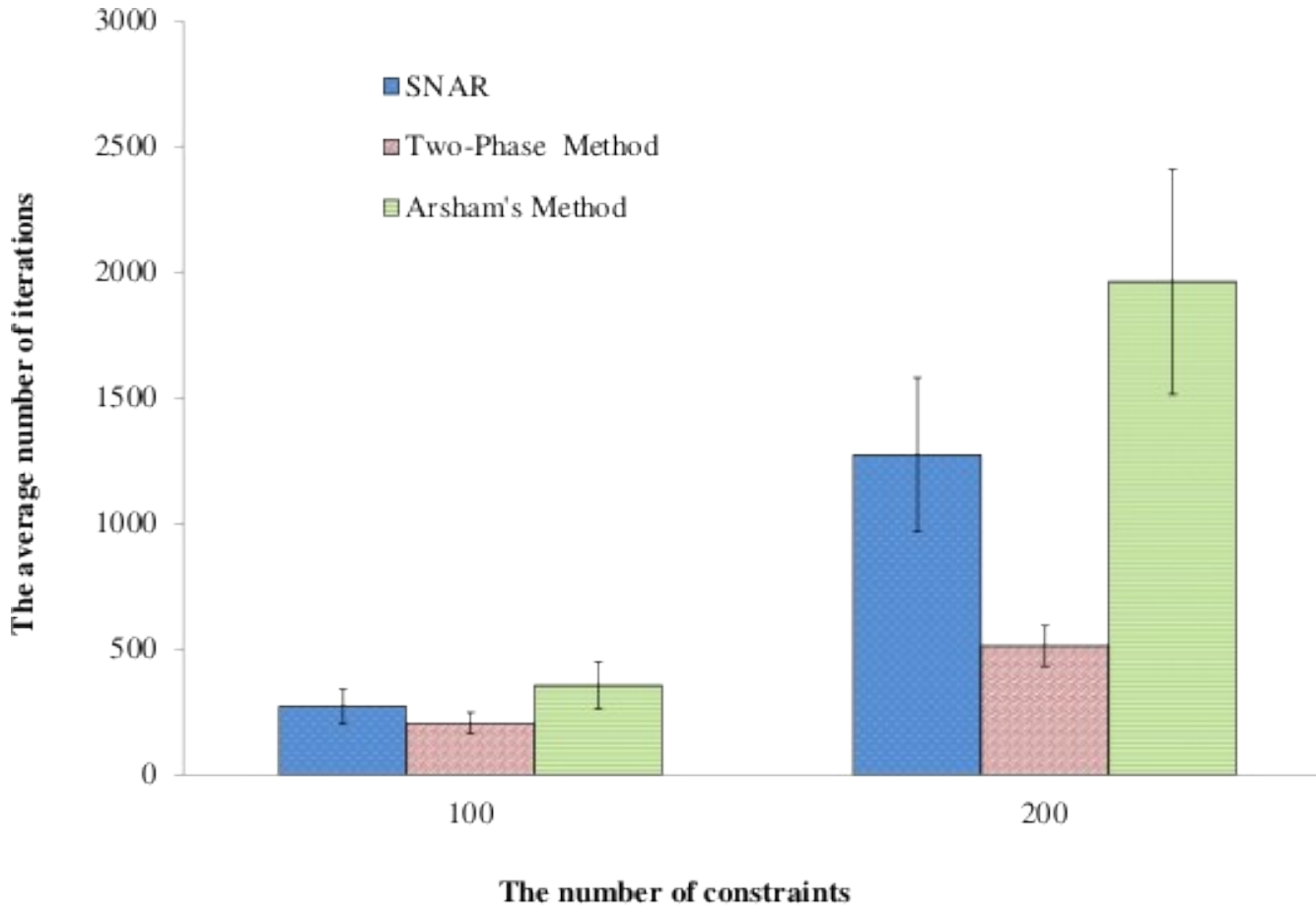


Figure : The average number of iterations





Small # of constraints
 $n = 100, 200$ variables
(Running time)

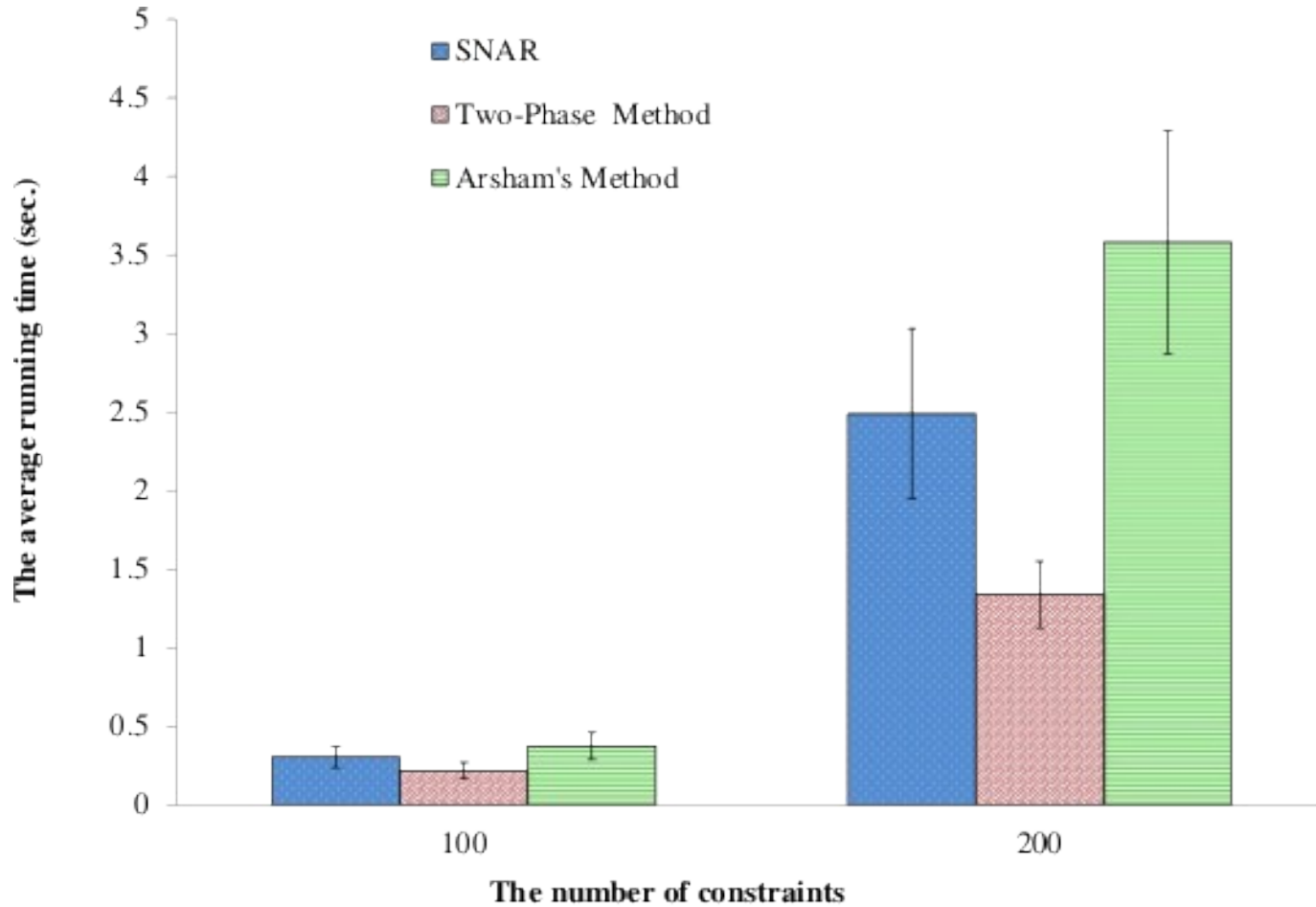


Figure : The average running time





Large # of constraints
 $n = 100$ variables
(Iterations)

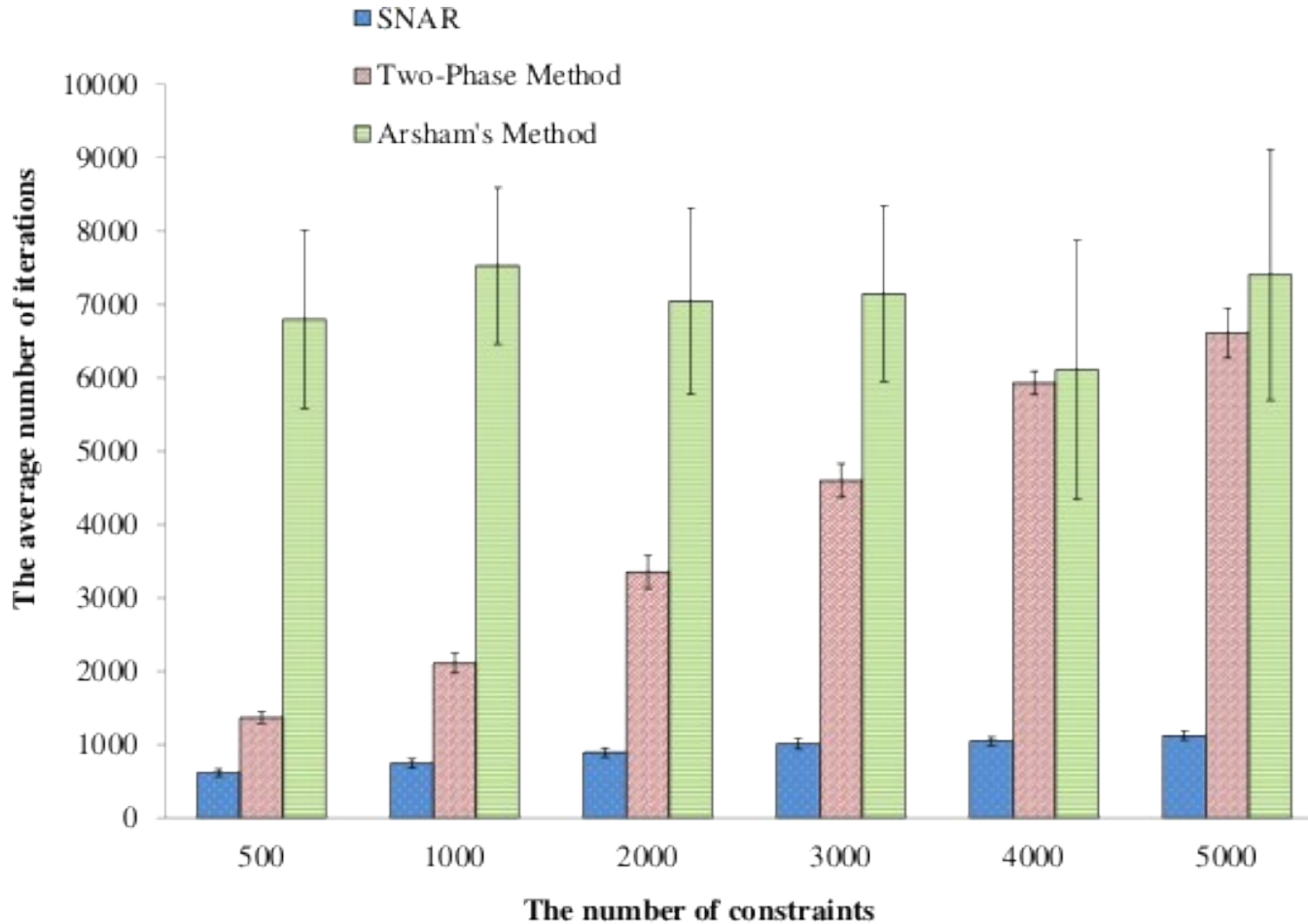
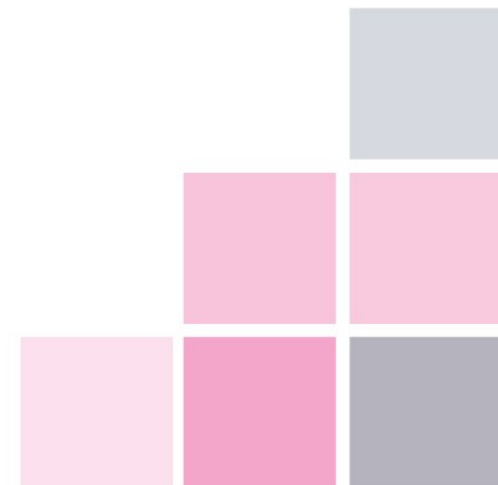


Figure : The average number of iterations





Large # of constraints
 $n = 100$ variables
(Running time)

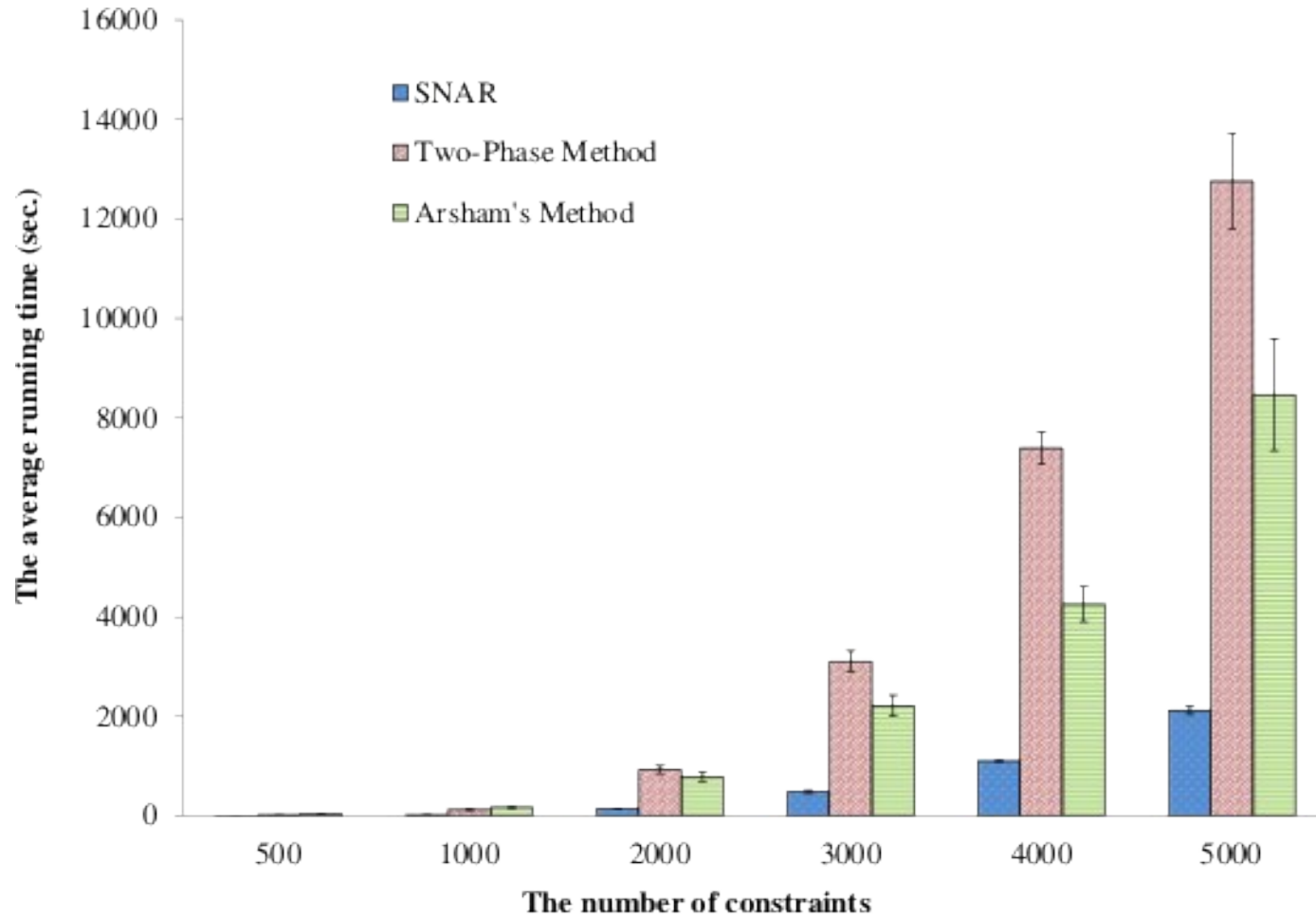
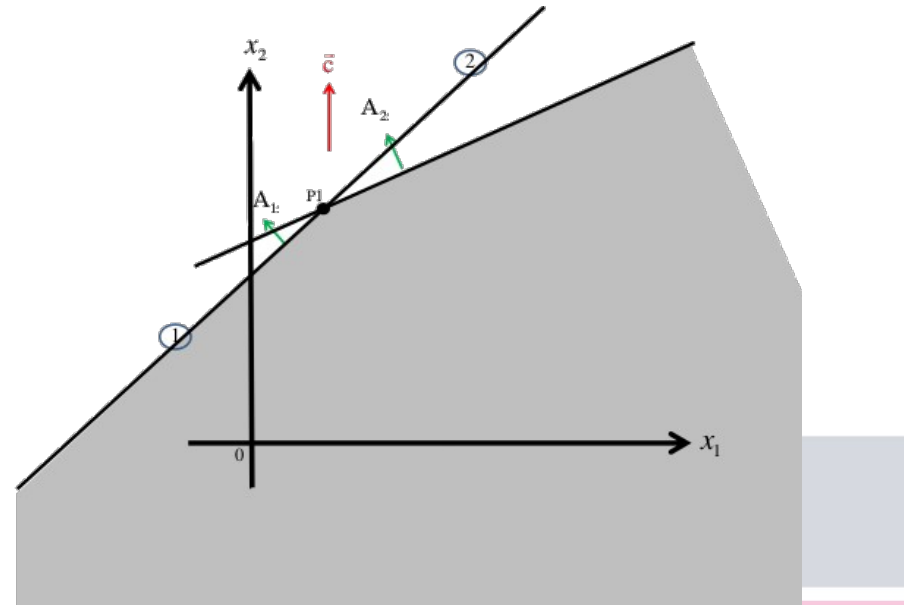
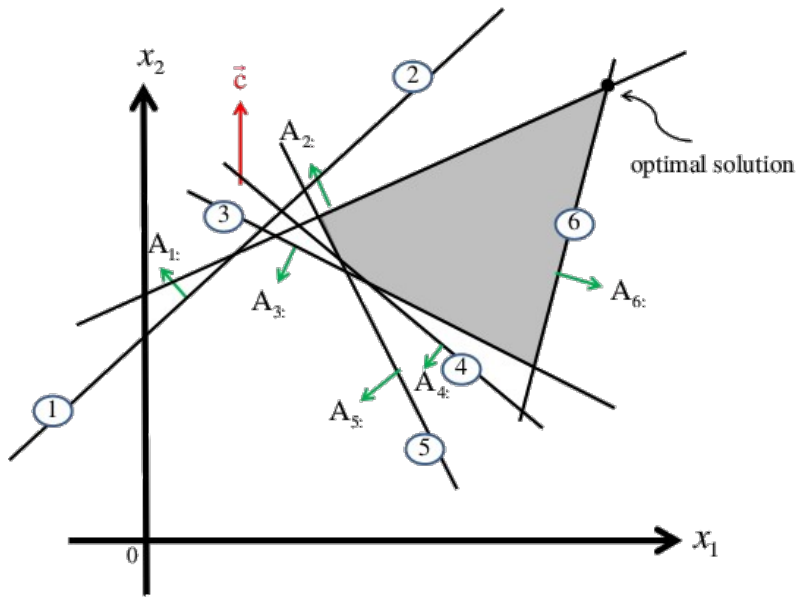
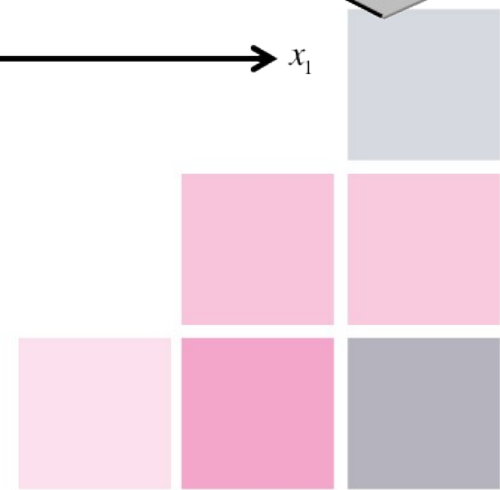
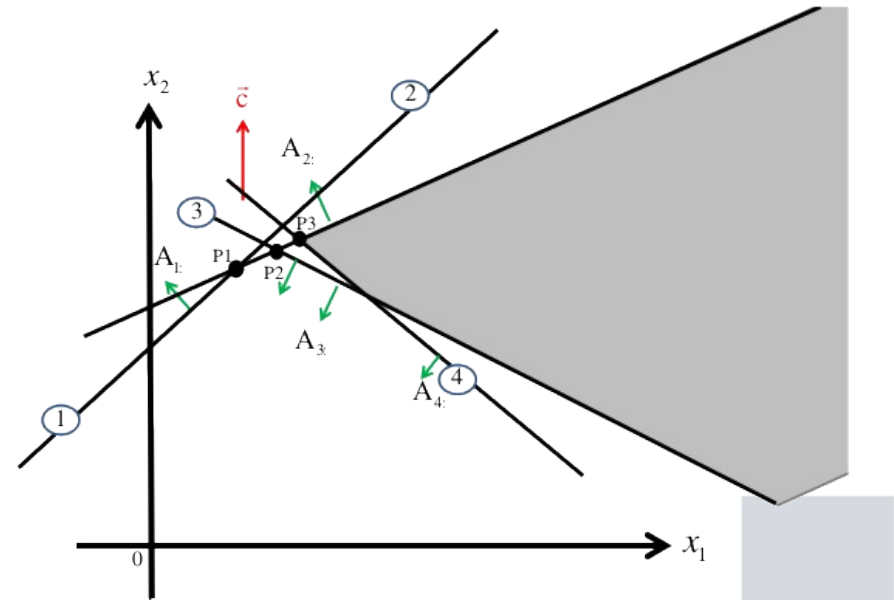
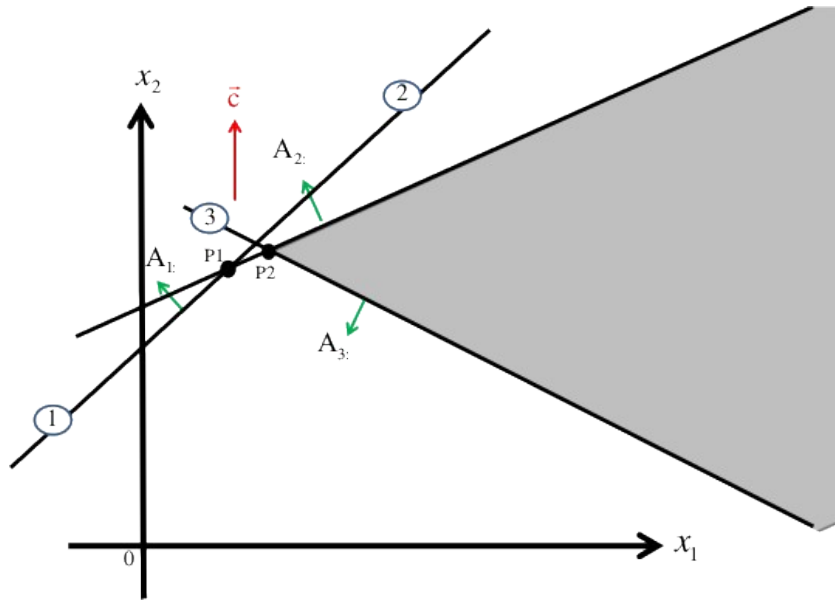
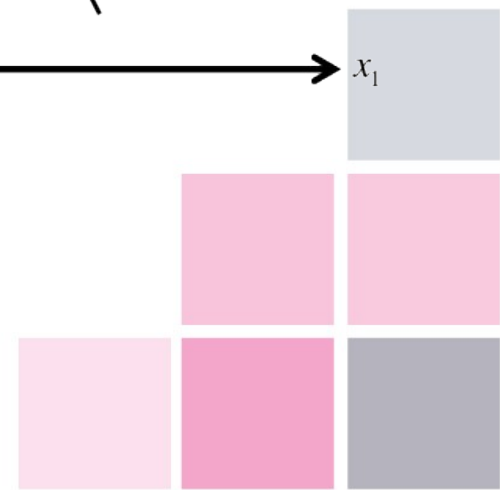
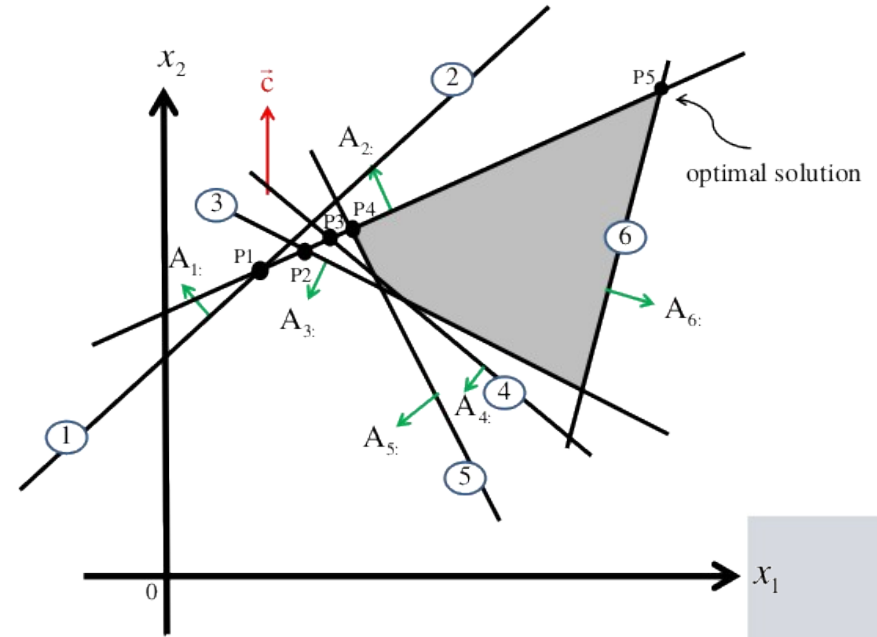
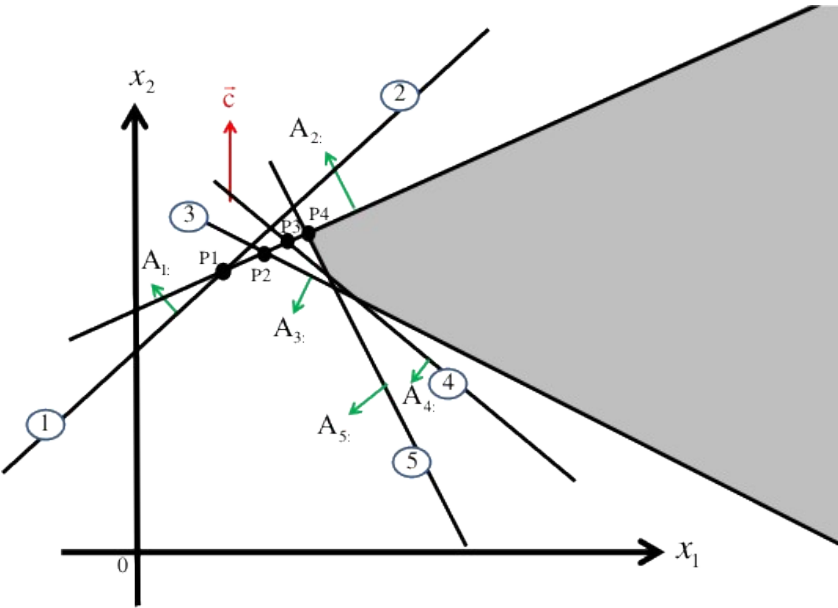


Figure : The average running time











SNAR

- Need no artificial variables (small number of variables)
- Solve the relaxation with less # constraints.
 - It accelerates when the optimal solution is in this relaxation
- Reinsertion after the optimal solution from NAR is easily by the dual simplex method.
 - When solving the relaxation gives the unbounded case, the method performance depends on the order of reinsertion of non-acute constraints.
- SNAR outperforms Two-Phase method and Arsham's method for large number of constraints.

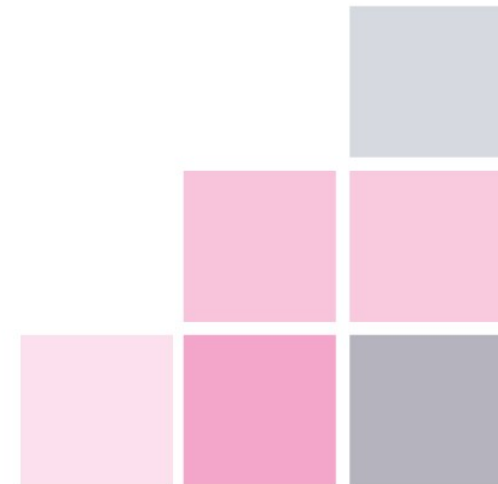




จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University
Pillar of the Kingdom

Page 50

Simplex algorithm with unrestricted variable problem





Unrestricted variable problem for LP

$$\begin{array}{ll} \text{Maximize} & \mathbf{c}^T \mathbf{x} \\ \text{subject to} & \mathbf{A} \mathbf{x} \leq \mathbf{b} \end{array}$$



$$\begin{array}{ll} \text{Maximize} & \mathbf{c}^T \mathbf{x}^+ - \mathbf{c}^T \mathbf{x}^- \\ \text{subject to} & \mathbf{A} \mathbf{x}^+ - \mathbf{A} \mathbf{x}^- \leq \mathbf{b} \\ & \mathbf{x}^+ \geq \mathbf{0}, \mathbf{x}^- \geq \mathbf{0} \end{array}$$

Unnecessary double the
number of variables



$$\begin{array}{ll} \text{Maximize} & \mathbf{c}^T \mathbf{x}^+ - \mathbf{c}^T \mathbf{x}^- + \mathbf{0}^T \mathbf{s} \\ \text{subject to} & \mathbf{A} \mathbf{x}^+ - \mathbf{A} \mathbf{x}^- + \mathbf{I} \mathbf{s} = \mathbf{b} \\ & \mathbf{x}^+ \geq \mathbf{0}, \mathbf{x}^- \geq \mathbf{0}, \mathbf{s} \geq \mathbf{0} \end{array}$$





Unrestricted variable problem for LP

$$\begin{aligned} &\text{Maximize} && \mathbf{c}^T \mathbf{x} \\ &\text{subject to} && \mathbf{A} \mathbf{x} \leq \mathbf{b} \end{aligned}$$



$$\begin{aligned} &\text{Maximize} && \mathbf{c}_U^T \mathbf{x}_U + \mathbf{c}_R^T \mathbf{x}_R \\ &\text{subject to} && \mathbf{A}_U \mathbf{x}_U + \mathbf{A}_R \mathbf{x}_R \leq \mathbf{b} \\ &&& \mathbf{x}_R \geq \mathbf{0} \end{aligned}$$



$$\begin{aligned} &\text{Maximize} && \mathbf{c}_U^T \mathbf{x}_U + \mathbf{c}_R^T \mathbf{x}_R + \mathbf{0}^T \mathbf{s} \\ &\text{subject to} && \mathbf{A}_U \mathbf{x}_U + \mathbf{A}_R \mathbf{x}_R + \mathbf{I} \mathbf{s} = \mathbf{b} \\ &&& \mathbf{x}_R \geq \mathbf{0}, \mathbf{s} \geq \mathbf{0} \end{aligned}$$

Handle unrestricted variable directly within the entering variable selection step

If unrestricted variable is changed to the basic variable, then that equation can be dropped.

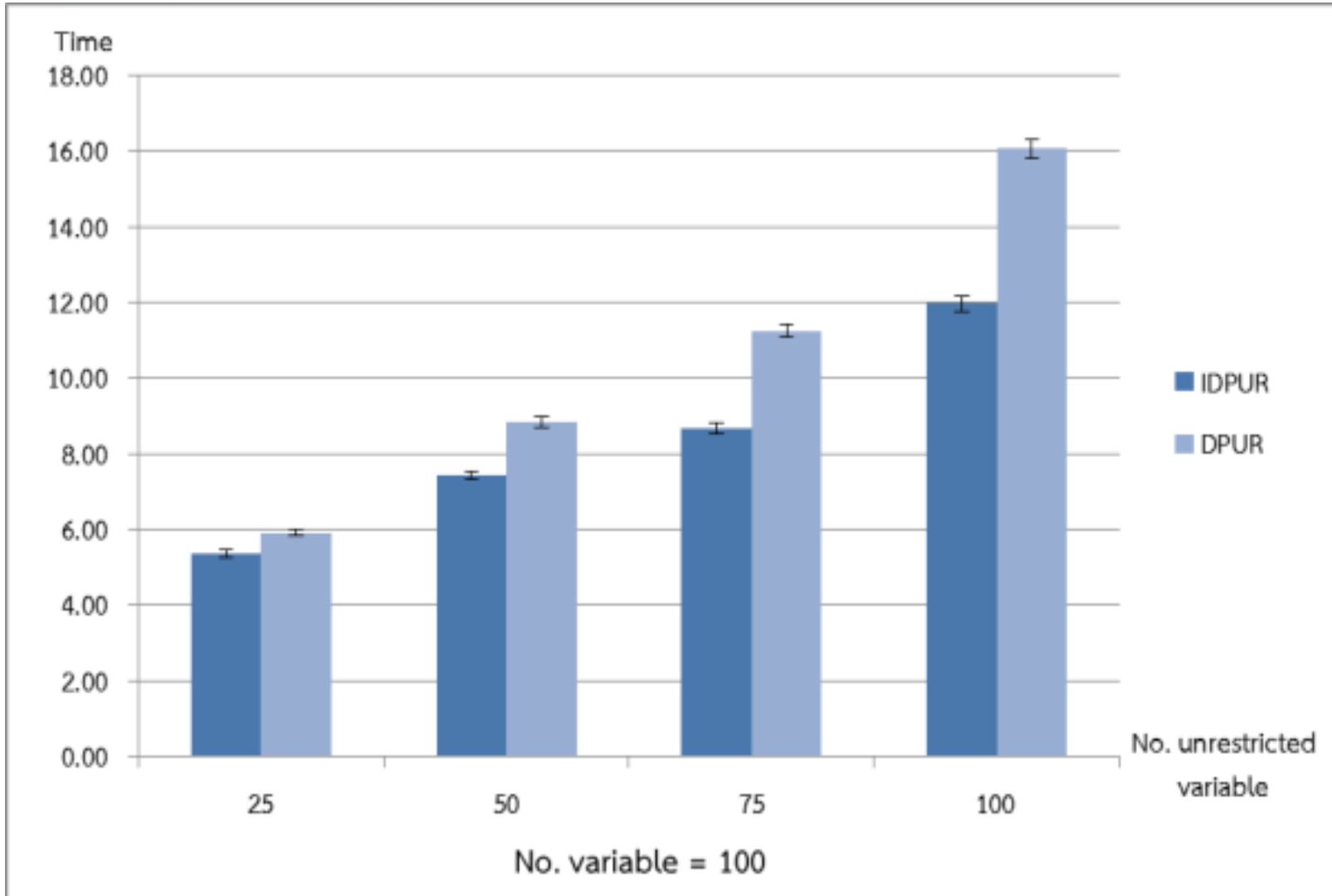




Problem size			Average time	
No. constraint	No. Variable	No. unrestricted variable	IDPUR	DPUR
100	100	25	5.354400	5.910860
		50	7.426180	8.833380
		75	8.658820	11.234920
		100	11.957900	16.057060
	200	50	22.824940	25.088420
		100	29.233360	35.537500
		150	22.153700	30.777500
		200	18.359740	29.345620
	300	75	41.144180	45.221960
		150	35.768360	45.431640
		225	29.496760	43.481600
		300	24.958480	41.754000

DPUR is the text-book method and IDPUR is our method.

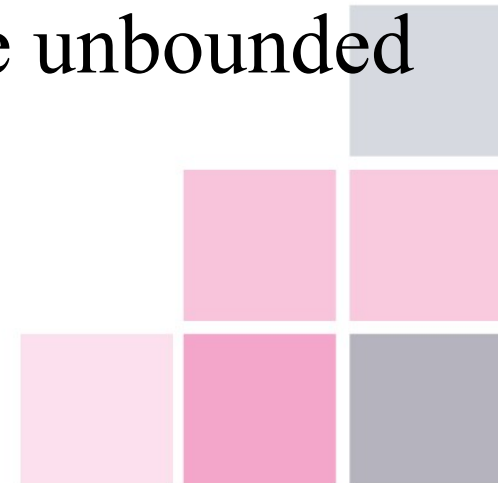




DPUR is the text-book method and IDPUR is our method.



- Revise the initial basic feasible solution \mathbf{x}_0 .
 - Using cosine angle
 - Using the directional vector
- Improve the artificial-free simplex algorithm
 - Handle unrestricted variable directly
 - Handle insertion of n linearly independent defining hyperplane instead when the unbounded solution of the relaxation is reached.





Assistant Professor Krung Sinapiromsaran

- Department of Mathematics and Computer Science
- 254 Phayathai Road, Pathumwan,
Bangkok Thailand. 10330
- <http://pioneer.netserv.chula.ac.th/~skrung>
- Krung.S@chula.ac.th
- 02-218-5150

