EFFECTIVE SOLUTION ALGORITHMS FOR BI-OBJECTIVE DISCRETE CONTINUOUS OPTIMIZATION PROBLEMS

Metin Türkay
College of Engineering
Koç University, Istanbul

AIChE Annual Meeting 2014
Atlanta, November 17, 2014
Paper 192d – In Honor of Ignacio E. Grossmann's 65th Birthday II

OUTLINE

- Multi-Objective Discrete-Continuous Optimization Problems
- Terminology
- Existing Methods
  - ✓ ϵ-constraint method and its extensions
  - ✓ CAN algorithm for finding extreme supported nodominated points
- Novel Algorithms
  - ✓ ϵ-*OA* for MINLP
  - ✓ EnpoBOMIP for MILP
- Performance Comparison
  - MINLP
  - MILP
- Summary

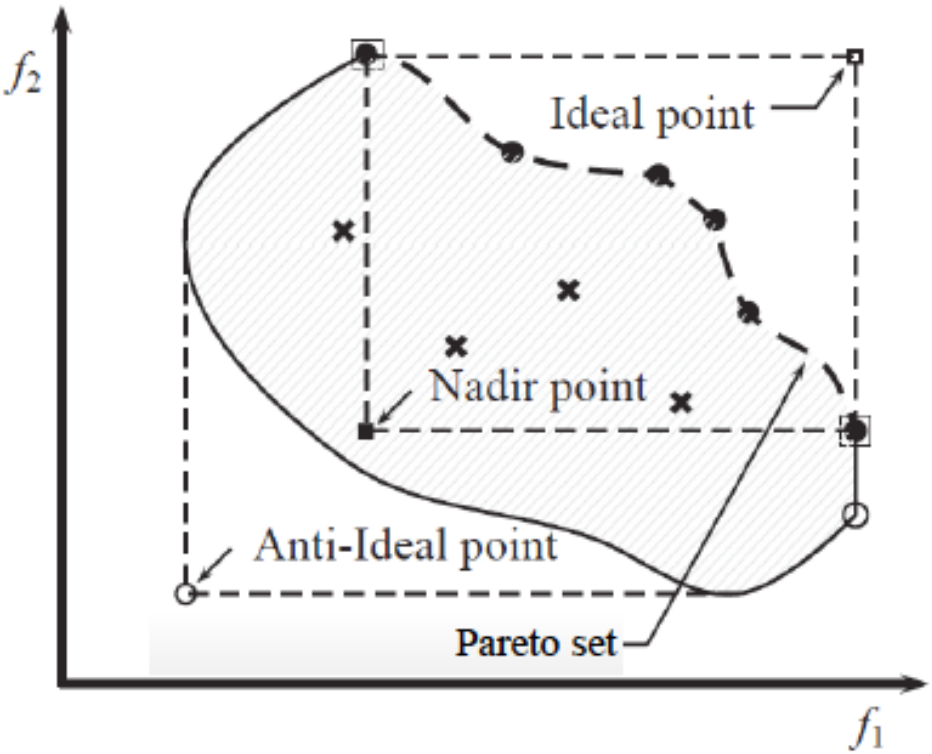
MULTI-OBJECTIVE OPTIMIZATION

- Multi-objective optimization problems (MOOP) involve optimizing simultaneously N objective functions $f_1, f_2, \ldots, f_N$ over a feasible set X.

$$\max \; F(x) = \left(f_1(x), \ldots, f_N(x) \right)$$
$$s.t. \qquad x \in X$$

- Many survey papers were published (Ulungu&Teghem, 1994; Ehrgott&Gandibleux, 2000; Ehrgott, 2005)
- Research on solution algorithms:
  - Continuous and convex: a variety of algorithms – maturing
  - Continuous and nonconvex: a few algorithms
  - Combinatorial: a number of papers in the last 5 years
  - Discrete-continuous – linear: a handful of papers in the last 3 years
  - Discrete-continuous – nonlinear: only 5 papers straighforward use of algorithms developed for continuous and convex problems

TERMINOLOGY



- Let $x, x' \in X$
- x dominates x'
  $f_n(x) \geq f_n(x') \quad \forall n = 1,\ldots,N$ and $\exists \tilde{n} \in \{1,\ldots,N\}$
  with $f_{\tilde{n}}(x) > f_{\tilde{n}}(x')$
- x strictly dominates x'
  $f_n(x) > f_n(x') \quad \forall n = 1,\ldots,N$
- x weakly dominates x'
  $f_n(x) \geq f_n(x') \quad \forall n = 1,\ldots,N$
- x is Pareto optimal or efficient
  $\forall x' \in X$ that does not dominate x

- *Ideal Point (Utopia Point):* all objectives are optimized simultaneously
- *Anti-Ideal Point:* all objectives are at their worst
- *Pareto set:* entire set of non-dominated solutions
- *Nadir Point:* lower bound of each objective in the Pareto set

ϵ-CONSTRAINT APPROACH

- Haimes et al., 1971
  - ✓ Presented the ϵ-constraint approach to solving MOOP.
  - ✓ The maximum and minimum values for all objectives are found separately
  - ✓ One of the objectives is retained and the rest of the objectives are converted into constraints
  - ✓ A virtual grid is constructed to include all N-1 objective functions.
  - ✓ Then the following sub-problem is solved iteratively for each i_j

$$
\begin{aligned}
&\max\ f_1(x) \\
&s.t. \\
&f_j(x) \geq Lb_j + i_j \epsilon_j \quad \forall j = 2, \ldots, N \\
&\qquad x \in X
\end{aligned}
$$

Lb_j: the lower bound on the objective j

ϵ_j: range of the objective j in the iteration i_j

AUGMENTED ϵ-CONSTRAINT APPROACH

- The ϵ-constraint approach may find weakly efficient solutions.
- Mavrotas, 2009
  - ✓ Modified the ϵ-constraint method by introducing a slack variable to the each objective that is converted into a constraint.
  - ✓ A penalty term with scalar μ (10^{-3}-10^{-6}) is added to the objective

$$\max\ f_1(x) + \mu \sum_{j=2}^{N} \frac{s_j}{r_j}$$

$$s.t.$$

$$f_j(x) - s_j = Lb_j + i_j \epsilon_j \quad \forall j = 2,\ldots,N$$

$$s_j \geq 0 \quad \forall j = 2,\ldots,N$$

$$x \in X$$

s_j: slack variable for each objective j that is converted into a constraint

r_j: the range of objective j

Lb_j: the lower bound on the objective j

ϵ_j: range of the objective j in the iteration i_j

CAN

- The CAN, presented by Cohon1978, Aneja&Nair1979, is an exact algorithm to find all Extreme Supported Nondominated (ESN) points of BOMILPs.

- In each iteration of the CAN two ESN points $(\dot{z},\check{z})$ are known and the algorithm searches for another ESN point between them.

$$P^{CAN}(\dot{z},\check{z}): \beta = \min z_2 - m z_1 \text{ s.t. } (x,y) \in S$$

where $m=(\dot{z}_2-\check{z}_2)/(\dot{z}_1-\check{z}_1)$.

- If $\beta = z_2 - z_1$, then there is no ESN between $(\dot{z},\check{z})$
- Otherwise, a new ESN is found, hence the algorithm continues to search between the new point and $\dot{z}$, and $\check{z}$.

LOGIC-BASED OA

$$\min \quad Z = (z_1, z_2) = \left(\sum_i c_i + f_1(x), \sum_i p_i + f_2(x) \right)$$

$$\text{s.t.} \quad g(x) \le 0,$$

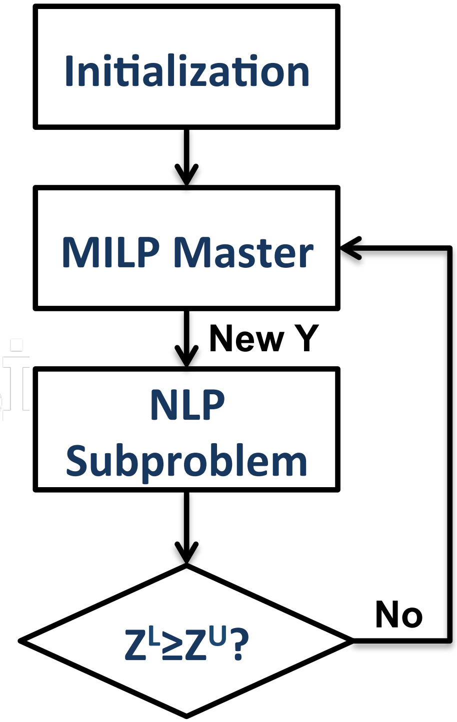
$$\begin{bmatrix} Y_i \\ h_i(x) \le 0 \\ c_i = \gamma_i \\ p_i = \eta_i \end{bmatrix} \vee \begin{bmatrix} \neg Y_i \\ B^i x = 0 \\ c_i = 0 \\ p_i = 0 \end{bmatrix} \quad \forall i \in D,$$

$$\Omega(Y) = True,$$

$$x \in \mathbb{R}^n, c \ge 0, p \ge 0, Y \in \{True, False\}^m.$$

Turkay&Grossmann, 1996

- Logic-Based OA
- Logic-Based Benders Decomposition



NLP SUBPROMLEMS & MILP MASTER

NLP Subproblems

$$\min \quad Z_U^{aug} = \sum_i c_i + f_1(x) - \frac{\mu}{r} s$$

$$\text{s.t.} \quad \sum_i p_i + f_2(x) + s = z_2^U - j\epsilon,$$

$$g(x) \leq 0,$$

$$\left.\begin{array}{l} h_i(x) \leq 0 \\ c_i = \gamma_i \\ p_i = \eta_i \end{array}\right\} \forall \overline{Y_i} = True,$$

$$\left.\begin{array}{l} B^i x = 0 \\ c_i = 0 \\ p_i = 0 \end{array}\right\} \forall \overline{Y_i} = False,$$

$$x \in \mathbb{R}^n, \; c \geq 0, \; p \geq 0.$$

MILP Master

$$\min \quad Z_L^{aug} = \sum_i \gamma_i y_i + \alpha_{oa} - \frac{\mu}{r} s$$

$$\text{s.t.} \quad \alpha_{oa} \geq f_1(x^l) + \nabla f_1(x^l)^T (x - x^l) \quad \forall l = 1, \ldots, L$$

$$g(x^l) + \nabla g(x^l)^T (x - x^l) \leq 0 \quad \forall l = 1, \ldots, L$$

$$\sum_i \eta_i y_i + f_2(x^l) + \nabla f_2(x^l)^T (x - x^l) + s = z_2^U - j\epsilon \quad \forall l = 1, \ldots, L$$

$$\nabla h_i(x^l)^T x \leq \left(-h_i(x^l) + \nabla h_i(x^l)^T x^l\right) y_i \quad \forall l = 1, \ldots, L, \; i \in D$$

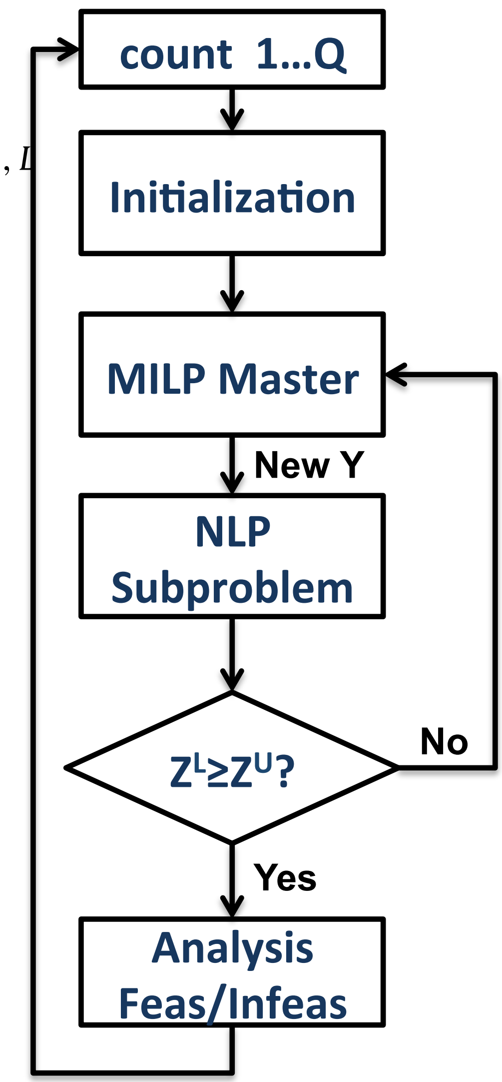
$$B^i x \leq M_i y_i \quad \forall i \in D$$

$$Ay \leq a$$

$$\alpha_{oa} \in \mathbb{R}^1, \; x \in \mathbb{R}^n, \; y \in \{0, 1\}^m.$$

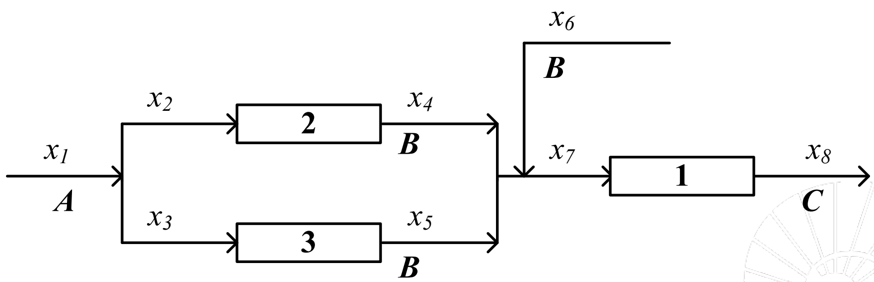
+ No good cuts

ϵ-OA

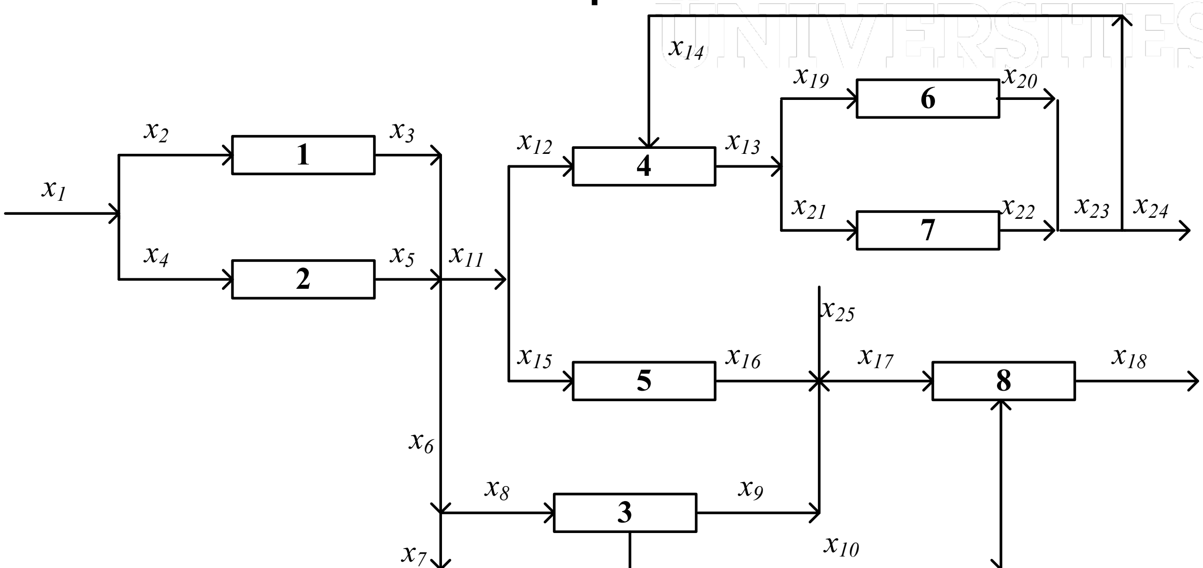


EXAMPLES

➢ 3 Process Example



➢ 8 Process Example



$$\min \quad Z = (z_1, z_2)$$

$$z_1 = c_1 + c_2 + c_3 + x_4 + 1.8x_1 + 1.2x_5 + 7x_6 - 11x_8$$

$$z_2 = p_1 + p_2 + p_3$$

$$+1.8x_1 + 2x_2 + 3x_3 + x_4 + 1.2x_5 + 7x_6 - 5x_7 - 11x_8$$

$$x_1 - x_2 - x_3 = 0,$$

$$x_7 - x_4 - x_5 - x_6 = 0,$$

$$x_5 \leq 5,$$

$$x_8 \leq 1$$

$$\begin{bmatrix} Y_1 \\ x_8 = 0.9x_7 \\ c_1 = 3.5 \\ p_1 = 3.5 \end{bmatrix} \vee \begin{bmatrix} \neg Y_1 \\ x_7 = x_8 = 0 \\ c_1 = 0 \\ p_1 = 0 \end{bmatrix},$$

$$\begin{bmatrix} Y_2 \\ x_4 = \ln(1 + x_2) \\ c_2 = 1 \\ p_2 = 1 \end{bmatrix} \vee \begin{bmatrix} \neg Y_2 \\ x_2 = x_4 = 0 \\ c_2 = 0 \\ p_2 = 0 \end{bmatrix},$$

$$\begin{bmatrix} Y_3 \\ x_5 = 1.2\ln(1 + x_3) \\ c_3 = 1.5 \\ p_3 = 12 \end{bmatrix} \vee \begin{bmatrix} \neg Y_3 \\ x_3 = x_5 = 0 \\ c_3 = 0 \\ p_3 = 0 \end{bmatrix}$$

$$Y_1 \Rightarrow Y_2 \vee Y_3 \vee (\neg Y_2 \wedge \neg Y_3),$$

$$Y_2 \Rightarrow Y_1,$$

$$Y_3 \Rightarrow Y_1,$$

$$\neg Y_2 \vee \neg Y_3$$

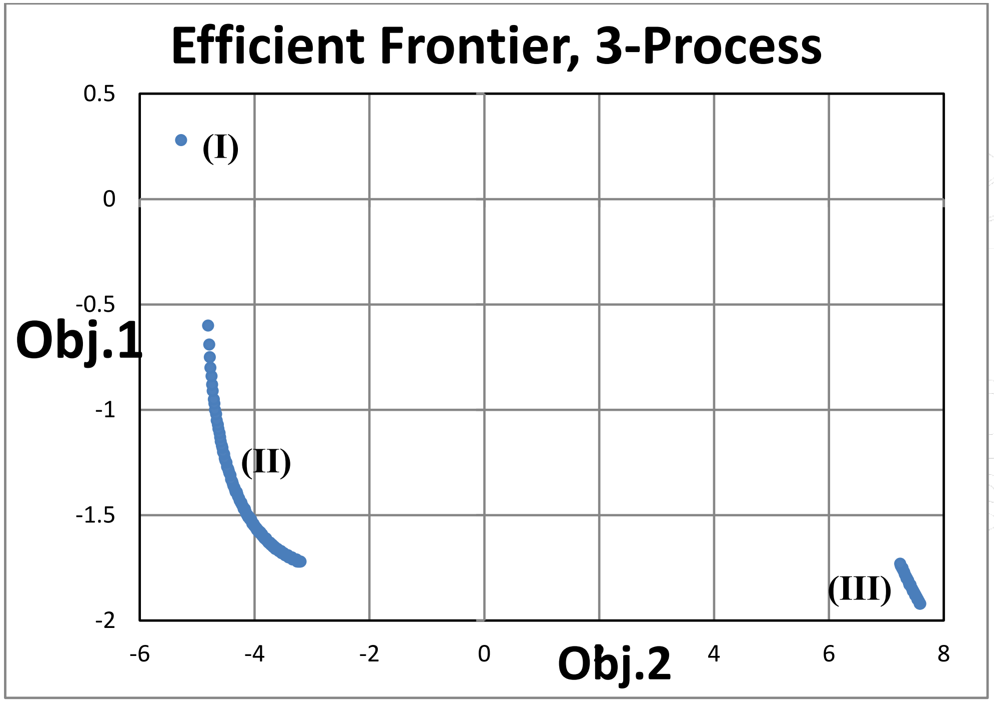
$$x_k \geq 0 \quad \forall k = 1, \ldots, 8,$$

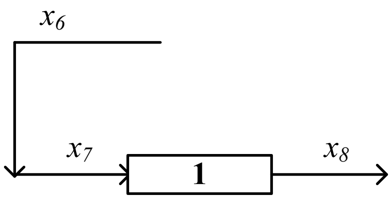
$$c_i, p_i \geq 0 \quad \forall i = 1, 2, 3,$$

$$Y_i \in \{True, False\} \quad \forall i = 1, 2, 3$$

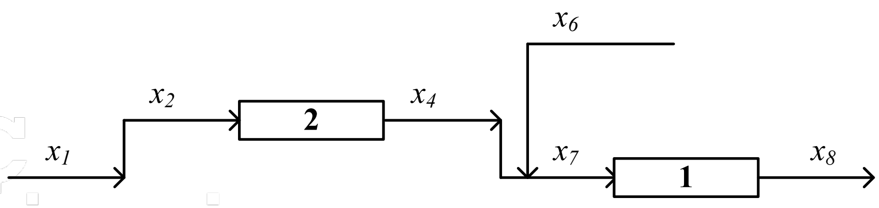
PARETO SOLUTIONS-1

- Example 1: 3 process

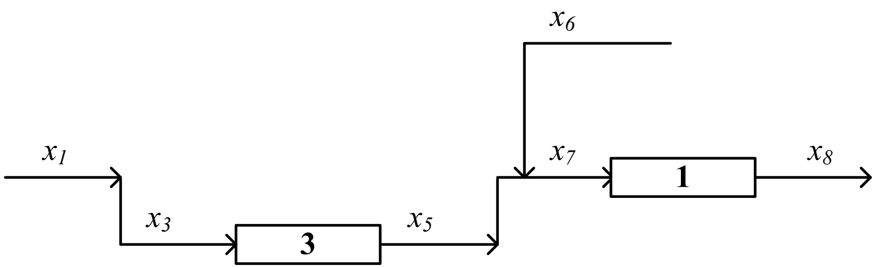




a) Topology for section (I)



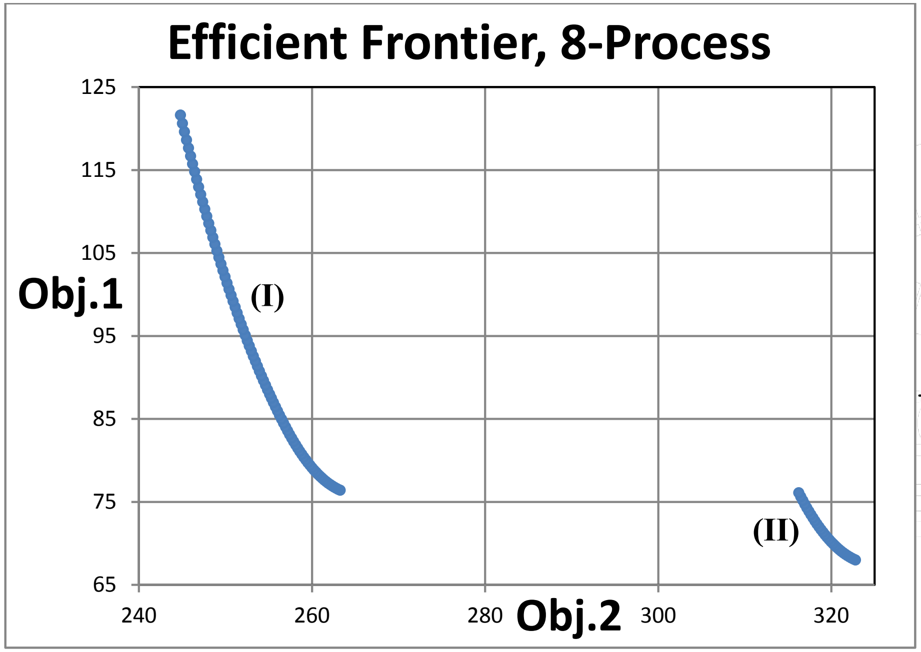
b) Topology for section (II)

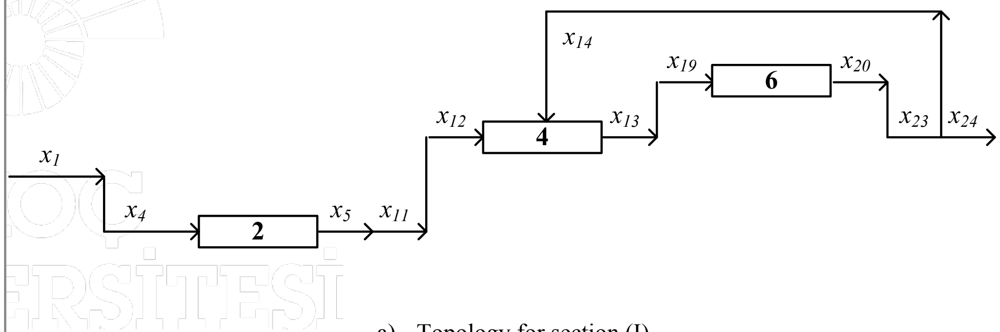


c) Topology for section (III)

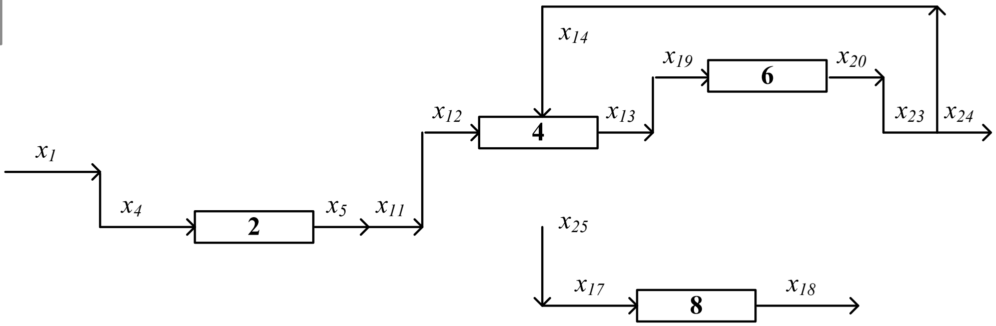
PARETO SOLUTIONS-2

- Example 1: 8 process





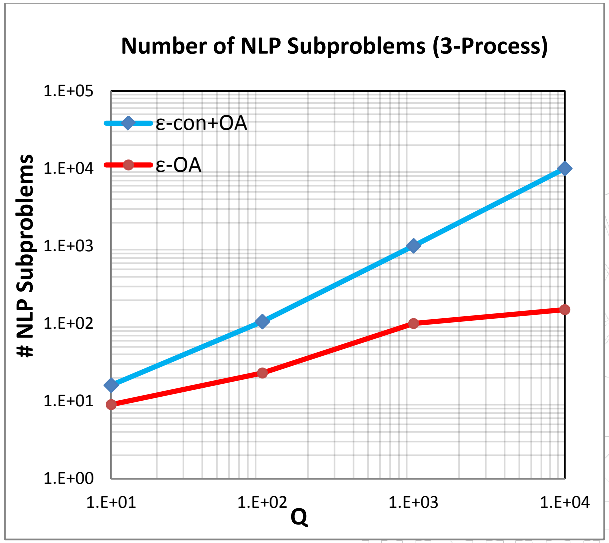
a) Topology for section (I)

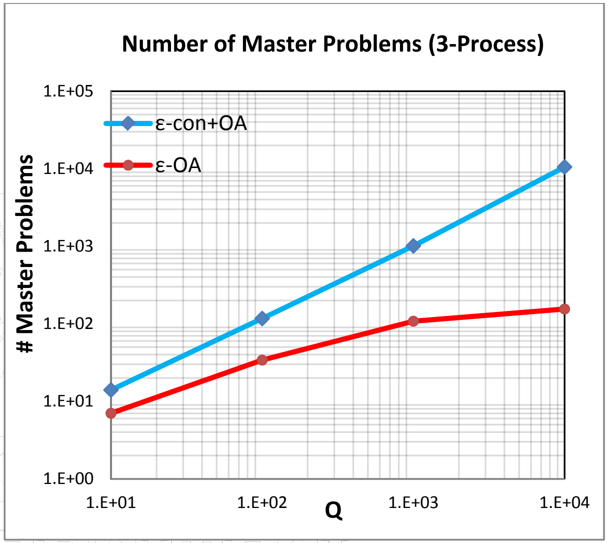


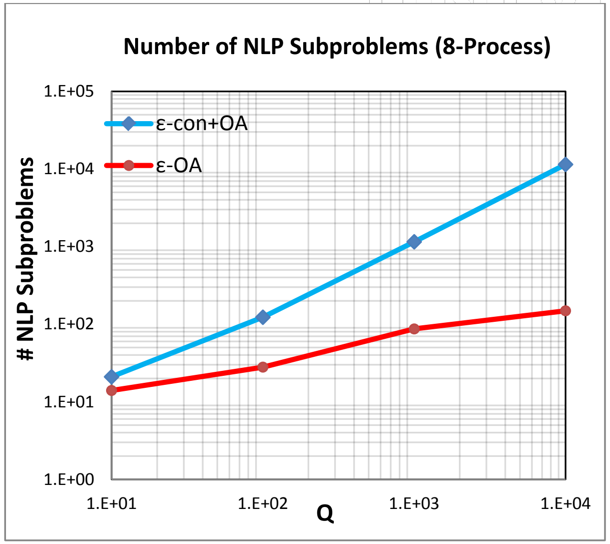
b) Topology for section (II)

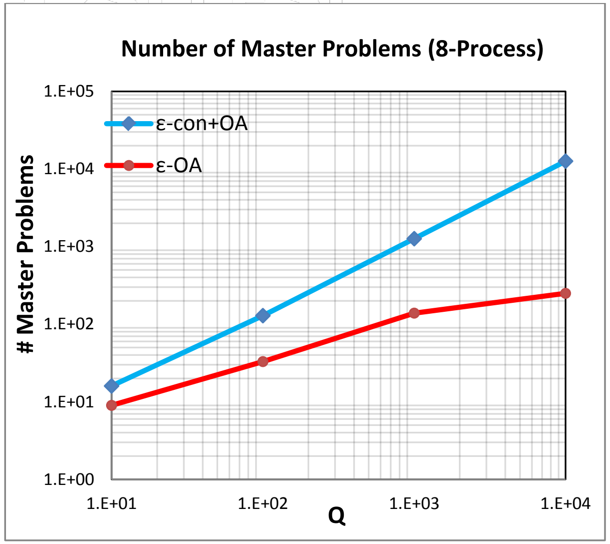
ITERATIONS

➢ The number of NLP subproblems and MILP master problems



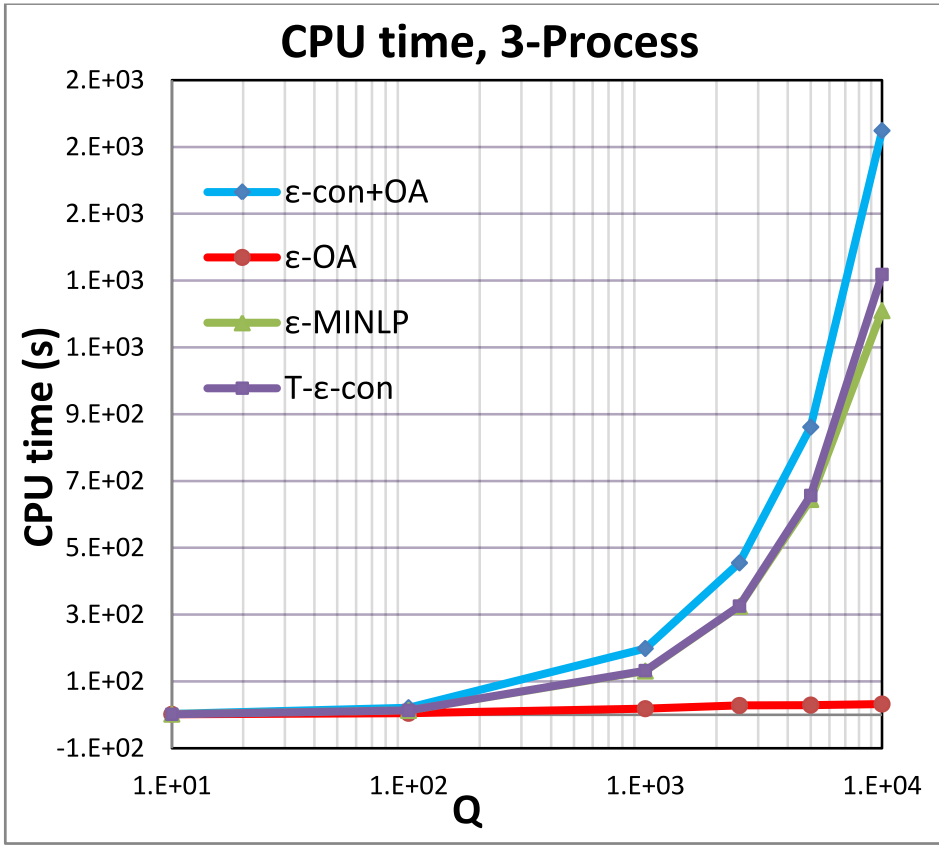


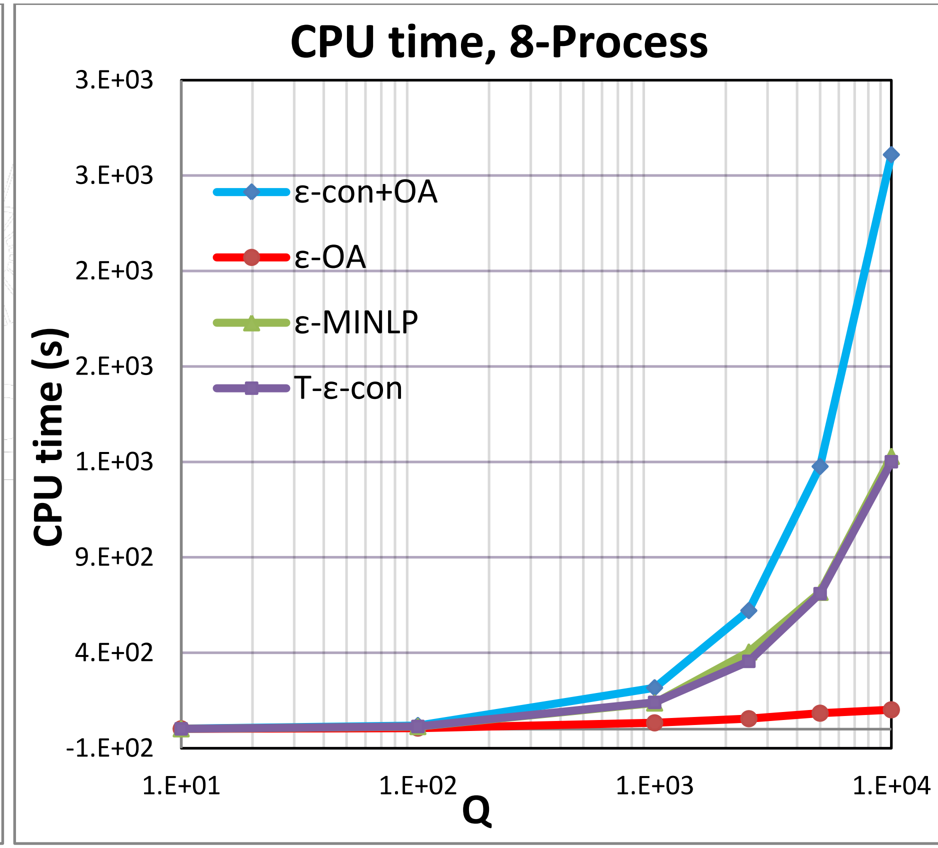




CPU TIMES

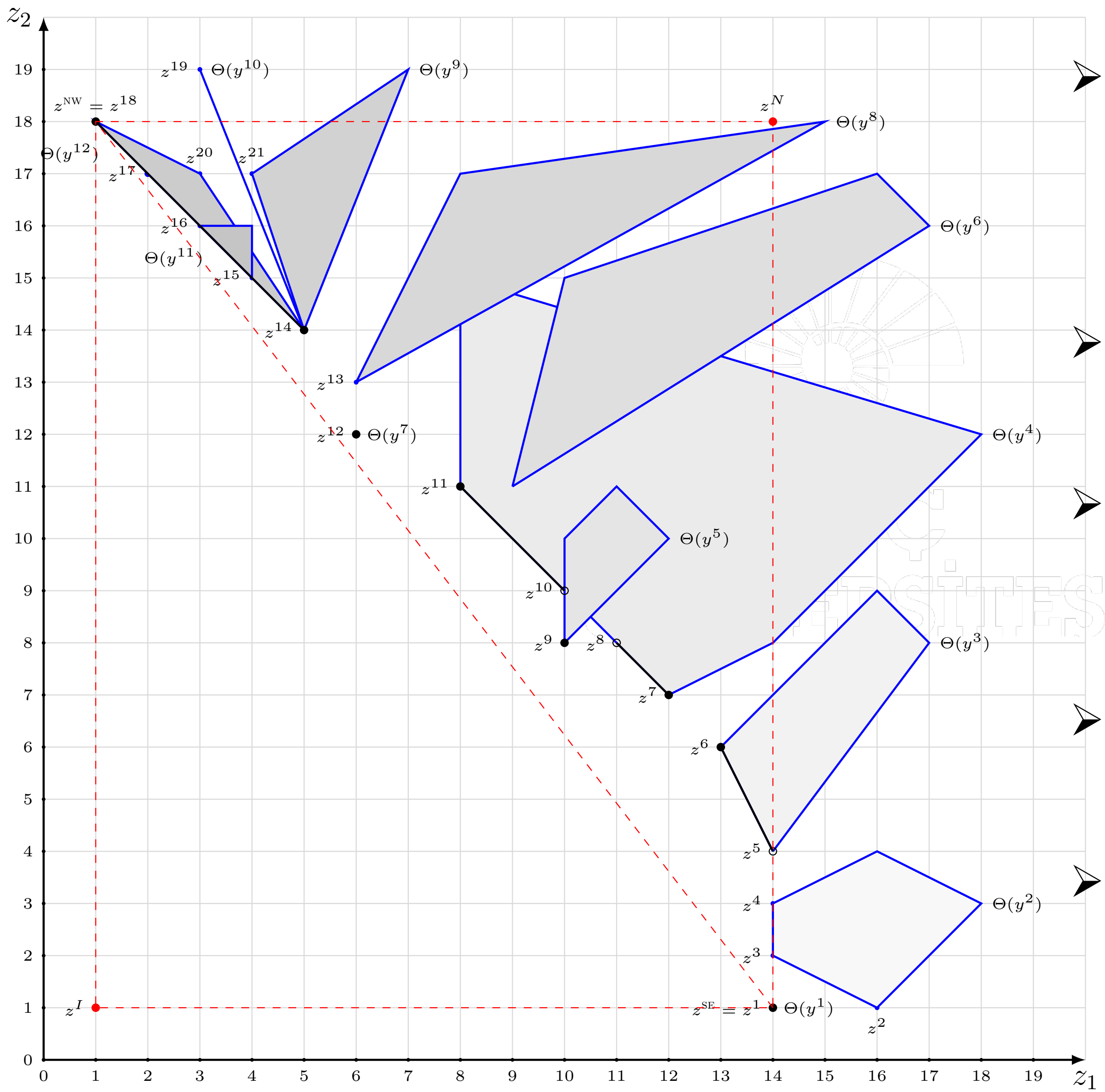
- CPU time comparison





Fattahi&Turkay, Computers & Chem. Eng. 72(2), 199-209 (2015).

ENPOBOMIP



- All possible complications with the Pareto set is represented in the figure -that we could think of
- Nonconvexity of the Pareto set is the main issue
- Incremental search algorithms fail to generate the Pareto set
- Finding the exact Pareto set is the main purpose
- Theoretical analysis to guarantee optimality

ENPOBOMIP

Algorithm 4 $\Phi^* \Leftarrow$ ENPOBOMIP(P)

$\chi^{CP} \Leftarrow LX(P^{SE}_{LX1}, P^{SE}_{LX2})$, Set $\Phi^* \leftarrow \{\chi^{CP}\}$, $Y^{EXC} \leftarrow \{\}$, $\xi \leftarrow$ TRUE
loop
  if $\phi\,(\Leftarrow LP^{NEP}(y^{CP})) = z_1^{CP}$ **then**
    Set $Y^{EXC} \leftarrow Y^{EXC} \cup \{y^{CP}\}$
    if $z_2^{NP} \Leftarrow P^{NP}_{LX1}(Y^{EXC}, z_1^{CP})$ is feasible **then**
      $\chi^{NP} \Leftarrow P^{NP}_{LX2}(z_2^{NP})$
      if $z_1^{NP} < z_1^{CP}$ **or** $z^{NP} = z^{CP}$ **then**
        Set $\Phi^* \leftarrow \Phi^* \cup \{\chi^{NP}\}$, $\xi \leftarrow$ TRUE , $\chi^{CP} \leftarrow \chi^{NP}$
      else Set $\chi^{CP} \leftarrow \chi^{NP}$, $\xi \leftarrow$ FALSE
      end if
    else RETURN Φ^*
    end if
  else
    $\chi^{NEP} \Leftarrow$ FINDNEP(χ^{CP}), Set $\tilde{Y}^{EXC} \leftarrow \{\}$, $\tilde{\Phi} \leftarrow \{\}$
    loop
      if $\breve{z}_2 \Leftarrow P^{NCC}_{LX1}(Y^{EXC} \cup \tilde{Y}^{EXC} \cup y^{CP}, z^{CP}, z^{NEP})$ is feasible **then**
        $\breve{\chi} \Leftarrow P^{NCC}_{LX2}(Y^{EXC} \cup \tilde{Y}^{EXC} \cup y^{CP}, \breve{z}_2)$
        if $\breve{z}_1 < z_1^{CP}$ **then**
          Update $\tilde{\Phi}$, Set $\Phi^* \leftarrow \Phi^* \cup (\chi^{CP}, \bar{\chi}) \cup \{\breve{\chi}\} \cup \tilde{\Phi}$, $\chi^{CP} \leftarrow \breve{\chi}$, $\xi \leftarrow$ TRUE, **break loop**
        else
          if $\xi =$ TRUE **then** Set $\Phi^* \leftarrow \Phi^* \cup \{\breve{\chi}\}$
          end if
          if $\phi\,(\Leftarrow LP^{NEP}(\breve{y})) < z_1^{CP}$ **then**
            $\bar{\chi} \Leftarrow$ FINDNEP$(\breve{\chi})$
            if $\frac{\bar{z}_2 - z_2^{CP}}{\bar{z}_1 - z_1^{CP}} > \frac{z_2^{NEP} - z_2^{CP}}{z_1^{NEP} - z_1^{CP}}$ **then**
              Set $\tilde{Y}^{EXC} \leftarrow \tilde{Y}^{EXC} \cup \{y^{CP}\}$, $\chi^{CP} \leftarrow \breve{\chi}$, $\chi^{NEP} \leftarrow \bar{\chi}$, $\tilde{\Phi} \leftarrow \{\}$
            else if $\frac{\bar{z}_2 - z_2^{CP}}{\bar{z}_1 - z_1^{CP}} < \frac{z_2^{NEP} - z_2^{CP}}{z_1^{NEP} - z_1^{CP}}$ **then** Set $\tilde{Y}^{EXC} \leftarrow \tilde{Y}^{EXC} \cup \{\breve{y}\}$
            else if $z_1^{NEP} \geq \bar{z}_1$ **then** Set $\tilde{Y}^{EXC} \leftarrow \tilde{Y}^{EXC} \cup \{\breve{y}\}$, $\tilde{\Phi} \leftarrow \tilde{\Phi} \cup (\breve{\chi}, \bar{\chi})$
            else Set $\tilde{Y}^{EXC} \leftarrow \tilde{Y}^{EXC} \cup \{y^{CP}\}$, update $\tilde{\Phi}$, Set $\tilde{\Phi} \leftarrow \tilde{\Phi} \cup (\chi^{CP}, \bar{\chi}^{NEP})$, $\chi^{CP} \leftarrow \breve{\chi}$, $\chi^{NEP} \leftarrow \bar{\chi}$,
            end if
          else Set $Y^{EXC} \leftarrow Y^{EXC} \cup \{\breve{y}\}$
          end if
        end if
      else
        Set $\Phi^* \leftarrow \Phi^* \cup (\chi^{CP}, \chi^{NEP}] \cup \tilde{\Phi}$, $\chi^{CP} \leftarrow \chi^{NEP}$, $\xi \leftarrow$ TRUE, **break loop**
      end if
    end loop
  end if
end loop

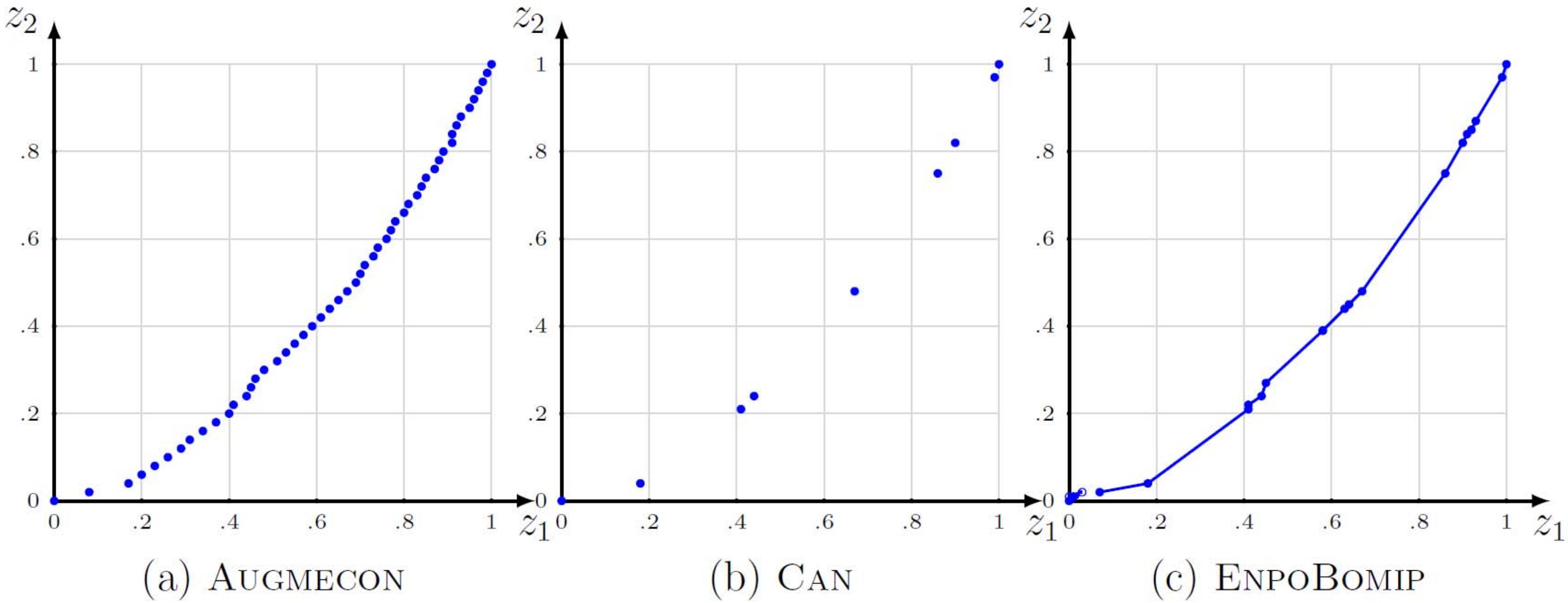
- All cases are embedded in the algorithm
- The exact Pareto set is guaranteed
- Implementation in GAMS

EXPERIMENTS

- Comparative analysis of AUGMECON, CAN and ENPOBOMIP
- AUGMECON: Q =10,000 and $\mu=10^{-4}$
- Three well-known problems:
  1. SCPN: Synthesis of Complex Production Networks: The benchmark problems are outlined in (Grossmann, Drabbant, Jain, 1982)
  2. FLP: Facility Location Problem: Instances are generated using the algorithm given by Stidsen et al., 2014, m=5, 10. 15, 20, 25, 30, 35
  3. FCNDP: Fixed Charge Network Design Problem: Instances are generated using the algorithm given by Stidsen et al., 2014, n=3, …,9

RESULTS: SCPN

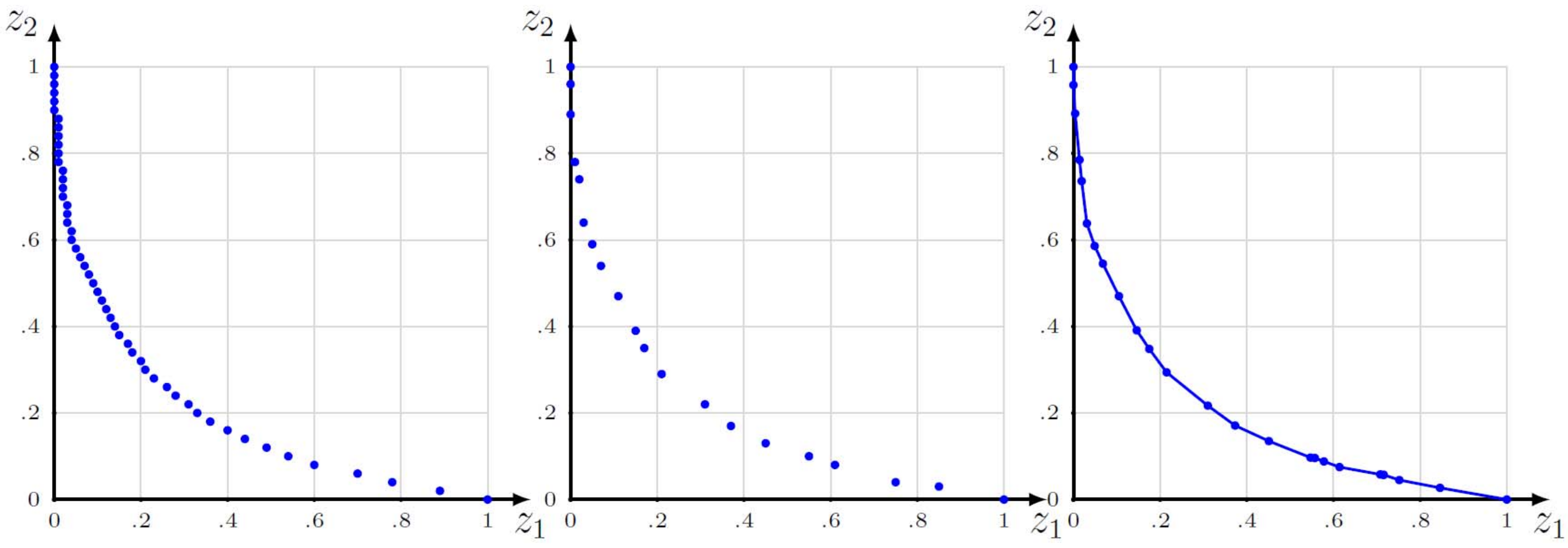
The resulting nondominated frontiers found for biobjective SCPN using the three algorithms



Interfaces: Ignacio shared the model and data that was published in 1982!

RESULTS: FLP

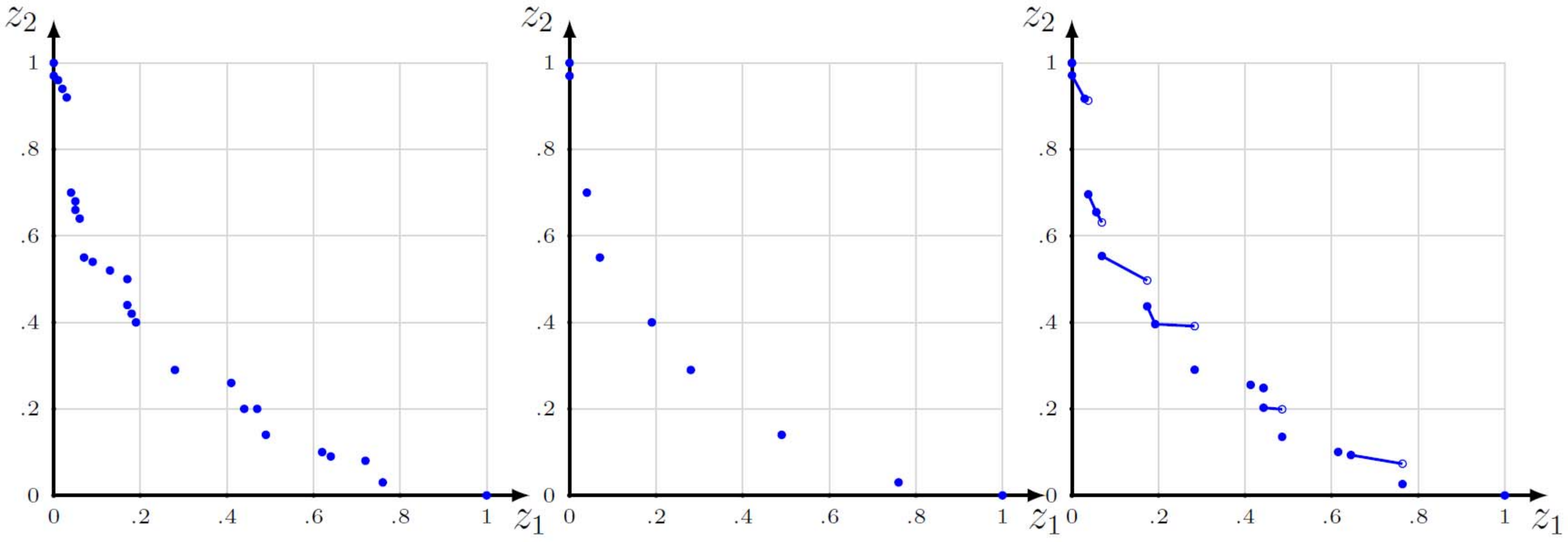
The resulting nondominated frontiers found for biobjective m=5 FLP using the three algorithms



(d) AUGMECON (e) CAN (f) ENPOBOMIP

RESULTS: FCNDP

The resulting nondominated frontiers found for biobjective n=5 FCNDP using the three algorithms



(g) AUGMECON (h) CAN (i) ENPOBOMIP

RESULTS: QUALITY

	AUG (NDP)	CAN (ESN)	ENP(NDLS,NDP)
SCPN	10,001	9	(22,0)
FLP			
m=5	10,001	20	(24,0)
m=10	10,001	47	(68,0)
m=15	9,885	89	(113,0)
m=20	9,962	121	(201,0)
m=25	10,001	176	(255,0)
m=30	9,854	207	(293,0)
m=35	9,766	273	(380,0)
FCNDP			
n=3	1,317	6	(4,4)
n=4	8	3	(0,8)
n=5	2,512	9	(11,6)
n=6	8,346	20	(69,7)
n=7	7,745	24	(94,0)
n=8	8,354	32	(169,0)
n=9	7,708	44	(144,0)
FCNDP($\times$.1)			
n=3	1	1	(0,1)
n=4	5	3	(0,5)
n=5	398	7	(5,10)
n=6	6,595	11	(29,11)
n=7	6,916	8	(31,4)
n=8	3,492	10	(47,6)
n=9	—*	14	(78,11)

AUG (NDP): number of nondominated points found by AUGMECON, CAN (ESN): number of ESN points found by CAN, ENP (NDLS,NDP): number of nondominated line segments and nondominated points found by ENPOBOMIP

$*$ the instance is not solved in 15,000 seconds

RESULTS: PERFORMANCE

	Augmecon		EnpoBomip				
	MILP	TT	MILP	T(MILP)	LP	T(LP)	TT
SCPN	10,003	1,483.76	45	7.15	114	12.45	19.66
FLP							
m=5	10,003	924.24	33	2.79	131	9.70	12.57
m=10	10,003	945.49	98	9.82	515	41.75	51.86
m=15	9,887	1,081.00	145	15.59	869	80.79	96.98
m=20	9,964	1,316.60	275	40.34	1,807	195.65	237.39
m=25	10,003	1,573.24	345	61.98	2,420	306.59	370.90
m=30	9,856	1,818.51	391	82.77	2,899	429.28	515.41
m=35	9,768	2,721.94	495	231.97	3,776	1,415.21	1,807.29
FCNDP							
n=3	1,319	225.67	21	2.24	16	1.61	3.86
n=4	10	2.02	17	2.20	8	0.87	3.08
n=5	2,514	359.36	40	5.94	39	4.09	10.06
n=6	8,348	1,396.80	189	40.23	338	40.31	80.90
n=7	7,747	2,031.47	212	68.32	494	65.19	134.18
n=8	8,356	4,153.38	405	231.72	1,289	192.91	426.86
n=9	7,710	3,925.49	305	228.00	992	182.99	413.25
FCNDP($\times$.1)							
n=3	2	0.74	3	0.35	1	0.11	0.46
n=4	7	1.32	11	1.36	5	0.48	1.84
n=5	400	66.18	32	4.98	27	2.56	7.57
n=6	6,597	2,066.65	98	44.74	144	16.15	61.05
n=7	6,918	4,697.94	82	71.37	135	19.21	90.78
n=8	3,494	12,848.55	130	495.32	195	27.62	523.35
n=9	—	—$*$	209	842.31	328	59.23	902.48

MILP: number of MILP resolutions, TT: total running time, T(MILP): total CPU time spent for MILP resolutions, LP: number of LP resolutions, T(LP): total CPU time spent for LP resolutions

$*$ the instance is not solved in 15,000 seconds

SUMMARY

- Generation of the Pareto set for bi-objective discrete-continuous optimization problems is challenging
- We have effective algorithms to generate the efficient set for bi-objective MINLP and MILP
- MILP
  - ✓ Exact Pareto set is guaranteed
  - ✓ Computationally superior than competing algorithms
  - ✓ Quality of solutions is much better
- MINLP
  - ✓ Approximation of the Pareto set
  - ✓ Network Synthesis Problems: the performance does not suffer from the second objective

ACKNOWLEDGEMENTS

- Ali Fattahi, MS Thesis
- IBM: SUR Award + Faculty Award
- TUBITAK: 104M322 Project
- European Commission: InTraRegio Project (Contract: 286975)

- Ignacio for being a mentor and inspiration always.

HAPPY 65th IGNACIO!