Mixed Integer Programming Models for Non-Separable Piecewise Linear Cost Functions

Juan Pablo Vielma Shabbir Ahmed George Nemhauser

H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology

INFORMS Annual Meeting 2008 – Washington, DC

Outline

- Introduction
- Modeling Piecewise Linear Functions
- Computational Results
- Conclusions

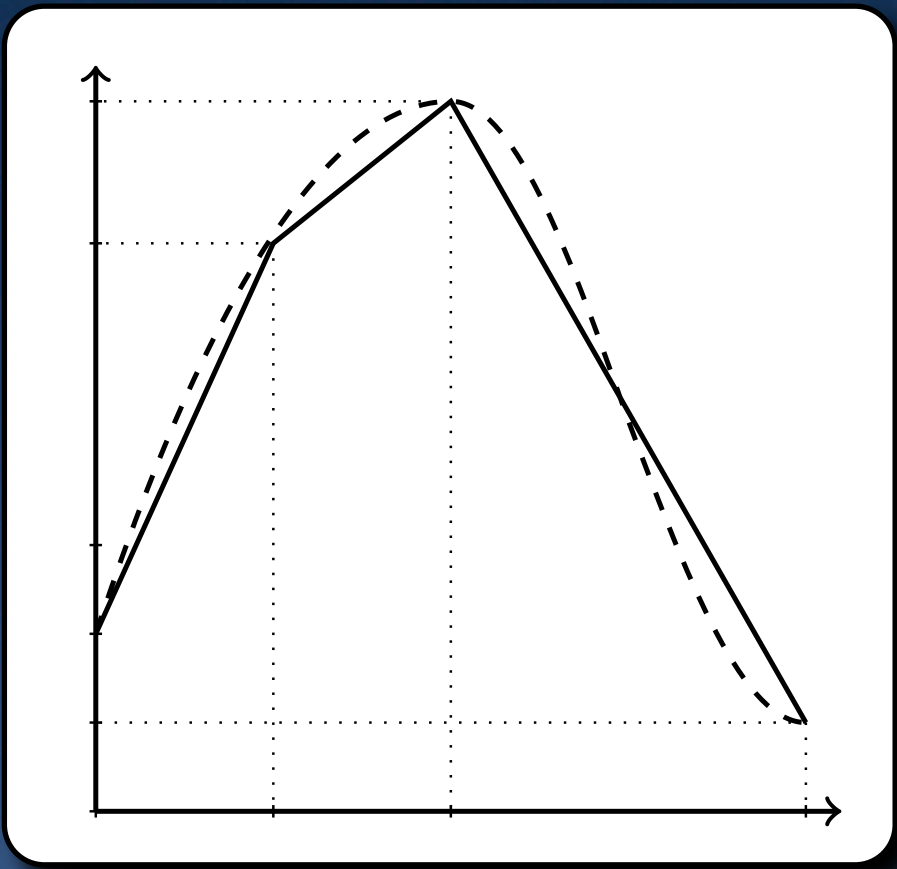
Piecewise Linear Optimization

$$
\begin{aligned}
\min \quad & f_0(x) \\
s.t. \quad & \\
& f_i(x) \leq 0 \qquad \forall i \in I \\
& x \in X \subset \mathbb{R}^n.
\end{aligned}
$$

- $f_i(x) : D \to \mathbb{R}$ is a piecewise linear function $\forall i \in \{0\} \cup I$.
- X is any compact set.

Piecewise Linear Functions (PLF)

- Approximate non-linearities, discounts for volume, etc.
- Many Applications.
- Convex = Linear Programming.
- Non-Convex = NP Hard.
- Specialized algorithms (Tomlin 1981, ..., de Farias et al. 2008) or Mixed Integer Programming Models (12+ papers)



Non-Separable = Multivariate

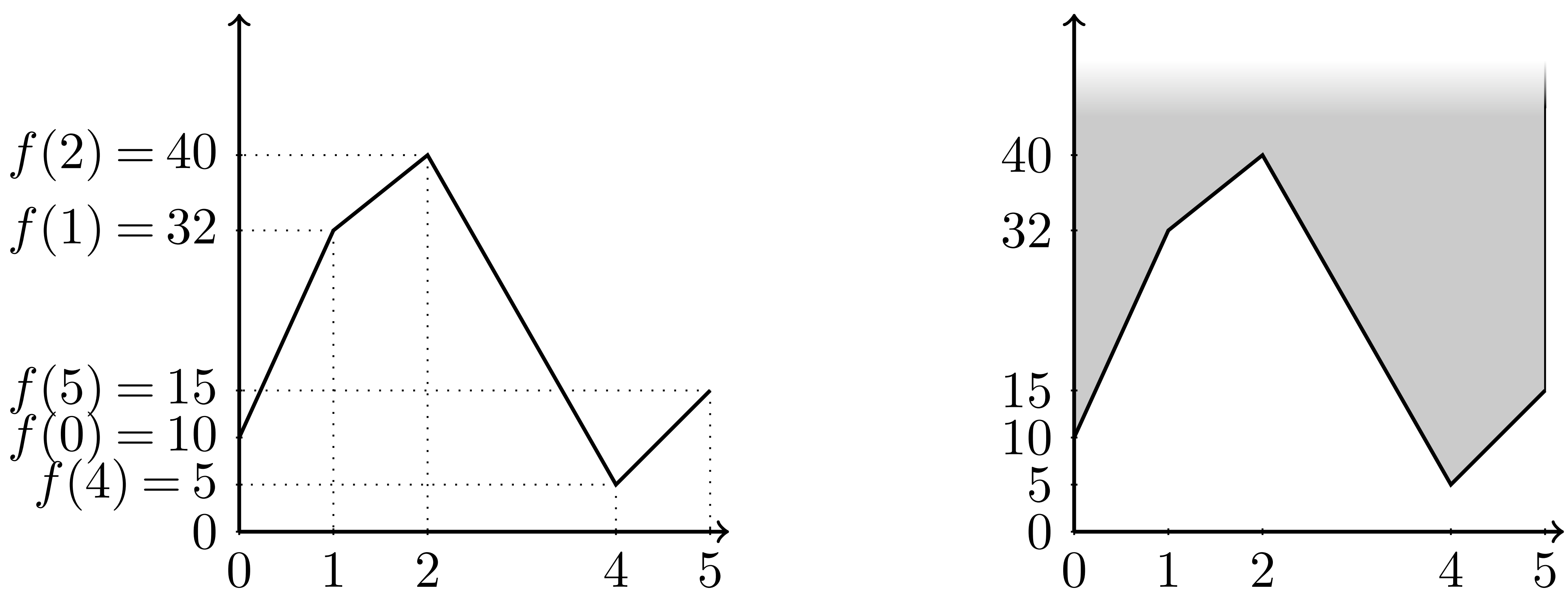
- Separable function:

$$f(x) = \sum_{j=1}^{n} f_j(x_j) \text{ for } f_j(x_j) : \mathbb{R} \rightarrow \mathbb{R}$$

- Functions can sometimes be separated:
  - Undesirable for numerical reasons and strength.
  - Not possible for interpolated functions.

Modeling Function = Epigraph

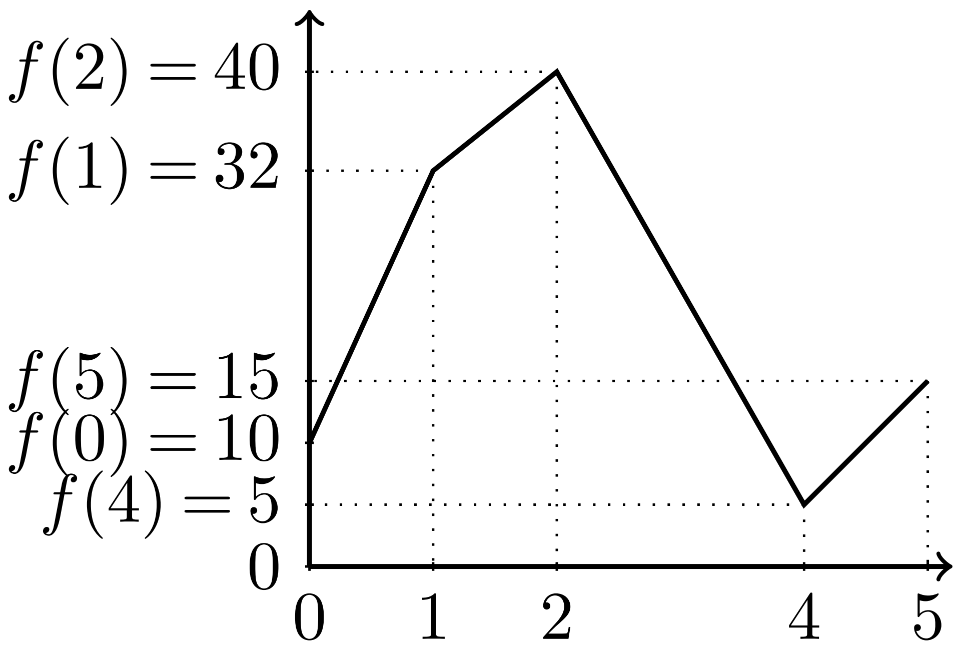
- $\text{epi}(f) := \{(x,z) \in D \times \mathbb{R} : f(x) \leq z\}.$



(a) f. (b) $\text{epi}(f)$.

- Example: $f(x) \leq 0 \Leftrightarrow (x,z) \in \text{epi}(f),\ z \leq 0$

Piecewise Linear Functions: Definition



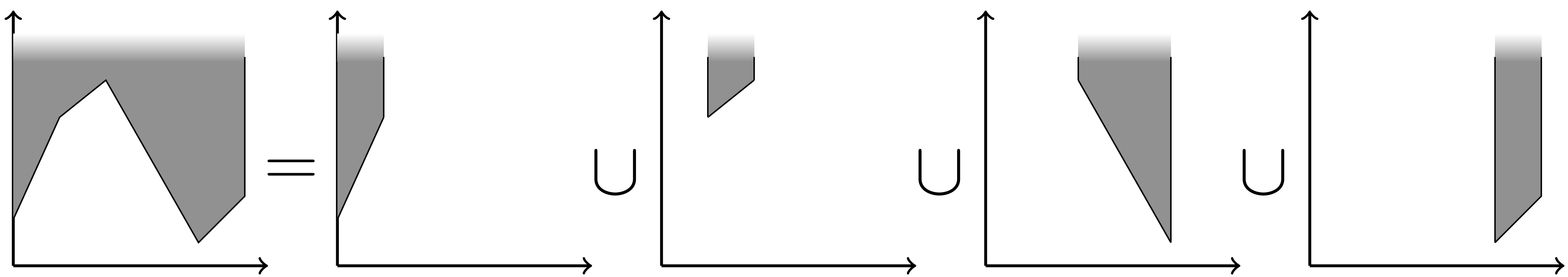
$$f(x) := \begin{cases} 22x + 10 & x \in [0,1] \\ 8x + 24 & x \in [1,2] \\ -17.5x + 75 & x \in [2,4] \\ 10x - 35 & x \in [4,5] \end{cases}$$

Definition 1. Piecewise Linear $f : D \subset \mathbb{R}^n \to \mathbb{R}$:

$$f(x) := \left\{ m_P x + c_P \quad x \in P \quad \forall P \in \mathcal{P}. \right.$$

for finite family of polytopes $\mathcal{P}$ such that $D = \bigcup_{P \in \mathcal{P}} P$

Epigraph of PLF is Union of Polyhedra

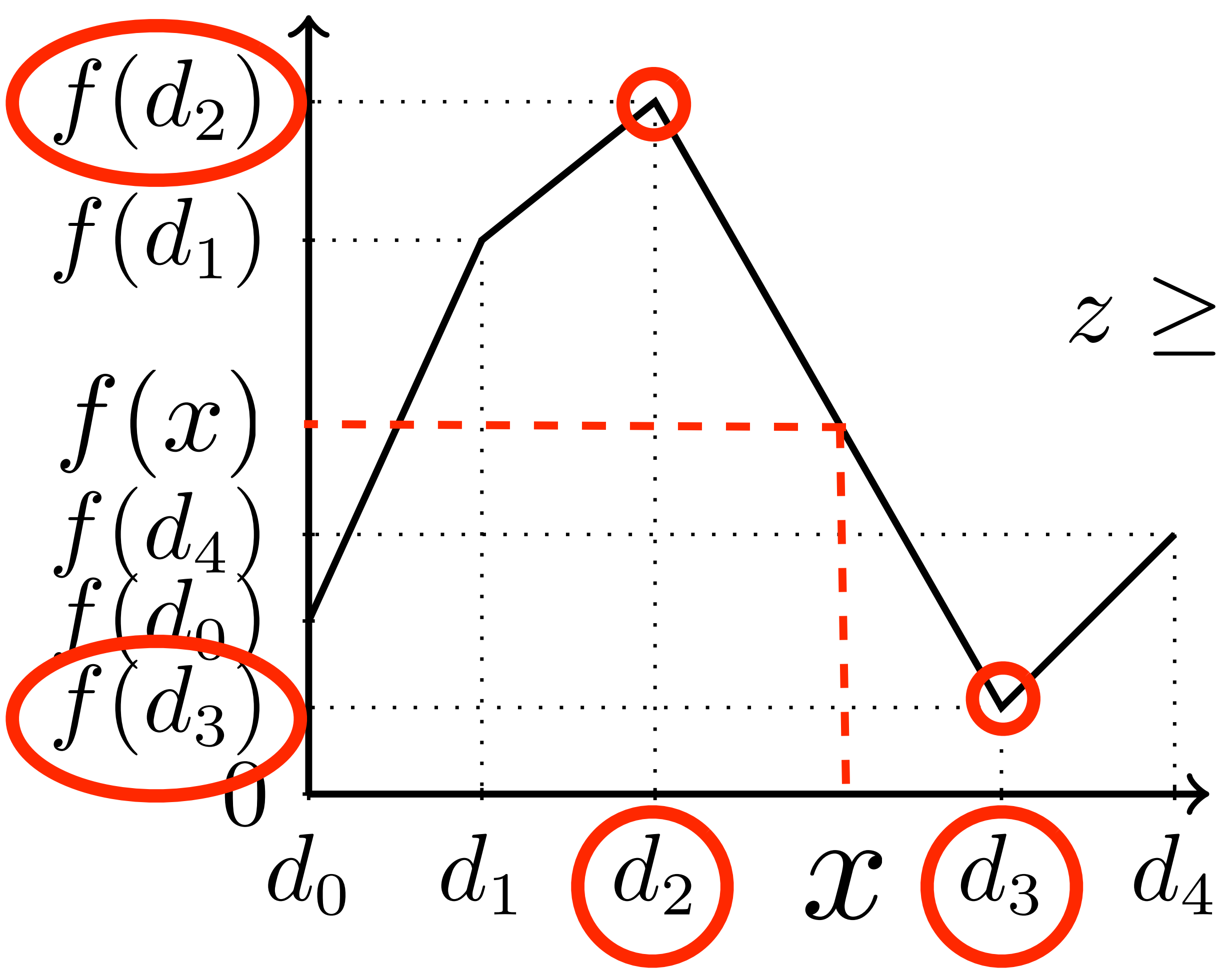


$$\operatorname{epi}(f) = C_n^+ + \bigcup_{P \in \mathcal{P}} \operatorname{conv}\left(\{(v, f(v))\}_{v \in V(P)}\right)$$

$$= C_n^+ + \bigcup_{P \in \mathcal{P}} \operatorname{conv}\left(\{(v, m_P v + c_P)\}_{v \in V(P)}\right)$$

$$C_n^+ := \{(0, z) \in \mathbb{R}^n \times \mathbb{R} \,:\, z \geq 0\}, \quad V(P) := \text{ vertices of } P.$$

Convex Combination Models



$$x = \lambda d_2 + (1 - \lambda) d_3$$

$$\Downarrow$$

$$z \geq f(x) = \lambda f(d_2) + (1 - \lambda) d_3$$

$$(x, z) \in \text{epi}(f)$$

Disaggregated Conv. Comb. (DCC)

$$\sum_{P\in\mathcal{P}}\sum_{v\in V(P)} \lambda_{P,v} v = x, \qquad \sum_{P\in\mathcal{P}}\sum_{v\in V(P)} \lambda_{P,v}\left(m_P v + c_P\right) \leq z$$

$$\lambda_{P,v} \geq 0 \quad \forall P\in\mathcal{P},\, v\in V(P), \qquad \sum_{v\in V(P)} \lambda_{P,v} = y_P \quad \forall P\in\mathcal{P}$$

$$\sum_{P\in\mathcal{P}} y_P = 1, \qquad y_P \in \{0,1\} \quad \forall P\in\mathcal{P}.$$

- Croxton et al. (2003a), Jeroslow (1987), Jeroslow and Lowe (1984), Lowe (1984), Meyer (1976) and Sherali (2001)

Logarithmic DCC (DLog)

$$\sum_{P\in\mathcal{P}}\sum_{v\in V(P)}\lambda_{P,v}v = x, \qquad \sum_{P\in\mathcal{P}}\sum_{v\in V(P)}\lambda_{P,v}\left(m_P v + c_P\right)\le z$$

$$\lambda_{P,v}\ge 0 \quad \forall P\in\mathcal{P},\, v\in V(P), \quad \sum_{P\in\mathcal{P}}\sum_{v\in V(P)}\lambda_{P,v} = 1$$

$$\sum_{P\in\mathcal{P}^{+}(B,l)}\sum_{v\in V(P)}\lambda_{P,v}\le y_l, \sum_{P\in\mathcal{P}^{0}(B,l)}\sum_{v\in V(P)}\lambda_{P,v}\le (1-y_l), y_l\in\{0,1\} \quad \forall l\in L(\mathcal{P})$$

where $B:\mathcal{P}\to\{0,1\}^{\lceil\log_2|\mathcal{P}|\rceil}$ is any injective function, $L(\mathcal{P}) := \{1,\ldots,\lceil\log_2|\mathcal{P}|\rceil\}$,

$\mathcal{P}^{+}(B,l) := \{P\in\mathcal{P} : B(P)_l = 1\}$ and $\mathcal{P}^{0}(B,l) := \{P\in\mathcal{P} : B(P)_l = 0\}$.

- New? Direct from ideas in Ibaraki (1976), Vielma and Nemhauser (2008)

Logarithmic DCC (DLog)

$$\sum_{P\in\mathcal{P}}\sum_{v\in V(P)}\lambda_{P,v}v = x, \qquad \sum_{P\in\mathcal{P}}\sum_{v\in V(P)}\lambda_{P,v}\left(m_P v + c_P\right) \le z$$

$$\lambda_{P,v}\ge 0 \quad \forall P\in\mathcal{P},\ v\in V(P), \quad \sum_{P\in\mathcal{P}}\sum_{v\in V(P)}\lambda_{P,v} = 1$$

$$\sum_{P\in\mathcal{P}^+(B,l)}\sum_{v\in V(P)}\lambda_{P,v}\le y_l, \quad \sum_{P\in\mathcal{P}^0(B,l)}\sum_{v\in V(P)}\lambda_{P,v}\le (1-y_l), y_l\in\{0,1\} \quad \forall l\in L(\mathcal{P})$$

where $B:\mathcal{P}\to\{0,1\}^{\lceil \log_2|\mathcal{P}|\rceil}$ is any injective function, $L(\mathcal{P}) := \{1,\ldots,\lceil \log_2|\mathcal{P}|\rceil\}$,

$\mathcal{P}^+(B,l) := \{P\in\mathcal{P} : B(P)_l = 1\}$ and $\mathcal{P}^0(B,l) := \{P\in\mathcal{P} : B(P)_l = 0\}$.

- New? Direct from ideas in Ibaraki (1976), Vielma and Nemhauser (2008)

Convex Combination (CC)

$$\sum_{v \in \mathcal{V}(\mathcal{P})} \lambda_v v = x, \qquad \sum_{v \in \mathcal{V}(\mathcal{P})} \lambda_v \left(m_P v + c_P \right) \leq z$$

$$\lambda_v \geq 0 \quad \forall v \in \mathcal{V}(\mathcal{P}), \qquad \sum_{v \in \mathcal{V}(\mathcal{P})} \lambda_v = 1$$

$$\lambda_v \leq \sum_{P \in \mathcal{P}(v)} y_P \quad \forall v \in \mathcal{V}(\mathcal{P}),$$

$$\sum_{P \in \mathcal{P}} y_P = 1, \qquad y_P \in \{0,1\} \quad \forall P \in \mathcal{P},$$

where $\mathcal{P}(v) := \{P \in \mathcal{P} : v \in P\}$.

- Dantzig (1963, 1960), Garfinkel and Nemhauser (1972), Jeroslow and Lowe (1985), Keha et al. (2004), Lee and Wilson (2001), Lowe (1984), Nemhauser and Wolsey (1988), Padberg (2000) and Wilson (1998)

Logarithmic Conv. Comb. (Log)

$$\sum_{v\in\mathcal{V}(\mathcal{P})} \lambda_v v = x, \qquad \sum_{v\in\mathcal{V}(\mathcal{P})} \lambda_v \left(m_P v + c_P\right) \le z$$

$$\lambda_v \ge 0 \quad \forall v \in \mathcal{V}(\mathcal{P}), \qquad \sum_{v\in\mathcal{V}(\mathcal{P})} \lambda_v = 1$$

$$\sum_{v\in L_s} \lambda_v \le y_s, \qquad \sum_{v\in R_s} \lambda_v \le (1-y_s), \qquad y_s \in \{0,1\} \quad \forall s \in \mathcal{S}.$$

- Requires Independent Branching Scheme.
- Vielma and Nemhauser (2008).

Logarithmic Conv. Comb. (Log)

$$\sum_{v\in\mathcal{V}(\mathcal{P})} \lambda_v v = x, \qquad \sum_{v\in\mathcal{V}(\mathcal{P})} \lambda_v \left(m_P v + c_P\right) \leq z$$

$$\lambda_v \geq 0 \quad \forall v \in \mathcal{V}(\mathcal{P}), \qquad \sum_{v\in\mathcal{V}(\mathcal{P})} \lambda_v = 1$$

$$\sum_{v\in L_s} \lambda_v \leq y_s, \qquad \sum_{v\in R_s} \lambda_v \leq (1-y_s), \qquad y_s \in \{0,1\} \quad \forall s \in \mathcal{S}.$$

- Requires Independent Branching Scheme.
- Vielma and Nemhauser (2008).

Multiple Choice (MC)

$$\sum_{P\in\mathcal{P}} x^P = x, \qquad \sum_{P\in\mathcal{P}} \left(m_P x^P + c_P y_P\right) \le z$$

$$A_P x^P \le y_P b_P \quad \forall P \in \mathcal{P}$$

$$\sum_{P\in\mathcal{P}} y_P = 1, \qquad y_P \in \{0,1\} \quad \forall P \in \mathcal{P},$$

where $A_P x \le b_P$ is the set of linear inequalities describing P.

- Balakrishnan and Graves (1989), Croxton et al. (2003a), Jeroslow and Lowe (1984) and Lowe (1984)

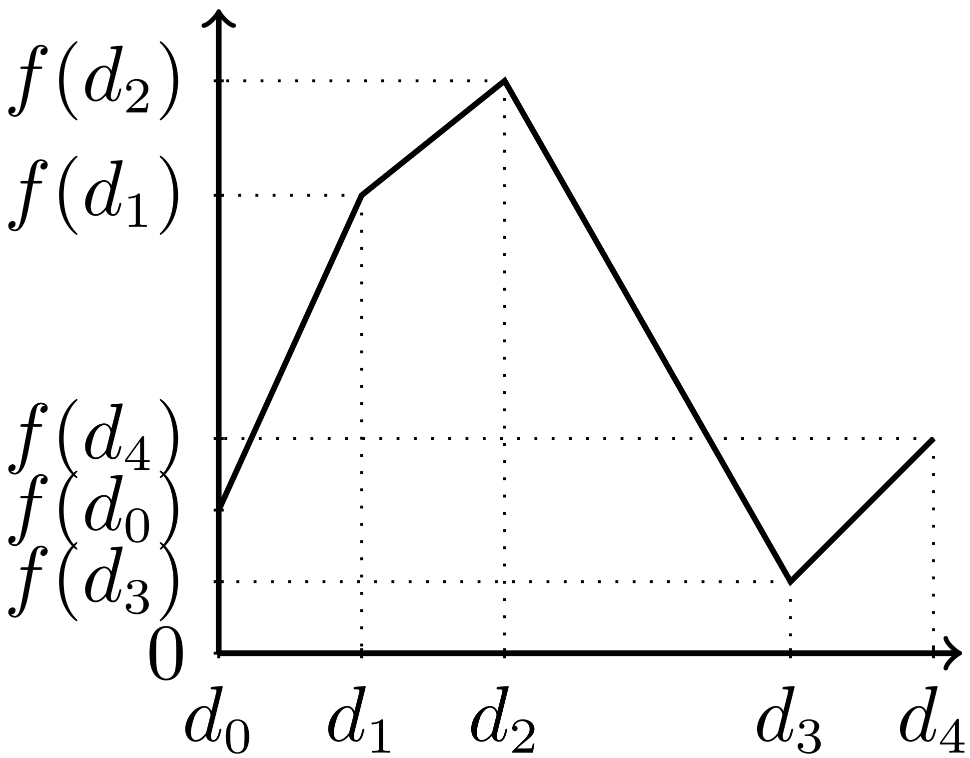
Incremental or Delta (Inc)

$$d_0 + \sum_{k=1}^{K} \delta_k \left(d_k - d_{k-1}\right) = x$$

$$f(d_0) + \sum_{k=1}^{K} \delta_k \left(f(d_k) - f(d_{k-1})\right) \leq z$$

$$\delta_1 \leq 1, \quad \delta_K \geq 0, \quad \delta_{k+1} \leq y_k \leq \delta_k,$$

$$y_k \in \{0,1\} \quad \forall k \in \{1,\ldots,K-1\}.$$



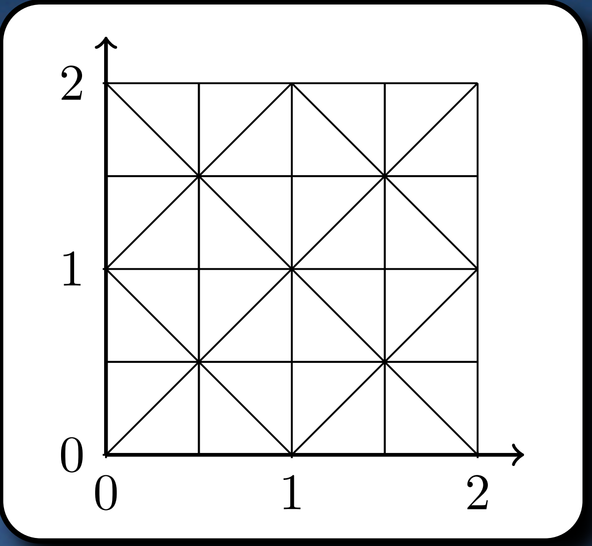
- Similar for multivariate functions.
- Croxton et al. (2003a), Dantzig (1963, 1960), Keha et al. (2004), Markowitz and Manne (1957), Padberg (2000), Sherali (2001), Vajda (1964) and Wilson (1998).

Strength of the Models

- All models give the same LP relaxation bound:
  - LP relaxation is model of lower convex envelope (Sharp).
- In the absence of other constraints:
  - All models except for CC have integral vertices (Locally Ideal).

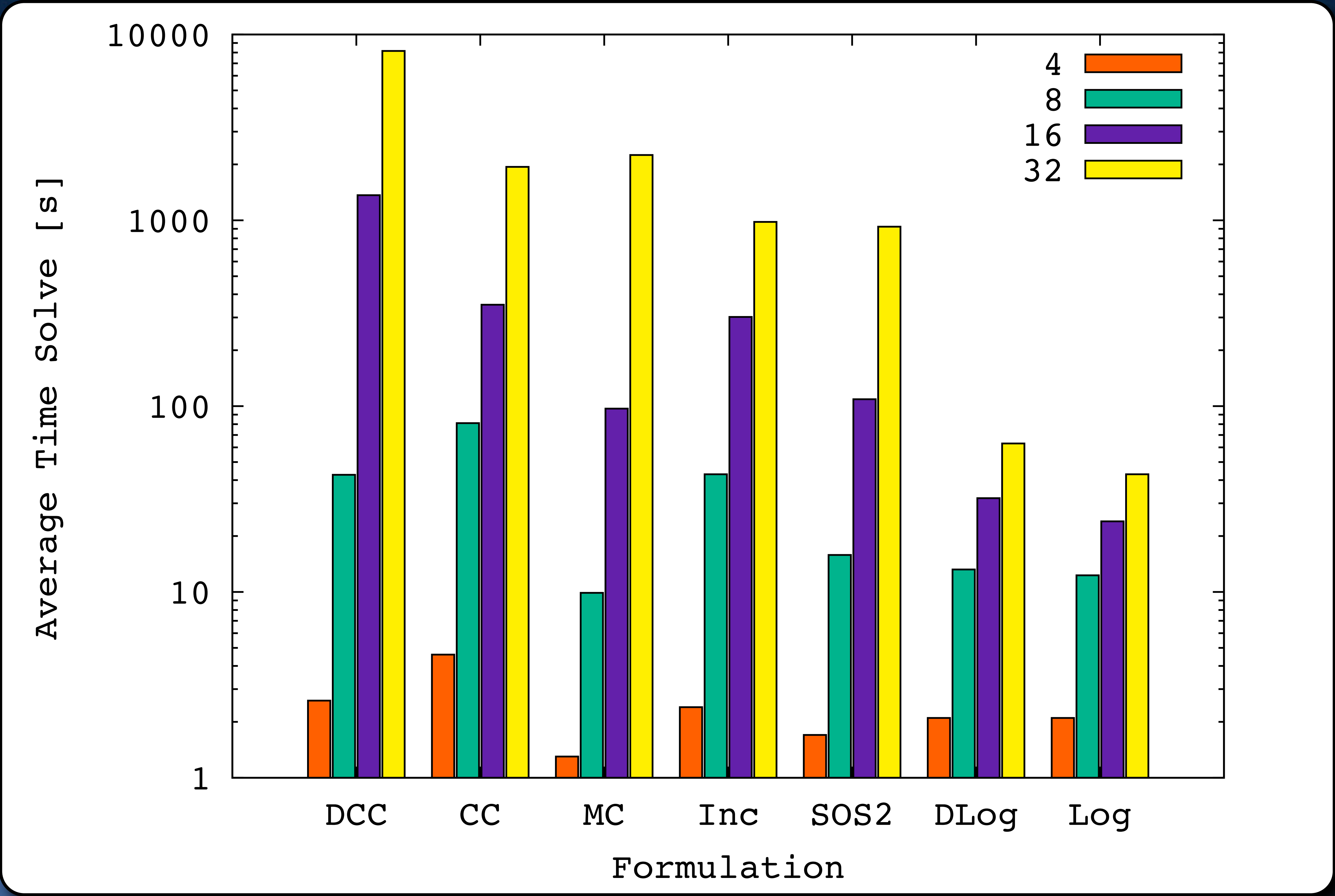
Instances and Solvers

- Instances
  - Transportation problems (10x10 & 5x2).
  - Univariate: Concave Separable Objective.
  - Multivariate: Multi-commodity function.
  - Functions are affine in k segments or in a k x k grid triangulation (100 instances per each k=4, 8, 16, 32).

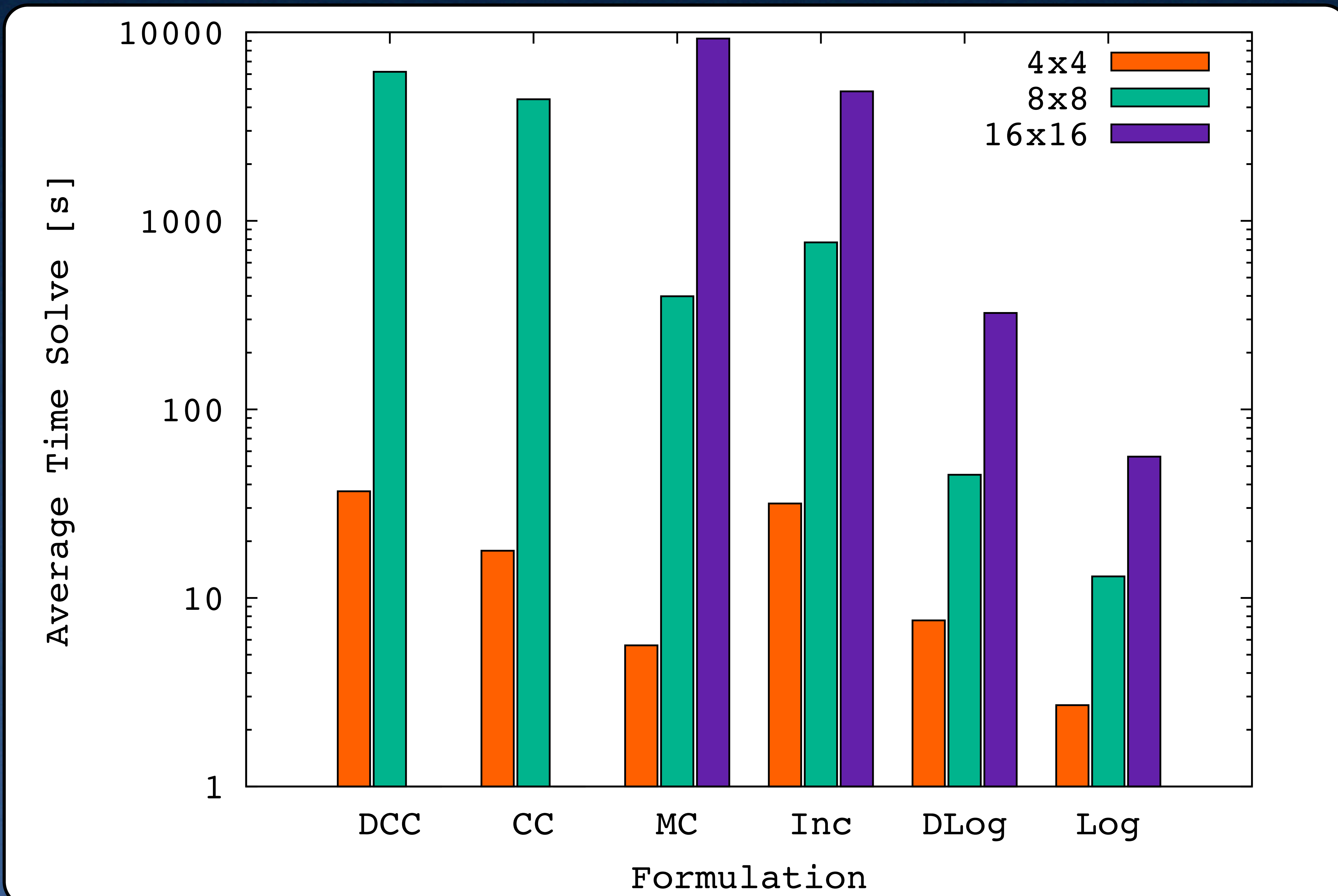


- Solver: CPLEX 11 on 2.4Ghz machine.

Univariate Case (Separable)



Multivariate Case (Non-Separable)



Conclusions and Other Results

- Suggestions
  - For small k use MC or Inc instead of DCC or CC.
  - For large k use DLog or Log.
- DLog, DCC and MC can also be also used for Lower Semicontinuous Functions.