Mixed-Integer programming approaches for some non-convex and combinatorial optimization problems.

Srikrishna Sridhar

Computer Sciences

University of Wisconsin-Madison

http://www.cs.wisc.edu/∼srikris/

Advised By

Jeff Linderoth, James Luedtke, and Stephen Wright

Committee

Jeff Linderoth, James Luedtke, Christopher Ré, Thomas Rutherford, and Stephen Wright

Life since 2009

- Part 1: Mixed integer programming (MIP) techniques with applications in:

Life since 2009

- Part 1: Mixed integer programming (MIP) techniques with applications in:
  - Extraction of natural resources like oil and natural gas.
  - Chemical processes design.
  - Compressors scheduling in petroleum reservoirs.
  - Hydro turbine performance modelling.

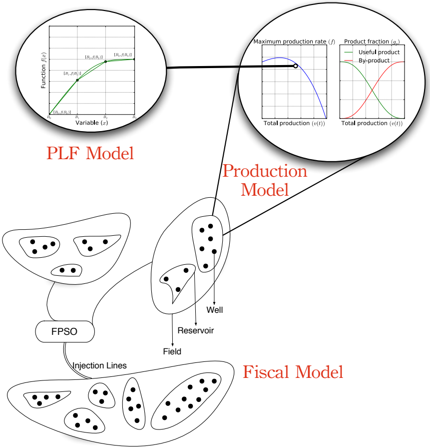
Life since 2009

- Part 1: Mixed integer programming (MIP) techniques with applications in:
  - Extraction of natural resources like oil and natural gas. Sample Application!
  - Chemical processes design.
  - Compressors scheduling in petroleum reservoirs.
  - Hydro turbine performance modelling.

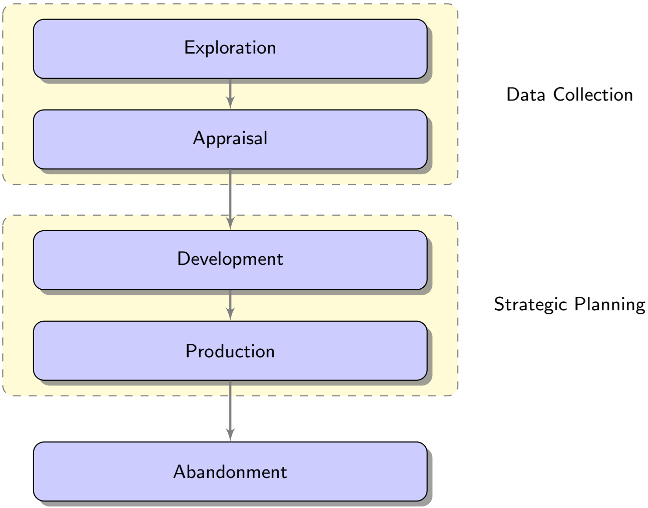
Life since 2009

- Part 1: Mixed integer programming (MIP) techniques with applications in:
  - Extraction of natural resources like oil and natural gas.
  - Chemical processes design.
  - Compressors scheduling in petroleum reservoirs.
  - Hydro turbine performance modelling.
- Part 2: Approximation algorithms for combinatorial problems using approximate LP Rounding.

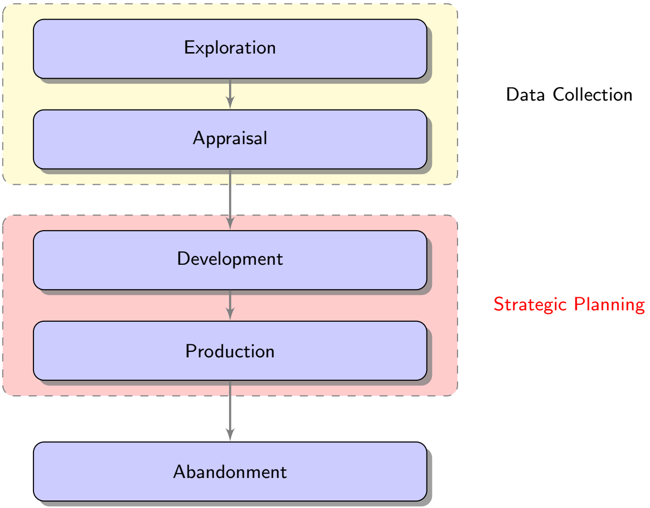
Part 1: MIP techniques for some non-convex problems.



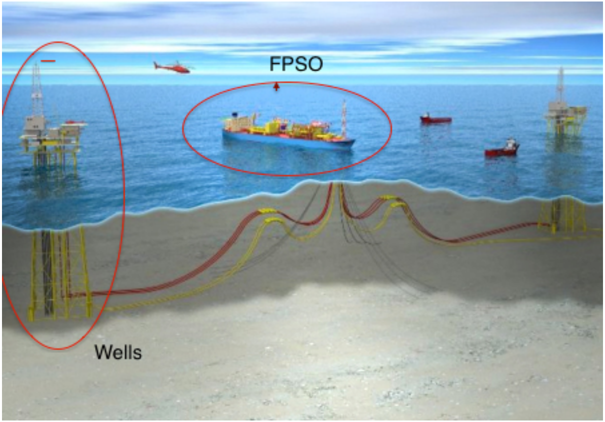
Oil Field Development Life Cycle



Oil Field Development Life Cycle

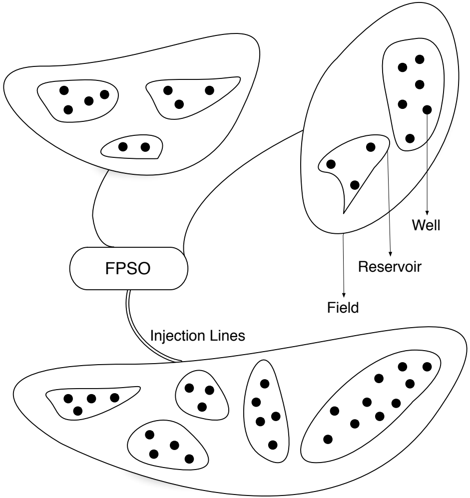


Oil Field Development Infrastructure[1]



[1] Statoil Peregrino Field

Oil Field Development Infrastructure[1]



[1] Tarhan, Grossmann, and Goel, Industrial & Engineering Chemistry Research (2009)

Why do we care?

FPSO development & installation. $\approx$ \$5B[1]

[1] Estimate: Do not start an oil company based on these estimates.

Why do we care?

	FPSO development & installation.	$\approx$ \$5B[1]
	Oil field development & installation.	$\approx$ \$1B[1]

[1] Estimate: Do not start an oil company based on these estimates.

Why do we care?

FPSO development & installation. $\approx \$5$B[1]

Oil field development & installation. $\approx \$1$B[1]

20 year operational expenses. $\approx \$5$B[1]

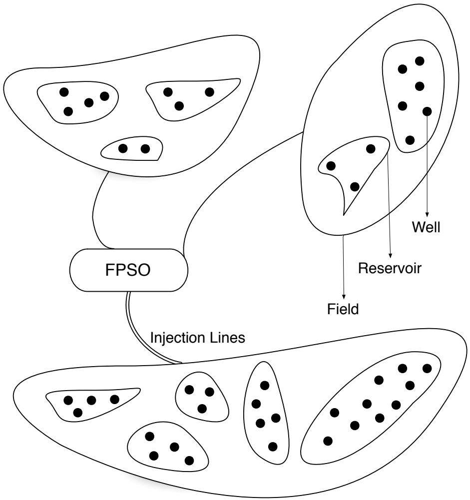
[1] Estimate: Do not start an oil company based on these estimates.

Why do we care?

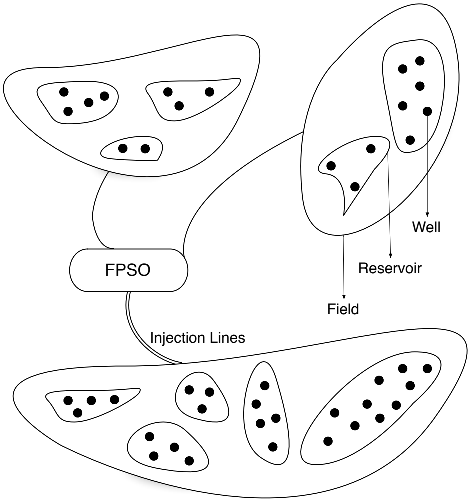
	FPSO development & installation.	≈ \$5B[1]
	Oil field development & installation.	≈ \$1B[1]
	20 year operational expenses.	≈ \$5B[1]
	Knowing optimal strategic & operational decisions for a 20 year horizon.	Priceless[2]

[1] Estimate: Do not start an oil company based on these estimates.
[2] Accurate: You may start a company based on this estimate.

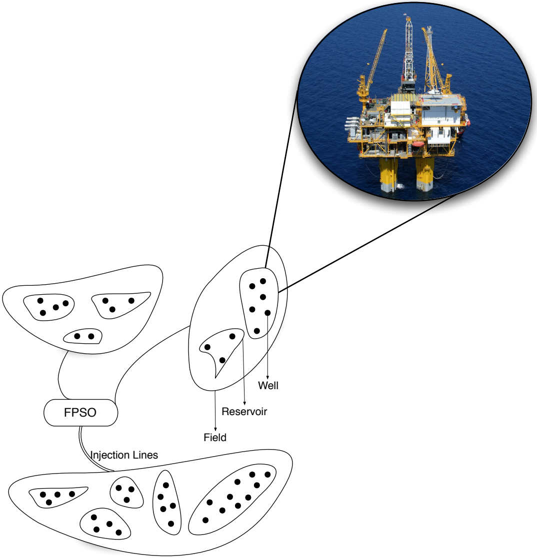
Supermodel...



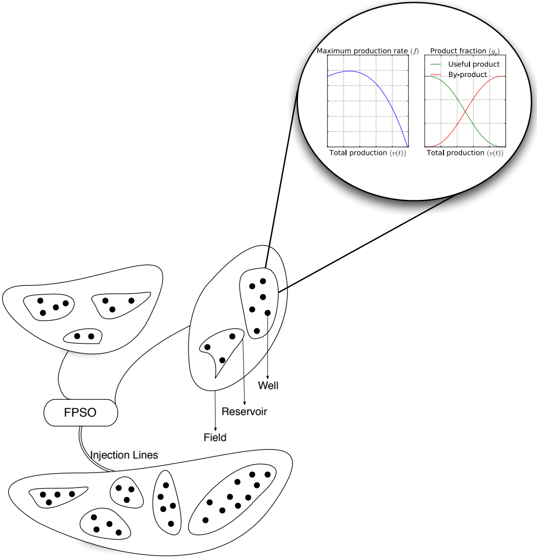
Dissected Supermodel...



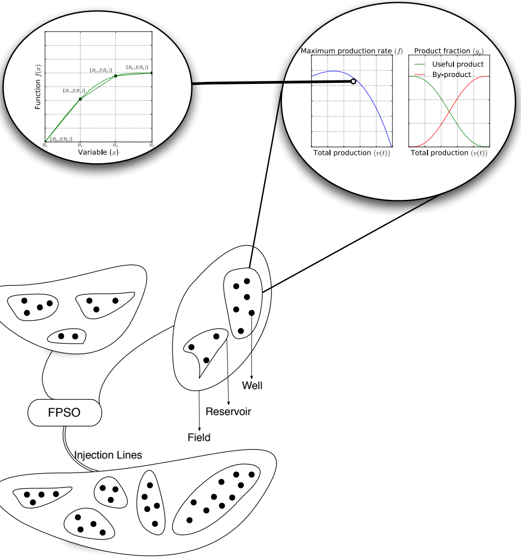
Dissected Supermodel...



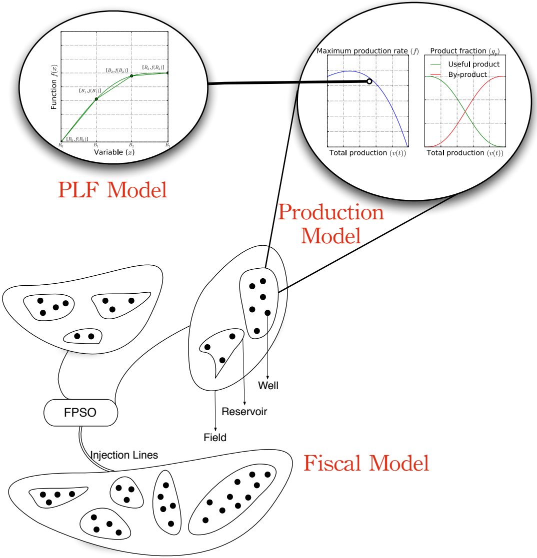
Dissected Supermodel...



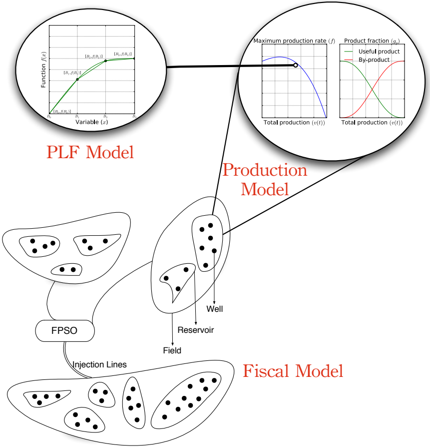
Dissected Supermodel...



Dissected Supermodel...



Key Challenges while Modelling Oil Field Infrastructure Planning



Contributions

- PLF Model: *Perfect* MIP models for piecewise linear functions (PLFs) with indicator variables.[1]

[1] Sridhar, Linderoth, and Luedtke Operations Research Letters (2013)

Contributions

- PLF Model: *Perfect* MIP models for piecewise linear functions (PLFs) with indicator variables.[1]

- Production Model: Convex reformulation of the production planning process to eliminate the bilinear terms from the MIP model. [2]

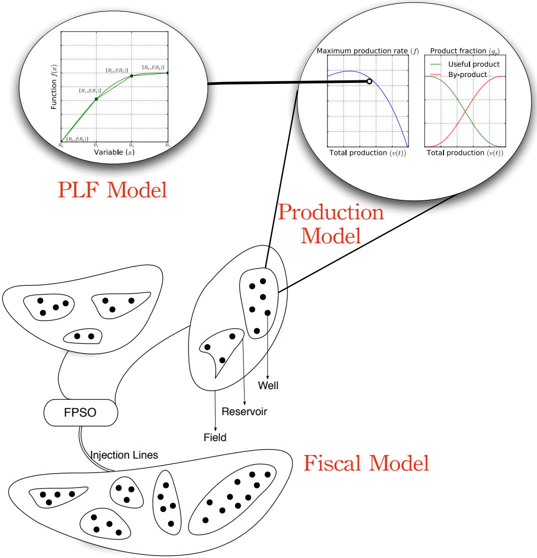
[1]Sridhar, Linderoth, and Luedtke Operations Research Letters (2013)
[2]Sridhar, Linderoth, and Luedtke Journal of Global Optimization (2014) – under review

Contributions

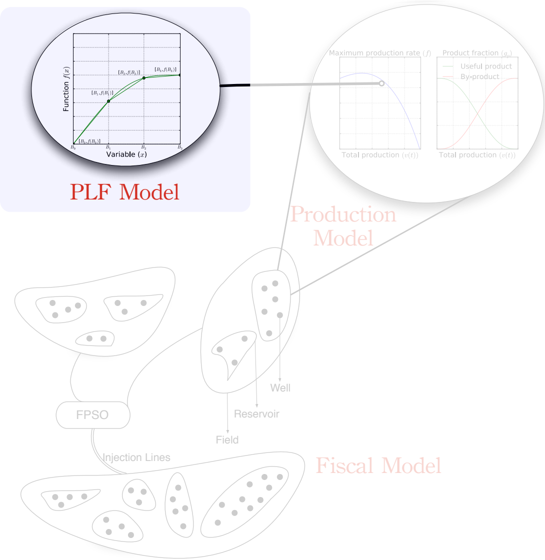
- PLF Model: *Perfect* MIP models for piecewise linear functions (PLFs) with indicator variables.[1]
- Production Model: Convex reformulation of the production planning process to eliminate the bilinear terms from the MIP model. [2]
- Fiscal Model: MIP models, solution techniques, and algorithms for production planning problems in the presence of complex fiscal objectives. [3]

[1] Sridhar, Linderoth, and Luedtke Operations Research Letters (2013)
[2] Sridhar, Linderoth, and Luedtke Journal of Global Optimization (2014) – under review
[2] Sridhar, Linderoth, Luedtke, and Wright – in preparation

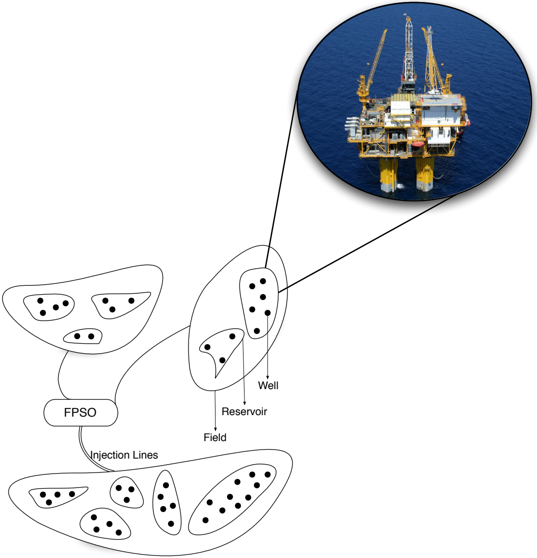
Challenge I: MIP Formulations for PLFs with Indicator Variables.



Challenge I: MIP Formulations for PLFs with Indicator Variables.



Challenge I: MIP Formulations for PLFs with Indicator Variables.

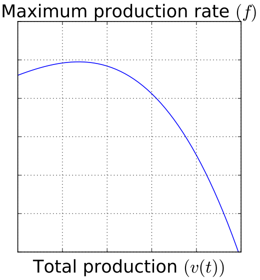


Challenge I: Nonlinear Production Functions

- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.

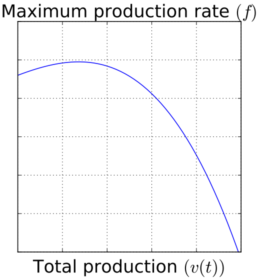
Challenge I: Nonlinear Production Functions

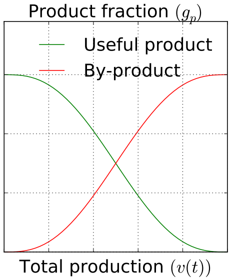
- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.
- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production $v(t)$.



Challenge I: Nonlinear Production Functions

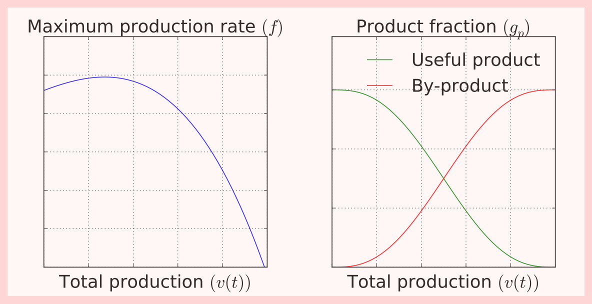
- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.
- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production $v(t)$.
- Product fraction functions $g_p(\cdot)$ evolve monotonically as a function of the total production $v(t)$.





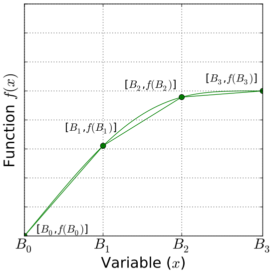
Challenge I: Nonlinear Production Functions

- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.
- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production $v(t)$.
- Product fraction functions $g_p(\cdot)$ evolve monotonically as a function of the total production $v(t)$.



Challenge I: Nonlinear Production Functions

Contribution: Piecewise linear functions (PLFs) with indicator variables to tackle nonlinearity![1]



[1] Sridhar, Linderoth, and Luedtke Operations Research Letters (2013)

Challenge I: MIP Formulations for PLFs with Indicator Variables

MIP formulations for PLFs that are evaluated when an indicator variable (z) is turned on.

$$\underbrace{z = 0}_{\text{Binary variable}} \quad \Rightarrow \quad \underbrace{x = 0}_{\text{Function argument}}, \; \underbrace{f(x) = 0}_{\text{PLF}}.$$

Challenge I: MIP Formulations for PLFs with Indicator Variables

MIP formulations for PLFs that are evaluated when an indicator variable (z) is turned on.

$$\underbrace{z = 0}_{\text{Binary variable}} \quad \Rightarrow \quad \underbrace{x = 0}_{\text{Function argument}} \quad , \; \underbrace{f(x) = 0}_{\text{PLF}} .$$

MIP formulations for PLFs that are evaluated when an indicator variable (z) is turned on.

$$\underbrace{z = 0}_{\text{Binary variable}} \quad \Rightarrow \quad \underbrace{x = 0}_{\text{Function argument}} \quad , \quad \underbrace{f(x) = 0}_{\text{PLF}} .$$

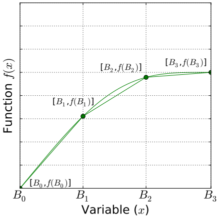
Application	Reference
Gas network optimization	Martin et al. (2006)
Transmissions expansion planning	Alguacil et al. (2003)
Thermal unit commitment	Carrion et al. (2006)
Oil field development	Gupta et al. (2012)
Hydro Scheduling	Borghetti et al. (2008)
Sales resource allocation	Lodish et al. (1971)

Challenge I: MIP Formulations for PLFs with Indicator Variables

MIP formulations for PLFs that are evaluated when an indicator variable (z) is turned on.

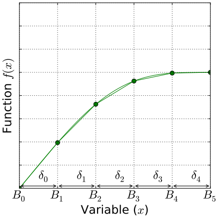
$$\underbrace{z = 0}_{\text{Binary variable}} \quad \Rightarrow \quad \underbrace{x = 0}_{\text{Function argument}} \quad , \; \underbrace{f(x) = 0}_{\text{PLF}}.$$

SOS2 Model



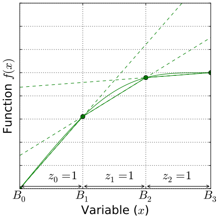
Beale and Tomlin (1970)

Incremental Model



Markowitz and Manne (1957)

Multiple Choice



Balakrishnan and Graves (1989)

Challenge I: MIP Formulations for PLFs with Indicator Variables

MIP formulations for PLFs that are evaluated when an indicator variable (z) is turned on.

$$\underbrace{z = 0}_{\text{Binary variable}} \quad \Rightarrow \quad \underbrace{x = 0}_{\text{Function argument}}, \ \underbrace{f(x) = 0}_{\text{PLF}}.$$

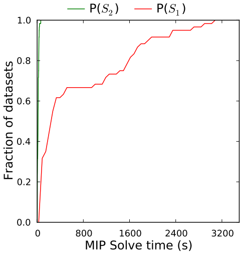
Theoretical Results[1]

- Our proposed formulations is locally ideal.
- Previously proposed formulations are not locally ideal. (counter example)

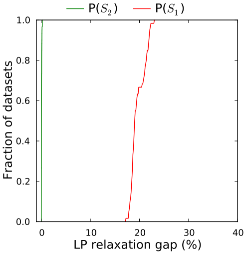
[1] Sridhar, Linderoth, and Luedtke Operations Research Letters (2013)

Challenge I: Numerical Experiments

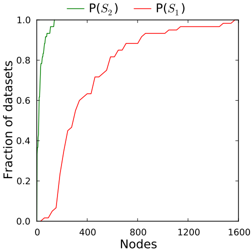
Solve time



Root LP gap



Nodes explored

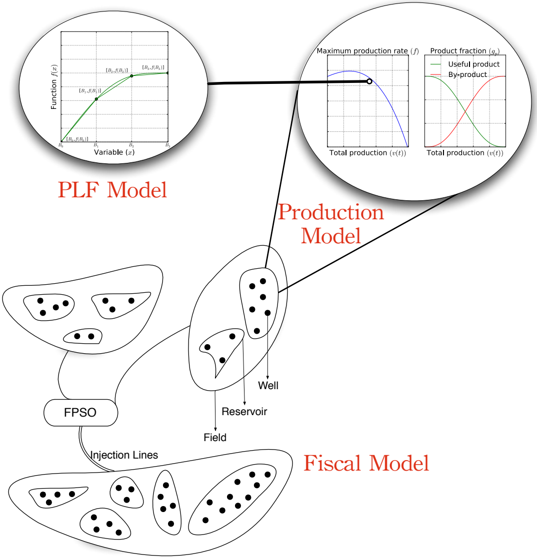


Average Performance Improvement[1]

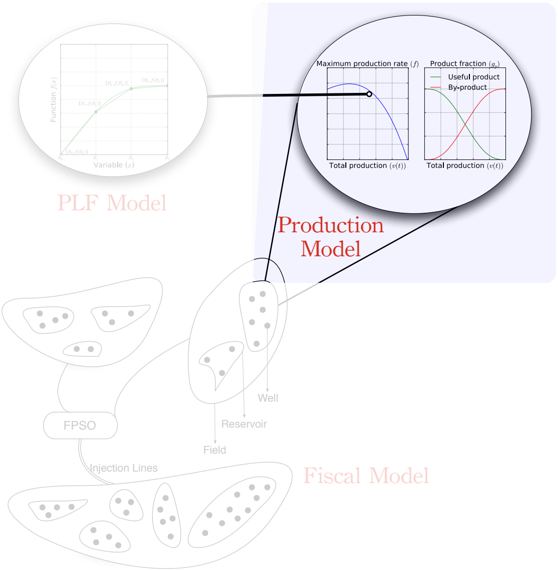
- 40x faster solve times.
- 15x fewer nodes explored.
- 20% better root node gaps.

[1] Sridhar, Linderoth, and Luedtke Operations Research Letters (2013)

Challenge II: Bilinear Terms in Production Process



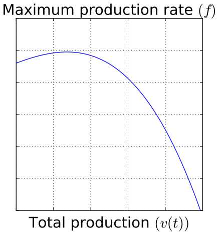
Challenge II: Bilinear Terms in Production Process

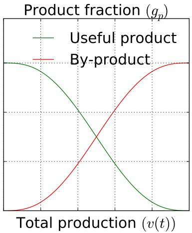


Challenge II: Bilinear Terms in Production Process

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$





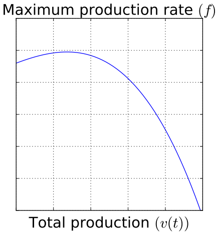
Challenge II: Bilinear Terms in Production Process

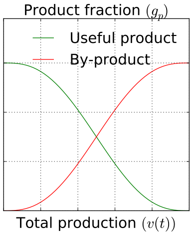
Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

Production rate is limited by a production function $f(\cdot)$

$$x(t) \leq f(v(t))$$





Challenge II: Bilinear Terms in Production Process

Cumulative production $v(t)$ is calculated using production rate x(t)

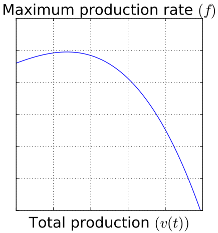
$$v(t) = \int_0^t x(s)\mathrm{d}s$$

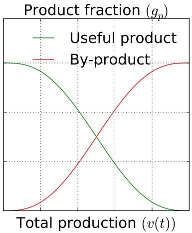
Production rate is limited by a production function $f(\cdot)$

$$x(t) \leq f(v(t))$$

Product production rates $y_p(t)$ calculated by fraction functions $g_p(\cdot)$

$$y_p(t) = \; x(t)\; g_p(v(t))$$





Challenge II: Bilinear Terms in Production Process

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

Production rate is limited by a production function $f(\cdot)$

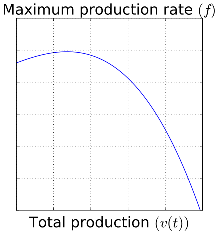
$$x(t) \leq f(v(t))$$

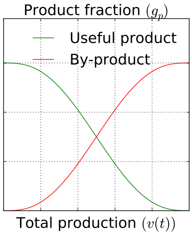
Product production rates $y_p(t)$ calculated by fraction functions $g_p(\cdot)$

$$y_p(t) = \ x(t)\ g_p(v(t))$$

Production profiles are active only after the start time $z(t)$

$$z(t) = 0 \Rightarrow v(t) = 0$$





Challenge II: Bilinear Terms in Production Process

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

Mixture production rate is limited by production function $f(\cdot)$

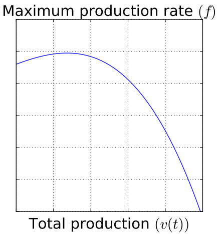
$$x(t) \leq f(v(t))$$

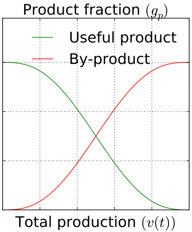
Product production rates $y_p(t)$ calculated by fraction functions$g_p(\cdot)$

$$y_p(t) = \underbrace{x(t)\ g_p(v(t))}_{\text{non-convex bilinear terms}}$$

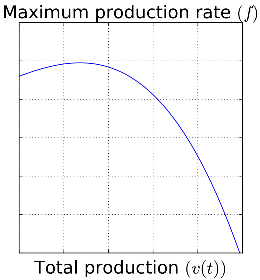
Production profiles are active only after the time $z(t)$

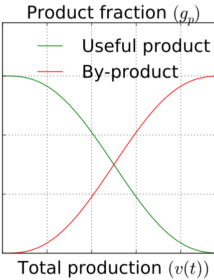
$$z(t) = 0 \Rightarrow v(t) = 0$$





Production Functions: Applications

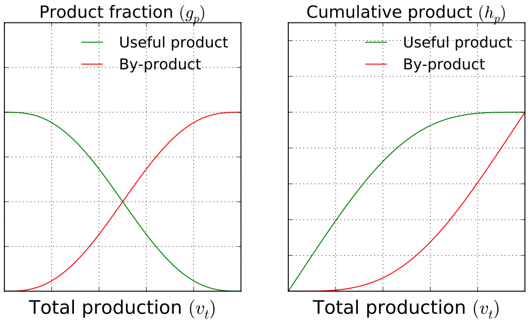




Application	Reference
Oil & Natural gas	Iyer et al. (1998) Tarhan et al. (2009)
Gas network optimization	Martin et al. (2006)
Hydro Scheduling	Borghetti et al. (2008)
Compressor Scheduling	Camponogara et al. (2011)

Main Results[1]

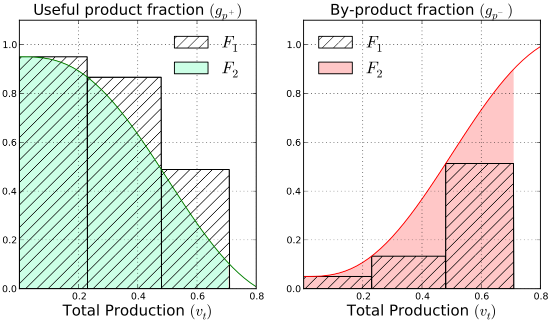
- Reformulate based on cumulative product production!



[1] Sridhar, Linderoth, and Luedtke Journal of Global Optimization (2014) – under review

Main Results[1]

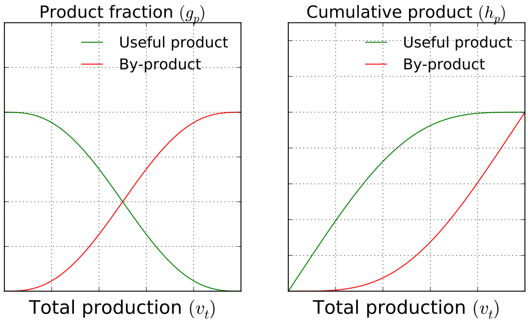
- Reformulate based on cumulative product production!
- Up to 30% more accurate than Tarhan et al. (2009).



[1] Sridhar, Linderoth, and Luedtke Journal of Global Optimization (2014) – under review

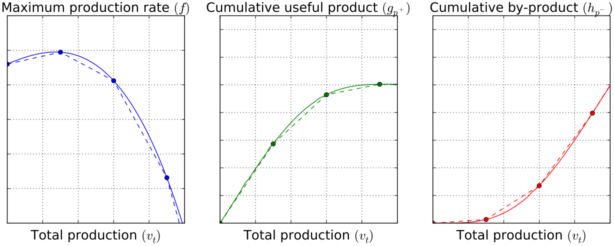
Main Results[1]

- Reformulate based on cumulative product production!
- Up to 30% more accurate than Tarhan et al. (2009).
- Order of magnitude faster because it deals with convex functions while Tarhan et. al (2009) deals with bilinear terms.



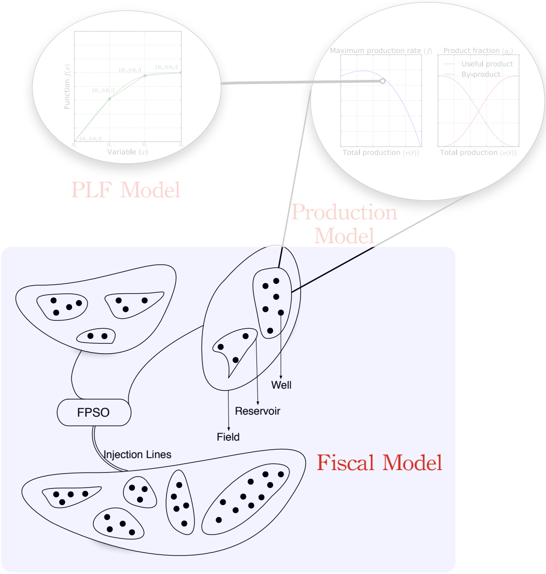
[1] Sridhar, Linderoth, and Luedtke Journal of Global Optimization (2014) – under review

Piecewise Linear Approximation (PLA)

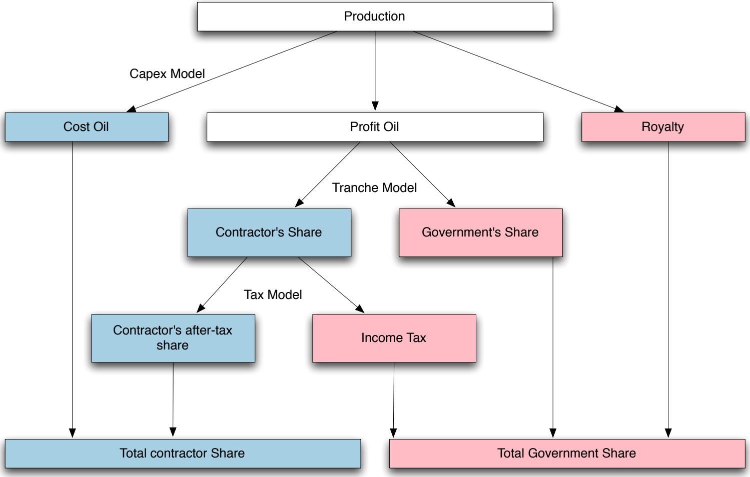


Approximate all the nonlinear production functions using PLFs.

Challenge III: Production Planning Problems with Complex Fiscal Terms.

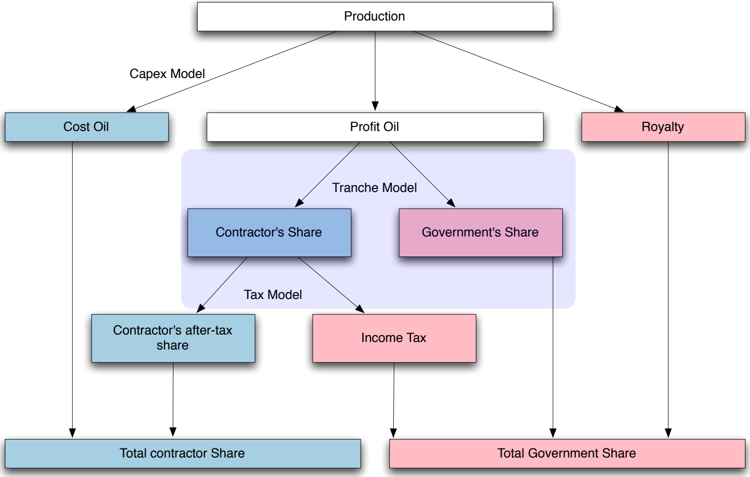


Challenge III: non-convex Objective Function[1]



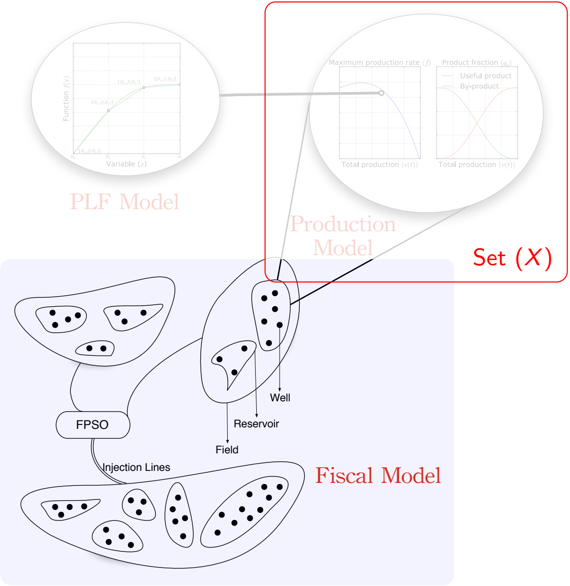
[1] Image source: World Bank Survey

Challenge III: non-convex Objective Function[1]



[1] Image source: World Bank Survey

Dissected Supermodel...



Problem Setup

- Multi period planning problem X with $\mathcal{T} := \{1, 2 \ldots T\}$ time periods.

Problem Setup

- Multi period planning problem X with $\mathcal{T} := \{1, 2 \ldots T\}$ time periods.
- Continuous operational decision variables $x \in \mathbb{R}^m \times \{0,1\}^n$ which produce a set of cash flows for each time period f_t.

Problem Setup

- Multi period planning problem X with $\mathcal{T} := \{1, 2 \dots T\}$ time periods.
- Continuous operational decision variables $x \in \mathbb{R}^m \times \{0,1\}^n$ which produce a set of cash flows for each time period f_t.
- The cash flows are broken down into revenue streams $r_t \quad \forall t \in \mathcal{T}$ and expense streams $c_t \quad \forall t \in \mathcal{T}$.

$$f_t = r_t - c_t \quad \forall t \in \mathcal{T}$$

Problem Setup

- Multi period planning problem X with $\mathcal{T} := \{1, 2 \ldots T\}$ time periods.
- Continuous operational decision variables $x \in \mathbb{R}^m \times \{0,1\}^n$ which produce a set of cash flows for each time period f_t.
- The cash flows are broken down into revenue streams $r_t \quad \forall t \in \mathcal{T}$ and expense streams $c_t \quad \forall t \in \mathcal{T}$.

$$f_t = r_t - c_t \quad \forall t \in \mathcal{T}$$

Complex fiscal terms like can be viewed as optimizing a discontinuous, non-convex function $G : \mathbb{R}^T \times \mathbb{R}^T \to \mathbb{R}$.

$$\max \quad \underbrace{G(r,c)}_{\text{Noncovex function}} \qquad \text{subject to} \qquad \underbrace{(r,c,x) \in X}_{\text{MIP Production Model}}$$

Net Present Value (NPV)

Given a time series of cash flows ($f_{[1,t]}$) and rate of interest ($\hat{q}$)

$$h(\hat{q}, f_{[1,t]}) = \underbrace{\sum_{s=1}^{t} \frac{f_s}{(1+\hat{q})^s}}_{\text{Present value of money}}$$

Net Present Value (NPV)

Given a time series of cash flows ($f_{[1,t]}$) and rate of interest ($\hat{q}$)

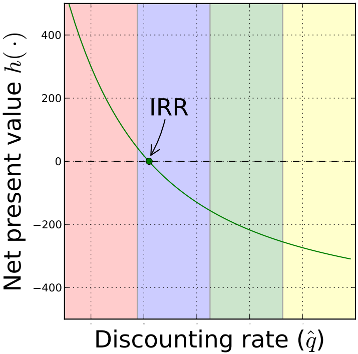
$$h(\hat{q}, f_{[1,t]}) = \underbrace{\sum_{s=1}^{t} \frac{f_s}{(1+\hat{q})^s}}_{\text{Present value of money}}$$

Internal rate of return (IRR)

Rate of return for which the NPV function is zero.

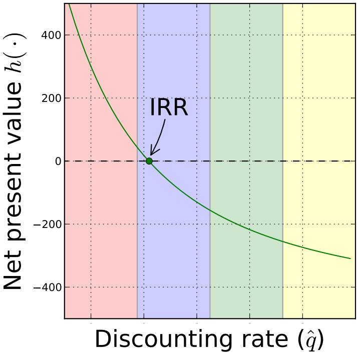
$$\underbrace{h(q_t, f_{[1,t]}) = 0}_{\text{Solve in each time period}}$$

Illustration: Production Sharing Contracts



Production Sharing Contracts (PSC)

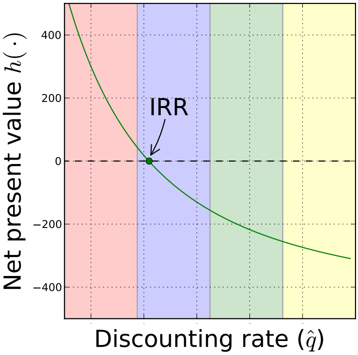
- IRR scale into tranches ($\mathcal{K} := \{1, 2 \ldots K\}$)



Production Sharing Contracts (PSC)

- IRR scale into tranches ($\mathcal{K} := \{1, 2 \dots K\}$)
- Contractors to retain μ_k fraction of the profit during time period t, if associated with tranche k.

Illustration: Production Sharing Contracts



Production Sharing Contracts (PSC)

- IRR scale into tranches ($\mathcal{K} := \{1, 2 \ldots K\}$)
- Contractors to retain μ_k fraction of the profit during time period t, if associated with tranche k.

Applications

- Oil & natural gas field infrastructure planning.
- Portfolio optimization.
- Production planning.

Example PSC

Time (t)	Cost (c_t)	Revenue (p_t)	IRR (q_t)	Tranche ($k(t)$)	Contractor share ($\mu_{k(t)}$)	Cash flow ($\mu_{k(t)} r_t - c_t$)
1	4000	0	n.a	1	70%	-4000
2	0	2300	-49.8%	1	70%	1610
3	0	6000	20%	1	70%	4200
4	0	5000	46.7%	2	60%	3000
5	0	4000	49.0%	3	15%	600

Example PSC

Time (t)	Cost (c_t)	Revenue (p_t)	IRR (q_t)	Tranche ($k(t)$)	Contractor share ($\mu_{k(t)}$)	Cash flow ($\mu_{k(t)} r_t - c_t$)
1	4000	0	n.a	1	70%	-4000
2	0	2300	-49.8%	1	70%	1610
3	0	6000	20%	1	70%	4200
4	0	5000	46.7%	2	60%	3000
5	0	4000	49.0%	3	15%	600

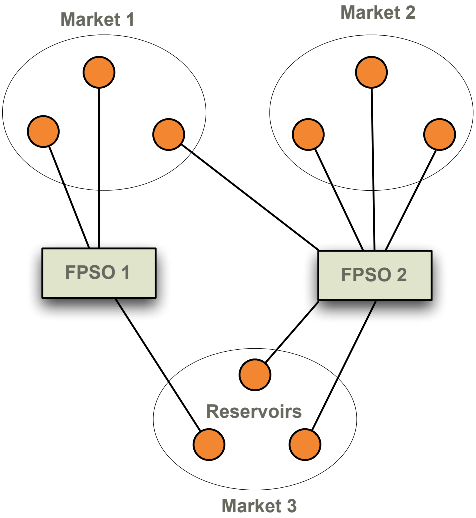
Example PSC

Time (t)	Cost (c_t)	Revenue (p_t)	IRR (q_t)	Tranche ($k(t)$)	Contractor share ($\mu_{k(t)}$)	Cash flow ($\mu_{k(t)} r_t - c_t$)
1	4000	0	n.a	1	70%	-4000
2	0	2300	-49.8%	1	70%	1610
3	0	6000	20%	1	70%	4200
4	0	5000	46.7%	2	60%	3000
5	0	4000	49.0%	3	15%	600

Example PSC

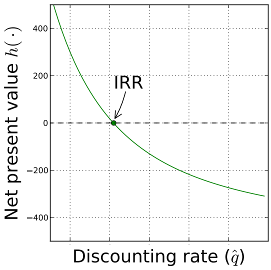
Time (t)	Cost (c_t)	Revenue (p_t)	IRR (q_t)	Tranche ($k(t)$)	Contractor share ($\mu_{k(t)}$)	Cash flow ($\mu_{k(t)} r_t - c_t$)
1	4000	0	n.a	1	70%	-4000
2	0	2300	-49.8%	1	70%	1610
3	0	6000	20%	1	70%	4200
4	0	5000	46.7%	2	60%	3000
5	0	4000	49.0%	3	15%	600

Markets (Ringfences)



Solution Restrictions

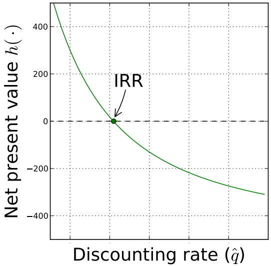
1. Given a series of cash flows $f_{[1,t]}$, the equation $h(\hat{q}, f_{[1,t]}) = 0$ always has at most one root.[1]

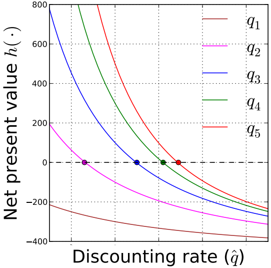


[1] Sufficient conditions: Single sign change test & the Norstrøm condition (1972).

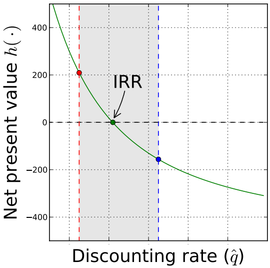
Solution Restrictions

1. Given a series of cash flows $f_{[1,t]}$, the equation $h(\hat{q}, f_{[1,t]}) = 0$ always has at most one root.[1]
2. The sequence of tranches associated with each time period is non-decreasing.[2]





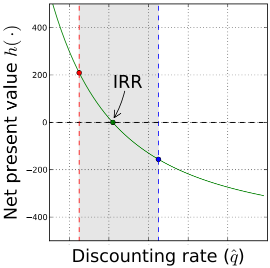
[1] Sufficient conditions: Single sign change test & the Norstrøm condition (1972).
[2] Sufficient conditions: Single sign change test.



NPV

$$\boxed{h(\hat{q}, f_{[1,t]})} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\hat{q})^s}$$

Tranche Model: Key Idea



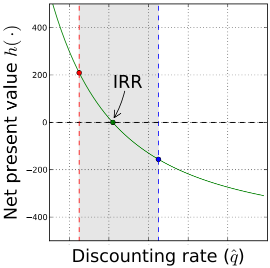
NPV

$$\boxed{h(\hat{q}, f_{[1,t]})} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\hat{q})^s}$$

$$h(\boxed{q_t}, f_{[1,t]}) = 0$$

IRR

Tranche Model: Key Idea



NPV

$$\boxed{h(\hat{q}, f_{[1,t]})} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\hat{q})^s}$$

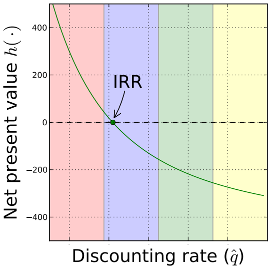
$$h(q_t, f_{[1,t]}) = 0$$

IRR

$\boxed{q_t \in [l, u]}$ if $h(l, f_{[1,t]}) \geq 0$ and $h(u, f_{[1,t]}) < 0$

IRR between [l,u]

Tranche Model: Key Idea



NPV

$$\boxed{h(\hat{q}, f_{[1,t]})} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\hat{q})^s}$$

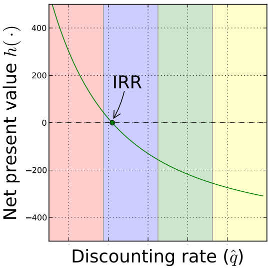
$$h(\boxed{q_t}, f_{[1,t]}) = 0$$

IRR

$$\boxed{q_t \in [l, u]} \text{ if } h(l, f_{[1,t]}) \geq 0 \text{ and } h(u, f_{[1,t]}) < 0$$

IRR between [l,u]

Tranche Model: MIP Formulation



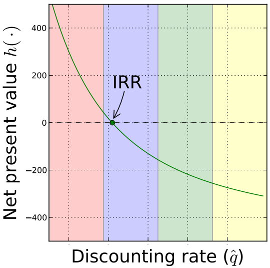
NPV at λ_k

$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



NPV at λ_k

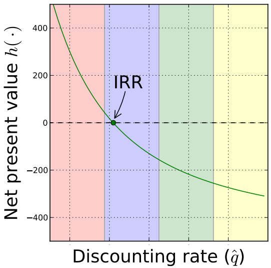
$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$b_{a,k,t} = \begin{cases} 0 & h_{a,k,t-1} \leq 0 \\ 1 & \text{otherwise} \end{cases}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



NPV at λ_k

$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$b_{a,k,t} = \begin{cases} 0 & h_{a,k,t-1} \leq 0 \\ 1 & \text{otherwise} \end{cases}$$

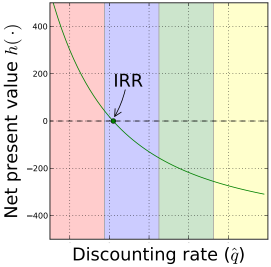
$$\boxed{f_{a,t}} = \begin{cases} \mu_k r_{a,t} - c_{a,t} & b_{a,k,t} = 1,\ b_{a,k+1,t} = 0 \\ \mu_1 r_{a,t} - c_{a,t} & \text{otherwise} \end{cases}$$

Cash flow retained by contractor

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



NPV at λ_k

$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$b_{a,k,t} = \begin{cases} 0 & h_{a,k,t-1} \leq 0 \\ 1 & \text{otherwise} \end{cases}$$

$$\boxed{f_{a,t}} = \begin{cases} \mu_k r_{a,t} - c_{a,t} & b_{a,k,t} = 1,\ b_{a,k+1,t} = 0 \\ \mu_1 r_{a,t} - c_{a,t} & \text{otherwise} \end{cases}$$

Cash flow retained by contractor

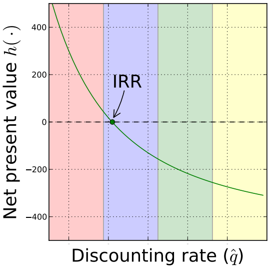
$$(r, c, x) \quad \in \boxed{X}$$

Production model

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



NPV at λ_k

$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$\boxed{b_{a,k,t} = \begin{cases} 0 & h_{a,k,t-1} \leq 0 \\ 1 & \text{otherwise} \end{cases}}$$

$$\boxed{f_{a,t}} = \begin{cases} \mu_k r_{a,t} - c_{a,t} & b_{a,k,t} = 1,\ b_{a,k+1,t} = 0 \\ \mu_1 r_{a,t} - c_{a,t} & \text{otherwise} \end{cases}$$

Cash flow retained by contractor

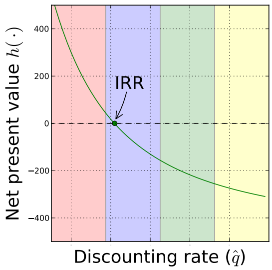
$$(r, c, x) \quad \in \boxed{X}$$

Production model

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



NPV at λ_k

$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$\boxed{\underbrace{\mathrm{m}^h_{a,k,t-1}}_{\text{Min NPV}} (1 - b_{a,k,t}) \leq h_{a,k,t-1} \leq \underbrace{\mathrm{M}^h_{a,k,t-1}}_{\text{Max NPV}} b_{a,k,t}}$$

$$\boxed{f_{a,t}} = \begin{cases} \mu_k r_{a,t} - c_{a,t} & b_{a,k,t} = 1,\ b_{a,k+1,t} = 0 \\ \mu_1 r_{a,t} - c_{a,t} & \text{otherwise} \end{cases}$$

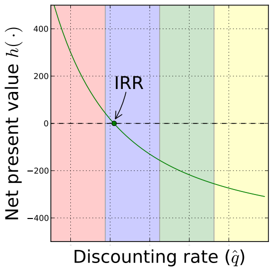
Cash flow retained by contractor

$$(r, c, x) \quad \in \boxed{X}$$ Production model

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



NPV at λ_k

$$\boxed{h_{a,k,t}} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$\underbrace{\mathrm{m}^h_{a,k,t-1}}_{Min\ \ NPV} (1 - b_{a,k,t}) \leq h_{a,k,t-1} \leq \underbrace{\mathrm{M}^h_{a,k,t-1}}_{Max\ \ NPV} b_{a,k,t}$$

$$\boxed{f_{a,t}} = \begin{cases} \mu_k r_{a,t} - c_{a,t} & b_{a,k,t} = 1,\ b_{a,k+1,t} = 0 \\ \mu_1 r_{a,t} - c_{a,t} & \text{otherwise} \end{cases}$$

Cash flow retained by contractor

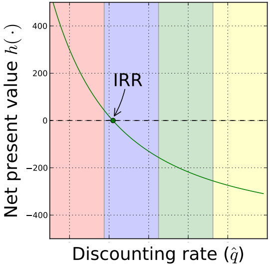
$$(r, c, x) \quad \in \boxed{X}$$

Production model

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



$$h_{a,k,t} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$\underbrace{\mathrm{m}^h_{a,k,t-1}}_{Min\ \ NPV}(1 - b_{a,k,t}) \leq h_{a,k,t-1} \leq \underbrace{\mathrm{M}^h_{a,k,t-1}}_{Max\ \ NPV} b_{a,k,t}$$

$$p_{a,k,t} \leq \underbrace{\mathrm{M}^p_{a,k,t}}_{Max\ \ Profit} (b_{a,k,t} - b_{a,k+1,t})$$

$$r_{a,t} = \sum_{k \in \mathcal{K}} \mu_k p_{a,k,t}$$

$$b_{a,k,t} \geq b_{a,k+1,t}$$

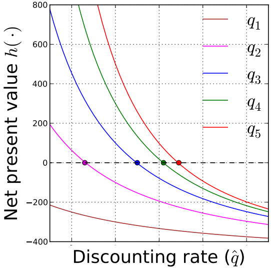
$$f_{a,t} = r_{a,t} - c_{a,t}$$

$$(r, c, x) \in X$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Tranche Model: MIP Formulation



$$h_{a,k,t} = \sum_{s=1}^{t} \frac{f_{a,s}}{(1+\lambda_k)^s}$$

$$\underbrace{\mathrm{m}^h_{a,k,t-1}}_{\textit{Min NPV}}(1 - b_{a,k,t}) \leq h_{a,k,t-1} \leq \underbrace{\mathrm{M}^h_{a,k,t-1}}_{\textit{Max NPV}} b_{a,k,t}$$

$$p_{a,k,t} \leq \underbrace{\mathrm{M}^p_{a,k,t}}_{\textit{Max Profit}} (b_{a,k,t} - b_{a,k+1,t})$$

$$r_{a,t} = \sum_{k \in \mathcal{K}} \mu_k p_{a,k,t}$$

$$b_{a,k,t} \geq b_{a,k+1,t}$$

$$f_{a,t} = r_{a,t} - c_{a,t}$$

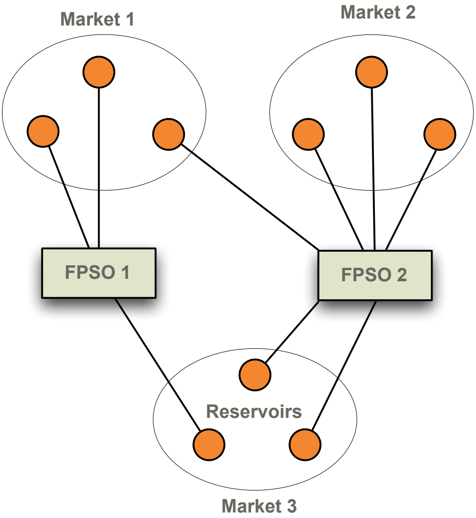
$$(r, c, x) \in X$$

$$\boxed{b_{a,k,t} \geq b_{a,k,t+1}}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Markets are Everything!



Markets are hard![1]

Markets	# Variables (binary)	# Constraints	Solve time (s)	Gap (%)	Nodes
1	15132 (1457)	22101	680.39	-	1323
2	15263 (1514)	22401	1117.4	-	2934
3	15389 (1568)	22701	3670.3	-	12136
4	15515 (1622)	23001	-	8.24	28208
5	15646 (1679)	23301	-	34.4	22381
6	15762 (1727)	23601	-	66.2	10515

[1] Solution statistics for for a single instance of the MIP for a sample application problem solved using Gurobi 5.0.1 with 2 threads for 7200 seconds.

Markets are hard![1]

Markets	# Variables (binary)	# Constraints	Solve time (s)	Gap (%)	Nodes
1	15132 (1457)	22101	680.39	-	1323
2	15263 (1514)	22401	1117.4	-	2934
3	15389 (1568)	22701	3670.3	-	12136
4	15515 (1622)	23001	-	8.24	28208
5	15646 (1679)	23301	-	34.4	22381
6	15762 (1727)	23601	-	66.2	10515

[1] Solution statistics for for a single instance of the MIP for a sample application problem solved using Gurobi 5.0.1 with 2 threads for 7200 seconds.

Markets are hard![1]

Markets	# Variables (binary)	# Constraints	Solve time (s)	Gap (%)	Nodes
1	15132 (1457)	22101	680.39	-	1323
2	15263 (1514)	22401	1117.4	-	2934
3	15389 (1568)	22701	3670.3	-	12136
4	15515 (1622)	23001	-	8.24	28208
5	15646 (1679)	23301	-	34.4	22381
6	15762 (1727)	23601	-	66.2	10515

[1] Solution statistics for for a single instance of the MIP for a sample application problem solved using Gurobi 5.0.1 with 2 threads for 7200 seconds.

Contributions[1]

Complex fiscal terms like can be viewed as optimizing a discontinuous, non-convex function $G : \mathbb{R}^T \times \mathbb{R}^T \to \mathbb{R}$.

$$\max \quad \underbrace{G(r, c)}_{\text{Noncovex function}} \quad \text{subject to} \quad \underbrace{(r, c, x) \in X}_{\text{MIP Production Model}}$$

[1] Sridhar, Linderoth, Luedtke, and Wright – in preparation

Contributions[1]

Complex fiscal terms like can be viewed as optimizing a discontinuous, non-convex function $G : \mathbb{R}^T \times \mathbb{R}^T \to \mathbb{R}$.

$$\max \quad \underbrace{G(r,c)}_{\text{Noncovex function}} \quad \text{subject to} \quad \underbrace{(r,c,x) \in X}_{\text{MIP Production Model}}$$

1. Search algorithms for finding high-quality feasible solutions.

[1] Sridhar, Linderoth, Luedtke, and Wright – in preparation

Contributions[1]

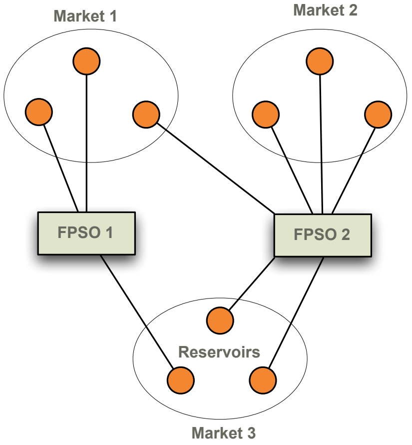
Complex fiscal terms like can be viewed as optimizing a discontinuous, non-convex function $G : \mathbb{R}^T \times \mathbb{R}^T \to \mathbb{R}$.

$$\max \quad \underbrace{G(r,c)}_{\text{Noncovex function}} \quad \text{subject to} \quad \underbrace{(r,c,x) \in X}_{\text{MIP Production Model}}$$

1. Search algorithms for finding high-quality feasible solutions.
2. Decomposition approaches for finding solution bounds.

[1] Sridhar, Linderoth, Luedtke, and Wright – in preparation

Finding Feasible Solutions



Definition: Tranche configuration $\kappa := \{t_k\}$ where t_k is the last time period where the system moves from tranche $k-1$ to k.

Time (t)	Cost (c_t)	Revenue (p_t)	IRR q_t	Tranche ($k(t)$)	Contractor share ($\mu_{k(t)}$)	Cash flow (f_t)
1	4000	0	n.a	1	70%	-4000
2	0	2300	-43%	1	70%	1610
3	0	3000	20%	1	70%	2110
4	0	2800	44%	2	60%	1680
5	0	1000	48%	3	15%	150

Example

The tranche configuration is [0 3 4] (Note: Separate for each market).

Function arguments: Tranche configurations

The function uses tranche configurations of the form [0,3,4] as input.

Fixed Tranche Problem

Function arguments: Tranche configurations

The function uses tranche configurations of the form [0,3,4] as input.

Function evaluation: Operational planning problem

Solve a fixed-tranche version of the operational problem by forcing signs on the NPV at each time period.

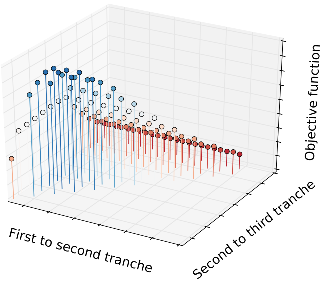
Fixed Tranche Problem

Function arguments: Tranche configurations

The function uses tranche configurations of the form [0,3,4] as input.

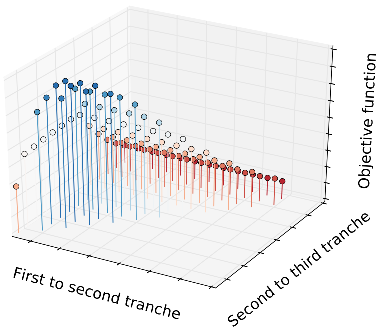
Function evaluation: Operational planning problem

Solve a fixed-tranche version of the operational problem by forcing signs on the NPV at each time period.



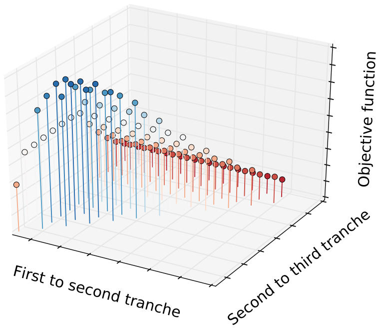
Discrete Domain Search

Pattern search for finding tranche configurations over a discrete space?



Discrete Domain Search

Pattern search for finding tranche configurations over a discrete space?



Search Directions

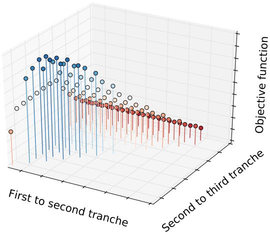
- Coarse.
- Computationally Expensive.

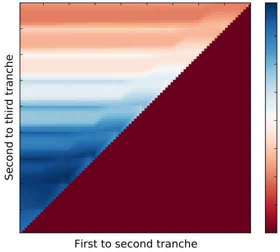
Continuous Tranche Space

Question: What does it mean for the system to to move from tranche 1 to 2 at time 2.9. That would define a tranche configuration [0, 2.9, 4]!

Continuous Tranche Space

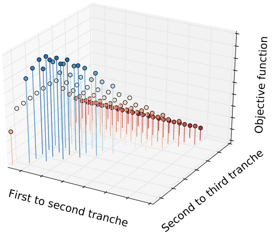
Question: What does it mean for the system to to move from tranche 1 to 2 at time 2.9. That would define a tranche configuration [0, 2.9, 4]!

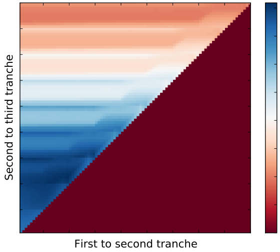




Continuous Tranche Space

Question: What does it mean for the system to to move from tranche 1 to 2 at time 2.9. That would define a tranche configuration [0, 2.9, 4]!

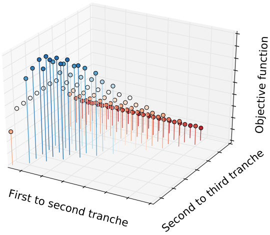


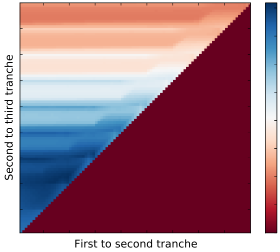


- Accurate search directions using finite difference.

Continuous Tranche Space

Question: What does it mean for the system to to move from tranche 1 to 2 at time 2.9. That would define a tranche configuration [0, 2.9, 4]!

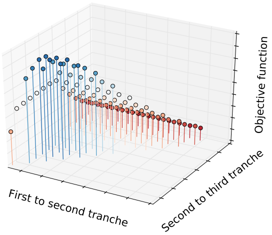


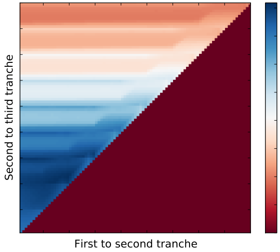


- Accurate search directions using finite difference.
- Search directions are computationally cheaper!

Continuous Tranche Space

Question: What does it mean for the system to to move from tranche 1 to 2 at time 2.9. That would define a tranche configuration [0, 2.9, 4]!



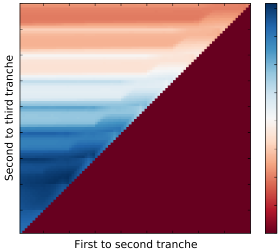


- Accurate search directions using finite difference.
- Search directions are computationally cheaper!
- Convert the fractional tranche configuration to feasible solution.

Contractor Share

Question: What does it mean for the system to to move to tranche 2 at time 2.9. That would define a tranche configuration [0, 2.9, 4]!

Tranche k	IRR Range (%) $[\lambda_k, \lambda_{k+1})$	Contractor Share μ_k
1	[$-\infty$, 10)	70%
2	[10, 25)	50%
3	[25, ∞)	30%



Contractor share rate for time period 3

$$\underbrace{\boxed{0.9 \times 70\%}}_{\text{Tranche 1}} + \underbrace{\boxed{0.1 \times 50\%}}_{\text{Tranche 2}} \approx 68\%.$$

Continuous NPV[2]

Discrete NPV

Continuous discounting

$$\hat{h}(f_{[1,t+1]}, \lambda, t + \delta) = \boxed{\sum_{s=1}^{t} \frac{f_s}{(1+\lambda)^s}} + \boxed{(\delta f_{t+1}) e^{-\lambda_c(\delta + t)}}$$

Integer

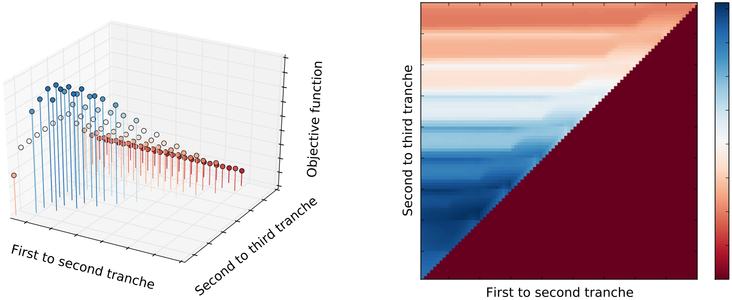
Fractional

where $\delta \in [0, 1]$, $\underbrace{\lambda_c = \log(1 + \lambda)}_{\textit{Continuous discounting rate}}$.

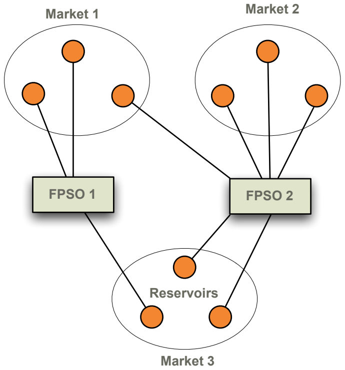
.

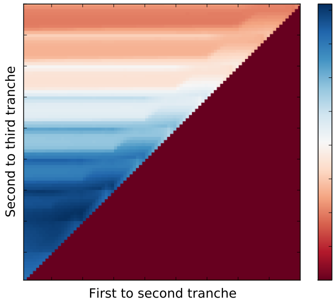
[1]The continuous NPV function $\hat{h}(\cdot)$ is consistent with the NPV function $h(\cdot)$ when $\delta \in \{0, 1\}$

Search Algorithm

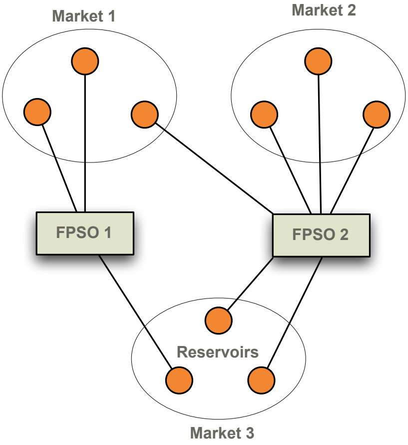


Search Algorithm

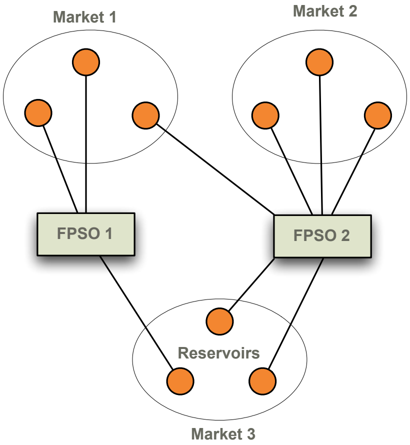




Solution Bounds: Market Based Decomposition



Market Based Decomposition



Observations

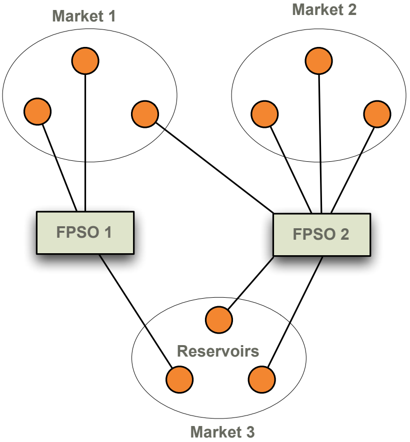
- A single market problem can solve in < 10 minutes.

Decompose by markets?

- Lagrangian decomposition.
- ADMM (Progressive Hedging).[1]

[1] Boyd et al. Foundations and Trends in Machine Learning (2009).

Market Based Decomposition

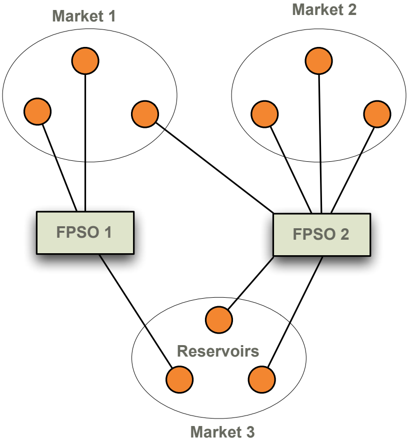


$$\max \sum_{a \in \mathcal{A}} \sum_{t \in \mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Market Based Decomposition



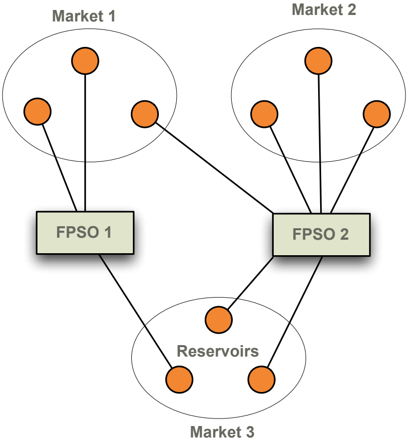
$$\max \sum_{a \in \mathcal{A}} \sum_{t \in \mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

$$\underbrace{(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a}_{\textit{Fiscal model}} \qquad \forall a \in \mathcal{A}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Market Based Decomposition



$$\max \sum_{a \in \mathcal{A}} \sum_{t \in \mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

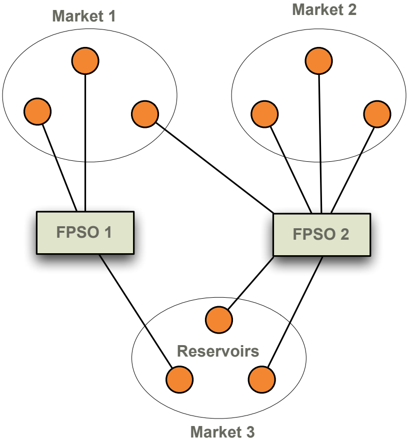
$$\underbrace{(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a}_{\textit{Fiscal model}} \qquad \forall a \in \mathcal{A}$$

$$\underbrace{(\mathbf{c_a}, \mathbf{r_a}, \mathbf{x_a}, \mathbf{w}) \in X_a}_{\textit{Production model}} \qquad \forall a \in \mathcal{A}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Market Based Decomposition



$$\max \sum_{a\in\mathcal{A}} \sum_{t\in\mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

$$\underbrace{(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a}_{\textit{Fiscal model}} \qquad \forall a \in \mathcal{A}$$

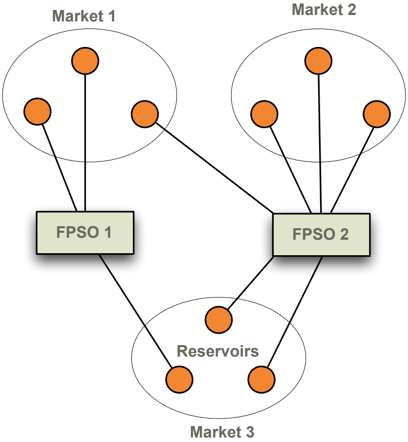
$$\underbrace{(\mathbf{c_a}, \mathbf{r_a}, \mathbf{x_a}, \mathbf{w}) \in X_a}_{\textit{Production model}} \qquad \forall a \in \mathcal{A}$$

$$\underbrace{w \in W}_{\textit{FPSO Constraints}}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Market Based Decomposition



$$\max \sum_{a\in\mathcal{A}} \sum_{t\in\mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

$$\underbrace{(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a}_{\textit{Fiscal model}} \qquad \forall a \in \mathcal{A}$$

$$\underbrace{(\mathbf{c_a}, \mathbf{r_a}, \mathbf{x_a}, \mathbf{w}) \in X_a}_{\textit{Production model}} \qquad \forall a \in \mathcal{A}$$

$$\underbrace{w \in W}_{\textit{FPSO Constraints}}$$

$$\underbrace{\sum_{a\in\mathcal{A}} C_a x_a + Dw \leq d}_{\textit{Aggregate constraints}}$$

Notation

Markets: $a \in \mathcal{A}$, Tranches: $k \in \mathcal{K}$, Timeperiods: $t \in \mathcal{T}$

Original Problem

$$\max \sum_{a \in \mathcal{A}} \sum_{t \in \mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

$$(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a \quad \forall a \in \mathcal{A}$$

$$(\mathbf{c_a}, \mathbf{r_a}, \mathbf{x_a}, \mathbf{w}) \in \underbrace{X_a}_{\textit{Relax}} \quad \forall a \in \mathcal{A}$$

$$\underbrace{w}_{\textit{Make copies}} \in \underbrace{W}_{\textit{Relax}}$$

$$\sum_{a \in \mathcal{A}} C_a x_a + Dw \leq d$$

Market Based Decomposition

Original Problem

$$\max \sum_{a \in \mathcal{A}} \sum_{t \in \mathcal{T}} \pi_t f_{a,t} \quad \text{subject to}$$

$$(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a \quad \forall a \in \mathcal{A}$$

$$(\mathbf{c_a}, \mathbf{r_a}, \mathbf{x_a}, \mathbf{w}) \in \underbrace{X_a}_{\textit{Relax}} \quad \forall a \in \mathcal{A}$$

$$\underbrace{w}_{\textit{Make copies}} \in \underbrace{W}_{\textit{Relax}}$$

$$\sum_{a \in \mathcal{A}} C_a x_a + Dw \leq d$$

Relaxation

$$\max \sum_{a \in \mathcal{A}} \sum_{t \in \mathcal{T}} \pi_t f_{a,t} \quad \text{s.t}$$

$$(\mathbf{b_a}, \mathbf{c_a}, \mathbf{f_a}, \mathbf{h_a}, \mathbf{p_a}, \mathbf{r_a}) \in Y_a \qquad \forall a \in \mathcal{A}$$

$$(\mathbf{c_a}, \mathbf{r_a}, \mathbf{x_a}, \mathbf{w_a}) \in \mathrm{LP}(X_a) \qquad \forall a \in \mathcal{A}$$

$$w_a \in \mathrm{LP}(W) \qquad \forall a \in \mathcal{A}$$

$$\frac{1}{A} \sum_{a' \in \mathcal{A}} w_{a'} = w_a \qquad \forall a \in \mathcal{A}$$

$$\sum_{a \in \mathcal{A}} C_a \mathbf{x_a} + D w_a \leq d \qquad \forall a \in \mathcal{A}$$

Define the Lagrangian as

$$L_a(\widetilde{\mathbf{x}}_\mathbf{a}, \widetilde{\mathbf{y}}_\mathbf{a}; \mathbf{\Theta}) := \sum_{t \in \mathcal{T}} \pi_t f_{a,t} - \underbrace{\Delta_a^T w_a}_{Relax\ \ copy\ \ equality} - \underbrace{\theta^T \left(\frac{d}{A} - C_a x_a - \frac{D w_a}{A} \right)}_{Aggregate\ \ constraints\ \ relaxed}.$$

Lagrangian Decomposition

Define the Lagrangian as

$$L_a(\widetilde{\mathbf{x}}_\mathbf{a}, \widetilde{\mathbf{y}}_\mathbf{a}; \boldsymbol{\Theta}) := \sum_{t \in \mathcal{T}} \pi_t f_{a,t} - \underbrace{\Delta_a^T w_a}_{\textit{Relax copy equality}} \underbrace{- \theta^T \left(\frac{d}{A} - C_a x_a - \frac{D w_a}{A} \right)}_{\textit{Aggregate constraints relaxed}}.$$

where for each market $a \in \mathcal{A}$, we solve

$$L_a^D(\boldsymbol{\Theta}) := \underbrace{\max_{\widetilde{\mathbf{x}}_\mathbf{a}, w_a \widetilde{\mathbf{y}}_\mathbf{a}} \left\{ L_a(\widetilde{\mathbf{x}}_\mathbf{a}, \widetilde{\mathbf{y}}_\mathbf{a}; \boldsymbol{\Theta}) \ \text{ s.t. } \ \widetilde{\mathbf{x}}_\mathbf{a} \in \mathrm{LP}(X_a), \mathbf{w}_\mathbf{a} \in \mathrm{LP}(W), \ \widetilde{\mathbf{y}}_\mathbf{a} \in Y_a \right\}}_{\textit{Single market MIP}}.$$

Lagrangian Decomposition

Define the Lagrangian as

$$L_a(\widetilde{\mathbf{x}}_\mathbf{a}, \widetilde{\mathbf{y}}_\mathbf{a}; \mathbf{\Theta}) := \sum_{t \in \mathcal{T}} \pi_t f_{a,t} - \underbrace{\Delta_a^T w_a}_{\textit{Relax copy equality}} - \underbrace{\theta^T \left(\frac{d}{\mathsf{A}} - C_a x_a - \frac{D w_a}{\mathsf{A}} \right)}_{\textit{Aggregate constraints relaxed}}.$$

where for each market $a \in \mathcal{A}$, we solve

$$L_a^D(\mathbf{\Theta}) := \underbrace{\max_{\widetilde{\mathbf{x}}_\mathbf{a}, w_a \widetilde{\mathbf{y}}_\mathbf{a}} \left\{ L_a(\widetilde{\mathbf{x}}_\mathbf{a}, \widetilde{\mathbf{y}}_\mathbf{a}; \mathbf{\Theta}) \ \text{ s.t. } \ \widetilde{\mathbf{x}}_\mathbf{a} \in \mathrm{LP}(X_a), \mathbf{w}_\mathbf{a} \in \mathrm{LP}(W), \ \widetilde{\mathbf{y}}_\mathbf{a} \in Y_a \right\}}_{\textit{Single market MIP}}.$$

Lagrangian Dual problem:

Solve for θ

$$\boxed{G_D^* := \min_{\mathbf{\Theta}} \left\{ \sum_{a \in \mathcal{A}} L_a^D(\mathbf{\Theta}) : \theta \geq 0 \right\}}$$

Augmented/Regularized Decomposition

Primal update

$$(\widetilde{\mathbf{x}}_{\mathbf{a}}, \widetilde{\mathbf{y}}_{\mathbf{a}})^{i+1} \leftarrow \underset{\widetilde{\mathbf{x}}_{\mathbf{a}}, w_a, \widetilde{\mathbf{y}}_{\mathbf{a}}}{\operatorname{argmax}} \Big\{ \underbrace{L_a(\widetilde{\mathbf{x}}_{\mathbf{a}}, \widetilde{\mathbf{y}}_{\mathbf{a}}; \boldsymbol{\Theta}) - \frac{\rho^i}{2} \|w_a - \bar{w}^i\|^2}_{\textit{Single market MIQP}} :$$

$$\widetilde{\mathbf{x}}_{\mathbf{a}} \in \mathrm{LP}(X_a), \mathbf{w} \in \mathrm{LP}(W), \ \widetilde{\mathbf{y}}_{\mathbf{a}} \in Y_a \Big\}$$

Augmented/Regularized Decomposition

Primal update

$$(\widetilde{\mathbf{x}}_{\mathbf{a}}, \widetilde{\mathbf{y}}_{\mathbf{a}})^{i+1} \leftarrow \underset{\widetilde{\mathbf{x}}_{\mathbf{a}}, w_a, \widetilde{\mathbf{y}}_{\mathbf{a}}}{\operatorname{argmax}} \Big\{ \underbrace{L_a(\widetilde{\mathbf{x}}_{\mathbf{a}}, \widetilde{\mathbf{y}}_{\mathbf{a}}; \boldsymbol{\Theta}) - \frac{\rho^i}{2} \| w_a - \bar{w}^i \|^2}_{\textit{Single market MIQP}} :$$

$$\widetilde{\mathbf{x}}_{\mathbf{a}} \in \mathrm{LP}(X_a), \mathbf{w} \in \mathrm{LP}(W), \ \widetilde{\mathbf{y}}_{\mathbf{a}} \in Y_a \Big\}$$

Dual update

$$\Delta_a^{i+1} \leftarrow \Delta_a^i + \rho^i \left(w_a^{i+1} - \bar{w}^{i+1} \right) \quad a \in \mathcal{A}$$

$$\theta_i^{i+1} \leftarrow \theta_i^i + \eta_d^i \left(d - \sum_{a \in \mathcal{A}} C_a x_a^{i+1} - D \sum_{a \in \mathcal{A}} \frac{w_a^{i+1}}{\mathrm{A}} \right)_+$$

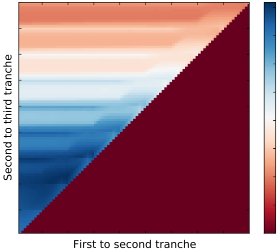
Step size

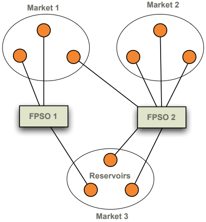
$$\delta_s^{i+1} \leftarrow \delta_s^0 / (i+1)$$

$$\rho^{e+1} \leftarrow \underbrace{\max\{\rho_d \rho^e, \rho_m\}}_{\textit{Geometrically decreasing}} \quad \text{where } \rho_d < 1, \ \rho_m > 0$$

Goals

- Compare the quality of the solutions obtained by the MIP with that obtained by the continuous domain formulation.
- Compare the quality of the solution bounds obtained by the MIP with the decomposition algorithms.





Feasible Solution Quality (5min)

Average gap to the best feasible solution (%)

Markets	TRMIP[1]			HEU+M[2]		
	t=300	t =1800	t =7200	t=300	t =1800	t =7200
2	3.89	0.01	0.00	0.77	0.17	0.15
3	2.76	0.37	0.00	0.70	0.22	0.18
4	5.34	1.02	0.23	0.67	0.27	0.17
5	8.47	1.39	0.30	0.62	0.33	0.10
6	8.23	1.57	0.79	0.68	0.41	0.25
Average	5.74	0.87	0.26	0.69	0.28	0.17

[1] TRMIP: Tranche MIP formulation.
[2] HEU+M: Search algorithm (multi-start).

Feasible Solution Quality (30min)

Average gap to the best feasible solution (%)

Markets	TRMIP[1]			HEU+M[2]		
	t=300	t =1800	t =7200	t=300	t =1800	t =7200
2	3.89	0.01	0.00	0.77	0.17	0.15
3	2.76	0.37	0.00	0.70	0.22	0.18
4	5.34	1.02	0.23	0.67	0.27	0.17
5	8.47	1.39	0.30	0.62	0.33	0.10
6	8.23	1.57	0.79	0.68	0.41	0.25
Average	5.74	0.87	0.26	0.69	0.28	0.17

[1]TRMIP: Tranche MIP formulation.
[2]HEU+M: Search algorithm (multi-start).

Feasible Solution Quality (2 hour)

Average gap to the best feasible solution (%)

Markets	TRMIP[1]			HEU+M[2]		
	t=300	t =1800	t =7200	t=300	t =1800	t =7200
2	3.89	0.01	0.00	0.77	0.17	0.15
3	2.76	0.37	0.00	0.70	0.22	0.18
4	5.34	1.02	0.23	0.67	0.27	0.17
5	8.47	1.39	0.30	0.62	0.33	0.10
6	8.23	1.57	0.79	0.68	0.41	0.25
Average	5.74	0.87	0.26	0.69	0.28	0.17

[1]TRMIP: Tranche MIP formulation.
[2]HEU+M: Search algorithm (multi-start).

Solution Bound Quality (30min)

Average gap to the best feasible solution (%)

Markets	TRMIP[1]		DD+LP[2]		D+LP[3]	
	t=1800	t =7200	t =1800	t=7200	t =1800	t =7200
2	0.23	0.01	2.41	2.13	2.87	2.17
3	14.93	0.03	3.24	2.57	3.34	2.59
4	43.25	11.65	4.49	2.99	3.81	2.92
5	66.43	36.77	6.04	3.54	3.93	3.05
6	81.20	51.86	8.80	4.74	4.60	3.53
Average	41.21	20.06	4.99	3.19	3.71	2.85

[1]TRMIP: Tranche MIP formulation.
[2]DD+LP: Lagrangian Decomposition.
[3]D+LP: Regularized Lagrangian Decomposition.

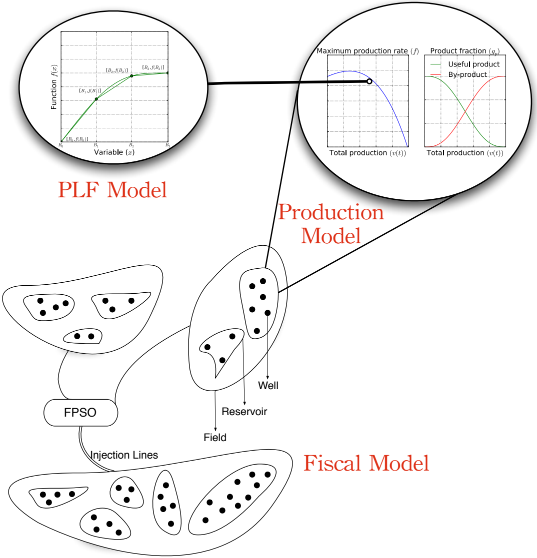
Solution Bound Quality (2 hour)

Average gap to the best feasible solution (%)

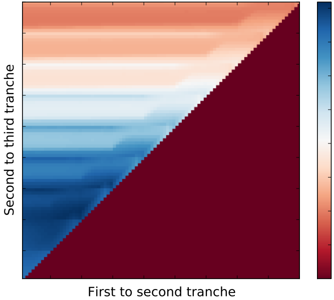
Markets	TRMIP[1]		DD+LP[2]		D+LP[3]	
	t=1800	t =7200	t =1800	t=7200	t =1800	t =7200
2	0.23	0.01	2.41	2.13	2.87	2.17
3	14.93	0.03	3.24	2.57	3.34	2.59
4	43.25	11.65	4.49	2.99	3.81	2.92
5	66.43	36.77	6.04	3.54	3.93	3.05
6	81.20	51.86	8.80	4.74	4.60	3.53
Average	41.21	20.06	4.99	3.19	3.71	2.85

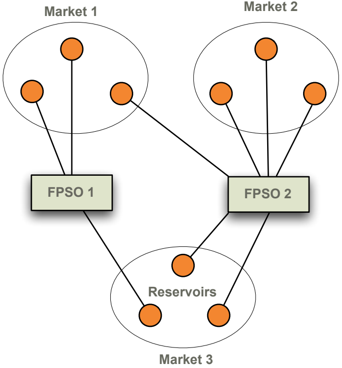
[1]TRMIP: Tranche MIP formulation.
[2]DD+LP: Lagrangian Decomposition.
[3]D+LP: Regularized Lagrangian Decomposition.

Part 1 Conclusion



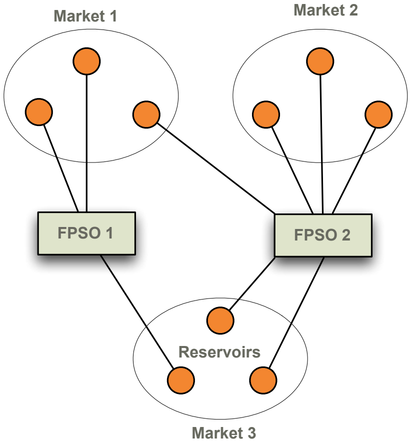
Part 1 Conclusion





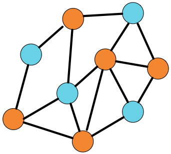
(Fiscal Model, Search Algorithm)

Part 1 Conclusion

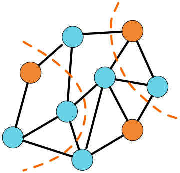


(Fiscal Model, Decomposition)

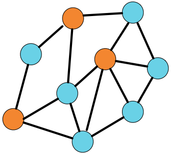
Part 2: Approximation algorithms for large combinatorial problems



vertex cover



multiway cut



independent set

One slide summary

Background

- Motivation: Sometimes approximate is good enough.

[1] Sridhar, Bittorf, Liu, Zhang, Ré, and Wright NIPS (2013)

One slide summary

Background

- Motivation: Sometimes approximate is good enough.
- Approximation algorithms via LP rounding.

[1] Sridhar, Bittorf, Liu, Zhang, Ré, and Wright NIPS (2013)

One slide summary

Background

- Motivation: Sometimes approximate is good enough.
- Approximation algorithms via LP rounding.

Contributions[1]

[1] Sridhar, Bittorf, Liu, Zhang, Ré, and Wright NIPS (2013)

One slide summary

Background

- Motivation: Sometimes approximate is good enough.
- Approximation algorithms via LP rounding.

Contributions[1]

- Rounding approximate LP solutions.

[1] Sridhar, Bittorf, Liu, Zhang, Ré, and Wright NIPS (2013)

One slide summary

Background

- Motivation: Sometimes approximate is good enough.
- Approximation algorithms via LP rounding.

Contributions[1]

- Rounding approximate LP solutions.
- Building a parallel solver to approximately solve large LPs.

[1] Sridhar, Bittorf, Liu, Zhang, Ré, and Wright NIPS (2013)

One slide summary

Background

- Motivation: Sometimes approximate is good enough.
- Approximation algorithms via LP rounding.

Contributions[1]

- Rounding approximate LP solutions.
- Building a parallel solver to approximately solve large LPs.
- Theoretical results bounding runtime and solution quality for approximating combinatorial problems.

[1] Sridhar, Bittorf, Liu, Zhang, Ré, and Wright NIPS (2013)

Motivating Applications

Problem	Applications
Set Covering	Classification (Bien 2009), Multi-object tracking (Wu 2012)
Set Packing	MAP-inference (Sanghavi 2007), Natural language (Kschischang 2001)
Multiway-cut	Computer vision (Yuri 2004), Entity resolution (Lee 2011)
Graphical Models	Semantic role labeling (Roth 2005), Clustering (Van Gael 2007)

Vertex Cover

Problem Statement

Given a graph $G = (V, E)$, find a subset of vertices $\bar{V} \subset V$ that covers all edges.

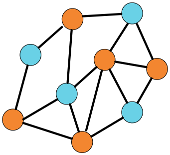
$$\min \sum_{v \in V} c_v x_v$$

$$\text{subject to} \quad \underbrace{x_u + x_v \geq 1}_{\textit{Chose one vertex}} \quad \forall (u, v) \in \boxed{E}$$

Edges

$$x_v \in \{0, 1\} \quad \forall v \in \boxed{V}$$

Vertices



vertex cover

Rounding 101

Integer program

$$\min \sum_{v \in V} c_v x_v$$
$$x_u + x_v \geq 1 \qquad \forall (u, v) \in E$$
$$x_v \in \{0, 1\} \qquad \forall v \in V$$

LP relaxation

$$\min \sum_{v \in V} c_v x_v$$
$$x_u + x_v \geq 1 \qquad \forall (u, v) \in E$$
$$x_v \in [0, 1] \qquad \forall v \in V$$

Rounding 101

Integer program

$$\min \sum_{v\in V} c_v x_v$$
$$x_u + x_v \geq 1 \qquad \forall (u,v)\in E$$
$$x_v \in \{0,1\} \qquad \forall v \in V$$

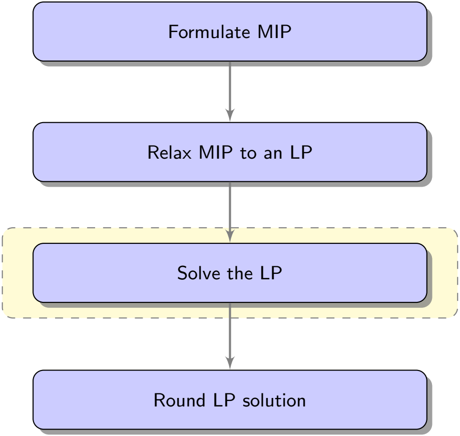
LP relaxation

$$\min \sum_{v\in V} c_v x_v$$
$$x_u + x_v \geq 1 \qquad \forall (u,v)\in E$$
$$x_v \in [0,1] \qquad \forall v \in V$$

Algorithm 1: 2 Approx

1. Compute an optimal solution x_{LP} of the LP relaxation.
2. Round x_{LP} to an integral feasible solution x_{IP}.
3. Clearly, $\underbrace{c' x_{IP} \leq 2c' x_{LP}}_{2 \ approx}$.

Rounding 101



Well Known LP Rounding Schemes [1]

Problem family	Approx factor	Applications
Set Covering	f (Hochbaum 1982)	Classification (Bien 2009), Multi-object tracking (Wu 2012)
Set Packing	$es + o(s)$ (Bansal 2012).	MAP-inference (Sanghavi 2007), Natural language (Kschischang 2001)
Multiway-cut	$3/2 - 1/k$ (Rabini 1998).	Computer vision (Yuri 2004), Entity resolution (Lee 2011).
Graphical Models	Heuristic	Semantic role labeling (Roth 2005), Clustering (Van Gael 2007)

[1] The parameter f refers to the frequency of the most frequent element; s refers to s-column sparse matrices; and k refers to the number of terminals. e refers to the Euler's constant.

Definitions

α-factor approx

An α-factor approximation provides a solution that is at most α times the cost of the true optimal solution.

Integer program

$$
\begin{aligned}
&\min \sum_{v \in V} c_v x_v \\
&x_u + x_v \geq 1 \qquad \forall (u, v) \in E \\
&x_v \in \{0, 1\} \qquad \forall v \in V
\end{aligned}
$$

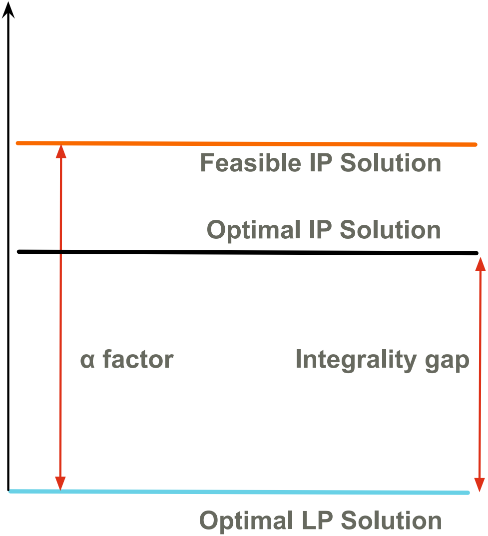
LP relaxation

$$
\begin{aligned}
&\min \sum_{v \in V} c_v x_v \\
&x_u + x_v \geq 1 \qquad \forall (u, v) \in E \\
&x_v \in [0, 1] \qquad \forall v \in V
\end{aligned}
$$

Integrality gap

The worst case ratio between the LP optimum and the integer programming solution is the integrally gap.

Definitions



Do we need the optimal LP solution?

Integer program

$$\min \sum_{v \in V} c_v x_v$$
$$x_u + x_v \geq 1 \qquad \forall (u, v) \in E$$
$$x_v \in \{0, 1\} \qquad \forall v \in V$$

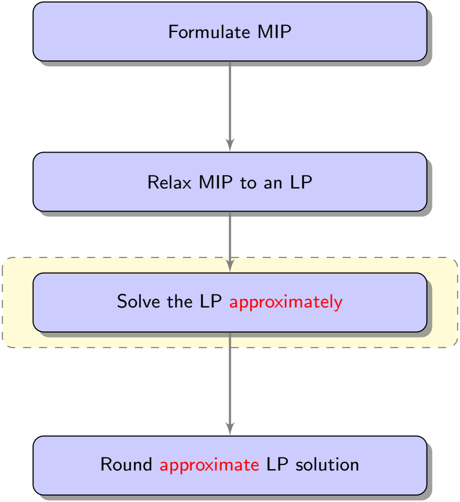
LP relaxation

$$\min \sum_{v \in V} c_v x_v$$
$$x_u + x_v \geq 1 \qquad \forall (u, v) \in E$$
$$x_v \in [0, 1] \qquad \forall v \in V$$

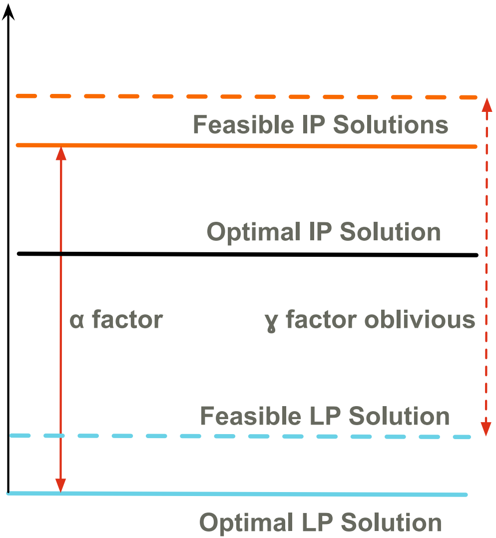
Algorithm 1: 2 Approx

1. Compute ~~an optimal~~ a feasible solution x_{LP} of the LP relaxation.
2. Round x_{LP} to an integral feasible solution x_{IP}.
3. Clearly, $\underbrace{c' x_{IP} \leq 2c' x_{LP}}_{2 \ \textit{approx}}$.

Main Result

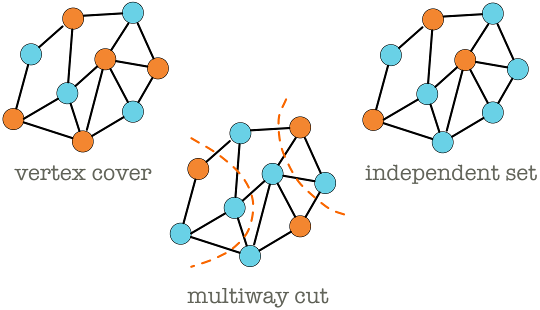

Formulate MIP
Relax MIP to an LP
Solve the LP approximately
Round approximate LP solution

Contribution: Oblivious Rounding



Oblivious LP Rounding Schemes

For a minimization problem Φ with an IP formulation P whose LP relaxation is denoted by $\mathrm{LP}(P)$, a γ-factor 'oblivious' rounding scheme converts any feasible point $x_f \in \mathrm{LP}(P)$ to an integral solution $x_I \in P$ with cost at most γ times the cost of $\mathrm{LP}(P)$ at x_f.



Oblivious Rounding + Feasible LP Solution = Feasible Integral

(ϵ, δ) approximate LP solutions

A point $\hat{x}$ is an (ϵ, δ) approximate solution of the LP

$$\boxed{\min c^T x \quad \text{s.t } Ax = b, x \geq 0}$$

if $\hat{x} \geq 0$ and $\exists \epsilon > 0$ and $\delta > 0$ such that

infeasibility

$$||A\hat{x} - b||_\infty \leq \boxed{\epsilon}$$

$$|c^T\hat{x} - c^T x^*| \leq \boxed{\delta c^T x^*}.$$

sub-optimality

Handling Infeasible LP Solutions

(ϵ, δ) approximate LP solutions

A point $\hat{x}$ is an (ϵ, δ) approximate solution of the LP

$$\boxed{\min c^T x \quad \text{s.t } Ax = b, x \geq 0}$$

if $\hat{x} \geq 0$ and $\exists \epsilon > 0$ and $\delta > 0$ such that

infeasibility

$$||A\hat{x} - b||_\infty \leq \boxed{\epsilon}$$
$$|c^T\hat{x} - c^T x^*| \leq \boxed{\delta c^T x^*}.$$

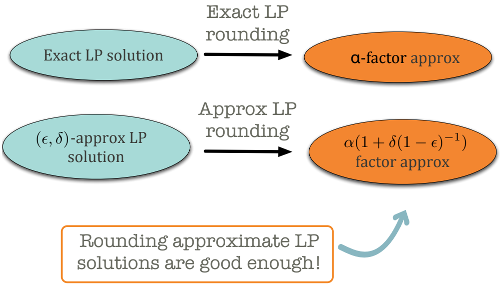
sub-optimality

Key Idea

Convert an $\underbrace{(\epsilon, \delta)}_{\textit{Infeasible}}$ LP solution to a $\underbrace{(0, \hat{\delta})}_{\textit{Feasible}}$ LP solution.

Main Results I: Approximate LP Rounding

Let $\hat{x}$ be an (ε, δ) approximate solution for the LP relaxation of vertex cover with $\varepsilon \in [0, 1)$. Then, $\tilde{x} = \Pi_{[0,1]^n}((1-\varepsilon)^{-1}\hat{x})$ is a $(0, \delta(1-\varepsilon)^{-1})$-approximate solution for vertex cover..



Extends to covering, packing and multiway-cuts!

Main Results II: Computing Approximate LP Solutions

LP

$$\boxed{\begin{array}{c} \min c^T x \quad \text{s.t} \\ Ax = b,\ x \geq 0 \end{array}}$$

$$\Downarrow$$

$$\boxed{x(\beta) := \arg\min_{x \geq 0}\ c^T x - \bar{u}^T (Ax - b) + \frac{\beta}{2}\|Ax - b\|^2 + \frac{1}{2\beta}\|x - \bar{x}\|^2}$$

Quadratic penalty formulation

Main Results II: Computing Approximate LP Solutions

$$x(\beta) := \arg\min_{x \geq 0} f_\beta(x) := c^T x - \underbrace{\boxed{\bar{u}^T (Ax - b)}}_{\text{Dual multiplier}} + \underbrace{\boxed{\frac{\beta}{2} \|Ax - b\|^2}}_{\text{Quadratic penalty}} + \underbrace{\boxed{\frac{1}{2\beta} \|x - \bar{x}\|^2}}_{\text{Regularizer}}$$

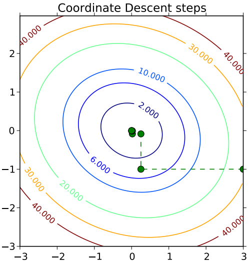
[1] Liu et al. Arxiv (2013)

[2] Depends on the conditioning of the underlying LP.

Main Results II: Computing Approximate LP Solutions

$$x(\beta) := \arg\min_{x \geq 0} f_\beta(x) := c^T x - \underbrace{\boxed{\bar{u}^T(Ax - b)}}_{\text{Dual multiplier}} + \underbrace{\boxed{\frac{\beta}{2}\|Ax - b\|^2}}_{\text{Quadratic penalty}} + \underbrace{\boxed{\frac{1}{2\beta}\|x - \bar{x}\|^2}}_{\text{Regularizer}}$$

ASCD[1] is ideal for large solving QPs!



[1] Liu et al. Arxiv (2013)

Main Results II: Computing Approximate LP Solutions

$$x(\beta) := \arg\min_{x \geq 0} f_\beta(x) := c^T x - \underbrace{\boxed{\bar{u}^T(Ax - b)}}_{\text{Dual multiplier}} + \underbrace{\boxed{\frac{\beta}{2}\|Ax - b\|^2}}_{\text{Quadratic penalty}} + \underbrace{\boxed{\frac{1}{2\beta}\|x - \bar{x}\|^2}}_{\text{Regularizer}}$$

- For a large enough β, the unique solution $x(\beta)$ of the QP approximation is an (ϵ, δ) approximate LP solution[2].

[1] Liu et al. Arxiv (2013)

[2] Depends on the conditioning of the underlying LP.

Main Results II: Computing Approximate LP Solutions

$$x(\beta) := \arg\min_{x \geq 0} f_\beta(x) := c^T x - \underbrace{\boxed{\bar{u}^T(Ax - b)}}_{\text{Dual multiplier}} + \underbrace{\boxed{\frac{\beta}{2}\|Ax - b\|^2}}_{\text{Quadratic penalty}} + \underbrace{\boxed{\frac{1}{2\beta}\|x - \bar{x}\|^2}}_{\text{Regularizer}}$$

- For a large enough β, the unique solution $x(\beta)$ of the QP approximation is an (ϵ, δ) approximate LP solution[2].
- ASCD[1] helps provide a $O(m^3 n^2 \epsilon^{-2})$ complexity estimate for the vertex cover problem for a graph with m edges and n vertices.

[1] Liu et al. Arxiv (2013)

[2] Depends on the conditioning of the underlying LP.

Results: Comparisons with Cplex-LP and Cplex-IP

Instance	# Vars	# Nonzeros	Speedup [1]	Quality [2]
frb59-26-1	0.12M	0.37M	2.8	1.04
Amazon	0.39M	1.17M	8.4	1.23
DBLP	0.37M	1.13M	8.3	1.25
Google+	0.71M	2.14M	9.0	1.21

[1] S: (time taken by Cplex-LP)/(time taken by our solver)
[2] Q: (our objective)/(Cplex-IP objective)

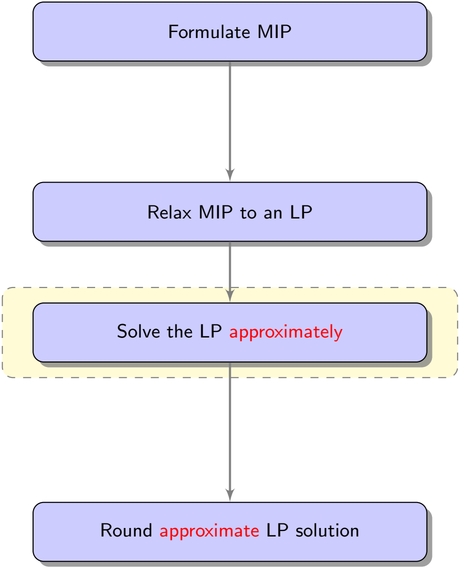
Results: Comparisons with Cplex-LP and Cplex-IP

Instance	# Vars	# Nonzeros	Speedup [1]	Quality [2]
frb59-26-1	0.12M	0.37M	2.8	1.04
Amazon	0.39M	1.17M	8.4	1.23
DBLP	0.37M	1.13M	8.3	1.25
Google+	0.71M	2.14M	9.0	1.21

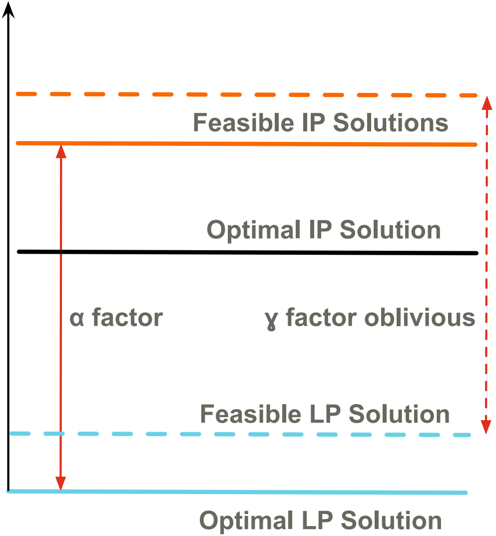
- Rounding approximate LP solutions produced feasible integral solutions of comparable quality with rounding exact LP solutions.
- On larger problems like multiway-cut, we found solutions within 2 minutes while Cplex-LP timed out!.
- On statistical inference problems, we obtain identical application level quality solutions in comparison with Cplex-LP and Cplex-IP.

[1] S: (time taken by Cplex-LP)/(time taken by our solver)
[2] Q: (our objective)/(Cplex-IP objective)

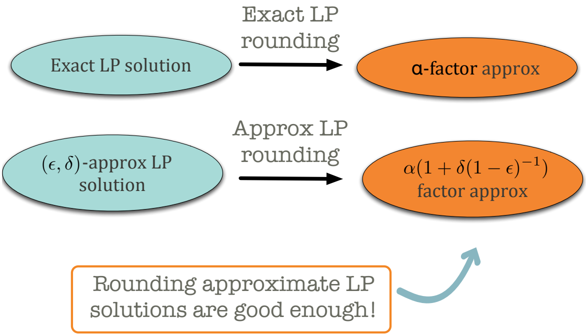
Part 2 Conclusion



Part 2 Conclusion



Part 2 Conclusion



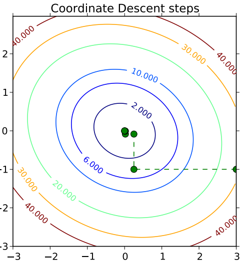
Part 2 Conclusion

LP

$$\min c^T x \quad \text{s.t}$$
$$Ax = b,\ x \geq 0$$

$$x(\beta) := \arg\min_{x \geq 0}\ c^T x - \bar{u}^T(Ax - b) + \frac{\beta}{2}\|Ax - b\|^2 + \frac{1}{2\beta}\|x - \bar{x}\|^2$$

Quadratic penalty formulation



Thats' all folks!

Backup Slides

Results: Vertex Cover

VC	Cplex-IP			Cplex-LP			Thetis		
(min)	t (secs)	BFS	Gap(%)	t (secs)	LP	RSol	t (secs)	LP	RSol
frb59-26-1	-	1475	0.7	2.48	767.0	1534	0.88	959.7	1532
frb59-26-2	-	1475	0.6	3.93	767.0	1534	0.86	979.7	1532
frb59-26-3	-	1475	0.5	4.42	767.0	1534	0.89	982.9	1533
Amazon	85.5	1.60×10^5	-	24.8	1.50×10^5	2.04×10^5	2.97	1.50×10^5	1.97×10^5
DBLP	22.1	1.65×10^5	-	22.3	1.42×10^5	2.08×10^5	2.70	1.42×10^5	2.06×10^5
Google+	-	1.06×10^5	0.01	40.1	1.00×10^5	1.31×10^5	4.47	1.00×10^5	1.27×10^5

Results: Vertex Cover

Cplex-LP

Instance	$\epsilon = 1 \times 10^{-1}$			$\epsilon = 1 \times 10^{-3}$			$\epsilon = 1 \times 10^{-5}$		
	t(s)	LP	RSol	t(s)	LP	RSol	t(s)	LP	RSol
frb59-26-1	2.48	767.0	1534	4.70	767.0	1534	4.59	767.0	1534
frb59-26-2	3.93	767.0	1534	4.61	767.0	1534	4.67	767.0	1534
frb59-26-3	4.42	767.0	1534	4.62	767.0	1534	4.76	767.0	1534
Amazon	24.8	1.50×10^5	2.04×10^5	21.0	1.50×10^5	1.99×10^5	46.7	1.50×10^5	1.99×10^5
DBLP	22.3	1.42×10^5	2.08×10^5	22.8	1.42×10^5	2.07×10^5	31.1	1.42×10^5	2.06×10^5
Google+	40.1	1.00×10^5	1.31×10^5	61.1	1.00×10^5	1.29×10^5	60.0	1.00×10^5	1.30×10^5

Thetis

Instance	$\epsilon = 1 \times 10^{-1}$			$\epsilon = 1 \times 10^{-3}$			$\epsilon = 1 \times 10^{-5}$		
	t(s)	LP	RSol	t(s)	LP	RSol	t(s)	LP	RSol
frb59-26-1	0.88	959.7	1532	13.7	767.0	1534	13.3	767.0	1534
frb59-26-2	0.86	979.7	1532	14.2	767.0	1534	14.1	767.0	1534
frb59-26-3	0.89	982.9	1533	12.9	767.0	1534	12.9	767.0	1534
Amazon	2.97	1.50×10^5	1.97×10^5	59.5	1.50×10^5	1.99×10^5	50.3	1.50×10^5	1.99×10^5
DBLP	2.70	1.42×10^5	2.06×10^5	39.2	1.42×10^5	2.07×10^5	59.1	1.42×10^5	2.07×10^5
Google+	4.47	1.00×10^5	1.27×10^5	1420.1	1.00×10^5	1.29×10^5	2818.2	1.00×10^5	1.30×10^5

Results: Inference

Task	Formulation	PV	NNZ	Method	**P**	**R**	**F1**	Rank
CoNLL	Skip-chain CRF	25M	51M	Cplex-IP	.87	.91	.89	10/13
				Thetis	.87	.90	.89	10/13
				Gibbs Sampling	.86	.90	.88	10/13
TAC-KBP	Factor graph	62K	115K	Cplex-IP	.80	.80	.80	6/17
				Thetis	.79	.79	.79	6/17
				Gibbs Sampling	.80	.80	.80	6/17

Results: Combinatorial Problems

VC (min)	Cplex-IP			Cplex-LP (default)			Thetis		
	t (secs)	BFS	Gap(%)	t (secs)	LP	RSol	t (secs)	LP	RSol
frb59-26-1	-	1475	0.7	4.59	767.0	1534	0.88	959.7	1532
Amazon	85.5	1.60×10^5	-	21.6	1.50×10^5	1.99×10^5	2.97	1.50×10^5	1.97×10^5
DBLP	22.1	1.65×10^5	-	23.7	1.42×10^5	2.07×10^5	2.70	1.42×10^5	2.06×10^5
Google+	-	1.06×10^5	0.01	60.0	1.00×10^5	1.30×10^5	4.47	1.00×10^5	1.27×10^5
MC (min)	Cplex-IP			Cplex-LP (default)			Thetis ($\epsilon = 0.1$)		
	t (secs)	BFS	Gap(%)	t (secs)	LP	RSol	t (secs)	LP	RSol
frb59-26-1	547.4	346	-	397.0	346	346	5.86	352.3	349
Amazon	-	12	NA	-	-	-	55.8	7.28	5
DBLP	-	15	NA	-	-	-	63.8	11.70	5
Google+	-	6	NA	-	-	-	109.9	5.84	5
MIS (max)	Cplex-IP			Cplex-LP (default)			Thetis ($\epsilon = 0.1$)		
	t (secs)	BFS	Gap(%)	t (secs)	LP	RSol	t (secs)	LP	RSol
frb59-26-1	-	50	18.0	4.88	767	16	0.88	447.7	18
Amazon	35.4	1.75×10^5	-	25.7	1.85×10^5	1.58×10^5	3.09	1.73×10^5	1.43×10^5
DBLP	17.3	1.52×10^5	-	24.0	1.75×10^5	1.41×10^5	2.72	1.66×10^5	1.34×10^5
Google+	-	1.06×10^5	0.02	68.8	1.11×10^5	9.40×10^4	4.37	1.00×10^5	8.67×10^4

Problem Description

Production Process

- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.

Production Process

- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.
- Decisions span a planning horizon $\mathcal{T}$.

Production Process

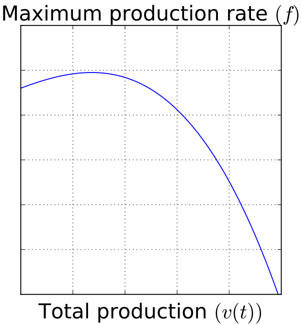
- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.
- Decisions span a planning horizon $\mathcal{T}$.
- Discrete decisions determine the start time of the production process.

Production Process

- The production process creates a mixture of useful products $\mathcal{P}^+$ and byproducts $\mathcal{P}^-$.
- Decisions span a planning horizon $\mathcal{T}$.
- Discrete decisions determine the start time of the production process.
- Continuous decisions determine the production profile evaluated by production functions $f(\cdot)$ and $g_p(\cdot)$.

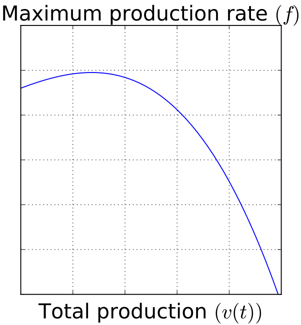
Production functions

- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production.



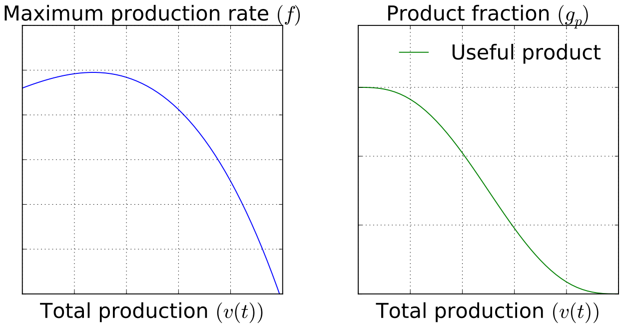
Production functions

- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production.
- Product fraction functions $g_p(\cdot)$ evolve monotonically as a function of the total production.



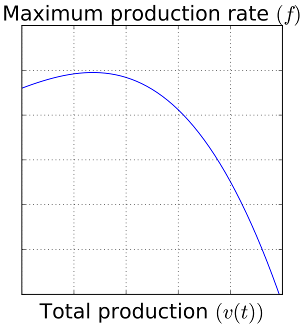
Production functions

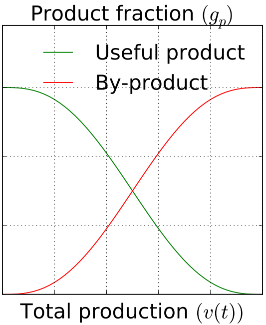
- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production.
- Product fraction functions $g_p(\cdot)$ evolve monotonically as a function of the total production.



Production functions

- Production function $f(\cdot)$ is a concave function that determines the maximum production rate as a function of total production.
- Product fraction functions $g_p(\cdot)$ evolve monotonically as a function of the total production.





Continous time formulation

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s) \mathrm{d}s$$

Continous time formulation

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

Mixture production rate is limited by production function $f(\cdot)$

$$x(t) \leq f(v(t))$$

Continous time formulation

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

Mixture production rate is limited by production function $f(\cdot)$

$$x(t) \leq f(v(t))$$

Product production rates $y_p(t)$ calculated by fraction functions $g_p(\cdot)$

$$y_p(t) = \; x(t) \; g_p(v(t))$$

Continous time formulation

Cumulative production $v(t)$ is calculated using production rate x(t)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

Mixture production rate is limited by production function $f(\cdot)$

$$x(t) \leq f(v(t))$$

Product production rates $y_p(t)$ calculated by fraction functions $g_p(\cdot)$

$$y_p(t) = \; x(t) \; g_p(v(t))$$

Production profiles are active only after the start time $z(t)$

$$v(t) \leq \mathrm{M} \; z(t)$$

Discrete Time Formulations

Discrete time formulations

A 'natural' discretization of this continuous time model (Tarhan 2009)

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = \; x(t) \; g_p(v(t))$$

$$v(t) \leq \mathrm{M} \; z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

A ‘natural’ discretization of this continuous time model (Tarhan 2009)

Continuous time formulation
(CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = \; x(t) \; g_p(v(t))$$

$$v(t) \leq \mathrm{M} \; z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

Discrete time formulations

A ‘natural’ discretization of this continuous time model (Tarhan 2009)

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \rightarrow \{0, 1\}, \text{increasing}$$

v_t Cumulative production up to time period $t \in \mathcal{T}$.

A 'natural' discretization of this continuous time model (Tarhan 2009)

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0,1\}, \text{increasing}$$

v_t	Cumulative production up to time period $t \in \mathcal{T}$.
x_t	Mixture production during time period $t \in \mathcal{T}$.

A ‘natural’ discretization of this continuous time model (Tarhan 2009)

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0,1\}, \text{increasing}$$

v_t Cumulative production up to time period $t \in \mathcal{T}$.

x_t Mixture production during time period $t \in \mathcal{T}$.

$y_{p,t}$ Product $p \in \mathcal{P}$ production during time period $t \in \mathcal{T}$.

Discrete time formulations

A 'natural' discretization of this continuous time model (Tarhan 2009)

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

v_t	Cumulative production up to time period $t \in \mathcal{T}$.
x_t	Mixture production during time period $t \in \mathcal{T}$.
$y_{p,t}$	Product $p \in \mathcal{P}$ production during time period $t \in \mathcal{T}$.
z_t	Facility on/off decision variable.

Past models have proposed a natural discretization of this continuous time model.

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

$\Longrightarrow$

Discrete time formulation (F_1)

Discrete time formulations

Past models have proposed a natural discretization of this continuous time model.

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\; g_p(v(t))$$

$$v(t) \leq \mathrm{M}\; z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

$$\Longrightarrow$$

Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

Discrete time formulations

Past models have proposed a natural discretization of this continuous time model.

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \rightarrow \{0, 1\}, \text{increasing}$$

$$\Longrightarrow$$

Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

Discrete time formulations

Past models have proposed a natural discretization of this continuous time model.

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

$$\Longrightarrow$$

Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t\ g_p(v_{t-1})$$

Discrete time formulations

Past models have proposed a natural discretization of this continuous time model.

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \rightarrow \{0, 1\}, \text{increasing}$$

$$\Longrightarrow$$

Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t\ g_p(v_{t-1})$$

$$v_t \leq \mathrm{M}\ z_t$$

Discrete time formulations

Past models have proposed a natural discretization of this continuous time model.

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{increasing}$$

$$\Longrightarrow$$

Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t\ g_p(v_{t-1})$$

$$v_t \leq \mathrm{M}\ z_t$$

$$z_t \geq z_{t-1},\ z_t \in \{0, 1\}$$

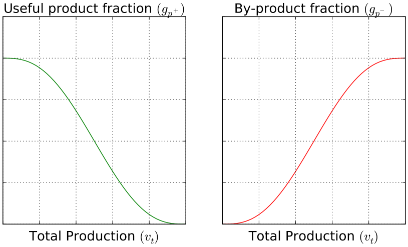
F_1 formulation

How much product is produced up to time t?

F_1 formulation

How much product is produced up to time t?

$$w_{p,t} := \sum_{s \leq t} y_{p,s}$$



Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t \;\; g_p(v_{t-1})$$

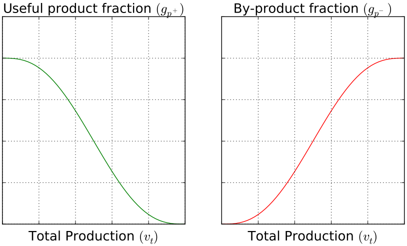
$$v_t \leq \mathrm{M} \;\; z_t$$

$$z_t \geq z_{t-1}, \;\; z_t \in \{0, 1\}$$

F_1 formulation

How much product is produced up to time t?

$$w_{p,t} := \sum_{s \le t} y_{p,s}$$
$$= \sum_{s \le t} x_s g_p(v_{s-1})$$



Discrete time formulation
(F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \le \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t \; g_p(v_{t-1})$$

$$v_t \le \mathrm{M} \; z_t$$

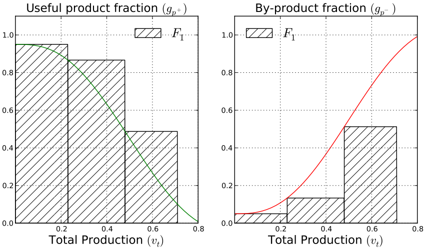
$$z_t \ge z_{t-1}, \; z_t \in \{0, 1\}$$

F_1 formulation

How much product is produced up to time t?

$$w_{p,t} := \sum_{s \leq t} y_{p,s}$$

$$= \sum_{s \leq t} x_s g_p(v_{s-1})$$



Discrete time formulation (F_1)

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t \; g_p(v_{t-1})$$

$$v_t \leq M \; z_t$$

$$z_t \geq z_{t-1}, \; z_t \in \{0, 1\}$$

Alternate formulation

Can we do better?

Can we do better?

Can we calculate exactly how much of product $p \in \mathcal{P}$ is produced up to and including time period t ?

Alternate formulation

Can we do better?

Can we calculate exactly how much of product $p \in \mathcal{P}$ is produced up to and including time period t ?

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0,1\}, \text{inc}$$

Alternate formulation

Can we do better?

Can we calculate exactly how much of product $p \in \mathcal{P}$ is produced up to and including time period t ?

$$w_{p,t} = \int_0^t y_p(s)\mathrm{d}s$$

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \rightarrow \{0,1\}, \text{inc}$$

Alternate formulation

Can we do better?

Can we calculate exactly how much of product $p \in \mathcal{P}$ is produced up to and including time period t ?

$$w_{p,t} = \int_0^t y_p(s)\mathrm{d}s$$

$$= \int_0^t x(s)\ g_p(v(s))\mathrm{d}s$$

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0,1\}, \text{inc}$$

Alternate formulation

Can we do better?

Can we calculate exactly how much of product $p \in \mathcal{P}$ is produced up to and including time period t ?

$$w_{p,t} = \int_0^t y_p(s)\mathrm{d}s$$

$$= \int_0^t x(s)\ g_p(v(s))\mathrm{d}s$$

$$= \int_0^{v_t} g_p(v)\mathrm{d}v$$

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\ g_p(v(t))$$

$$v(t) \leq \mathrm{M}\ z(t)$$

$$z(t) : \mathcal{T} \to \{0,1\}, \text{inc}$$

Alternate formulation

Can we do better?

Can we calculate exactly how much of product $p \in \mathcal{P}$ is produced up to and including time period t ?

$$
\begin{aligned}
w_{p,t} &= \int_0^t y_p(s)\mathrm{d}s \\
&= \int_0^t x(s)\; g_p(v(s))\mathrm{d}s \\
&= \int_0^{v_t} g_p(v)\mathrm{d}v \\
&:= h_p(v_t)
\end{aligned}
$$

Continuous time formulation (CNT)

$$v(t) = \int_0^t x(s)\mathrm{d}s$$

$$x(t) \leq f(v(t))$$

$$y_p(t) = x(t)\; g_p(v(t))$$

$$v(t) \leq \mathrm{M}\; z(t)$$

$$z(t) : \mathcal{T} \to \{0, 1\}, \text{inc}$$

Comparing Formulations

Which formulation is better?

Formulation F_1

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \le \Delta_t f(v_{t-1})$$

$$y_{p,t} = x_t \; g_p(v_{t-1})$$

$$v_t \le \mathrm{M} \; z_t$$

$$z_t \ge z_{t-1}$$

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \le \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

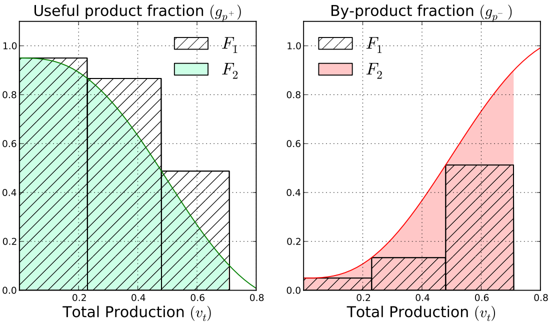
$$v_t \le \mathrm{M} \; z_t$$

$$z_t \ge z_{t-1}$$

Comparing Formulations

Which formulation is better?

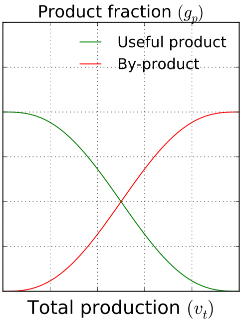
- F_2 is a more accurate formulation of CNT than F_1 .

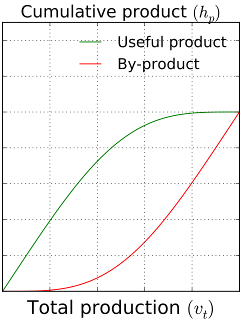


Comparing Formulations

Which formulation is better?

- F_2 is a more accurate formulation of CNT than F_1 .
- F_2 is computationally better because it deals with convex functions while F_1 deals with bilinear terms.



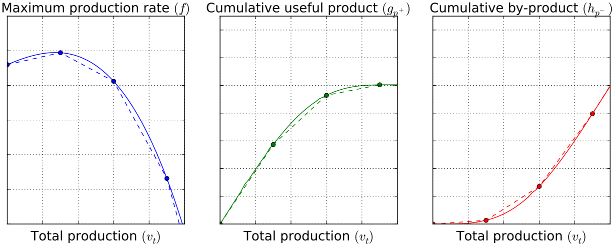


MIP Approximation & Relaxations

Approximations & Relaxations I

Piecewise Linear Approximation (PLA)

Approximate all the nonlinear production functions with piecewise linearizations.

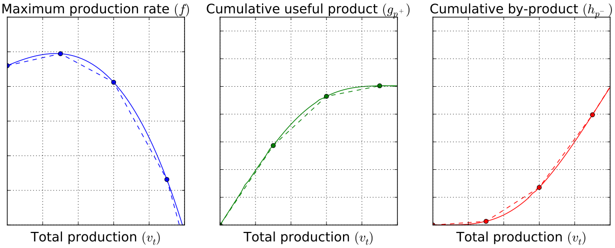


Approximations & Relaxations I

Piecewise Linear Approximation (PLA)

Approximate all the nonlinear production functions with piecewise linearizations.

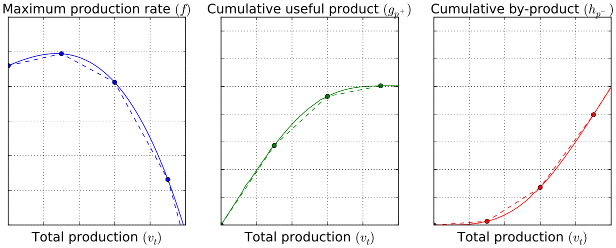
- Pros
  - 'Close' to a feasible solution of the MINLP formulation.



Piecewise Linear Approximation (PLA)

Approximate all the nonlinear production functions with piecewise linearizations.

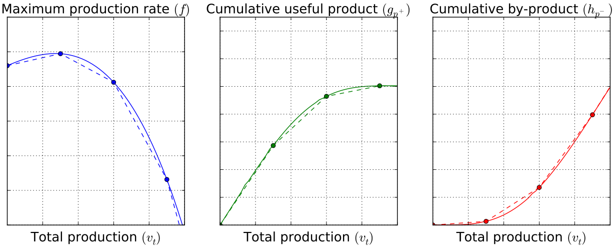
- Pros
  - 'Close' to a feasible solution of the MINLP formulation.
- Cons
  - Introduces additional SOS2 variables to branch on.



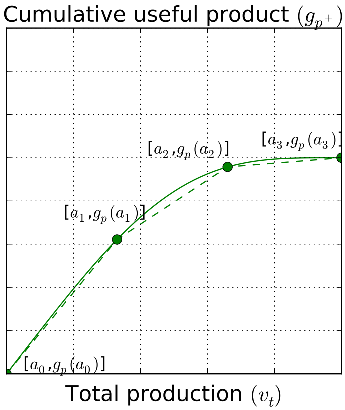
Piecewise Linear Approximation (PLA)

Approximate all the nonlinear production functions with piecewise linearizations.

- Pros
  - 'Close' to a feasible solution of the MINLP formulation.
- Cons
  - Introduces additional SOS2 variables to branch on.
  - NOT a relaxation of the original formulation.



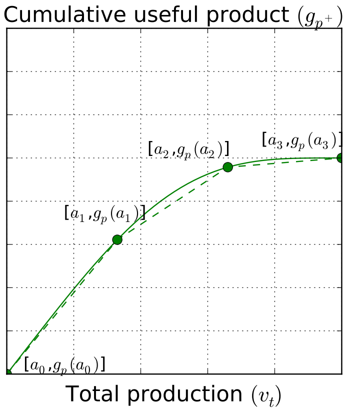
Specially ordered sets (SOS)



Approximating $g_p(v_t)$

$$g_p(v_t) \approx \sum_{o \in \mathcal{O}} \lambda_{t,o} g_p(a_o)$$

Specially ordered sets (SOS)

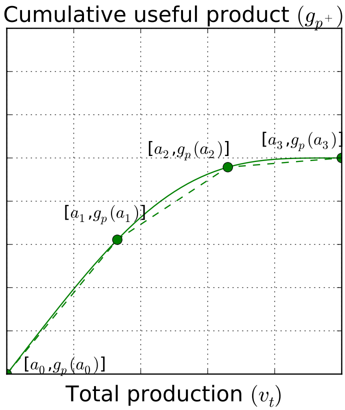


Approximating $g_p(v_t)$

$$g_p(v_t) \approx \sum_{o \in \mathcal{O}} \lambda_{t,o} g_p(a_o)$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

Specially ordered sets (SOS)



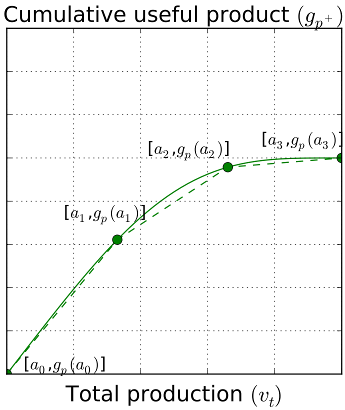
Approximating $g_p(v_t)$

$$g_p(v_t) \approx \sum_{o \in \mathcal{O}} \lambda_{t,o} g_p(a_o)$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

Structure: Only two adjacent non zeros.

Specially ordered sets (SOS)



Approximating $g_p(v_t)$

$$g_p(v_t) \approx \sum_{o \in \mathcal{O}} \lambda_{t,o} g_p(a_o)$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

Structure: Only two adjacent non zeros.

$$\{\lambda_{t,o} | o \in \mathcal{O}\} \in \text{S0S2}$$

Piecewise Linear Approximation (PLA)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

Piecewise Linear Approximation (PLA)

$$v_t = \sum_{s=0}^{t} x_s$$

Piecewise Linear Approximation (PLA)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

Piecewise Linear Approximation (PLA)

$$v_t = \sum_{s=0}^{t} x_s$$

$$v_t = \sum_{o \in \mathcal{O}} B_o \ \lambda_{t,o}$$

$$x_t \leq \Delta_t \sum_{o \in \mathcal{O}} \mathrm{F}_o \ \lambda_{t,o}$$

Piecewise Linear Approximation (PLA)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

Piecewise Linear Approximation (PLA)

$$v_t = \sum_{s=0}^{t} x_s$$

$$v_t = \sum_{o \in \mathcal{O}} B_o \, \lambda_{t,o}$$

$$x_t \leq \Delta_t \sum_{o \in \mathcal{O}} \mathrm{F}_o \, \lambda_{t,o}$$

$$y_{p,t} = w_{p,t} - w_{p,t-1}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o} \, \lambda_{t,o}$$

Piecewise Linear Approximation (PLA)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

$$v_t \leq \mathrm{M}\ z_t$$

$$z_t \geq z_{t-1}$$

Piecewise Linear Approximation (PLA)

$$v_t = \sum_{s=0}^{t} x_s$$

$$v_t = \sum_{o \in \mathcal{O}} B_o\ \lambda_{t,o}$$

$$x_t \leq \Delta_t \sum_{o \in \mathcal{O}} \mathrm{F}_o\ \lambda_{t,o}$$

$$y_{p,t} = w_{p,t} - w_{p,t-1}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o}\ \lambda_{t,o}$$

$$v_t \leq \mathrm{M}\ z_t$$

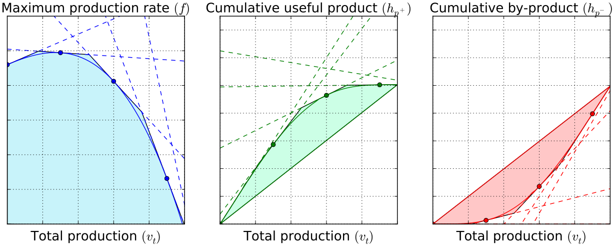
$$z_t \geq z_{t-1}$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

$$\{\lambda_{t,o} | o \in \mathcal{O}\} \in \mathrm{SOS2}$$

Secant Relaxation (1-SEC)

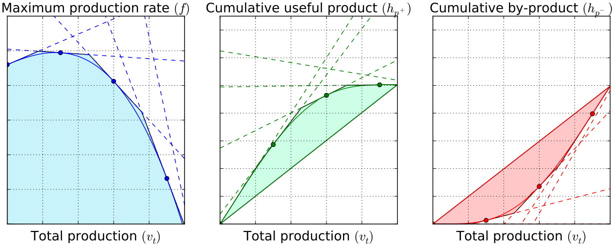
Relax all the nonlinear production functions using inner and outer approximations.



Secant Relaxation (1-SEC)

Relax all the nonlinear production functions using inner and outer approximations.

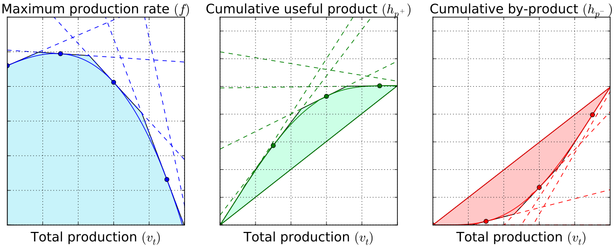
- Pros
  - Relaxation of the original formulation.



Secant Relaxation (1-SEC)

Relax all the nonlinear production functions using inner and outer approximations.

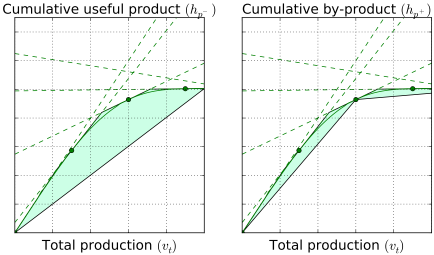
- Pros
  - Relaxation of the original formulation.
  - Does NOT introduce additional SOS2 variables.
- Cons
  - May not be 'close' to a feasible solution of the MINLP formulation.



Approximations & Relaxations III

Multiple Secant Relaxation (k-SEC)

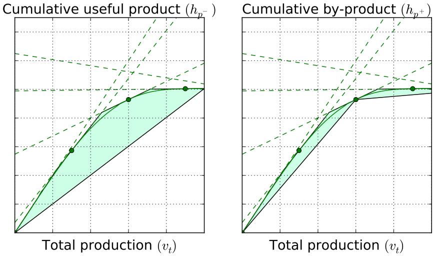
Relax all the nonlinear production functions using inner and outer approximations but use multiple secants instead of a just a single one.



Multiple Secant Relaxation (k-SEC)

Relax all the nonlinear production functions using inner and outer approximations but use multiple secants instead of a just a single one.

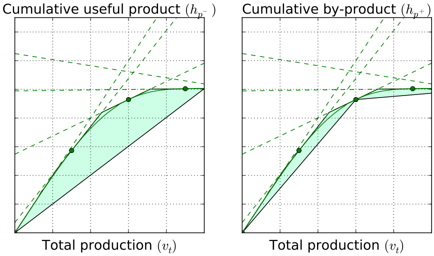
- Pros
  - ‘Close’ to a feasible solution of the MINLP formulation.
  - Relaxation of the original formulation.



Multiple Secant Relaxation (k-SEC)

Relax all the nonlinear production functions using inner and outer approximations but use multiple secants instead of a just a single one.

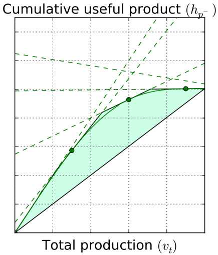
- Pros
  - 'Close' to a feasible solution of the MINLP formulation.
  - Relaxation of the original formulation.
- Cons

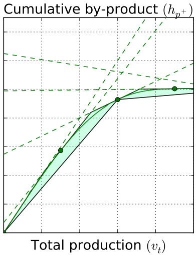


Multiple Secant Relaxation (k-SEC)

Relax all the nonlinear production functions using inner and outer approximations but use multiple secants instead of a just a single one.

- Pros
  - ‘Close’ to a feasible solution of the MINLP formulation.
  - Relaxation of the original formulation.
- Cons
  - Introduces additional SOS2 variables to branch on.





Key Idea

Production functions are positive only if the facility is open.

Key Idea

Production functions are positive only if the facility is open.

Original Formulation...

$$v_t = \sum_{o \in \mathcal{O}} B_o \; \lambda_{t,o}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o} \; \lambda_{t,o}$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

$$v_t \leq M z_t$$

Key Idea

Production functions are positive only if the facility is open.

Original Formulation...

$$v_t = \sum_{o \in \mathcal{O}} B_o \; \lambda_{t,o}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o} \; \lambda_{t,o}$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

$$v_t \leq M z_t$$

Stronger Formulation...

$$v_t = \sum_{o \in \mathcal{O}} B_o \; \lambda_{t,o}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o} \; \lambda_{t,o}$$

$$z_t = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

Locally Ideal formulations

How do we determine which formulation is better?

Locally Ideal formulations

How do we determine which formulation is better?

- Strength of LP relaxations in the absence of other constraints.

Locally Ideal formulations

How do we determine which formulation is better?

- Strength of LP relaxations in the absence of other constraints.

Locally Ideal: A strong property of a MIP model

Locally Ideal formulations

How do we determine which formulation is better?

- Strength of LP relaxations in the absence of other constraints.

Locally Ideal: A strong property of a MIP model

- Vertices of the LP relaxation of a MIP model always satisfy integrality [Padberg2000]

Locally Ideal formulations

How do we determine which formulation is better?

- Strength of LP relaxations in the absence of other constraints.

Locally Ideal: A strong property of a MIP model

- Vertices of the LP relaxation of a MIP model always satisfy integrality [Padberg2000]
- In a more general sense, locally ideal could also mean that you get a property like SOS2 also for free. [Keha2004]

Locally Ideal formulations

How do we determine which formulation is better?

- Strength of LP relaxations in the absence of other constraints.

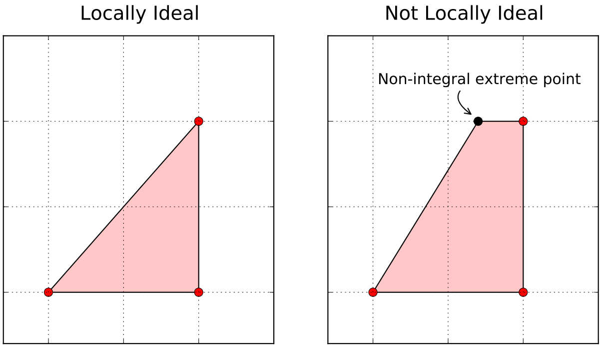
Locally Ideal: A strong property of a MIP model

- Vertices of the LP relaxation of a MIP model always satisfy integrality [Padberg2000]
- In a more general sense, locally ideal could also mean that you get a property like SOS2 also for free. [Keha2004]

Bottom line

- Locally ideal $\rightarrow$ we get integrality for free.

Locally Ideal formulations



Bottom line

- Locally ideal $\rightarrow$ we get integrality for free.

Properties of MIP formulations

Theorem 1

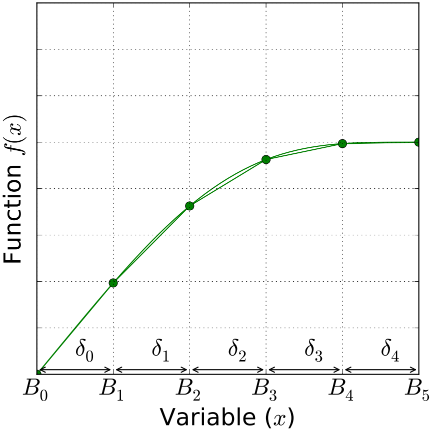
- F_1 is not locally ideal. (counter example)

Theorem 2

- F_2 is locally ideal.

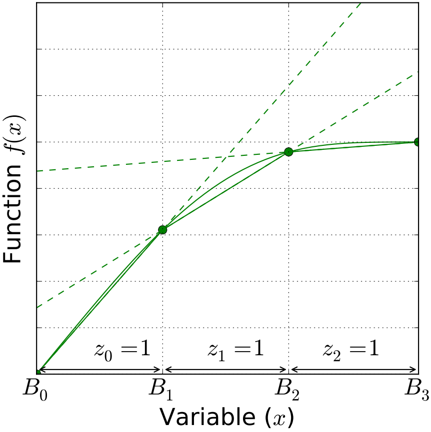
Other Piecewise Linear Formulations

Incremental Model



[Markowitz and Manne 1957]

Multiple Choice



[Balakrishna and Graves 1989]

Performance Evaluation

Experiments

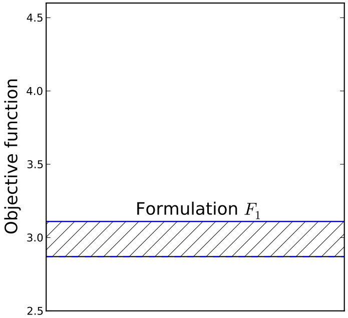
Goals

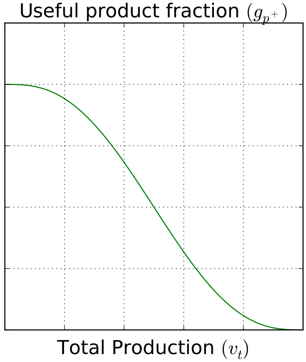
- Impact on formulation accuracy in going from F_1 to F_2
- Impact in solution time in going from F_1 to F_2 as solved by our models.
- Impact of stronger formulations on solving the MIP approximation/relaxations.

Sample Application

Transportation problem with production facilities manufacturing products for customers.

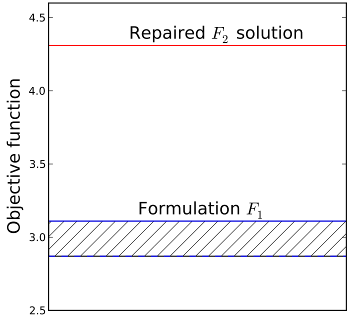
Accuracy

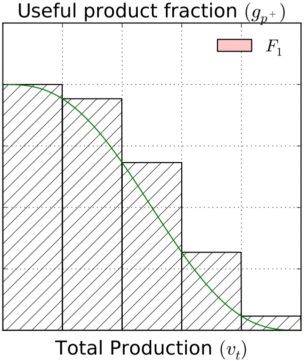




- F_1: Bilinear formulation of [Tarhan2009].
- F_2: Our formulation.

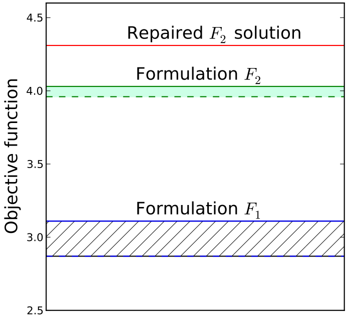
Accuracy

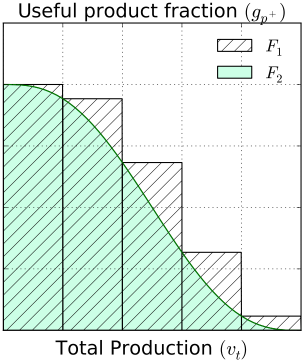




- F_1: Bilinear formulation of [Tarhan2009].
- F_2: Our formulation.

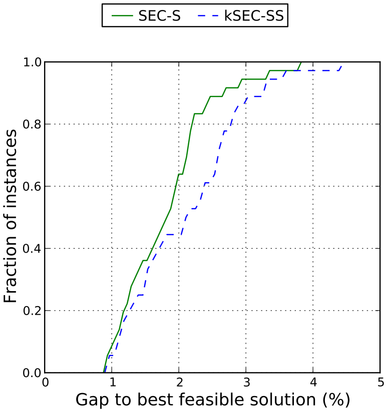
Accuracy



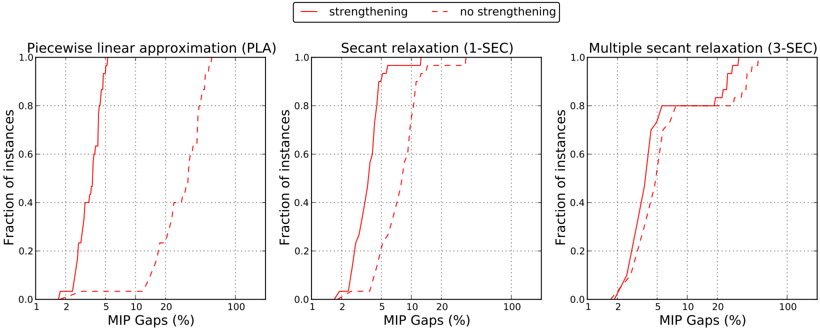


- F_1: Bilinear formulation of [Tarhan2009].
- F_2: Our formulation.

Formulations



Trix



strengthening
no strengthening
Piecewise linear approximation (PLA)
Secant relaxation (1-SEC)
Multiple secant relaxation (3-SEC)
Fraction of instances
MIP Gaps (%)
0.0
0.2
0.4
0.6
0.8
1.0
1
2
5
10
20
100

Conclusions

- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.

Conclusions

- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.
  - Ratio of byproducts to total production increases monotonically.

Conclusions

- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.
  - Ratio of byproducts to total production increases monotonically.
- **Methods**
  - Reformulated an existing formulation (F_1) to produce a more accurate formulation (F_2) based on the cumulative product production function.

Conclusions

- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.
  - Ratio of byproducts to total production increases monotonically.
- **Methods**
  - Reformulated an existing formulation (F_1) to produce a more accurate formulation (F_2) based on the cumulative product production function.
  - Devised scalable MIP approximations & relaxations (PLA, 1-SEC, k-SEC).

Conclusions

- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.
  - Ratio of byproducts to total production increases monotonically.
- **Methods**
  - Reformulated an existing formulation (F_1) to produce a more accurate formulation (F_2) based on the cumulative product production function.
  - Devised scalable MIP approximations & relaxations (PLA, 1-SEC, k-SEC).
- **Conclusions**

Conclusions

- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.
  - Ratio of byproducts to total production increases monotonically.
- **Methods**
  - Reformulated an existing formulation (F_1) to produce a more accurate formulation (F_2) based on the cumulative product production function.
  - Devised scalable MIP approximations & relaxations (PLA, 1-SEC, k-SEC).
- **Conclusions**
  - F_2 formulation is a more accurate evaluation of operations as compared to F_1 .

Conclusions

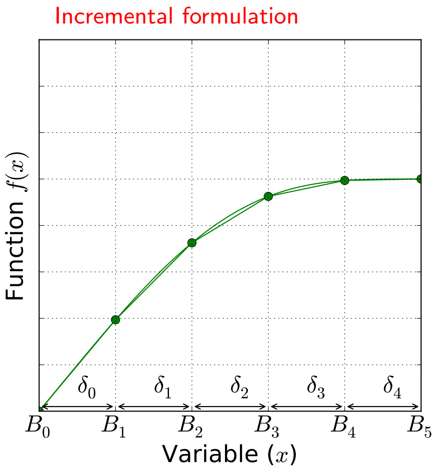
- **Problem Description**
  - Defined a non-convex production process involving desirable & undesirable products.
  - Ratio of byproducts to total production increases monotonically.
- **Methods**
  - Reformulated an existing formulation (F_1) to produce a more accurate formulation (F_2) based on the cumulative product production function.
  - Devised scalable MIP approximations & relaxations (PLA, 1-SEC, k-SEC).
- **Conclusions**
  - F_2 formulation is a more accurate evaluation of operations as compared to F_1 .
  - F_2 is computationally more tractable than F_1 .

Set structure

- Function $f(\cdot)$ is evaluated only if a binary variable z is on.

Set structure

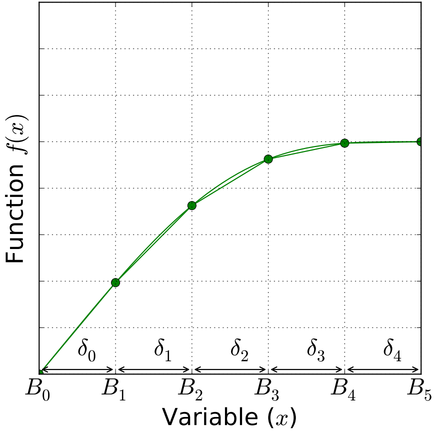
- Function $f(\cdot)$ is evaluated only if a binary variable z is on.



Set structure

- Function $f(\cdot)$ is evaluated only if a binary variable z is on.

Incremental formulation



Original Formulation (Δ_1)

$$x = \sum_{o \in \mathcal{O}} [B_o - B_{o-1}]\delta_o$$

$$y = \sum_{o \in \mathcal{O}} [F_o - F_{o-1}]\delta_o$$

$$\delta_1 \leq 1$$

$$\delta_n \geq 0$$

$$\delta_{i+1} \leq b_i \leq \delta_i \quad \forall o \in \mathcal{O}$$

$$b \in \{0,1\}^n$$

Incremental Formulation

Set structure

- Function $f(\cdot)$ is evaluated only if a binary variable z is on.

Original Formulation (Δ_1)

$$x = \sum_{o \in \mathcal{O}} [B_o - B_{o-1}] \delta_o$$

$$y = \sum_{o \in \mathcal{O}} [F_o - F_{o-1}] \delta_o$$

$$\delta_1 \leq 1$$

$$\delta_n \geq 0$$

$$\delta_{i+1} \leq b_i \leq \delta_i \quad \forall o \in \mathcal{O}$$

$$b \in \{0, 1\}^n$$

Incremental Formulation

Set structure

- Function $f(\cdot)$ is evaluated only if a binary variable z is on.

Stronger Formulation (Δ_2)

$$
\begin{aligned}
x &= \sum_{o \in \mathcal{O}} [B_o - B_{o-1}]\delta_o \\
y &= \sum_{o \in \mathcal{O}} [F_o - F_{o-1}]\delta_o \\
\delta_1 &\le z \\
\delta_n &\ge 0 \\
\delta_{i+1} &\le b_i \le \delta_i \quad \forall o \in \mathcal{O} \\
b &\in \{0,1\}^n
\end{aligned}
$$

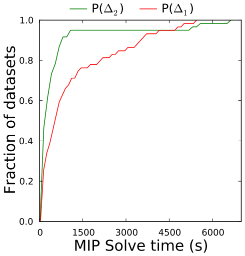
Original Formulation (Δ_1)

$$
\begin{aligned}
x &= \sum_{o \in \mathcal{O}} [B_o - B_{o-1}]\delta_o \\
y &= \sum_{o \in \mathcal{O}} [F_o - F_{o-1}]\delta_o \\
\delta_1 &\le 1 \\
\delta_n &\ge 0 \\
\delta_{i+1} &\le b_i \le \delta_i \quad \forall o \in \mathcal{O} \\
b &\in \{0,1\}^n
\end{aligned}
$$

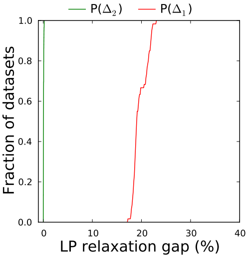
Numerical Experiments: Incremental Model

Numerical experiments performed on randomly generated instances of an advertising budget allocation problem.

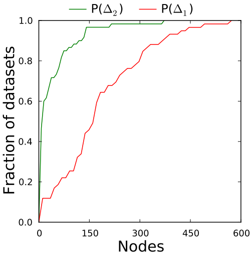
Solve time



Root LP gap



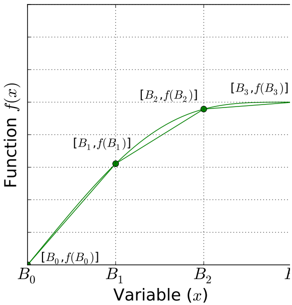
Nodes explored



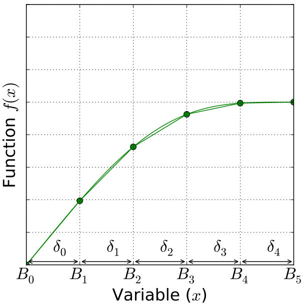
Other Piecewise Linear Formulations

Many formulations can incorporate binary indicators variables in a locally ideal way!

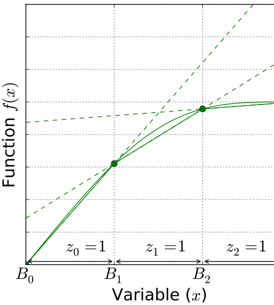
Convex Combination



Incremental Model

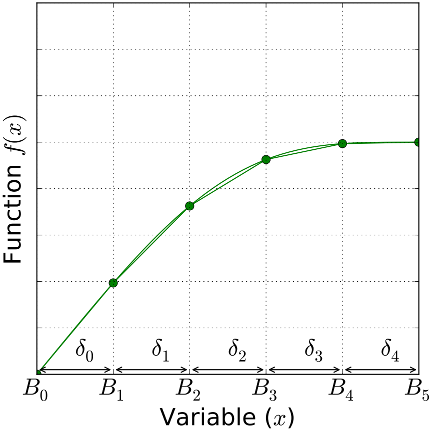


Multiple Choice



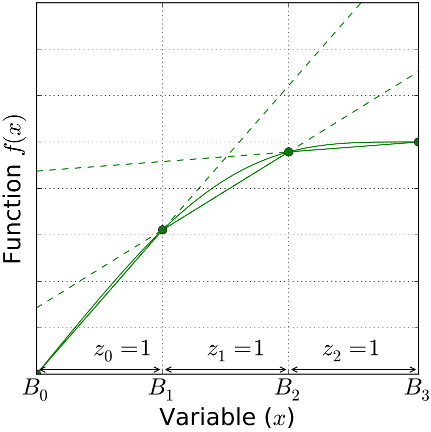
Other Piecewise Linear Formulations

Incremental Model



[Balakrishnan1989]

Multiple Choice

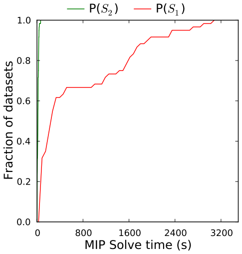


[Dantzig1960]

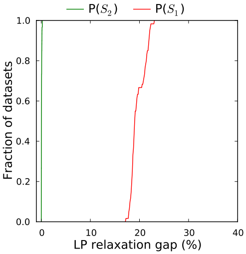
Numerical Experiments: SOS2 Model

Numerical experiments performed on randomly generated instances of an advertising budget allocation problem.

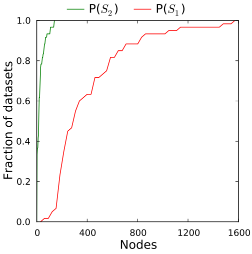
Solve time



Root LP gap

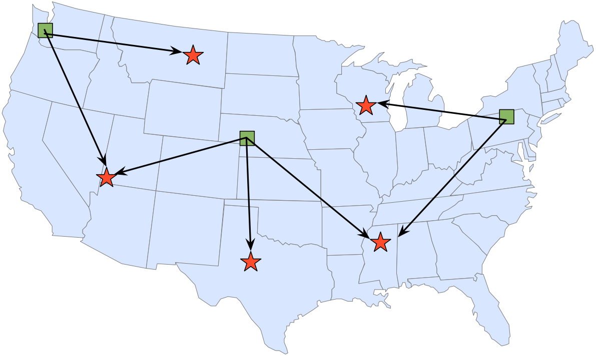


Nodes explored



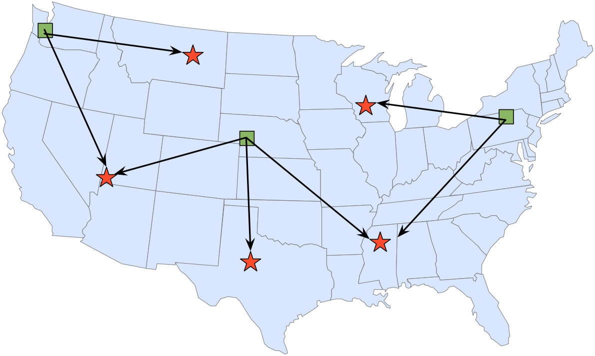
Performance Evaluation

- Transportation problem with production facilities $\mathcal{I}$ manufacturing products $\mathcal{P}^+$ for customers $\mathcal{J}$.



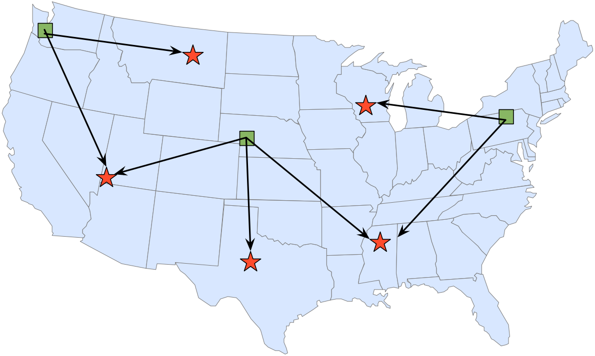
Performance Evaluation

- Transportation problem with production facilities $\mathcal{I}$ manufacturing products $\mathcal{P}^+$ for customers $\mathcal{J}$.
- Demand made by customers are known a priori.



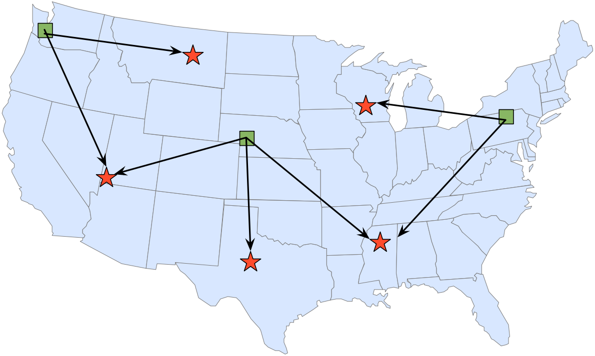
Performance Evaluation

- Transportation problem with production facilities $\mathcal{I}$ manufacturing products $\mathcal{P}^+$ for customers $\mathcal{J}$.
- Demand made by customers are known a priori.
- Facility operations follow known production functions.



Performance Evaluation

- Transportation problem with production facilities $\mathcal{I}$ manufacturing products $\mathcal{P}^+$ for customers $\mathcal{J}$.
- Demand made by customers are known a priori.
- Facility operations follow known production functions.
- Facilities incur fixed, operating, transportation and penalty costs.



Secant Relaxation (1-SEC)

Formulation F$_2$

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

Secant Relaxation (1-SEC)

$$v_t = \sum_{s=0}^{t} x_s$$

Secant Relaxation (1-SEC)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

Secant Relaxation (1-SEC)

$$v_t = \sum_{s=0}^{t} x_s$$

$$v_t = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o \ \lambda_{t,o}$$

$$x_t \leq \Delta_t \sum_{o \in \mathcal{O}} \hat{\mathrm{F}}_o \ \lambda_{t,o}$$

Secant Relaxation (1-SEC)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

Secant Relaxation (1-SEC)

$$v_t = \sum_{s=0}^{t} x_s$$

$$v_t = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o \; \lambda_{t,o}$$

$$x_t \leq \Delta_t \sum_{o \in \mathcal{O}} \hat{\mathrm{F}}_o \; \lambda_{t,o}$$

$$y_{p,t} = w_{p,t} - w_{p,t-1}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \hat{\mathrm{H}}_{p,o} \; \lambda_{t,o}$$

Secant Relaxation (1-SEC)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

$$v_t \leq \mathrm{M}\ z_t$$

$$z_t \geq z_{t-1}$$

Secant Relaxation (1-SEC)

$$v_t = \sum_{s=0}^{t} x_s$$

$$v_t = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o\ \lambda_{t,o}$$

$$x_t \leq \Delta_t \sum_{o \in \mathcal{O}} \hat{\mathrm{F}}_o\ \lambda_{t,o}$$

$$y_{p,t} = w_{p,t} - w_{p,t-1}$$

$$w_{p,t} = \sum_{o \in \mathcal{O}} \hat{\mathrm{H}}_{p,o}\ \lambda_{t,o}$$

$$v_t \leq \mathrm{M}\ z_t$$

$$z_t \geq z_{t-1}$$

$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

Multiple Secant Relaxation (k-SEC)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

Multiple Secant Relaxation (k-SEC)

$$v_t = \sum_{s=0}^{t} x_s = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o \; \lambda_{t,o}$$

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

Multiple Secant Relaxation (k-SEC)

$$v_t = \sum_{s=0}^{t} x_s = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o \; \lambda_{t,o}$$

Multiple Secant Relaxation (k-SEC)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \le \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

$$v_t \le \mathrm{M}\ z_t$$

$$z_t \ge z_{t-1}$$

Multiple Secant Relaxation (k-SEC)

$$v_t = \sum_{s=0}^{t} x_s = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o\ \lambda_{t,o}$$

$$y_{p,t} = w_{p,t} - w_{p,t-1}$$

$$\sum_{o \in \mathcal{O}} \mathrm{H}_{p,o}\ \lambda_{t,o} \le w_{p,t} \le \sum_{o \in \mathcal{O}} \hat{\mathrm{H}}_{p,o}\ \lambda_{t,o} \quad \forall p \in \mathcal{P}^+$$

$$\sum_{o \in \mathcal{O}} \hat{\mathrm{H}}_{p,o}\ \lambda_{t,o} \le w_{p,t} \le \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o}\ \lambda_{t,o} \quad \forall p \in \mathcal{P}^-$$

Multiple Secant Relaxation (k-SEC)

Formulation F_2

$$v_t = \sum_{s=0}^{t} x_s$$

$$x_t \leq \Delta_t f(v_{t-1})$$

$$y_{p,t} = h_p(v_t) - h_p(v_{t-1})$$

$$v_t \leq \mathrm{M}\ z_t$$

$$z_t \geq z_{t-1}$$

Multiple Secant Relaxation (k-SEC)

$$v_t = \sum_{s=0}^{t} x_s = \sum_{o \in \mathcal{O}} \hat{\mathrm{B}}_o\ \lambda_{t,o}$$

$$y_{p,t} = w_{p,t} - w_{p,t-1}$$

$$\sum_{o \in \mathcal{O}} \mathrm{H}_{p,o}\ \lambda_{t,o} \leq w_{p,t} \leq \sum_{o \in \mathcal{O}} \hat{\mathrm{H}}_{p,o}\ \lambda_{t,o} \quad \forall p \in \mathcal{P}^+$$

$$\sum_{o \in \mathcal{O}} \hat{\mathrm{H}}_{p,o}\ \lambda_{t,o} \leq w_{p,t} \leq \sum_{o \in \mathcal{O}} \mathrm{H}_{p,o}\ \lambda_{t,o} \quad \forall p \in \mathcal{P}^-$$

$$v_t \leq \mathrm{M}\ z_t$$

$$z_t \geq z_{t-1}$$

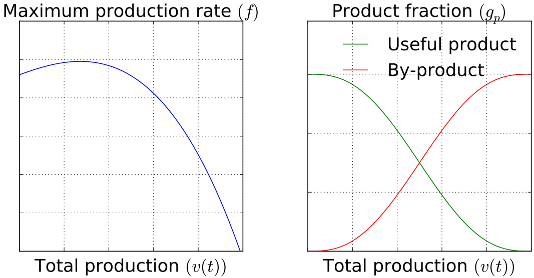
$$1 = \sum_{o \in \mathcal{O}} \lambda_{t,o}$$

$$\{\lambda_{t,o} | o \in \mathcal{O}\} \in \mathrm{SOS2}$$

Breakpoint Selection

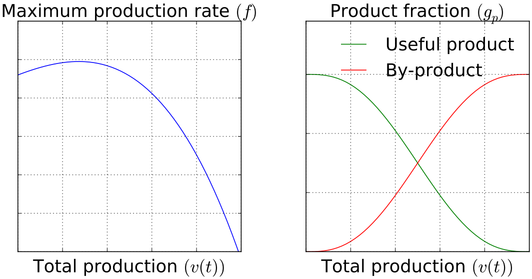
Multiple functions functions

- We consider a special case with multiple functions sharing the same domain.



Multiple functions functions

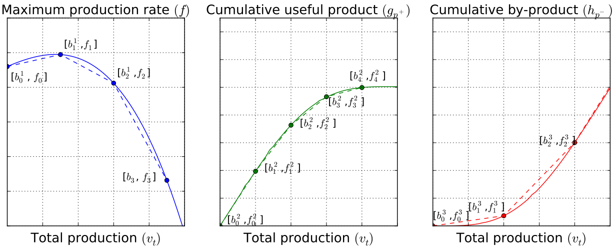
- We consider a special case with multiple functions sharing the same domain.



This structure occurs in models from various applications!

Model 1

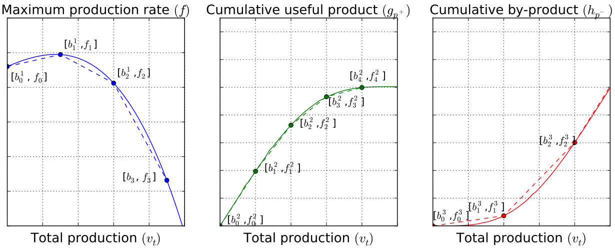
Separately approximate each of the nonlinear functions.



Model 1

Separately approximate each of the nonlinear functions.

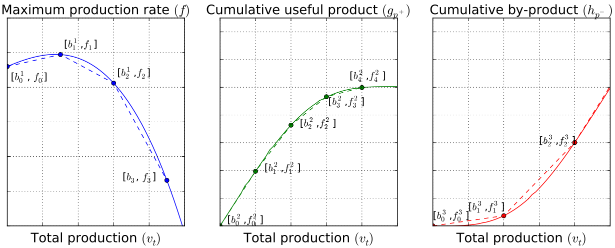
- Pros: Accurate



Model 1

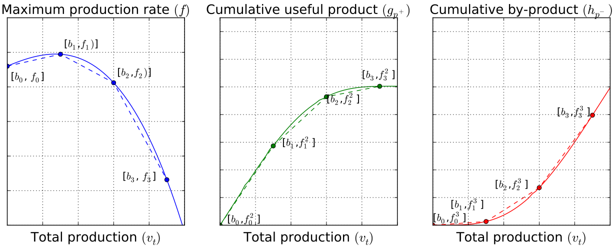
Separately approximate each of the nonlinear functions.

- Pros: Accurate
- Cons: Introduces additional SOS2 variables for each function.



Model 2

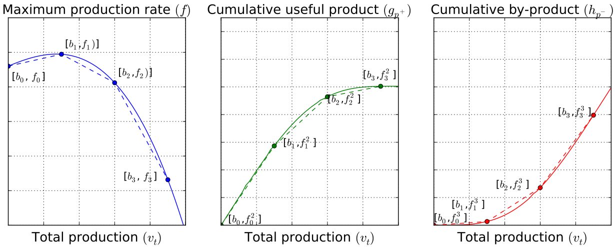
Use common SOS2 variables to approximate the nonlinear functions simultaneously.



Model 2

Use common SOS2 variables to approximate the nonlinear functions simultaneously.

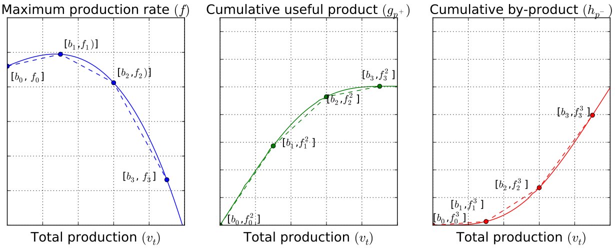
- Pros: Fewer branching entities (SOS2 variables)



Model 2

Use common SOS2 variables to approximate the nonlinear functions simultaneously.

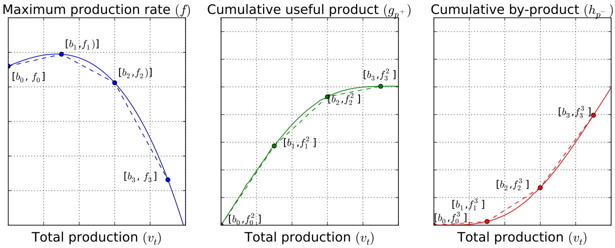
- Pros: Fewer branching entities (SOS2 variables)
- Cons: Each function approximation is less accurate.



Model 2

Use common SOS2 variables to approximate the nonlinear functions simultaneously.

- Pros: Fewer branching entities (SOS2 variables)
- Cons: Each function approximation is less accurate.



Applicable to piecewise linear relaxations

Choosing breakpoints

Key question

Is there a way to select breakpoints of piecewise linear approximations & relaxations which are:

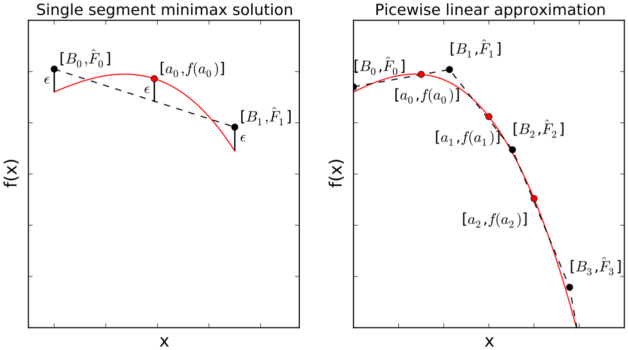
- As accurate as Model I.
- Use as few variables as Model II.

Contributions

- An NLP formulation to find the tightest possible piecewise linear relaxations or approximations of a single/multiple convex function(s).

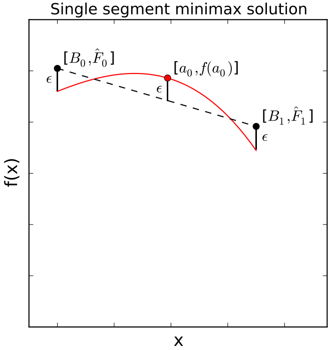
Single function approximation

Piecewise linear approximations that minimize maximum point-wise function evaluation error.



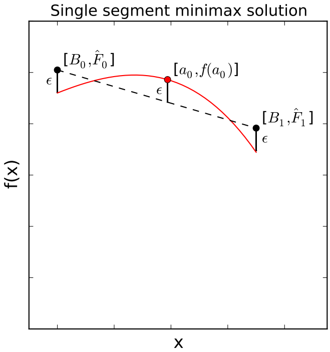
We extend previous work [Imamoto2008], [Rote1992] to multiple univariate convex/concave that share the same domain

Breakpoint selection: Single function approximation



$$\epsilon := \max_{x \in \mathcal{D}} |f(x) - \hat{f}(x)|$$

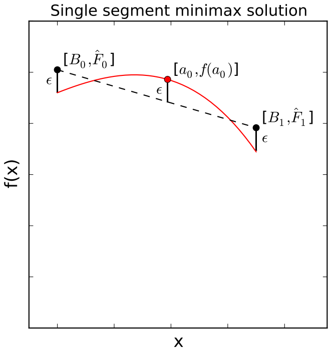
Breakpoint selection: Single function approximation



$$\epsilon := \max_{x \in \mathcal{D}} |f(x) - f(\hat{x})|$$

$$= |f(B_o) - \hat{F}_o|$$

Breakpoint selection: Single function approximation

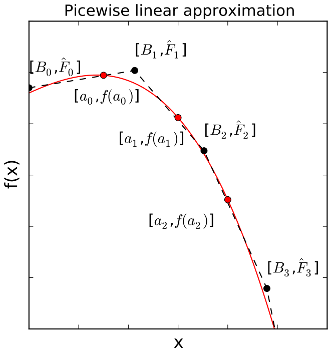


$$\epsilon := \max_{x \in \mathcal{D}} |f(x) - f(\hat{x})|$$

$$= |f(B_o) - \hat{F}_o|$$

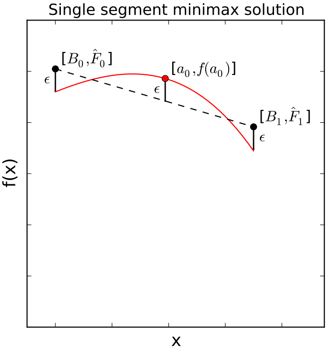
$$= |\hat{F}_o + f'(a_o)(a_o - B_o) - f(a_o)|$$

Breakpoint selection: Single function approximation



$$
\begin{aligned}
\epsilon &:= \max_{x \in \mathcal{D}} |f(x) - f(\hat{x})| \\
&= |f(B_o) - \hat{F}_o| && \forall o \in \mathcal{O} \\
&= |\hat{F}_o + f'(a_o)(a_o - B_o) - f(a_o)| && \forall o \in \mathcal{O}
\end{aligned}
$$

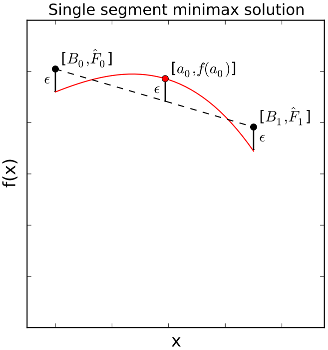
Single function approximation



MIP Formulation

$$\min_{B,a,\hat{F}} \quad \epsilon$$

Single function approximation

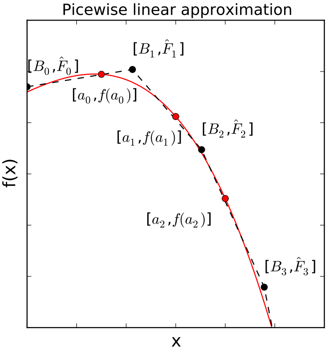


MIP Formulation

$$\min_{B,a,\hat{F}} \quad \epsilon$$

$$\hat{F}_{o+1} - \hat{F}_o = \; f'(a_o)(B_{o+1} - B_o)$$

Single function approximation



MIP Formulation

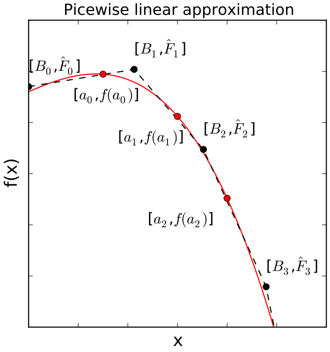
$$\min_{B,a,\hat{F}} \quad \epsilon$$

$$\hat{F}_{o+1} - \hat{F}_o = f'(a_o)(B_{o+1} - B_o)$$

$$\epsilon = \left| \hat{F}_{o+1} - f(B_o) \right|$$

$$\epsilon = \left| \hat{F}_o + f'(a_o)(a_o - B_o) - f(a_o) \right|$$

Single function approximation



MIP Formulation

$$\min_{B,a,\hat{F}} \quad \epsilon$$

$$\hat{F}_{o+1} - \hat{F}_o = f'(a_o)(B_{o+1} - B_o)$$

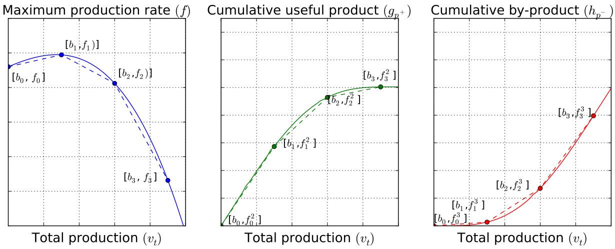
$$\epsilon = \left|\hat{F}_{o+1} - f(B_o)\right|$$

$$\epsilon = \left|\hat{F}_o + f'(a_o)(a_o - B_o) - f(a_o)\right|$$

$$a_o \leq a_{o+1}$$

$$a_{o-1} \leq B_o \leq a_o$$

Multiple functions

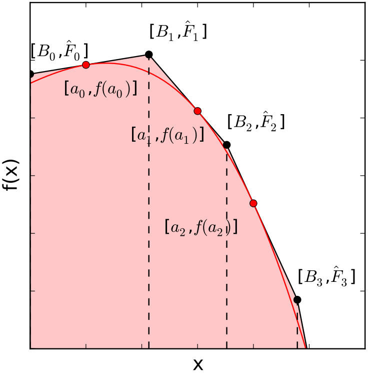


MIP Formulation

- Share the same variables $B_o \quad \forall o \in \mathcal{O}$.
- Objective function changed to a weighted error.

Single function relaxation

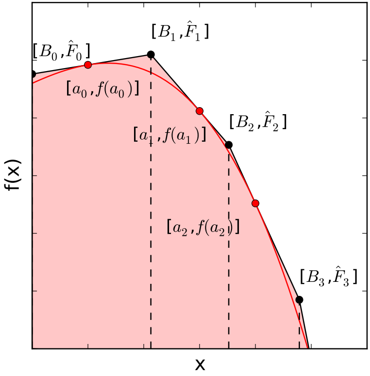
Piecewise linear relaxations that minimize area difference to the nonlinear function.



Relaxing convex function $f(\cdot)$

Single function relaxation

Piecewise linear relaxations that minimize area difference to the nonlinear function.

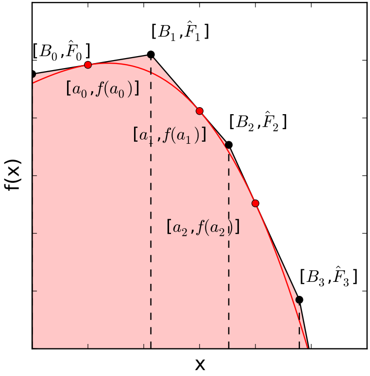


Relaxing convex function $f(\cdot)$

$$\min_{B,\hat{F},\hat{H},a} \sum_{o \in O} (\hat{F}_o + \hat{F}_{o-1})(B_o - B_{o-1}) - \int_0^{M_v} f(s)ds$$

Single function relaxation

Piecewise linear relaxations that minimize area difference to the nonlinear function.



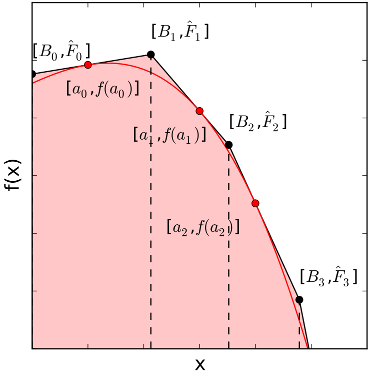
Relaxing convex function $f(\cdot)$

$$\min_{B,\hat{F},\hat{H},a} \sum_{o \in O} (\hat{F}_o + \hat{F}_{o-1})(B_o - B_{o-1})$$

$$\hat{F}_o - f(a_{o+1}) = f'(a_{o+1})(B_o - a_{o+1})$$
$$\hat{F}_o - f(a_o) = f'(a_o)(B_o - a_o)$$

Single function relaxation

Piecewise linear relaxations that minimize area difference to the nonlinear function.



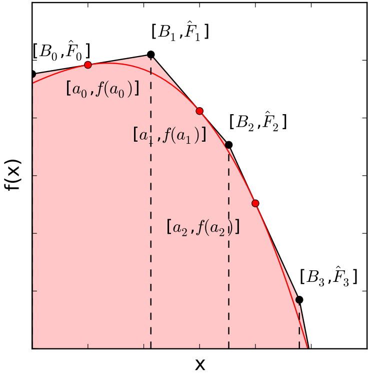
Relaxing convex function $f(\cdot)$

$$\min_{B,\hat{F},\hat{H},a} \sum_{o \in O} (\hat{F}_o + \hat{F}_{o-1})(B_o - B_{o-1})$$

$$\begin{aligned} \hat{F}_o - f(a_{o+1}) &= f'(a_{o+1})(B_o - a_{o+1}) \\ \hat{F}_o - f(a_o) &= f'(a_o)(B_o - a_o) \\ a_o &\leq a_{o+1} \\ a_{o-1} \leq B_o &\leq a_o \end{aligned}$$

Single function relaxation

Piecewise linear relaxations that minimize area difference to the nonlinear function.



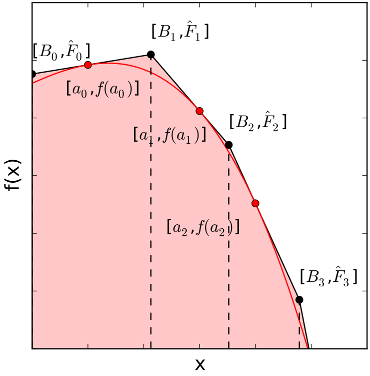
Relaxing convex function $f(\cdot)$

$$\min_{B, \hat{F}, \hat{H}, a} \sum_{o \in O} (\hat{F}_o + \hat{F}_{o-1})(B_o - B_{o-1})$$

$$\begin{aligned}
\hat{F}_o - f(a_{o+1}) &= f'(a_{o+1})(B_o - a_{o+1}) \\
\hat{F}_o - f(a_o) &= f'(a_o)(B_o - a_o) \\
a_o &\le a_{o+1} \\
a_{o-1} \le B_o &\le a_o \\
f(B_o) &\le \hat{F}_o
\end{aligned}$$

Single function relaxation

Piecewise linear relaxations that minimize area difference to the nonlinear function.



Relaxing convex function $f(\cdot)$

$$\min_{B,\hat{F},\hat{H},a} \sum_{o \in O} (\hat{F}_o + \hat{F}_{o-1})(B_o - B_{o-1})$$

$$\begin{aligned} \hat{F}_o - f(a_{o+1}) &= f'(a_{o+1})(B_o - a_{o+1}) \\ \hat{F}_o - f(a_o) &= f'(a_o)(B_o - a_o) \\ a_o &\leq a_{o+1} \\ a_{o-1} \leq B_o &\leq a_o \\ f(B_o) &\leq \hat{F}_o \end{aligned}$$

Extension to multiple functions similar to the approximation case.

Performance Evaluation

Piecewise linear approximation: Accuracy

Pointwise function evaluation error between the piecewise linear approximations and the non-linear functions over 592 instances.

Benchmarks

- Our method: NLP formulation using common breakpoints.

Piecewise linear approximation: Accuracy

Pointwise function evaluation error between the piecewise linear approximations and the non-linear functions over 592 instances.

Benchmarks

- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.

Piecewise linear approximation: Accuracy

Pointwise function evaluation error between the piecewise linear approximations and the non-linear functions over 592 instances.

Benchmarks

- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.
- Uniform: Uniformly spaced common breakpoints.

Piecewise linear approximation: Accuracy

Pointwise function evaluation error between the piecewise linear approximations and the non-linear functions over 592 instances.

Benchmarks

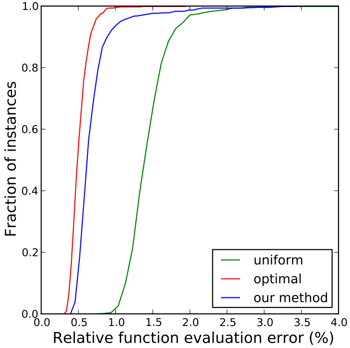
- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.
- Uniform: Uniformly spaced common breakpoints.

Metrics

- Maximum relative difference in function evaluation.

Piecewise linear approximation: Accuracy

Pointwise function evaluation error between the piecewise linear approximations and the non-linear functions over 592 instances.



Benchmarks

- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.
- Uniform: Uniformly spaced common breakpoints.

Metrics

- Maximum relative difference in function evaluation.

Piecewise linear relaxations: Accuracy

Relative area difference between the piecewise linear relaxations and the non-linear functions over 592 instances.

Benchmarks

- Our method: NLP formulation using common breakpoints.

Piecewise linear relaxations: Accuracy

Relative area difference between the piecewise linear relaxations and the non-linear functions over 592 instances.

Benchmarks

- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.

Piecewise linear relaxations: Accuracy

Relative area difference between the piecewise linear relaxations and the non-linear functions over 592 instances.

Benchmarks

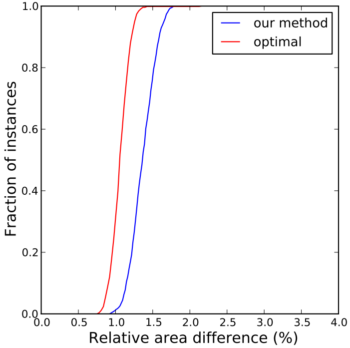
- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.

Metrics

- Relative difference in area under the curve.

Piecewise linear relaxations: Accuracy

Relative area difference between the piecewise linear relaxations and the non-linear functions over 592 instances.



Benchmarks

- Our method: NLP formulation using common breakpoints.
- Optimal: NLP formulation using separate breakpoints.

Metrics

- Relative difference in area under the curve.