Comparison of Bundle and Classical Column Generation

Olivier Briant **Philippe Meurdesoif** Nancy Perrot

Claude Lemaréchal **Sophie Michel** François Vanderbeck

Funded by Inria New Investigation Grant

INRIA - Rhônes-Alpes *Université Bordeaux 1* *Université de Genève*

"Convex Optimization and Dantzig-Wolfe Decomposition"

`http://www.math.u-bordeaux.fr/MAB/ODW/`

Solve the Dantzig-Wolfe reformulation of an IP

Solve the Dantzig-Wolfe reformulation of an IP

Original P: $$\min\{c\,x:\ A\,x\ \geq b, x \in X = \{x^i\}_i\}$$

Master IP: $$\min\{\sum_i c\,x^i\,\lambda_i : \sum_i A\,x^i\,\lambda_i \geq b,\ \sum_i \lambda_i = 1, \lambda_i \in \{0,1\}\ \forall i\}$$

Solve the Dantzig-Wolfe reformulation of an IP

Original P: $$\min\{c\,x : \; A\,x \;\geq b, x \in X = \{x^i\}_i\}$$

Master IP: $$\min\{\sum_i c\,x^i\,\lambda_i : \sum_i A\,x^i\,\lambda_i \geq b,\; \sum_i \lambda_i = 1, \lambda_i \in \{0,1\}\;\forall i\}$$

Relaxed P: $$\min\{c\,x : \; A\,x \;\geq b, x \in conv(X)\}$$

Master LP: $$\min\{\sum_i c\,x^i\,\lambda_i : \sum_i A\,x^i\,\lambda_i \geq b,\; \sum_i \lambda_i = 1, \lambda_i \geq 0\;\forall i\}$$

Solve the Dantzig-Wolfe reformulation of an IP

Original P: $\min\{c\,x : \; A\,x \;\geq b, x \in X = \{x^i\}_i\}$

Master IP: $\min\{\sum_i c\,x^i\,\lambda_i : \sum_i A\,x^i\,\lambda_i \geq b,\; \sum_i \lambda_i = 1, \lambda_i \in \{0,1\}\;\forall i\}$

Relaxed P: $\min\{c\,x : \; A\,x \;\geq b, x \in conv(X)\}$

Master LP: $\min\{\sum_i c\,x^i\,\lambda_i : \sum_i A\,x^i\,\lambda_i \underbrace{\geq}_{u} b,\; \sum_i \lambda_i \underbrace{=}_{r} 1, \lambda_i \geq 0\;\forall i\}$

Dual Master: $\max\{\underbrace{b\,u - r}_{\eta} : uA\,x^i - r \leq c\,x^i\;\forall i,\; u \geq 0\}$

$$\max\{\eta : \eta \leq \underbrace{c\,x^i + u\,(b - A\,x^i)}_{L(x^i, u)}\,\forall i,\; u \geq 0\}$$

Lagrangian D: $\max\{\theta(u) : \; u \geq 0\}$ with $\theta(u) := \min_i L(x^i, u)$

Standard column generation: primal view

Master Problem

$$\begin{aligned}
\text{min} \quad & \sum_{i \in I} (cx^i)\lambda_i \\
\text{s.t.} \quad & \sum_{i \in I} (Ax^i)\lambda_i \geq b \\
& \sum_{i \in I} \lambda_i = 1 \\
& \lambda_i \geq 0
\end{aligned}$$

Standard column generation: primal view

Master Problem

$$\begin{aligned} \text{min} \quad & \sum_{i \in I} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I} (Ax^i)\lambda_i \geq b \\ & \sum_{i \in I} \lambda_i = 1 \\ & \lambda_i \geq 0 \end{aligned}$$

Restricted Master Problem (iteration k): $\boxed{I^k \subset I}$

$$\begin{aligned} \text{min} \quad & \sum_{i \in I^k} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I^k} (Ax^i)\lambda_i \geq b \\ & \sum_{i \in I^k} \lambda_i = 1 \\ & \lambda_i \geq 0 \end{aligned}$$

Standard column generation: primal view

Master Problem

$$\begin{aligned}\min \quad & \sum_{i\in I}(cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i\in I}(Ax^i)\lambda_i \geq b \\ & \sum_{i\in I}\lambda_i = 1 \\ & \lambda_i \geq 0\end{aligned}$$

Restricted Master Problem
(iteration k): $\boxed{I^k \subset I}$

$$\begin{aligned}\min \quad & \sum_{i\in I^k}(cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i\in I^k}(Ax^i)\lambda_i \geq b \rightarrow u^k \\ & \sum_{i\in I^k}\lambda_i = 1 \rightarrow r^k \\ & \lambda_i \geq 0\end{aligned}$$

Standard column generation: primal view

Master Problem

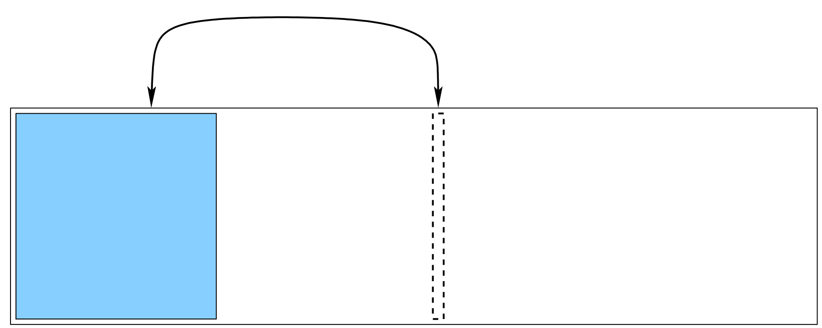
$$\begin{aligned} \min \quad & \sum_{i \in I} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I} (Ax^i)\lambda_i \geq b \\ & \sum_{i \in I} \lambda_i = 1 \\ & \lambda_i \geq 0 \end{aligned}$$

Restricted Master Problem

(iteration k): $\boxed{I^k \subset I}$

$$\begin{aligned} \min \quad & \sum_{i \in I^k} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I^k} (Ax^i)\lambda_i \geq b \rightarrow u^k \\ & \sum_{i \in I^k} \lambda_i = 1 \rightarrow r^k \\ & \lambda_i \geq 0 \end{aligned}$$

Subproblem (Oracle) : $\min_{i \in I} (c - u^k A)x^i = \min_{x \in X} (c - u^k A)x \ \rightarrow \ x^{k+1}$



Standard column generation: primal view

Master Problem

$$\begin{aligned} \text{min} \quad & \sum_{i \in I} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I} (Ax^i)\lambda_i \geq b \\ & \sum_{i \in I} \lambda_i = 1 \\ & \lambda_i \geq 0 \end{aligned}$$

Restricted Master Problem (iteration k): $\boxed{I^k \subset I}$

$$\begin{aligned} \text{min} \quad & \sum_{i \in I^k} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I^k} (Ax^i)\lambda_i \geq b \rightarrow u^k \\ & \sum_{i \in I^k} \lambda_i = 1 \rightarrow r^k \\ & \lambda_i \geq 0 \end{aligned}$$

Subproblem (Oracle) : $\min_{i \in I} (c - u^k A)x^i = \min_{x \in X} (c - u^k A)x$

If $((c - u^k A)x^{k+1} - r^k) < 0$, add $k+1$ to I^k. **Otherwise,** STOP.

Standard column generation: primal view

Master Problem

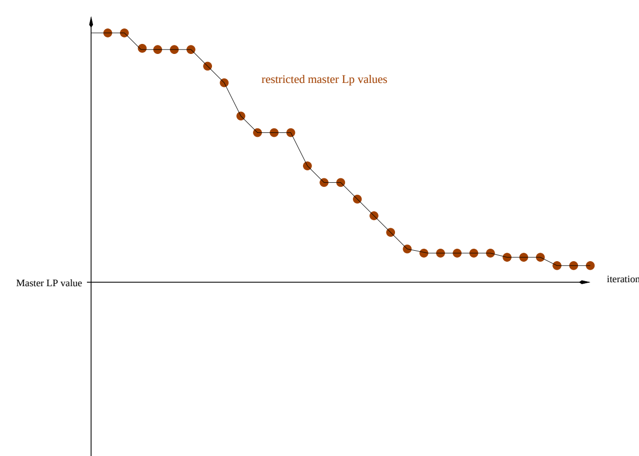
$$\begin{aligned} \min \quad & \sum_{i \in I} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I} (Ax^i)\lambda_i \geq b \\ & \sum_{i \in I} \lambda_i = 1 \\ & \lambda_i \geq 0 \end{aligned}$$

Restricted Master Problem (iteration k): $\boxed{I^k \subset I}$

$$\begin{aligned} \min \quad & \sum_{i \in I^k} (cx^i)\lambda_i \\ \text{s.t.} \quad & \sum_{i \in I^k} (Ax^i)\lambda_i \geq b \to u^k \\ & \sum_{i \in I^k} \lambda_i = 1 \to r^k \\ & \lambda_i \geq 0 \end{aligned}$$

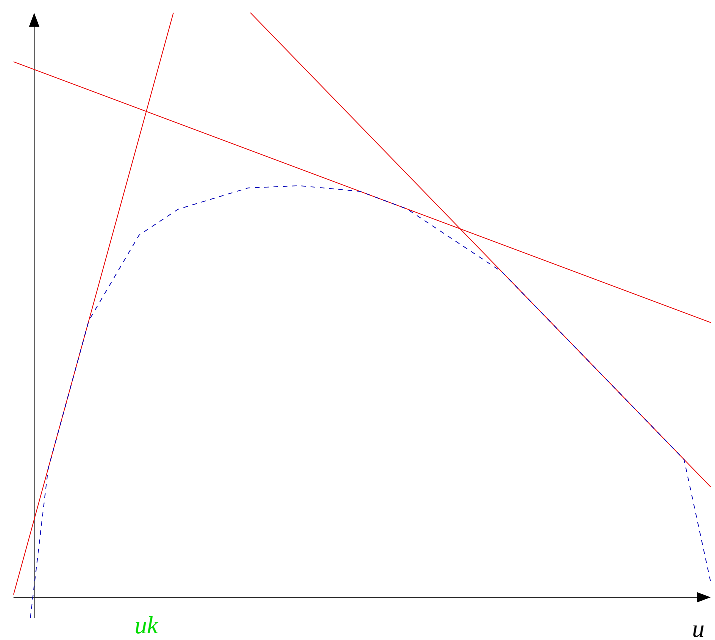
Subproblem (Oracle) : $\min_{i \in I} (c - u^k A)x^i = \min_{x \in X} (c - u^k A)x$

If $((c - u^k A)x^{k+1} - r^k) < 0$, add $k+1$ to I^k. **Otherwise**, STOP.



Standard column generation: dual view

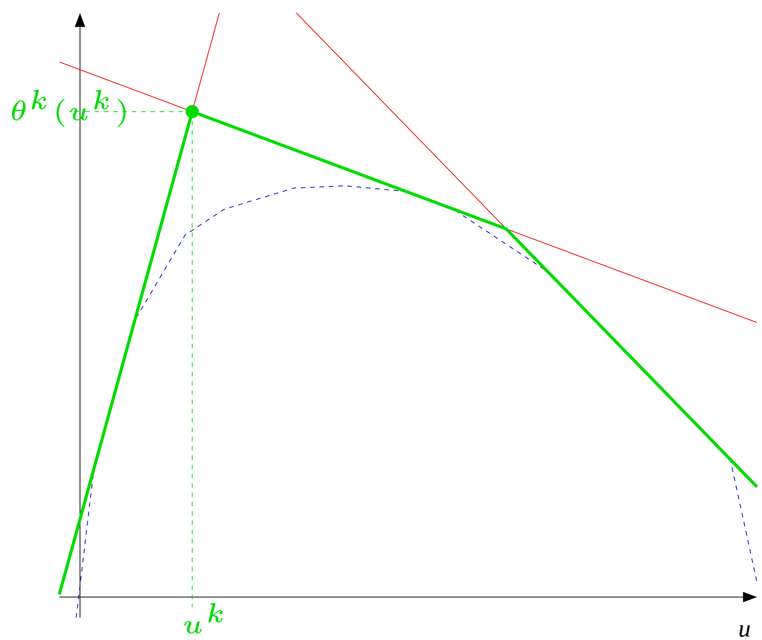
Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)} \,\forall i \in I^k,\ u \geq 0\}$



Standard column generation: dual view

Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)} \forall i \in I^k,\ u \geq 0\} \rightarrow (u^k)$

Approxim Dual Function : $\theta^k(u) = ub + \min_{i \in I^k}(c - uA)x^i$

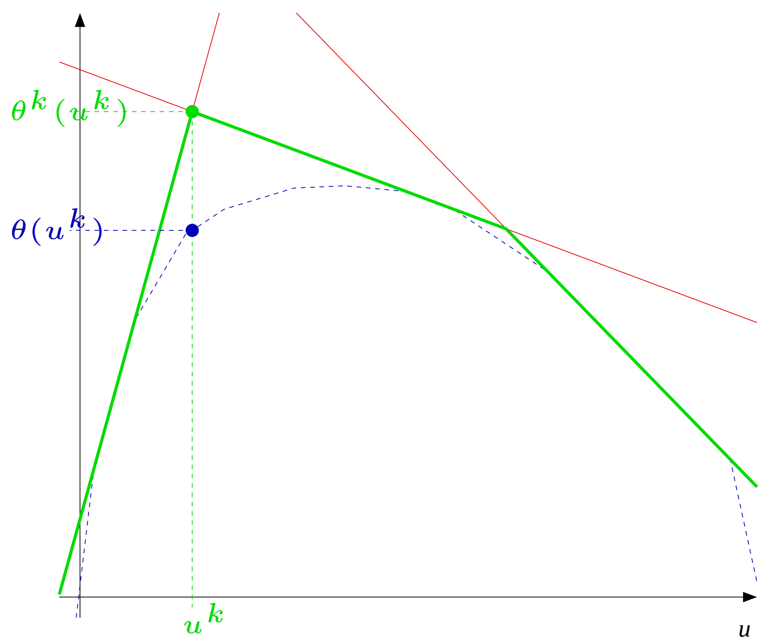


Standard column generation: dual view

Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)} \,\forall i \in I^k,\ u \geq 0\} \ \rightarrow\ (u^k)$

Approxim Dual Function : $\theta^k(u) = ub + \min_{i \in I^k} (c - uA)x^i$

Subproblem (Oracle) : $\theta(u^k) = u^k b + \min_{x \in X} (c - u^k A)x$

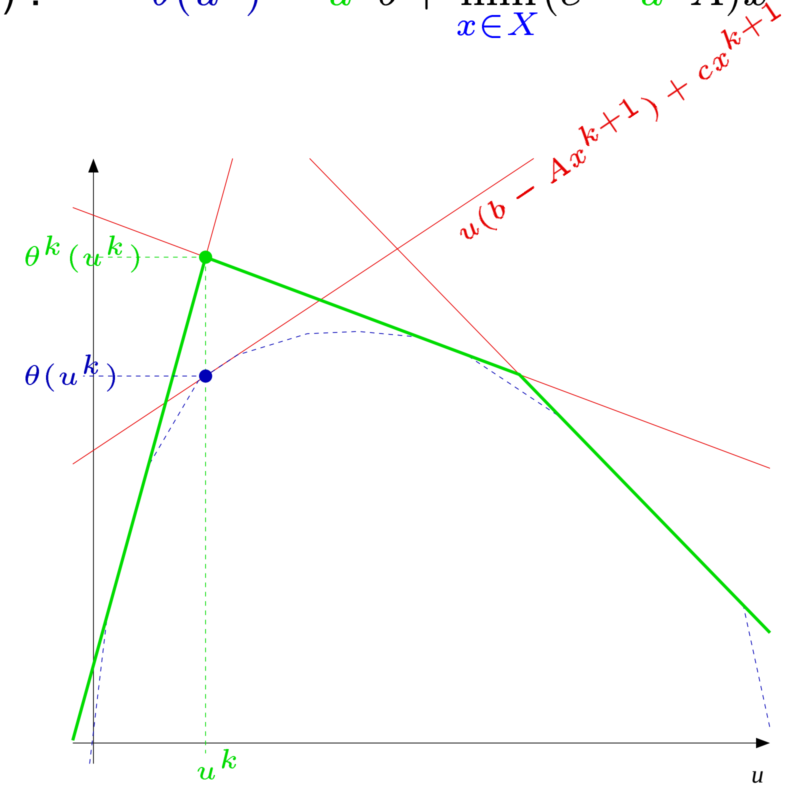


Standard column generation: dual view

Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)}\ \forall i \in I^k,\ u \geq 0\} \rightarrow (u^k)$

Approxim Dual Function : $\theta^k(u) = ub + \min_{i \in I^k}(c - uA)x^i$

Subproblem (Oracle) : $\theta(u^k) = u^k b + \min_{x \in X}(c - u^k A)x \rightarrow (x^{k+1})$

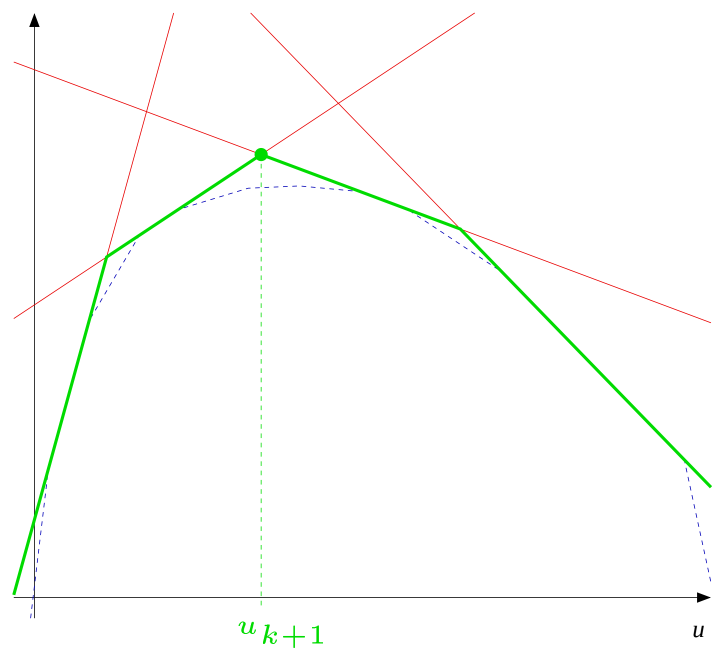


Standard column generation: dual view

Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)} \,\forall i \in I^k,\ u \geq 0\} \;\rightarrow\; (u^k)$

Approxim Dual Function : $\theta^k(u) = ub + \min_{i \in I^k}(c - uA)x^i$

Subproblem (Oracle) : $\theta(u^k) = u^k b + \min_{x \in X}(c - u^k A)x \;\rightarrow\; (x^{k+1})$

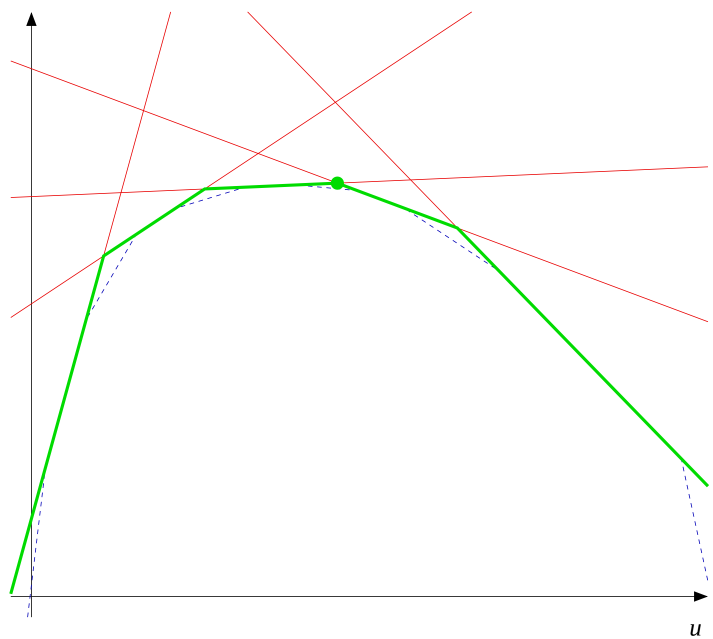


Standard column generation: dual view

Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)}\, \forall i \in I^k,\ u \geq 0\} \rightarrow (u^k)$

Approxim Dual Function : $\theta^k(u) = ub + \min_{i \in I^k} (c - uA)x^i$

Subproblem (Oracle) : $\theta(u^k) = u^k b + \min_{x \in X} (c - u^k A)x \rightarrow (x^{k+1})$

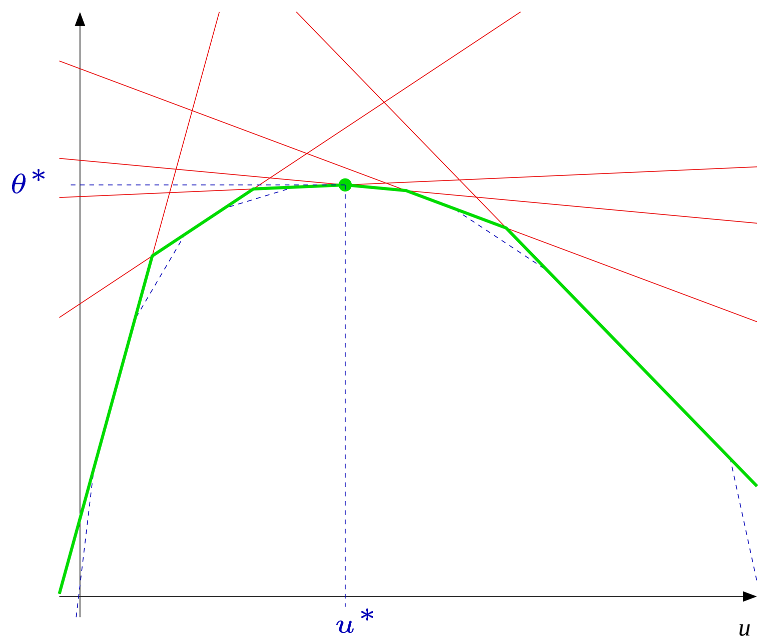


Standard column generation: dual view

Restricted Master Dual: $\max\{\eta : \eta \leq \underbrace{cx^i + u(b - A\,x^i)}_{L(x^i,u)}\ \forall i \in I^k,\ u \geq 0\} \rightarrow (u^k)$

Approxim Dual Function : $\theta^k(u) = ub + \min_{i \in I^k}(c - uA)x^i$

Subproblem (Oracle) : $\theta(u^k) = u^k b + \min_{x \in X}(c - u^k A)x \rightarrow (x^{k+1})$



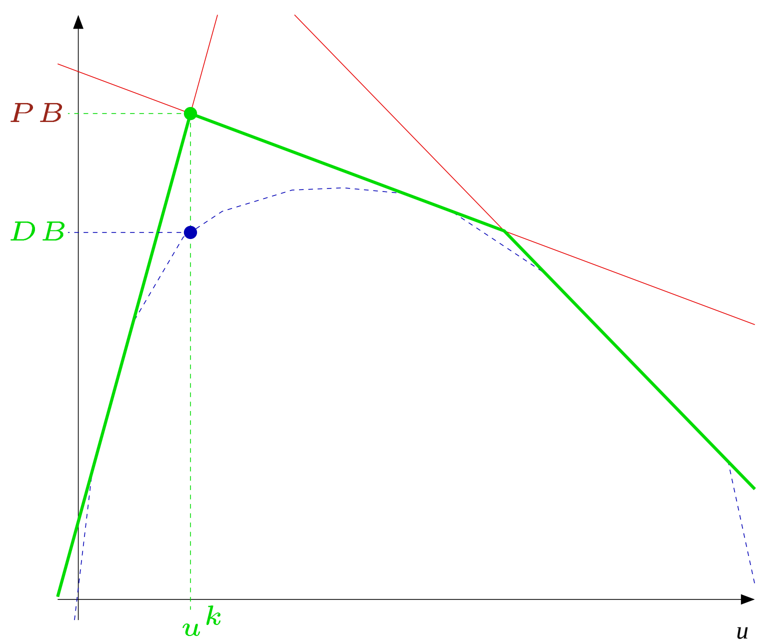
Kelley's cutting plane algorithm

Standard column generation: primal/ dual view

At iteration k, one has

A Primal Bound on the master: $PB \equiv \theta^k(u^k) = \sum_{i \in I^k} cx^i \lambda_i$

A Dual Bound on the master: $DB \equiv \theta(u^k) = PB - (c - u^k A)x^{k+1}$

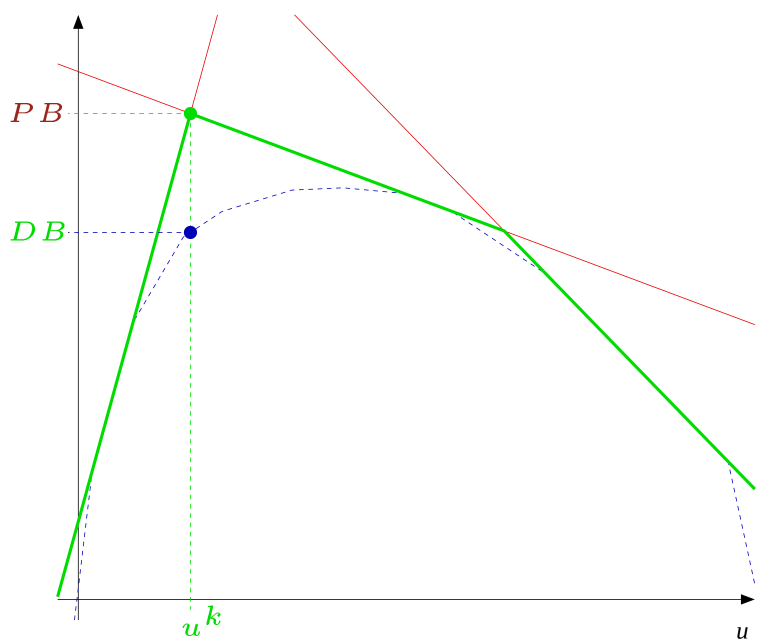


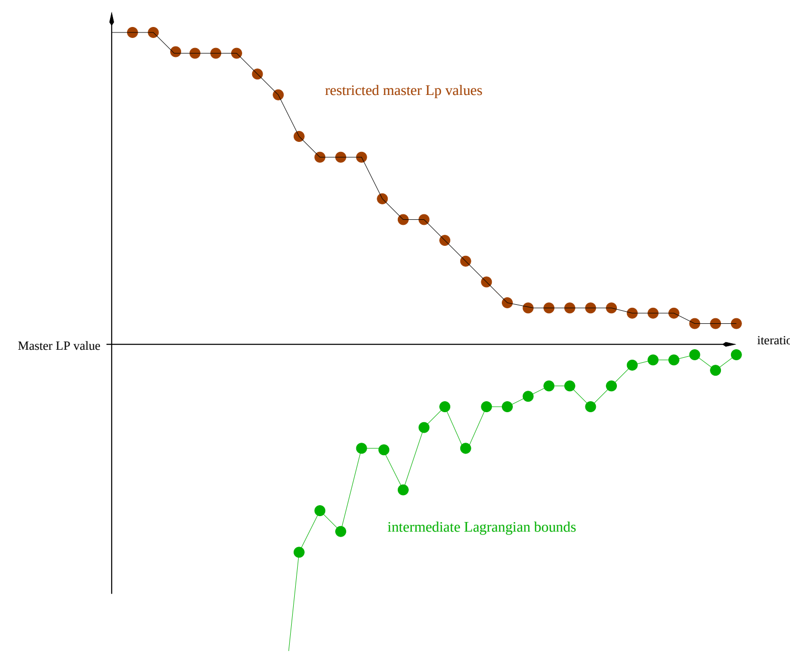
Standard column generation: primal/ dual view

At iteration k, one has

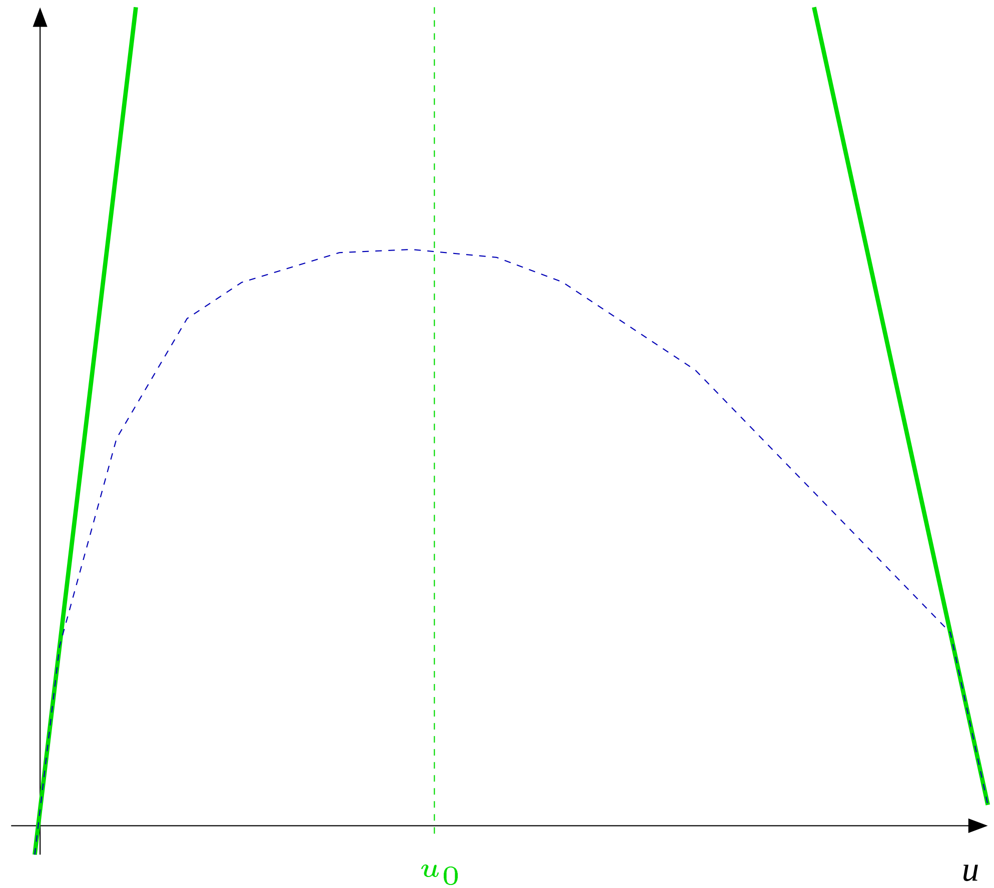
A Primal Bound on the master: $PB \equiv \theta^k(u^k) = \sum_{i \in I^k} cx^i \lambda_i$

A Dual Bound on the master: $DB \equiv \theta(u^k) = PB - (c - u^k A)x^{k+1}$

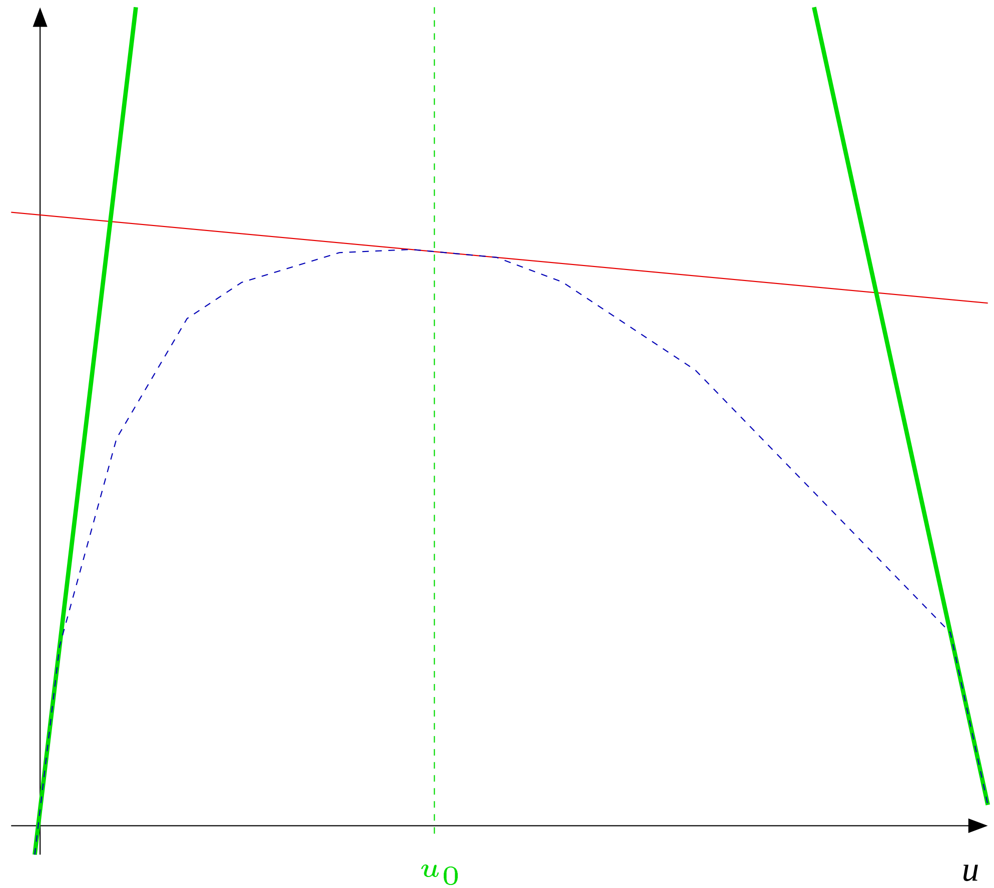




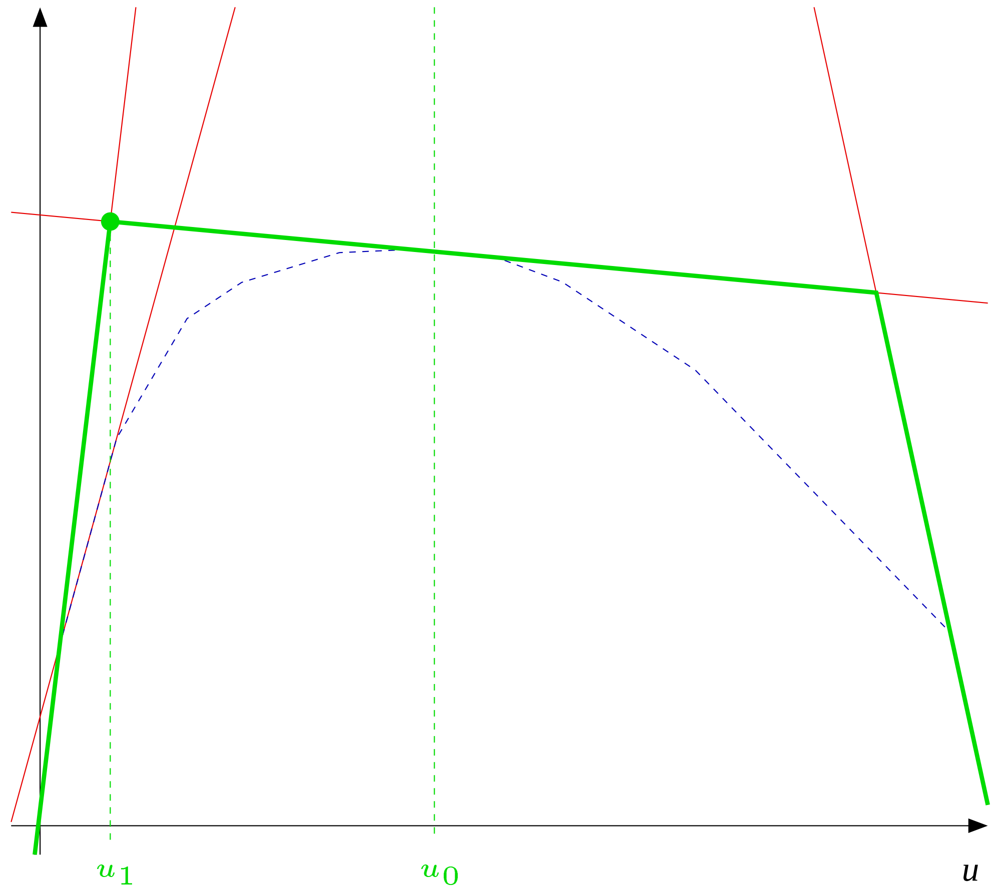
Slow convergence of Kelley's algorithm



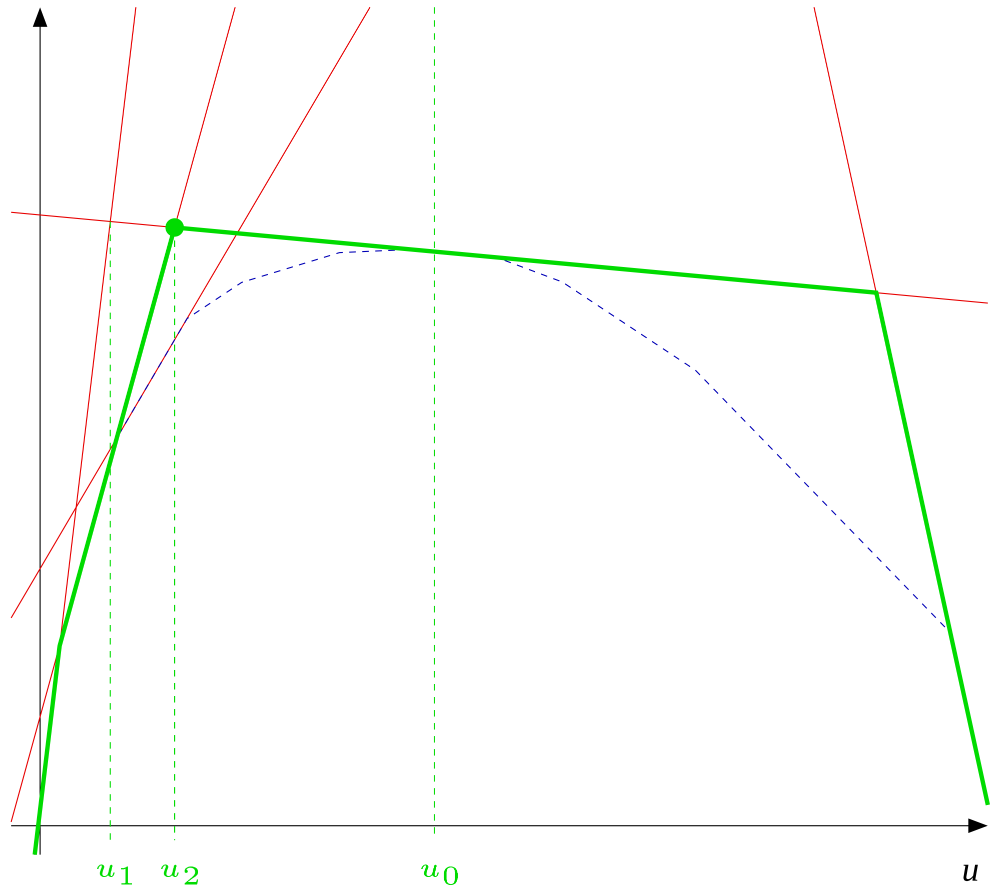
Slow convergence of Kelley's algorithm



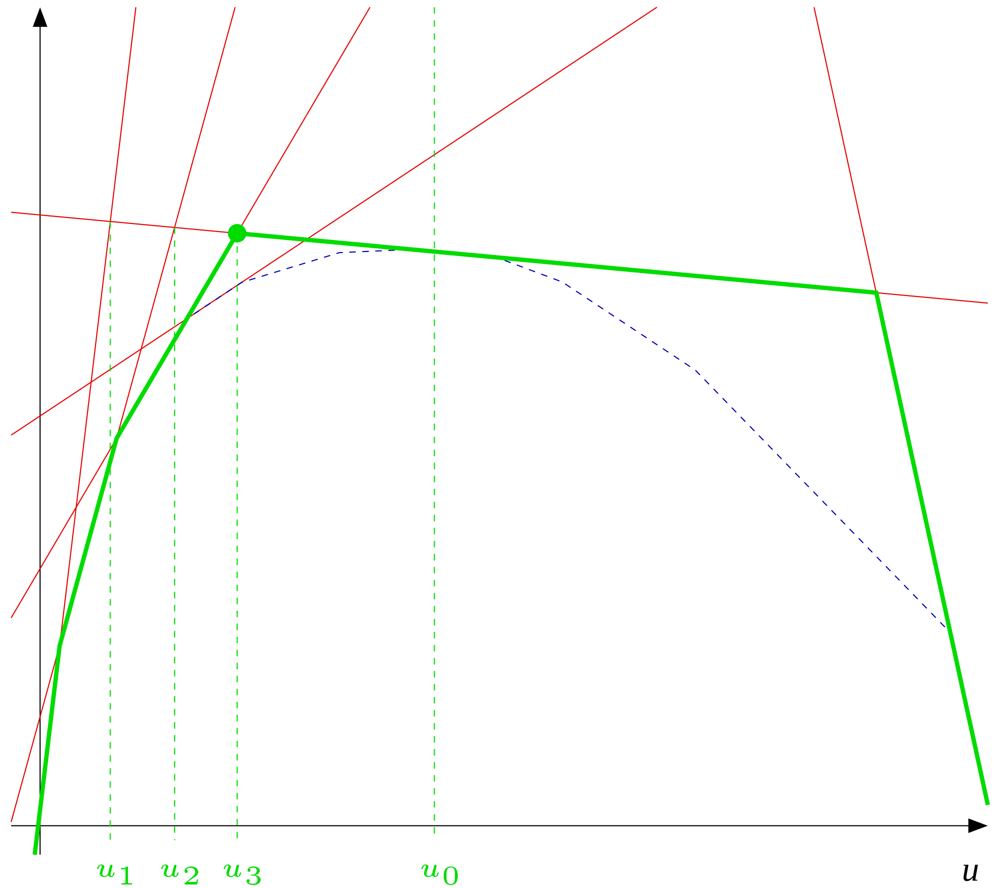
Slow convergence of Kelley's algorithm



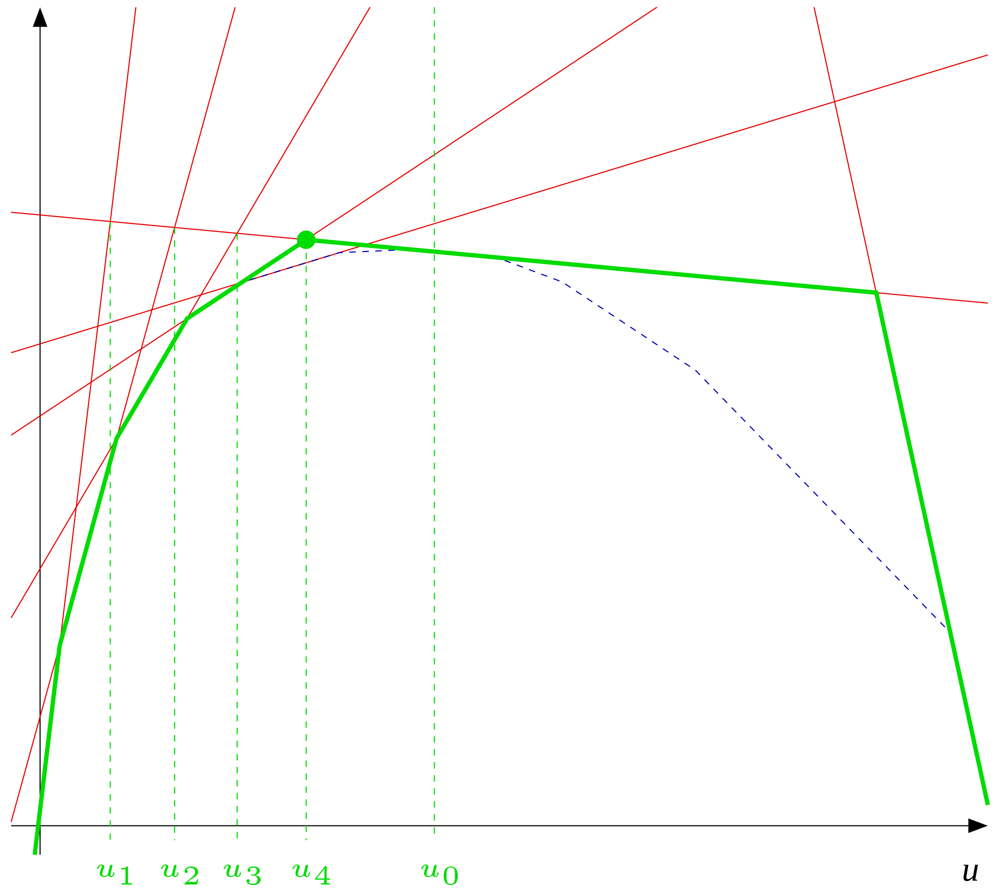
Slow convergence of Kelley's algorithm



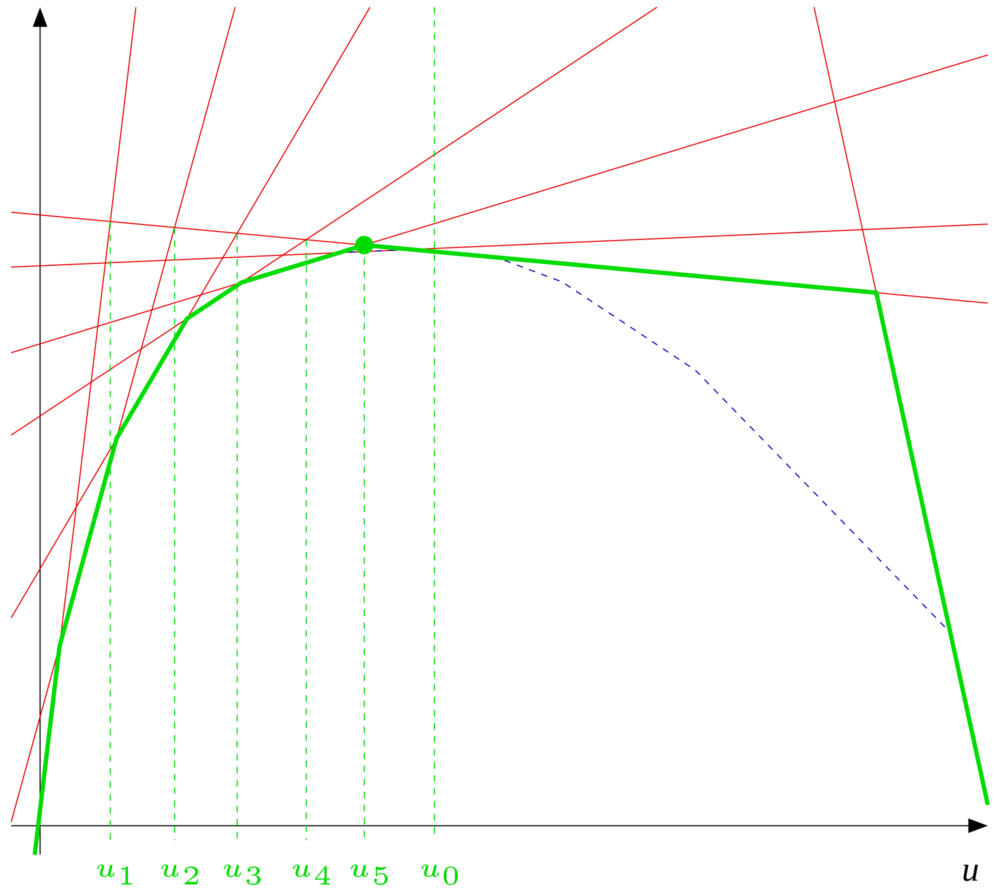
Slow convergence of Kelley's algorithm



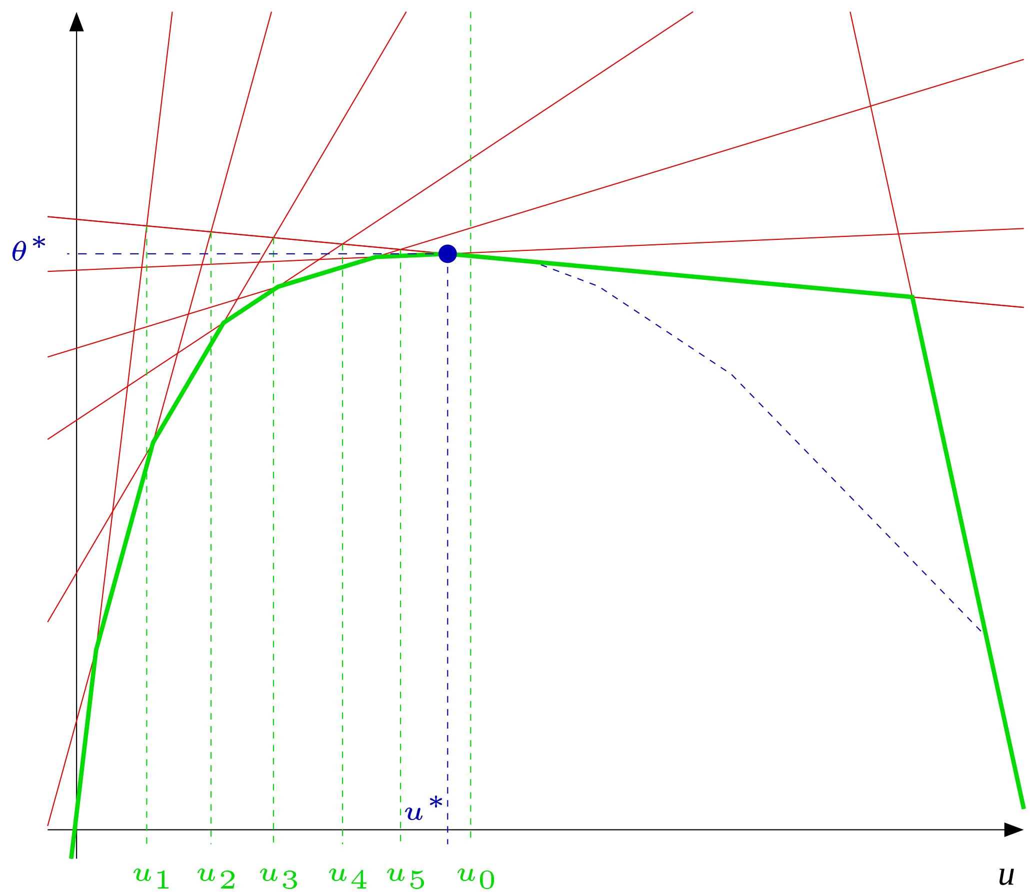
Slow convergence of Kelley's algorithm



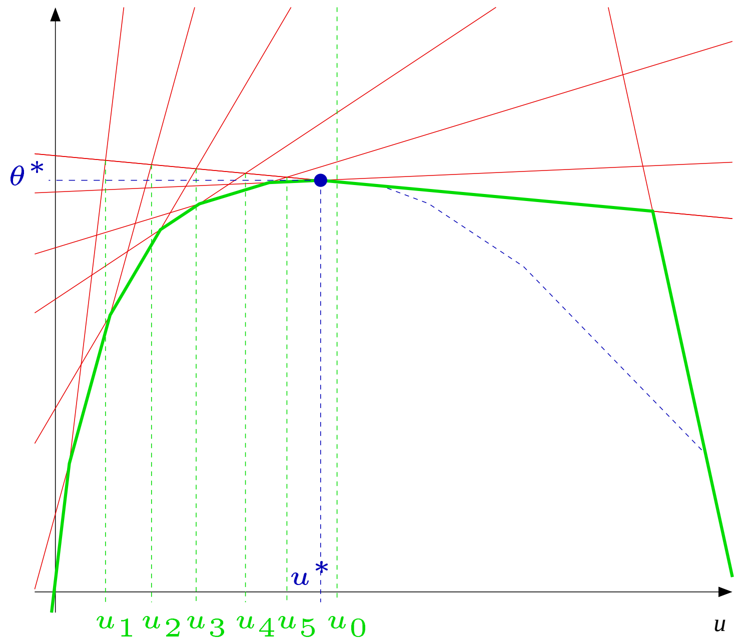
Slow convergence of Kelley's algorithm

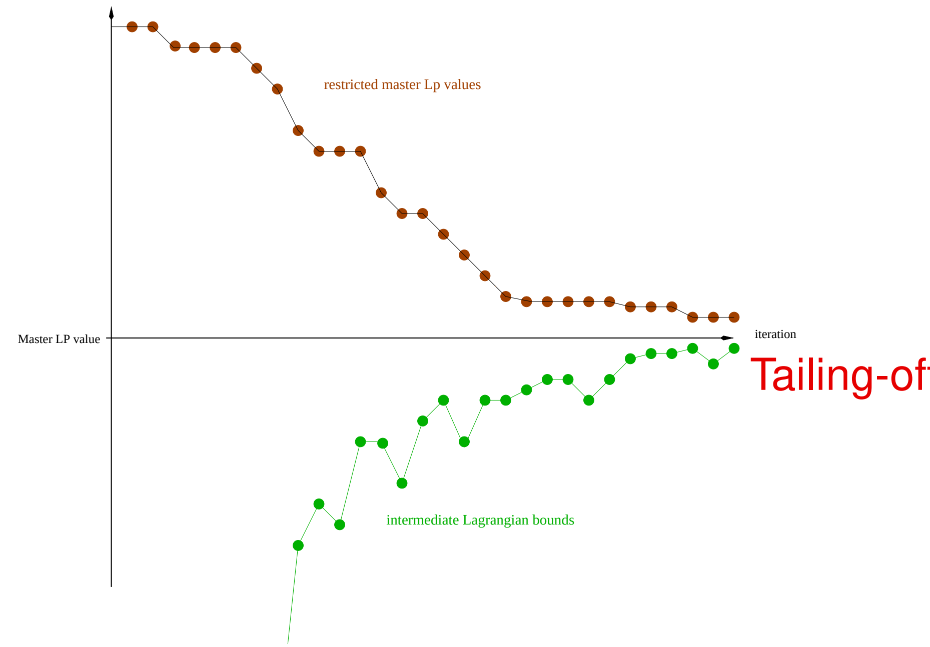


Slow convergence of Kelley's algorithm

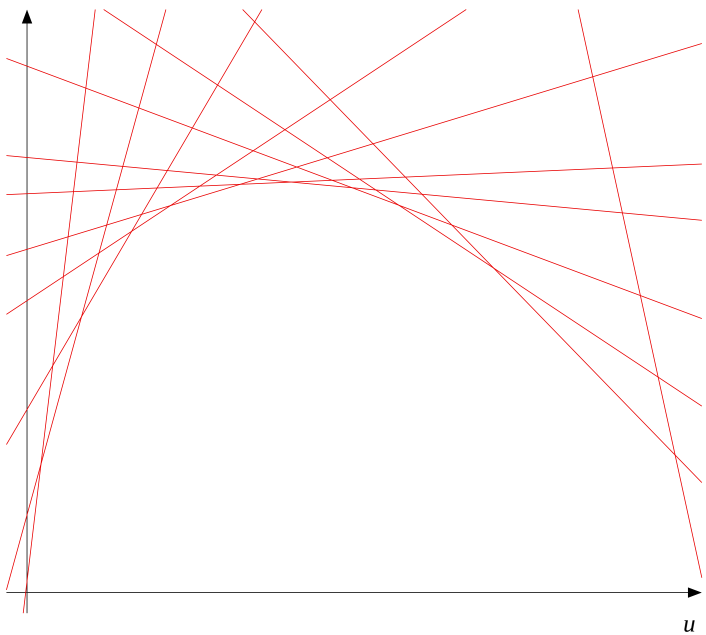


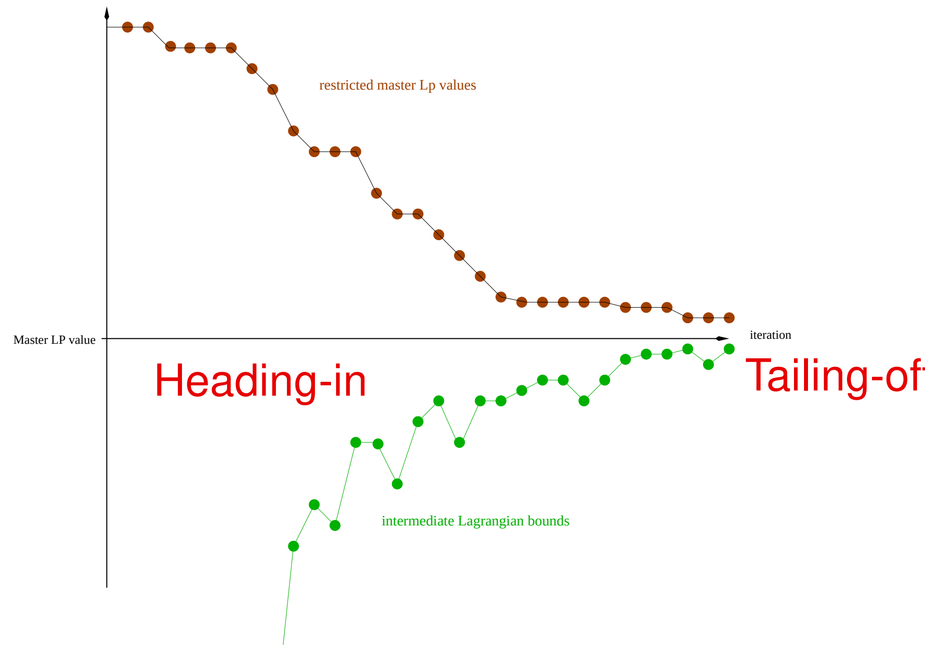
Instability: observed bad behaviors



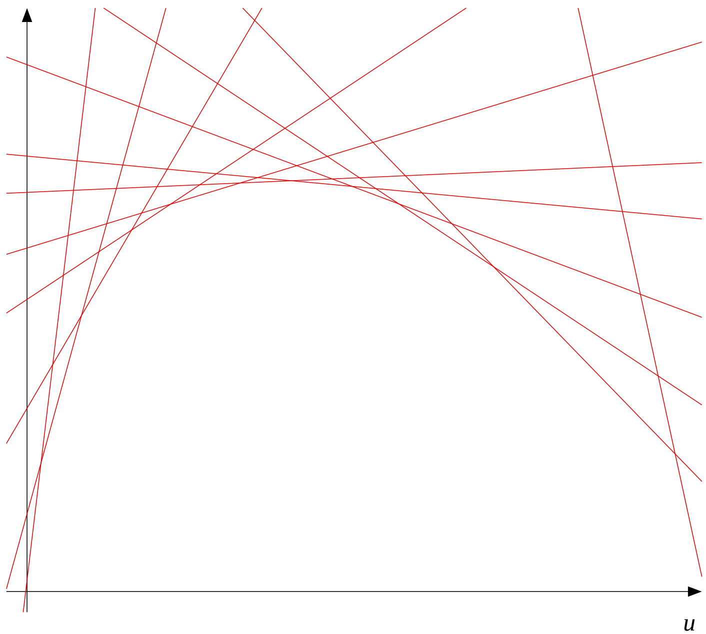


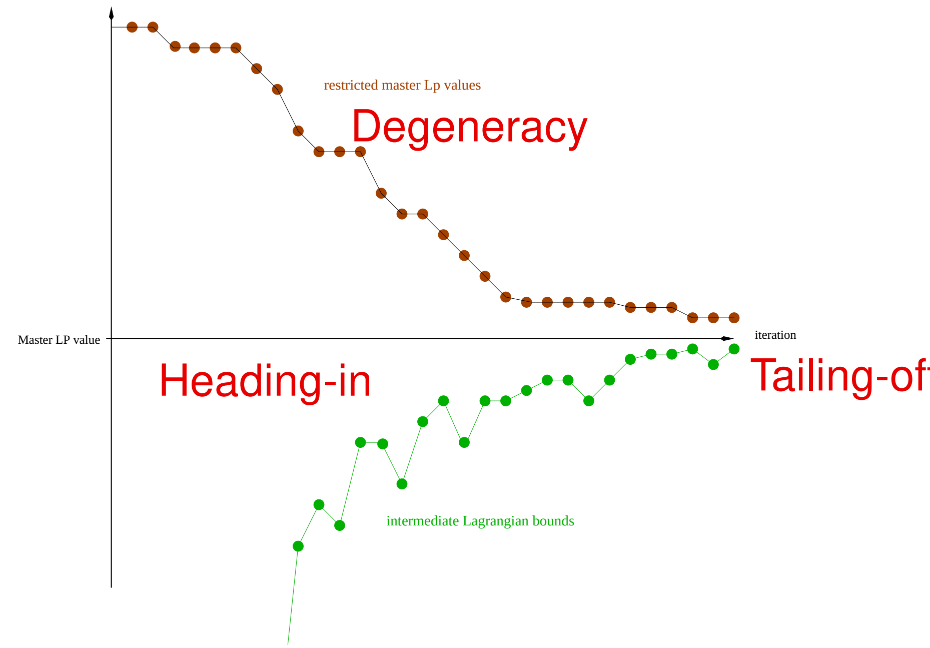
Instability: observed bad behaviors



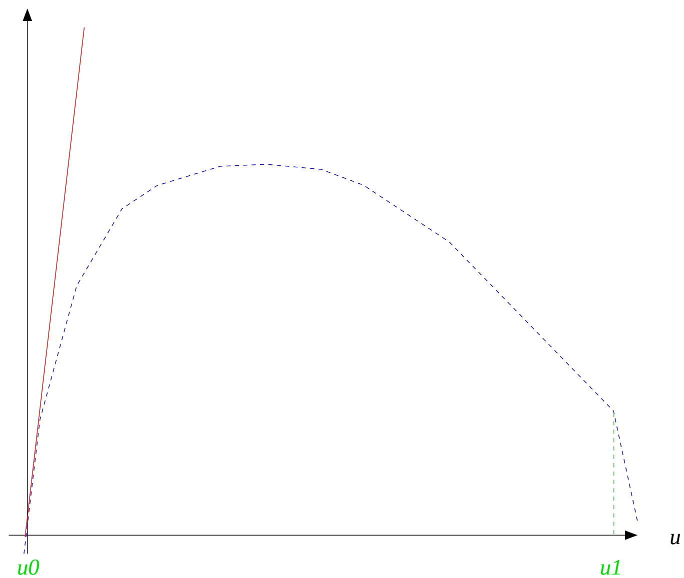


Instability: observed bad behaviors

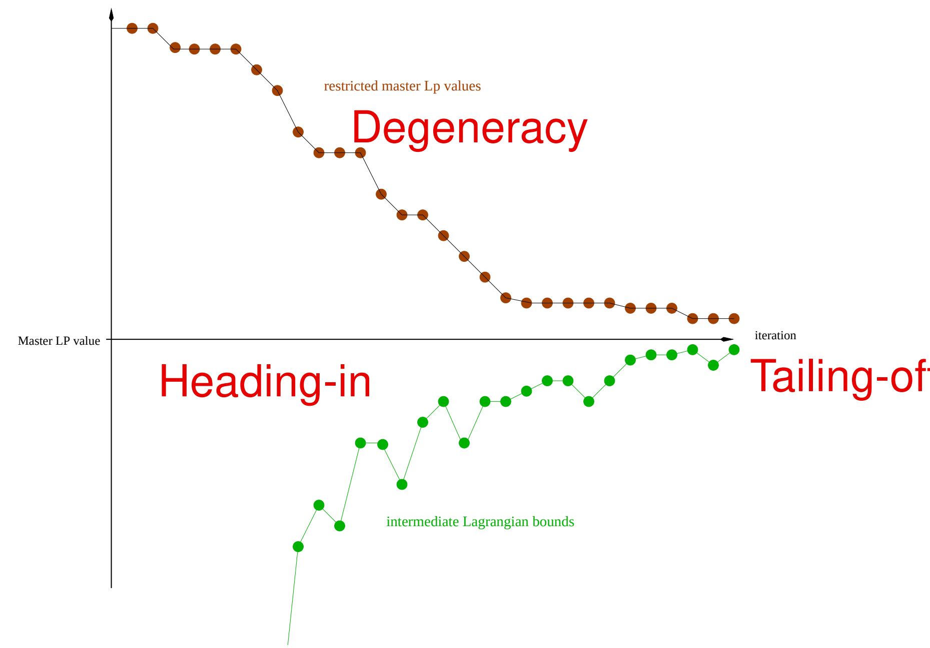




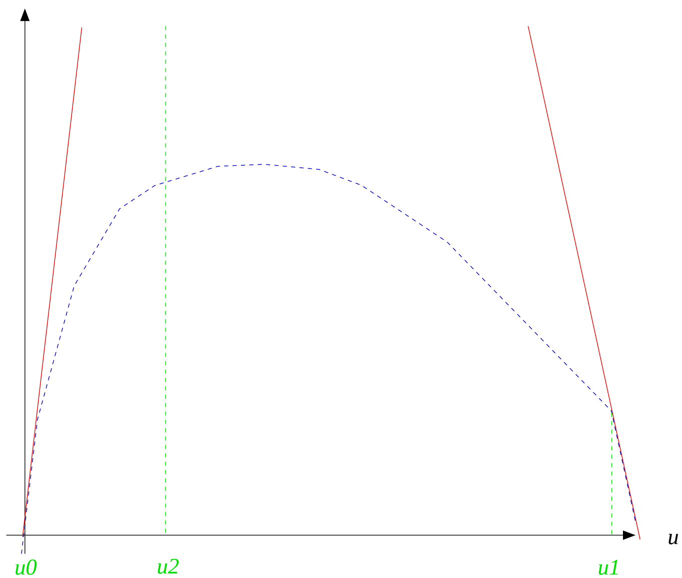
Instability: observed bad behaviors



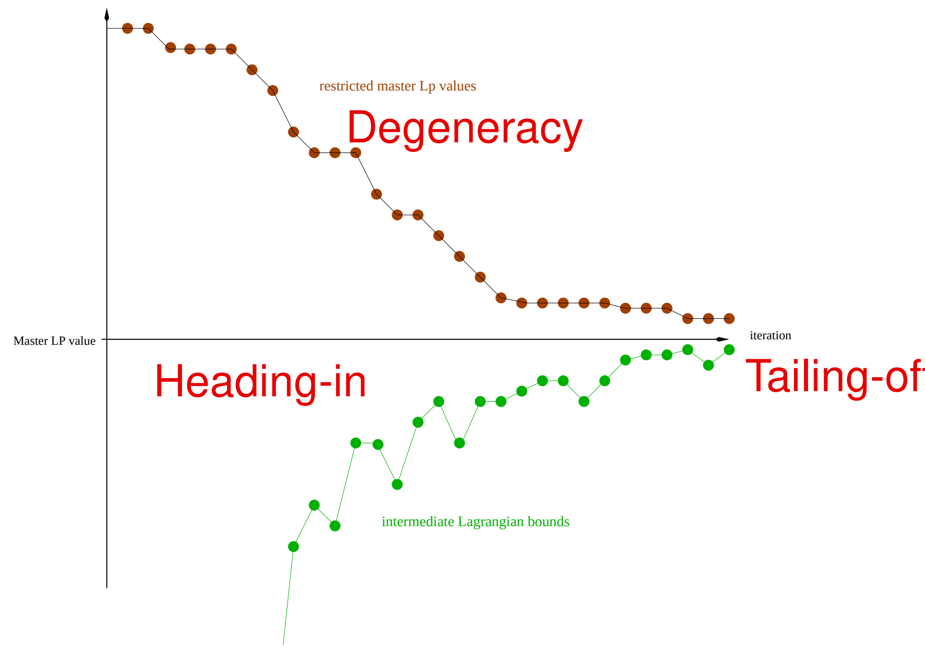
Bang-Bang Behavior



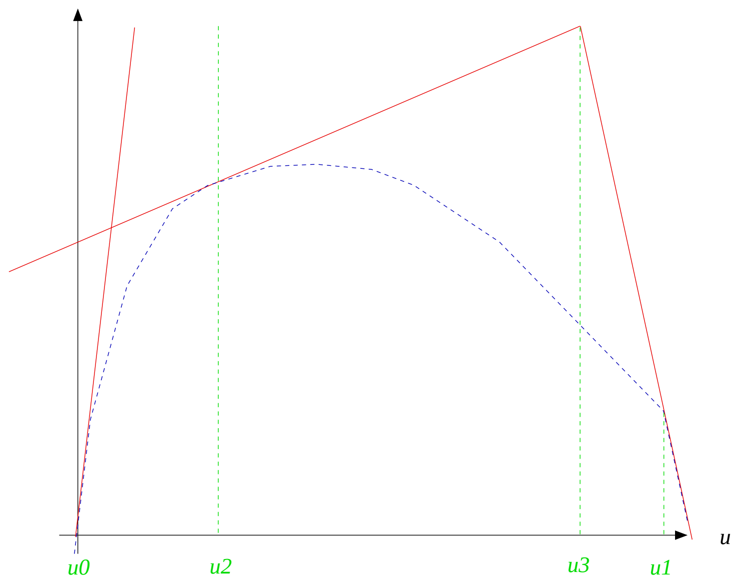
Instability: observed bad behaviors



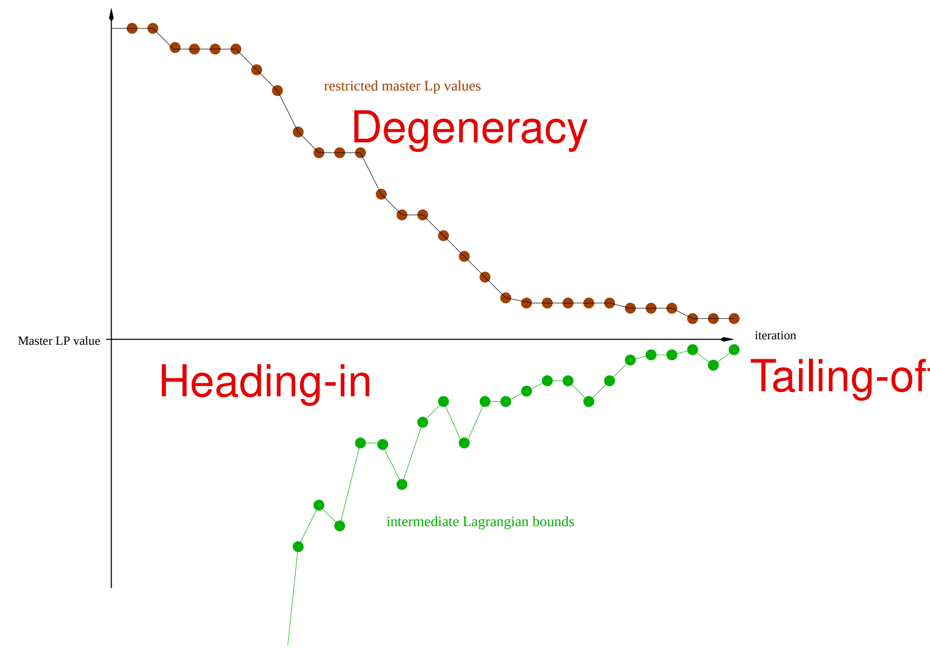
Bang-Bang Behavior



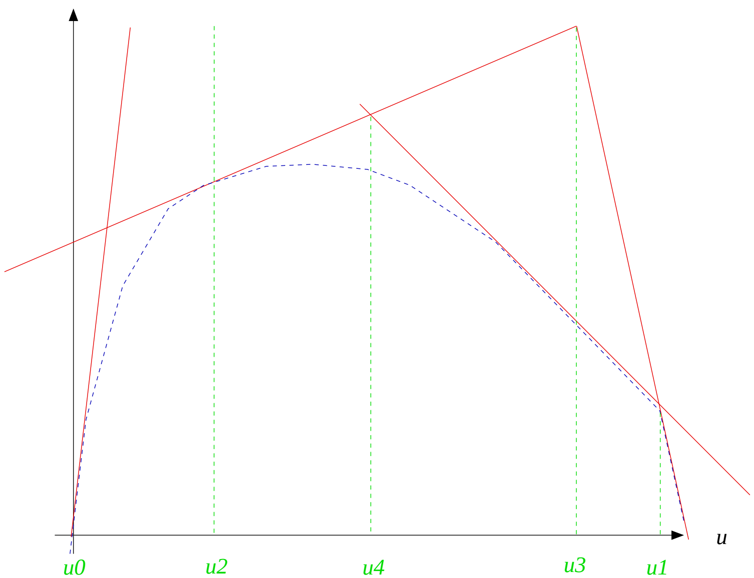
Instability: observed bad behaviors



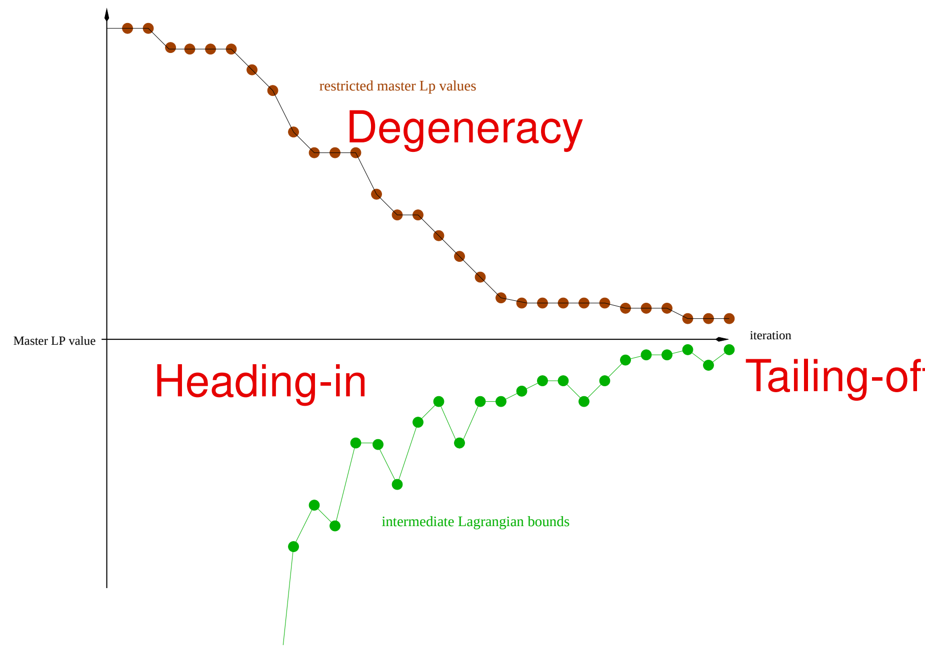
Bang-Bang Behavior



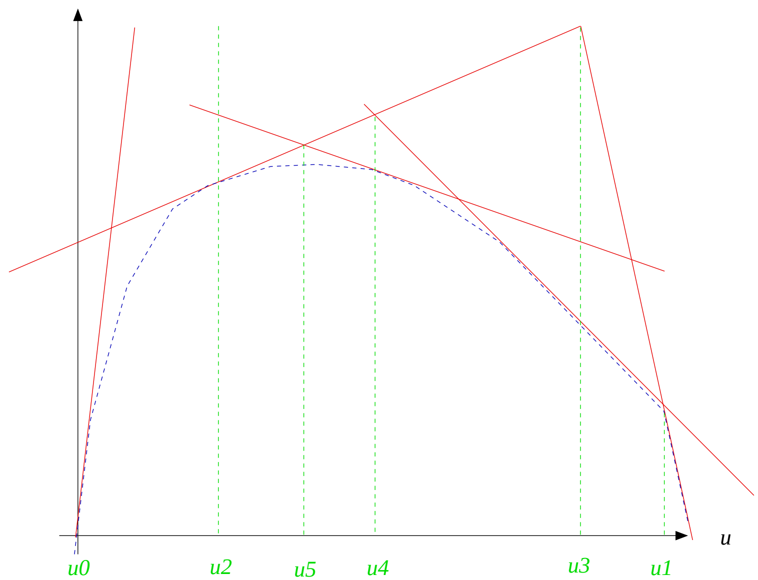
Instability: observed bad behaviors



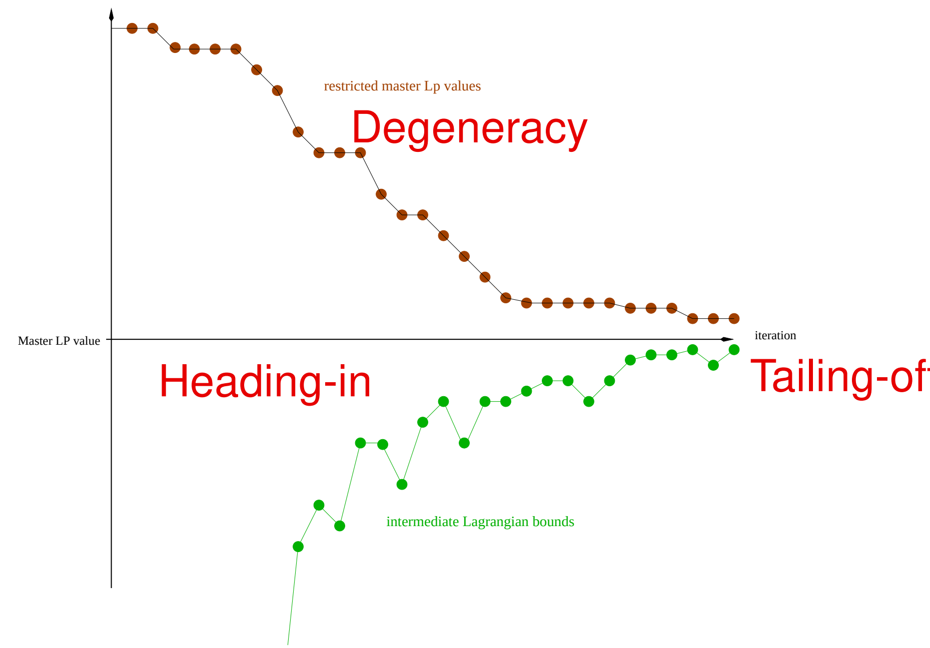
Bang-Bang Behavior



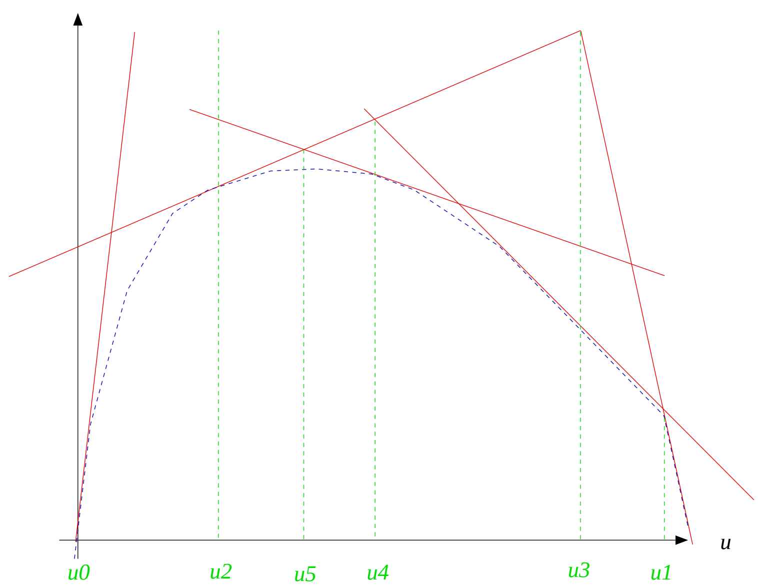
Instability: observed bad behaviors



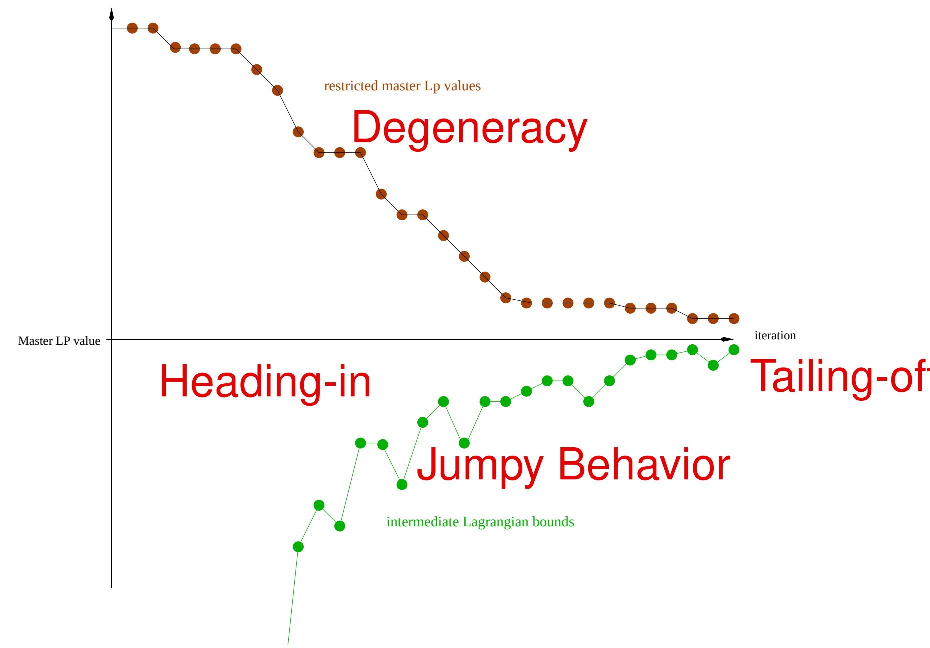
Bang-Bang Behavior



Instability: observed bad behaviors



Bang-Bang Behavior



Stabilization Methods -

- penalizing distance from a stability center $\hat{u}$

$$\hat{u} \rightarrow \text{ best dual bound}$$

Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

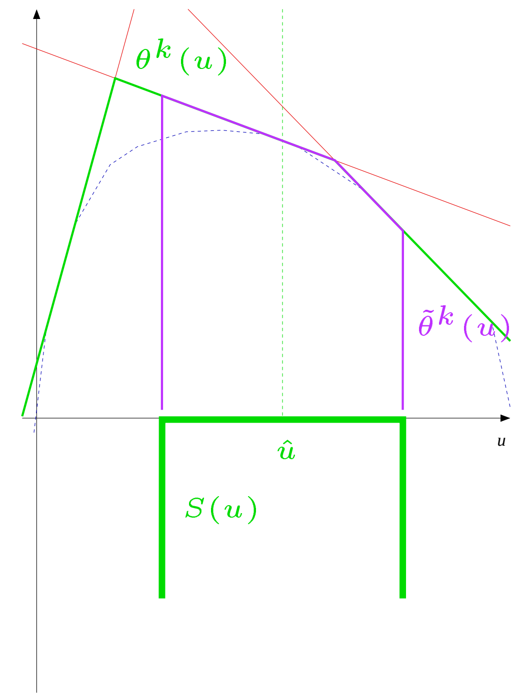
- penalizing distance from a stability center $\hat{u}$

Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

- penalizing distance from a stability center $\hat{u}$

  $S(u - \hat{u}) = \delta(||u - \hat{u}||_\infty \leq \epsilon)$ (box step)

  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975



Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

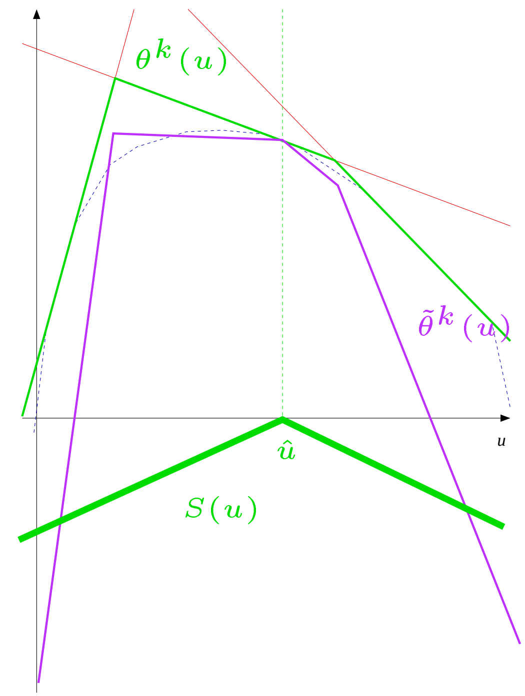
- penalizing distance from a stability center $\hat{u}$

  $S(u - \hat{u}) = \delta(||u - \hat{u}||_\infty \le \epsilon)$ (box step)

  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975

  $S(u - \hat{u}) = ||u - \hat{u}||_1$ (1 breakpoint)

  S.Kim, K.N.Chang and J.Y.Lee, 1995



Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

- penalizing distance from a stability center $\hat{u}$

  $S(u - \hat{u}) = \delta(||u - \hat{u}||_\infty \leq \epsilon)$ (box step)

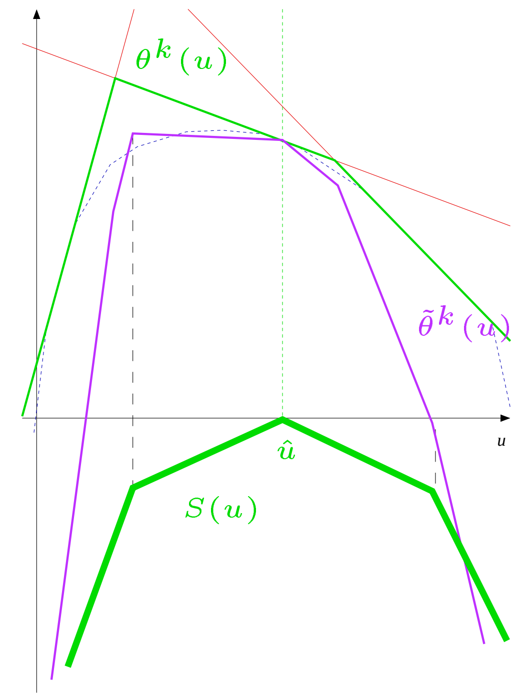
  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975

  $S(u - \hat{u}) = ||u - \hat{u}||_1$ (1 breakpoint)

  S.Kim, K.N.Chang and J.Y.Lee, 1995

  $S(u - \hat{u}) = ||u - \hat{u}||_1$+penalization outside boxstep (3 breakpoints)

  O.du Merle, D.Villeneuve, J.Desrosiers and P.Hansen, 1999



Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

- penalizing distance from a stability center $\hat{u}$

  $S(u - \hat{u}) = \delta(||u - \hat{u}||_\infty \leq \epsilon)$ (box step)

  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975

  $S(u - \hat{u}) = ||u - \hat{u}||_1$ (1 breakpoint)

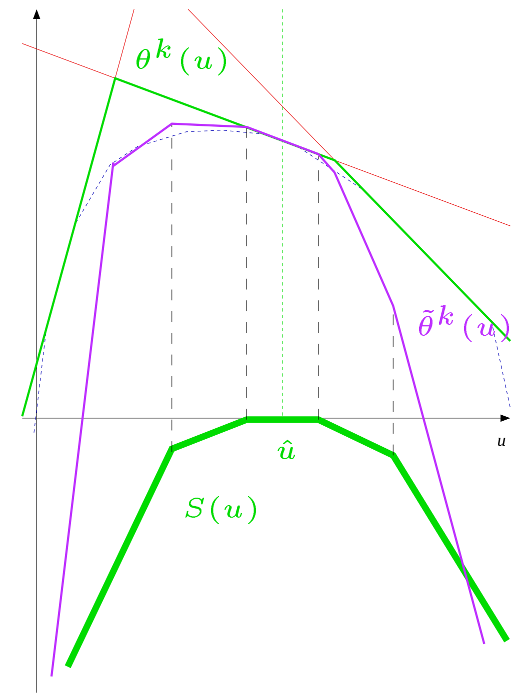
  S.Kim, K.N.Chang and J.Y.Lee, 1995

  $S(u-\hat{u}) = ||u-\hat{u}||_1 +$penalization outside boxstep (3 breakpoints)

  O.du Merle, D.Villeneuve, J.Desrosiers and P.Hansen, 1999

  $S(u-\hat{u})$ = penalization outside 2 boxsteps (4 breakpoints)

  H.Ben-Amor and J.Desrosiers, 2003



Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

- penalizing distance from a stability center $\hat{u}$

  $S(u - \hat{u}) = \delta(||u - \hat{u}||_\infty \leq \epsilon)$ (box step)

  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975

  $S(u - \hat{u}) = ||u - \hat{u}||_1$ (1 breakpoint)

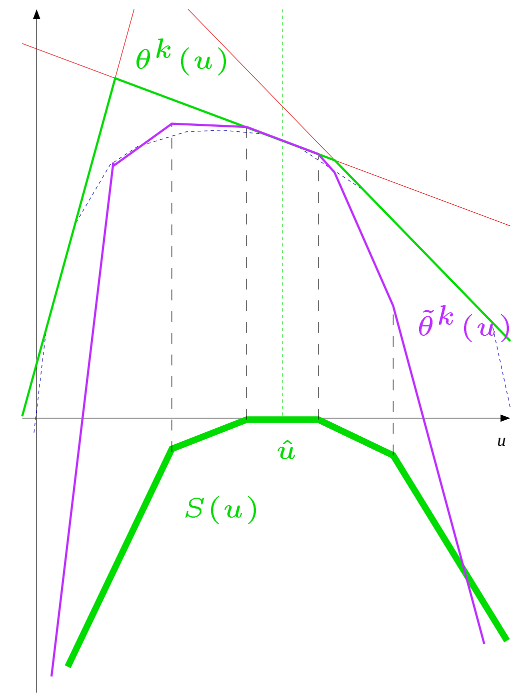
  S.Kim, K.N.Chang and J.Y.Lee, 1995

  $S(u - \hat{u}) = ||u - \hat{u}||_1 +$ penalization outside boxstep (3 breakpoints)

  O.du Merle, D.Villeneuve, J.Desrosiers and P.Hansen, 1999

  $S(u - \hat{u})$ = penalization outside 2 boxsteps (4 breakpoints)

  H.Ben-Amor and J.Desrosiers, 2003



$\Rightarrow$ As $S(u - \hat{u})$ is a concave and piecewise linear function, we can still use LP solvers

Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

- penalizing distance from a stability center $\hat{u}$

  $S(u - \hat{u}) = \delta(||u - \hat{u}||_\infty \leq \epsilon)$ (box step)

  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975

  $S(u - \hat{u}) = ||u - \hat{u}||_1$ (1 breakpoint)

  S.Kim, K.N.Chang and J.Y.Lee, 1995

  $S(u - \hat{u}) = ||u - \hat{u}||_1 +$ penalization outside boxstep (3 breakpoints)

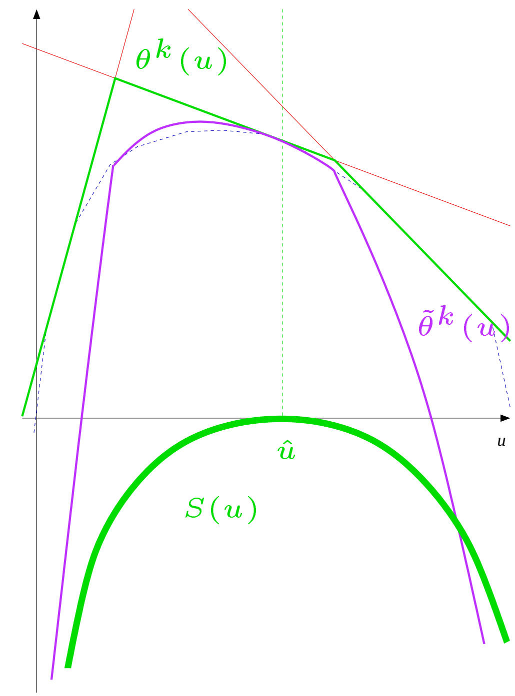
  O.du Merle, D.Villeneuve, J.Desrosiers and P.Hansen, 1999

  $S(u - \hat{u})$ = penalization outside 2 boxsteps (4 breakpoints)

  H.Ben-Amor and J.Desrosiers, 2003

  $S(u - \hat{u}) = \frac{1}{2t}||u - \hat{u}||^2 \Rightarrow$ **Bundle**

  C. Lemaréchal and C. Sagastizábal, 1997



Stabilization Methods - $\tilde{\theta}^k(u) = \theta^k(u) - S(u - \hat{u})$

- penalizing distance from a stability center $\hat{u}$

  $S(u-\hat{u}) = \delta(||u-\hat{u}||_\infty \leq \epsilon)$ (box step)

  R.E.Martsen, W.W.Hogan and J.W.Blankenship, 1975

  $S(u-\hat{u}) = ||u-\hat{u}||_1$ (1 breakpoint)

  S.Kim, K.N.Chang and J.Y.Lee, 1995

  $S(u-\hat{u}) = ||u-\hat{u}||_1 +$ penalization outside boxstep (3 breakpoints)

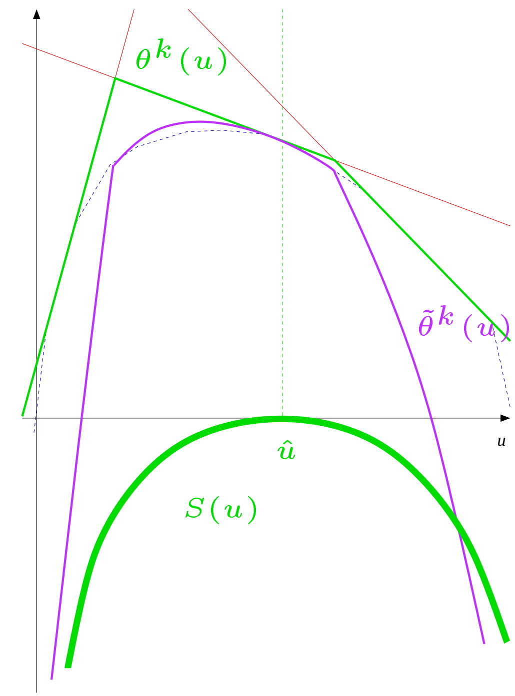
  O.du Merle, D.Villeneuve, J.Desrosiers and P.Hansen, 1999

  $S(u-\hat{u})$ = penalization outside 2 boxsteps (4 breakpoints)

  H.Ben-Amor and J.Desrosiers, 2003

  $S(u-\hat{u}) = \frac{1}{2t}||u-\hat{u}||^2 \Rightarrow$ **Bundle**

  C. Lemaréchal and C. Sagastizábal, 1997



$\Rightarrow$ As $S(u-\hat{u})$ is a quadratic concave function,
we must now use Quadratic solvers *K.C.Kiwiel, 1989, 1994, A.Frangioni, 1996*

Other Stabilization Methods

- smoothing dual value : $u^k = \alpha\, u^{Kelley} + (1-\alpha)\, \hat{u}$

  P.Wentges, 1997

Other Stabilization Methods

- smoothing dual value : $u^k = \alpha\, u^{Kelley} + (1-\alpha)\,\hat{u}$

  P.Wentges, 1997

- Using interior-point-like technique to generate dual solution

Other Stabilization Methods

- smoothing dual value : $u^k = \alpha\, u^{Kelley} + (1-\alpha)\,\hat{u}$

  P.Wentges, 1997

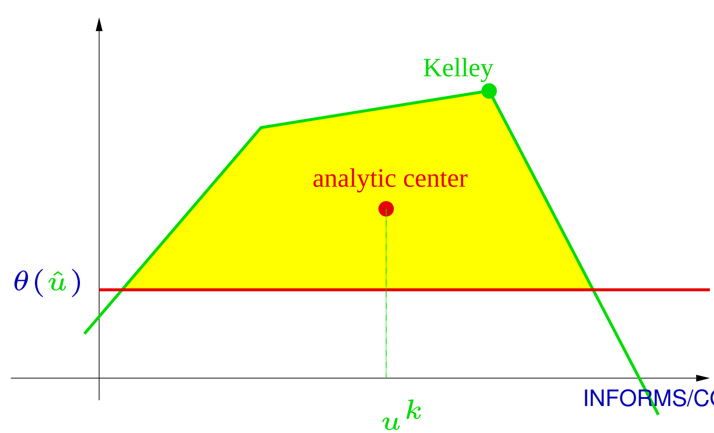
- Using interior-point-like technique to generate dual solution
  - Kelley solved by interior point method

    L-M Rousseau, M.Gendreau and D.Feillet, 2003

  - Analytic Center Cutting-Plane Method (ACCPM)

$$\tilde{\theta}^k(u) = \sum_{i \in I^k} \log(cx^i - u\,a\,x^i + r) + k\,\log(b\,u - r - \theta(\hat{u}))$$

J.L. Goffin, A.Haurie and J.Ph. Vial, 1992



Bi-Dual: Fenchel duality

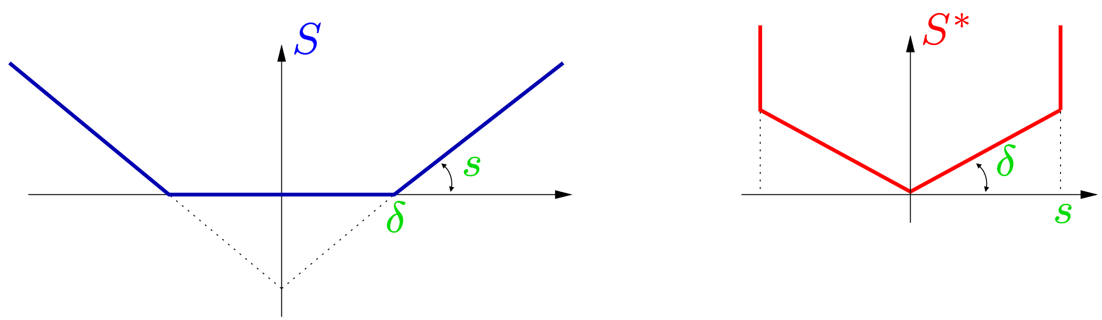
$$\begin{aligned} \max \theta^{k}(u) &\equiv \min\{c\,x : A\,x \geq b, x \in conv(X^{k})\} \\ \max \theta^{k}(u) - S(u - \hat{u}) &\equiv \quad \ldots \end{aligned}$$

Bi-Dual: Fenchel duality

$$
\begin{aligned}
\max \theta^{k}(u) \quad &\equiv \quad \min\{c\,x : \; A\,x \; \geq b, x \in conv(X^{k})\} \\
\max \theta^{k}(u) - S(u - \hat{u}) \quad &\equiv \quad \min\{c\,x + \hat{u}\; g + S^{*}(g) : \\
& \qquad A\,x \; + g \geq b, \; x \in conv(X^{k}), g \in \mathbb{R}^{m}\}
\end{aligned}
$$

Bi-Dual: Fenchel duality

$$
\begin{aligned}
\max \theta^k(u) \quad &\equiv \quad \min\{c\,x : \; A\,x \; \geq b, x \in conv(X^k)\} \\
\max \theta^k(u) - S(u - \hat{u}) \quad &\equiv \quad \min\{c\,x + \hat{u}\,g + S^*(g) : \\
& \qquad A\,x \; + g \geq b, \; x \in conv(X^k), g \in \mathbb{R}^m\}
\end{aligned}
$$



$$
\min \sum_{i=1}^{k} cx^i \lambda_i + \sum_{j=1}^{m} (\delta + \hat{u}_j) g_j^+ + \sum_{j=1}^{m} (\delta - \hat{u}_j) g_j^-
$$

$$
\begin{aligned}
\sum_{i=1}^{k} Ax^i \lambda_i + g_j^+ - g_j^- \quad &\geqslant \quad b_j \quad j = 1, \ldots, m\,, \\
s \geqslant g_j^+, \; g_j^- \quad &\geqslant \quad 0 \quad j = 1, \ldots, m\,, \\
\sum_{i=1}^{k} \lambda_i = 1, \;\; \lambda_i \quad &\geqslant \quad 0 \quad i\,.
\end{aligned}
$$

PART 2: Numerical Comparison

BaPCod : a generic Branch-And-Price Code

- $\exists$ '*tool-box*" codes : MINTO, ABACUS, BCP, or MAESTRO
- Our aim is to develop a '*black-box*" implementation (in the same way as Cplex or Xpress provide implementation of Branch-and-Bound)

BaPCod : a generic Branch-And-Price Code

- $\exists$ '*tool-box*" codes : MINTO, ABACUS, BCP, or MAESTRO
- Our aim is to develop a '*black-box*" implementation (in the same way as Cplex or Xpress provide implementation of Branch-and-Bound)
  - The user provides a MIP formulation of its problem
  - The user specifies what constraints make a subproblem
  - The code reformulates the problem for column generation
  - The code solves it with a default branch-and-price implementation

BaPCod : motivations

Practical:

- Provides a starting point to try out a column generation approach.
- Allows to test different decompositions: just change the subproblem definition.

Research Wise:

- Demands to generalize ad-hoc features to turn them into generic tools: initialization, preprocessing, branching, primal heuristic . . .
- Any algorithmic development is made available for all applications
- Permits to test algorithmic features across applications

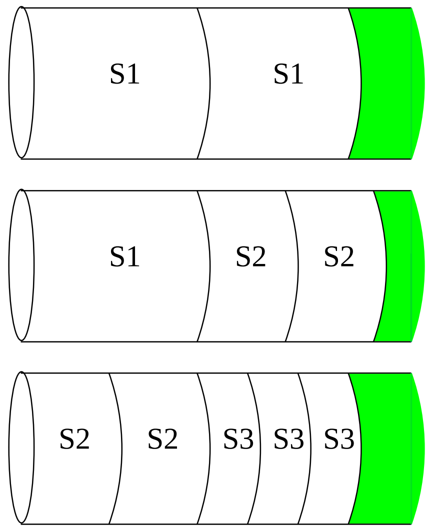
Tested variants

1. Kelley “Basic” initialized with 1 artificial column

2. Kelley “Rich” initialized with m artificial columns of cost $\alpha\hat{u}$ (with an heuristic $\hat{u}$)

3. Bundle initialized with $u_0 = \hat{u}$

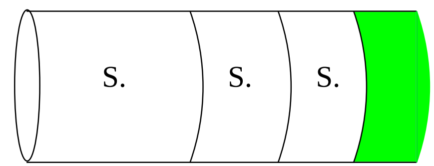
Cutting Stock Problem (CSP)



$\equiv$

$\sum_k x_{ik} \geq d_i \quad \forall i$

$+$



- $\min$ **number of rolls**

$$\begin{cases} \min & \sum_k y_k & & & \\ & \sum_k x_{ik} & \geq & d_i & \forall i \\ & \sum_i s_i x_{ik} & \leq & y_k & \forall k \end{cases}$$

- $\min$ **waste**

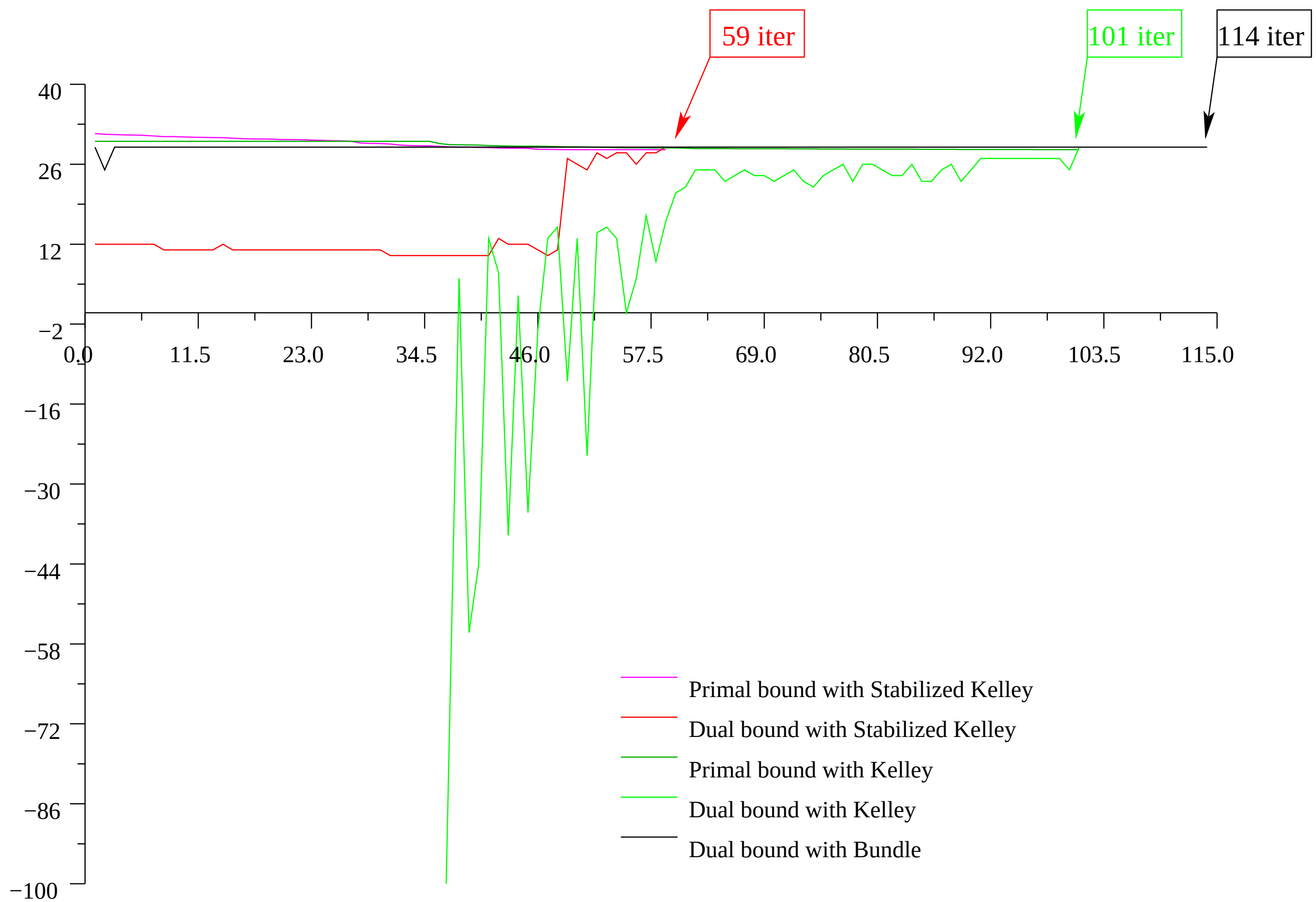
$$\begin{cases} \min & \sum_k (1 - \sum_i s_i x_{ik}) & & & \\ & \overline{d}_i \geq \sum_k x_{ik} & \geq & \underline{d}_i & \forall i \\ & \sum_i s_i x_{ik} & \leq & y_k & \forall k \end{cases}$$

CSP number of rolls: Comparative results

average on 10 or 20 instances

		Counters		Timers (ticks)		
Problem	Method	Iter	Col	Oracle	Master	total
realDat	kelley basic	45	46	54	2	0s64t
$i = 7 - 33$	kelley rich	29	29	422	2	4s30t
	bundle	51	38	1144	20	11s71t
randomDat	kelley basic	226	237	589	38	6s69t
$i = 50$	kelley rich	139	139	520	28	5s73t
	bundle	142	119	1703	260	19s89t

CSP number of rolls: primal and dual bounds

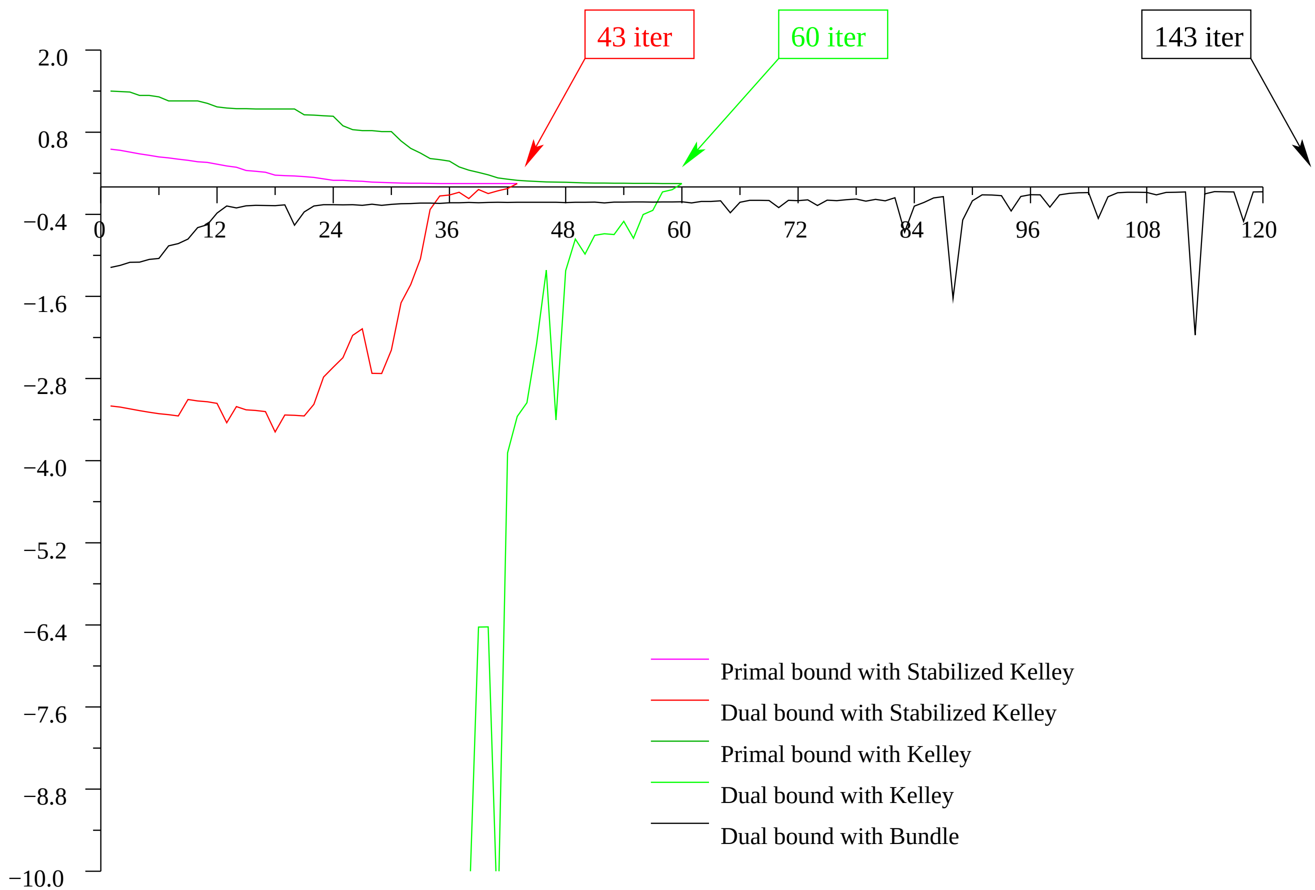


CSP Waste: Comparative results

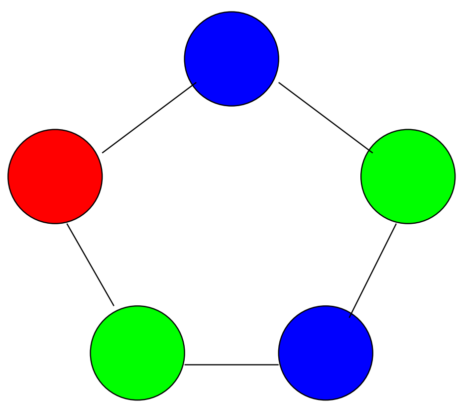
average on 10 or 20 instances

		Counters		Timers (ticks)		
Problem	Method	Iter	Col	Oracle	Master	total
realDat	kelley basic	43	43	239	3	2s53t
$i = 7 - 33$	kelley rich	31	26	842	3	8s53t
	bundle	60	35	1535	25	15s71t
randomDat	kelley basic	144	143	825	29	9s03t
$i = 50$	kelley rich	142	120	3280	29	33s56t
	bundle	200	86	5892	578	1m5s15t

CSP Waste: primal and dual bounds



Vertex Coloring problem (VCP)

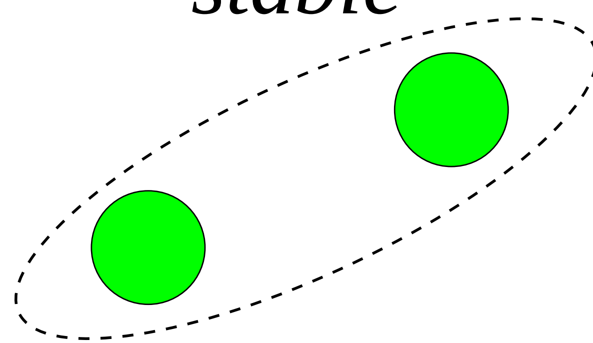


$\equiv$

$$\sum_k x_{ik} = 1 \quad \forall i$$

$+$

stable

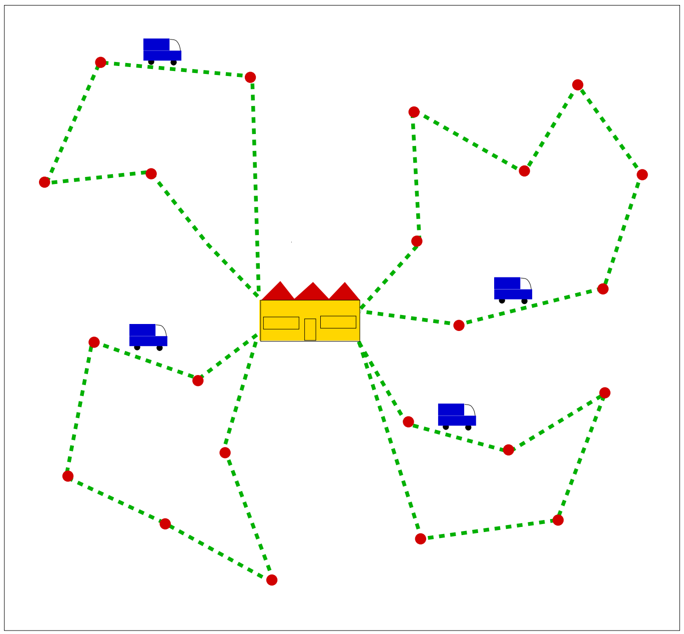


VCP : Comparative results

instances from DIMACS challenge
(mat.gsia.cmu.edu/COLOR/instances.html)

		Counters		Timers (ticks)		
Problem	Method	Iter	Col	Oracle	Master	total
Mycielski	kelley basic	94	94	6391	83	1m08s73t
$n = 5 - 47$	kelley rich	69	69	6612	66	1m11s54t
$m = 5 - 236$	bundle	41	22	3853	1949	1m01s90t
Queen	kelley basic	183	183	119174	267	20m32s12t
$n = 25 - 81$	kelley rich	128	128	88701	179	15m19s80t
$m = 160 - 2112$	bundle	50	46	13610	2380	15m22s31t

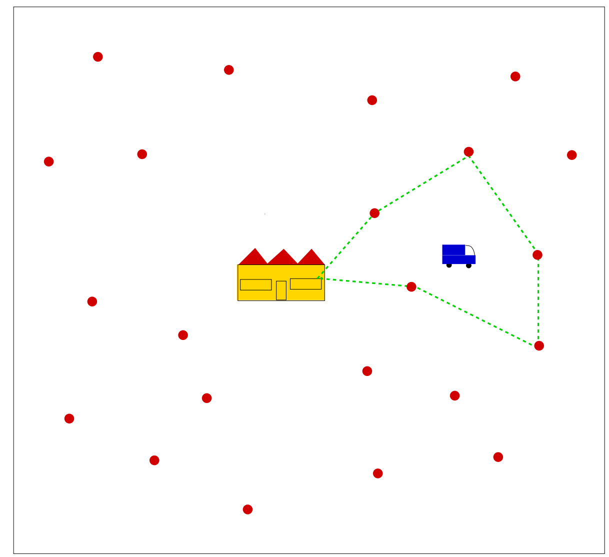
Capacitated Vehicle Routing Problem (CVRP)



$\equiv$

$$\sum_{k \in K} x_{ik} \geq 1 \quad \forall i \neq 0$$

$+$

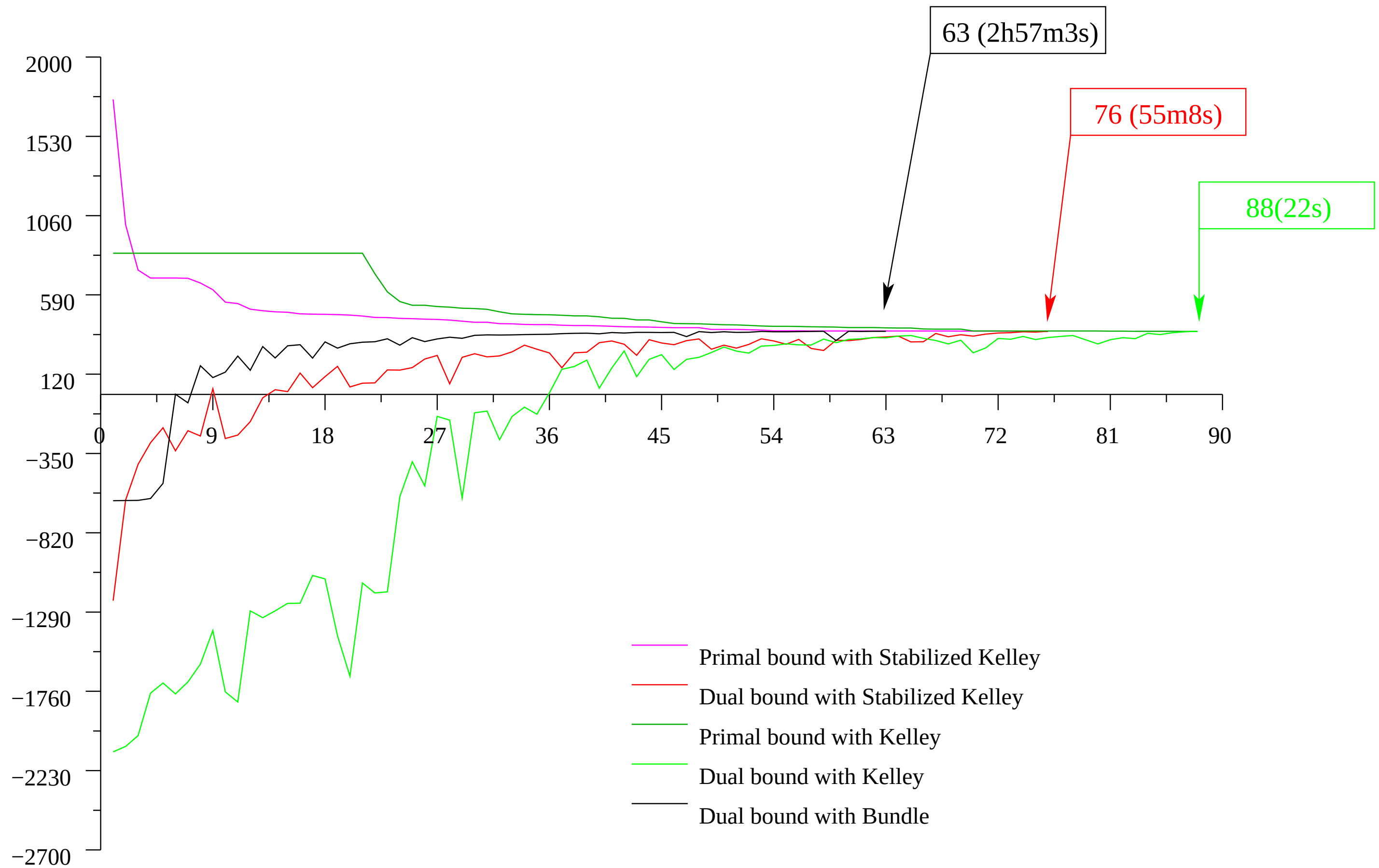


CVRP : Comparative results

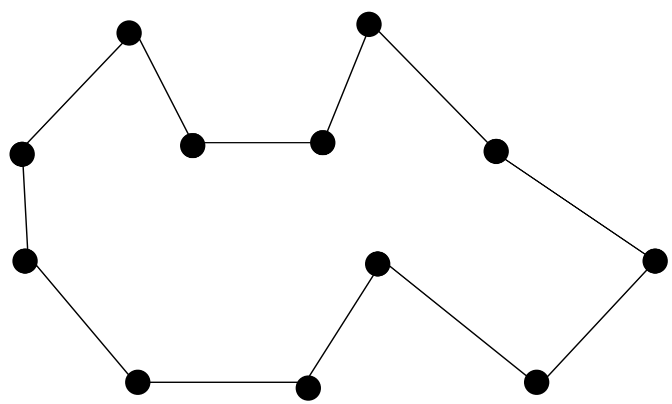
Instances and generated instances from CVRPLIB (www.branchandcut.org/VRP/data)

		Counters		Timers (ticks)		
Problem	Method	Iter	Col	Oracle	Master	total
P-k8	kelley basic	49	48	21	7	0s35t
average on 3 inst	kelley rich	42	40	22	11	0s43t
n $= 16 - 23$	bundle	59	32	76	33	1s21t
E-n22-k4	kelley basic	88	88	2144	24	21s84t
	kelley rich	76	75	330826	22	55m8s59t
	bundle	63	47	1062299	26	2h57m3s45t
gen-k2	kelley basic	34	35	4044	3	40s53t
average on 8 inst	kelley rich	28	28	3046	4	30s57t
n $= 8 - 14$	bundle	17	13	10490	6	1m45s0t

CVRP : E-n22-k4 : primal and dual bounds



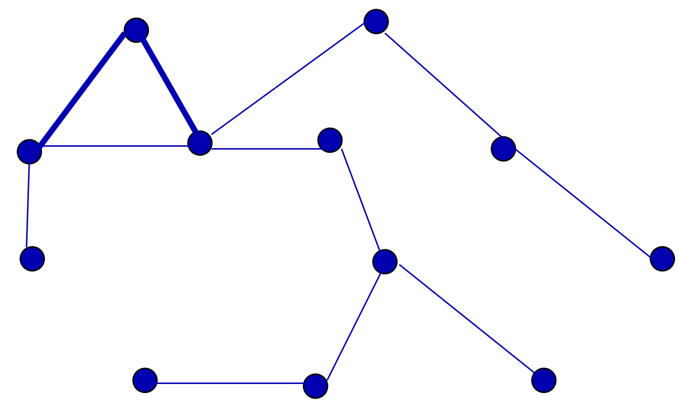
Traveling Salesman Problems (TSP)



$\equiv$

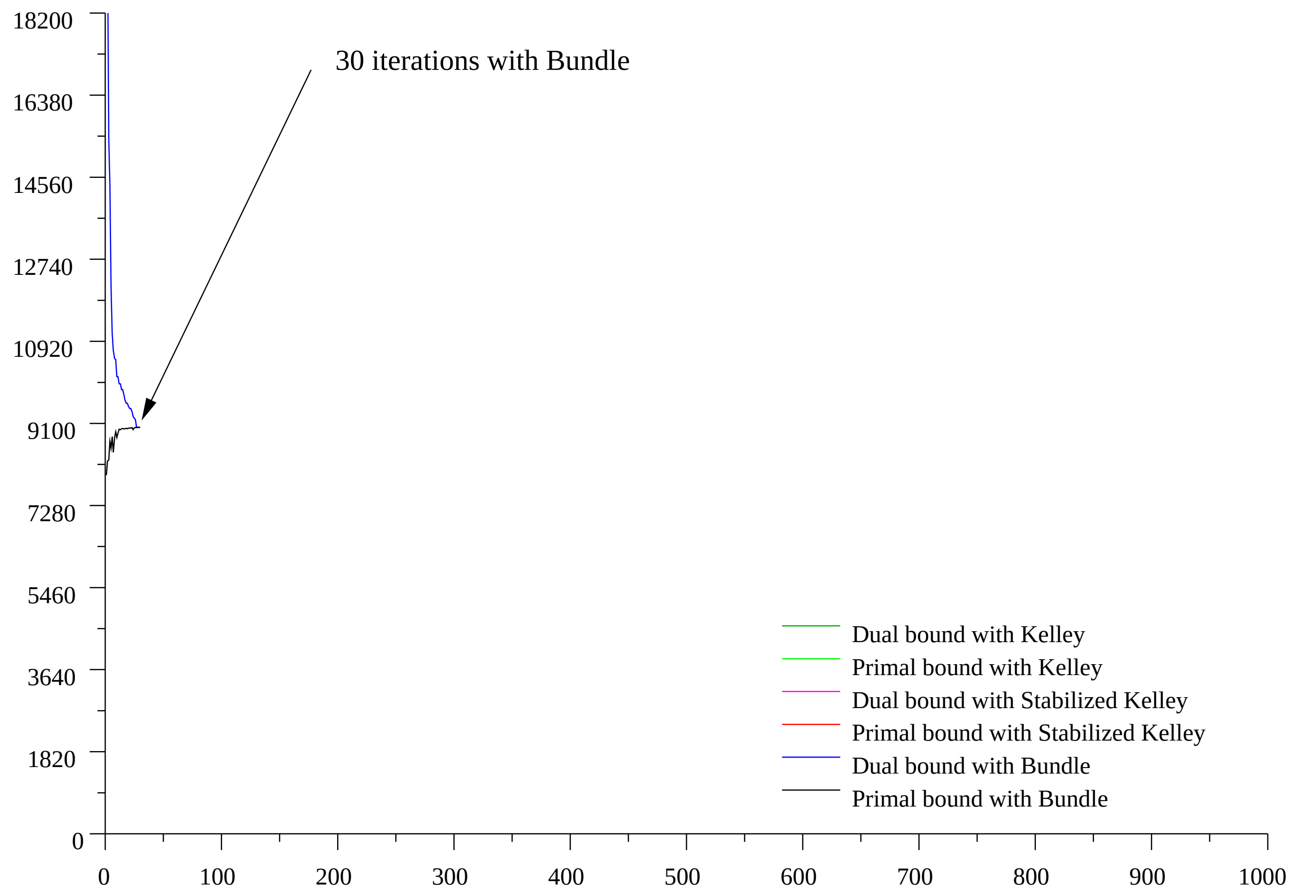
$\sum_{e \in \delta(i)} x_e = 2 \;\; \forall i$

$+$

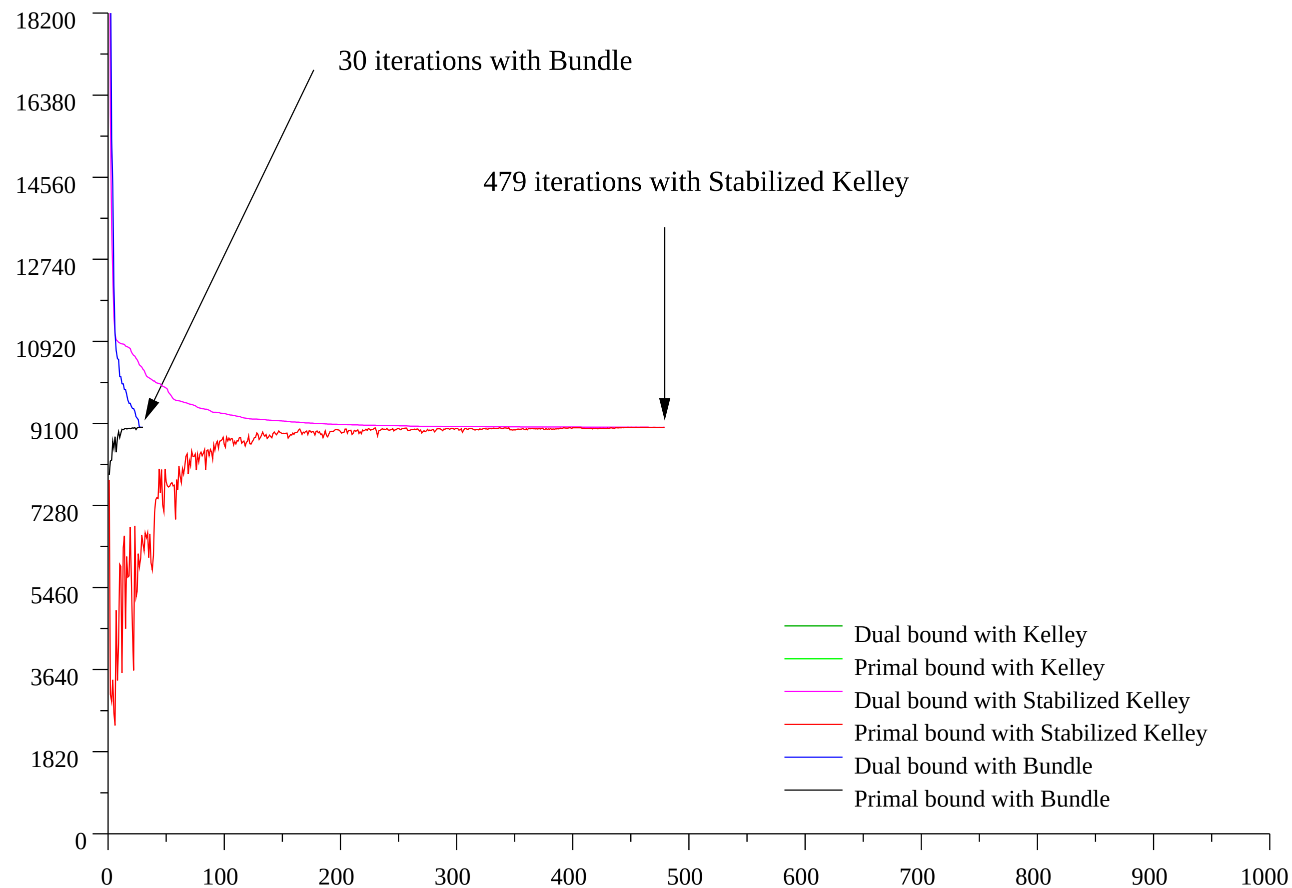


Instances from TSPLIB

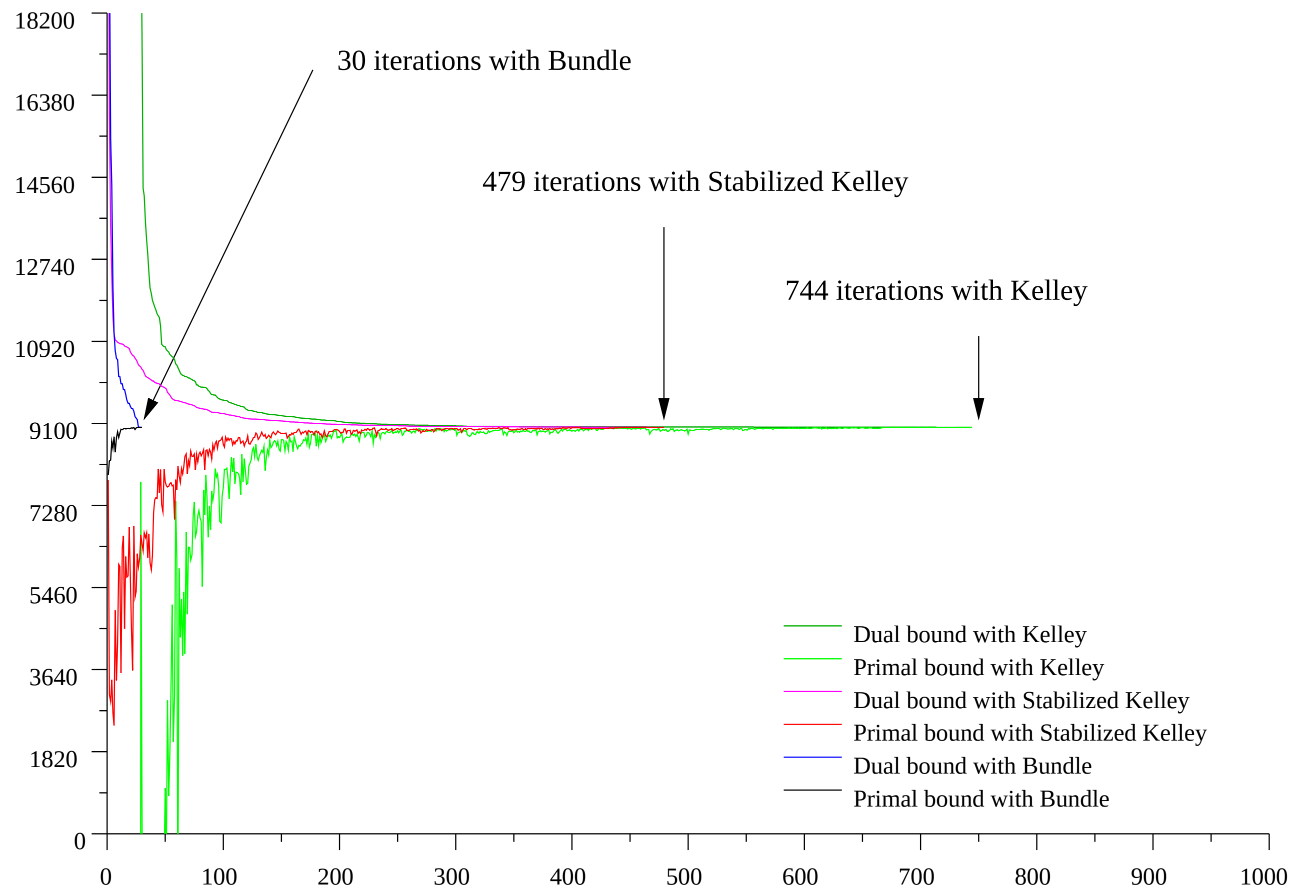
TSP : bays29 : primal and dual bounds



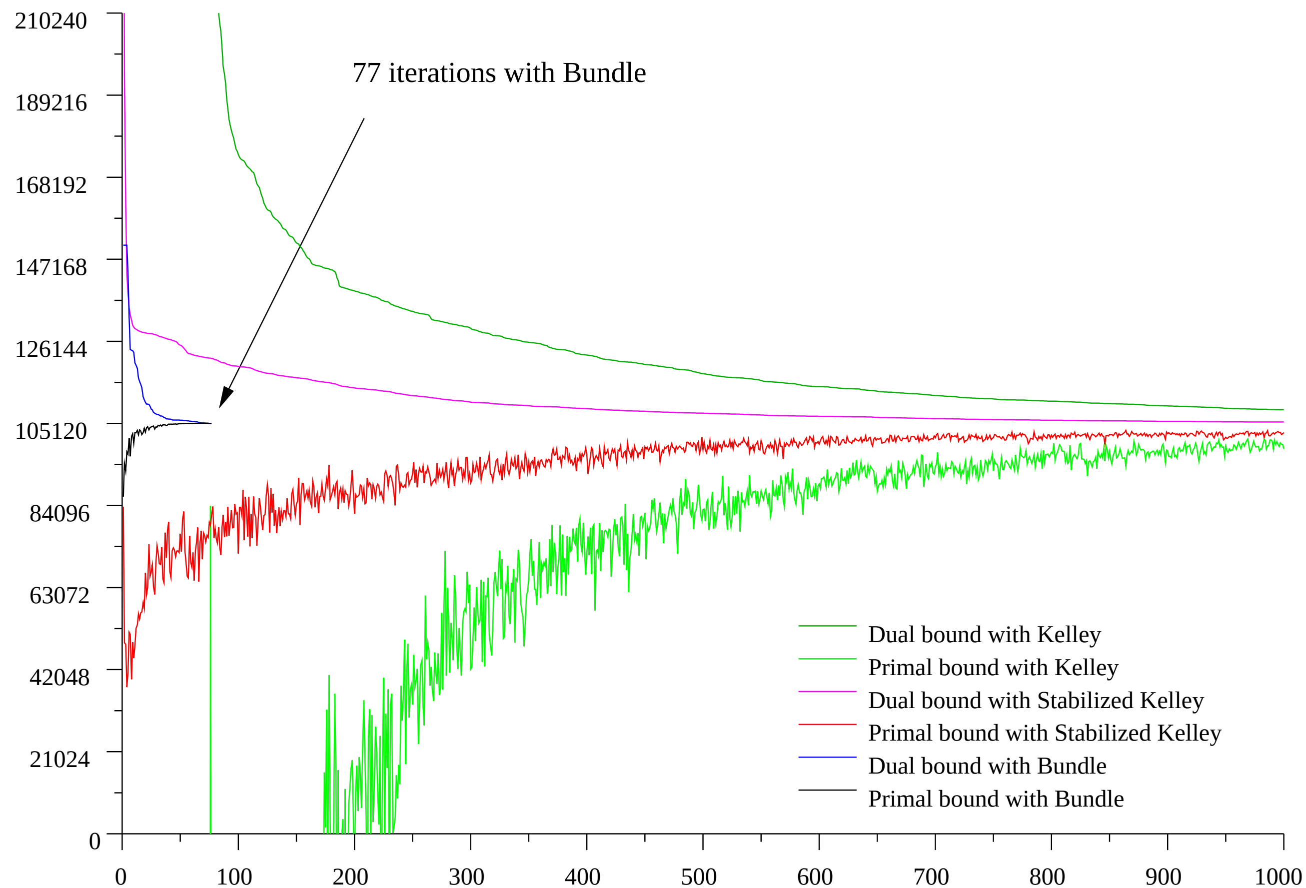
TSP : bays29 : primal and dual bounds



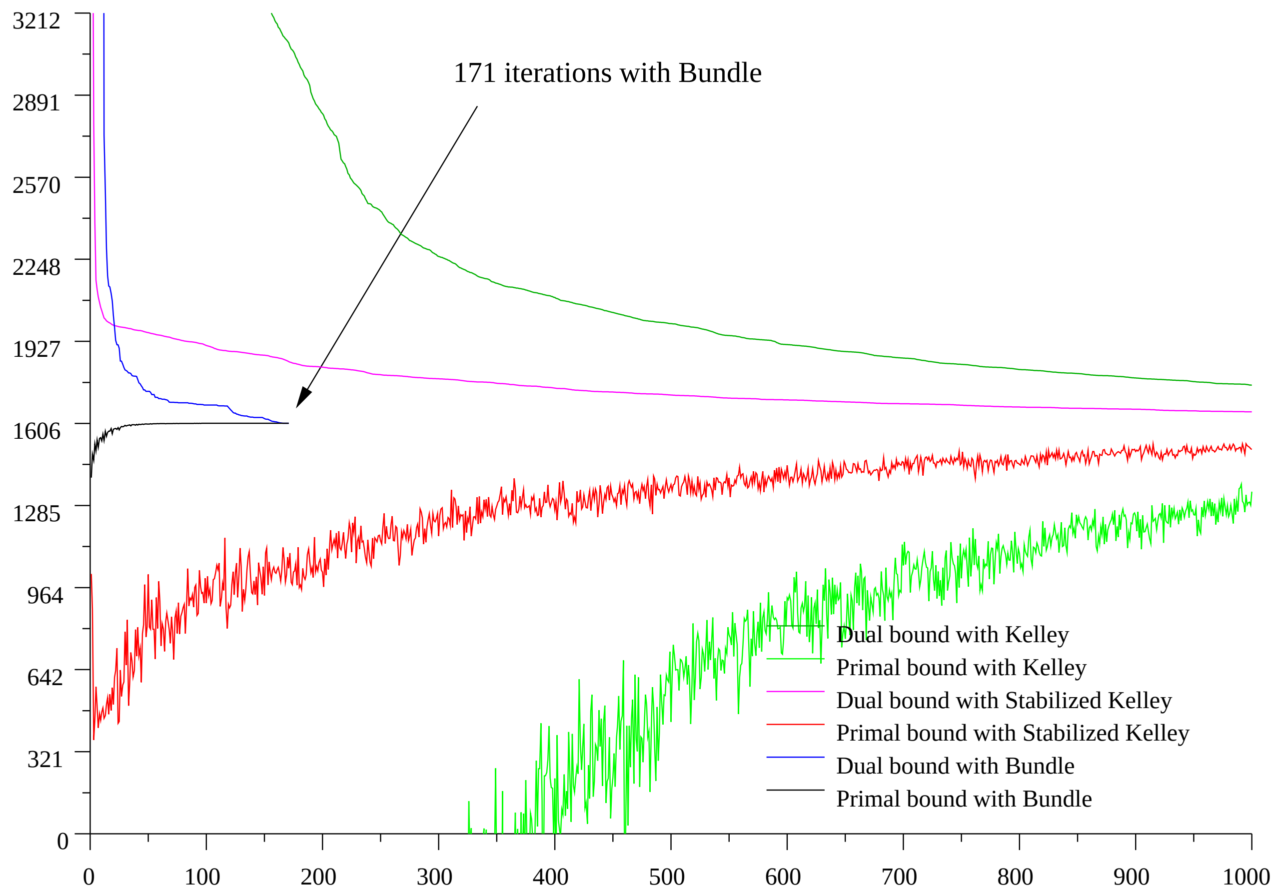
TSP : bays29 : primal and dual bounds



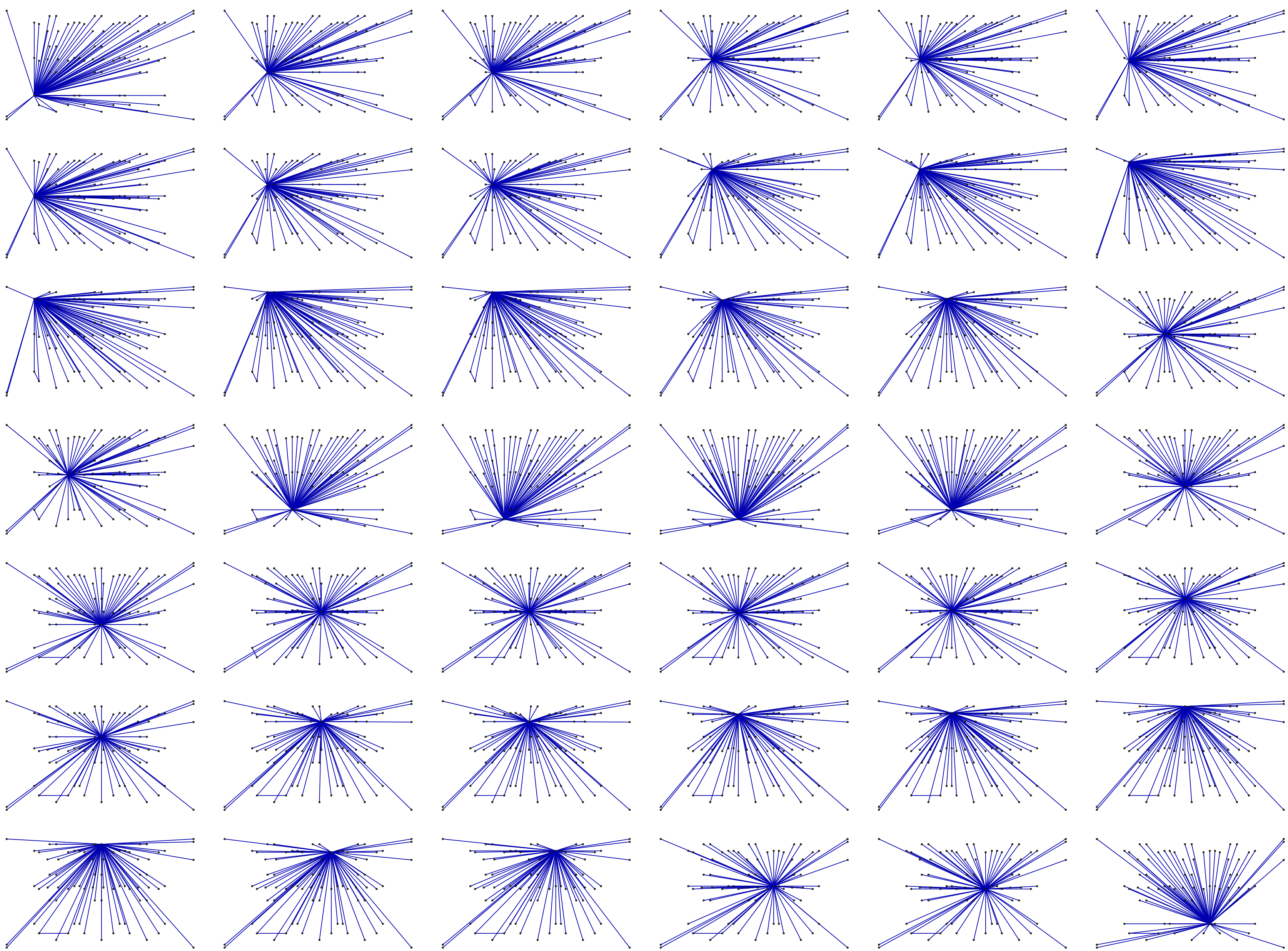
TSP : pr76 : primal and dual bounds



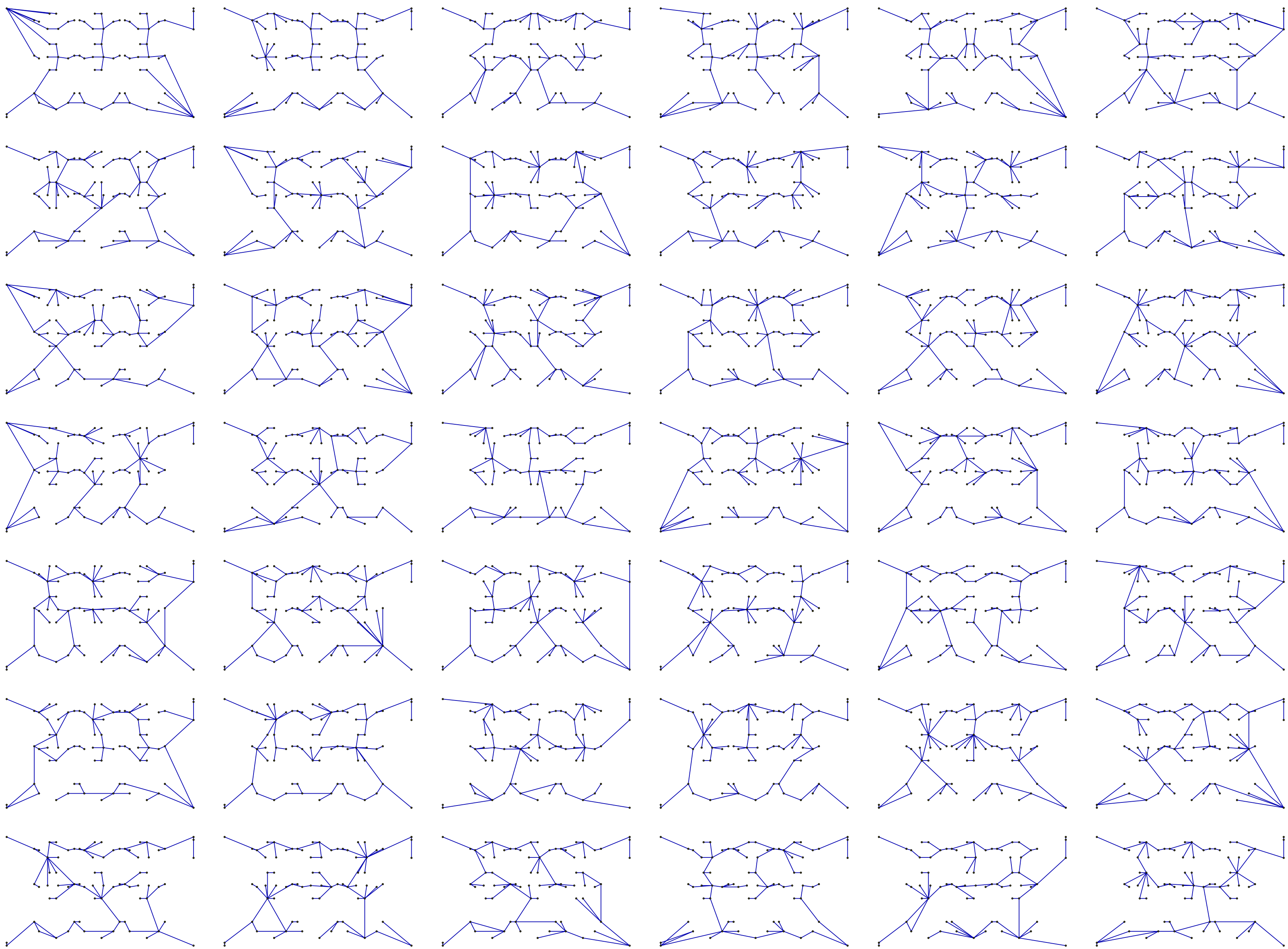
TSP : gr120 : primal and dual bounds



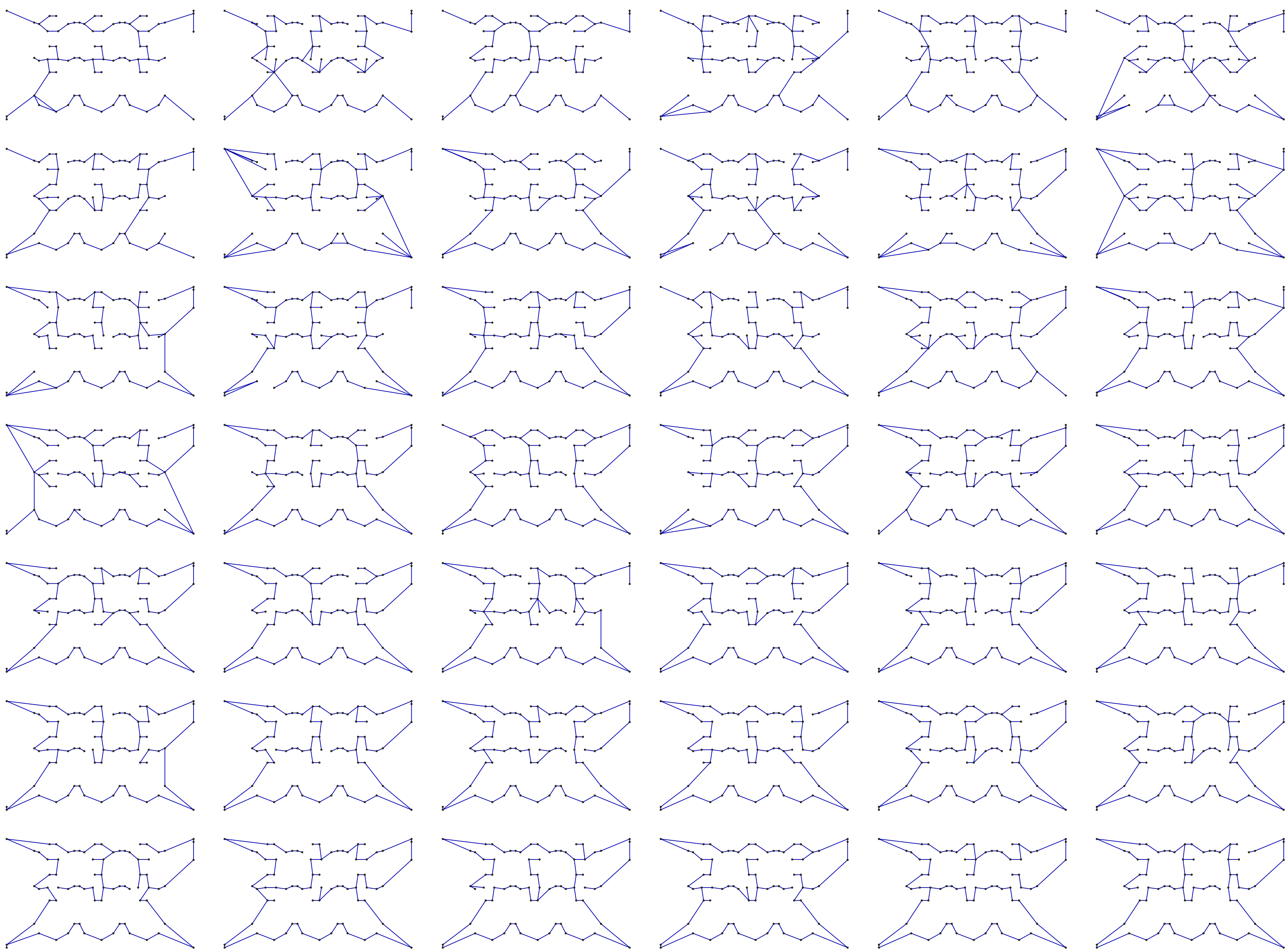
pr76 : first 1-trees with Kelley



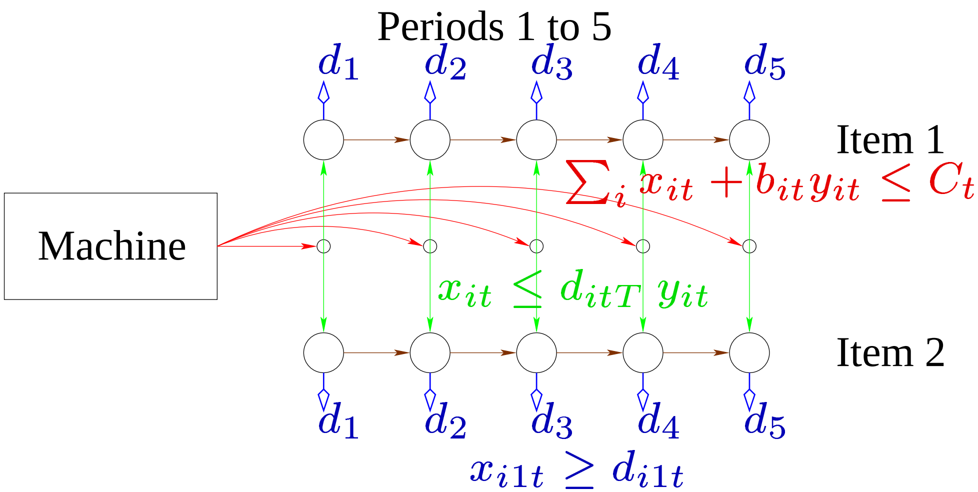
pr76 : first 1-trees with Stabilized Kelley



pr76 : first 1-trees with Bundle



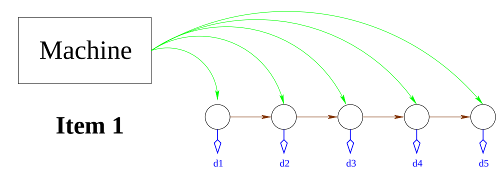
Multi-Item Lot-Sizing (MILS)



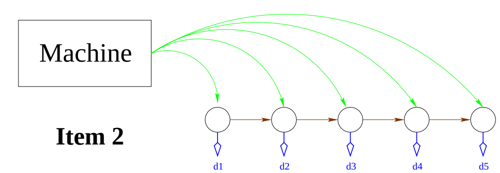
$$\equiv$$

$$\sum_i x_{it} + b_{it} y_{it} \leq C_t \quad \forall t$$

$$+$$



$$AND$$



MILS : Comparative results

average on 10 instances

		Counters			Timers (ticks)		
Problem	Method	Iter	SP	Col	Oracle	Master	total
i20-t60	kelley basic	9	164	117	60	3	5s08t
	kelley rich	15	244	80	86	7	4s45t
	bundle	34	672	75	326	610	15s73t

Observations

(based on preliminary results)

- Bundle takes far fewer iterations for some applications (like TSP), while not much is to win (compared to kelley) in other applications.

  ? Characterization ?

- Stabilization may imply harder oracles. Hence, there is a time tradeoff (warm start for the oracle).
- Kelley finds primal feasible solution earlier.
- Kelley accepts inexact oracles and multiple generation columns (to be implemented for Bundle).