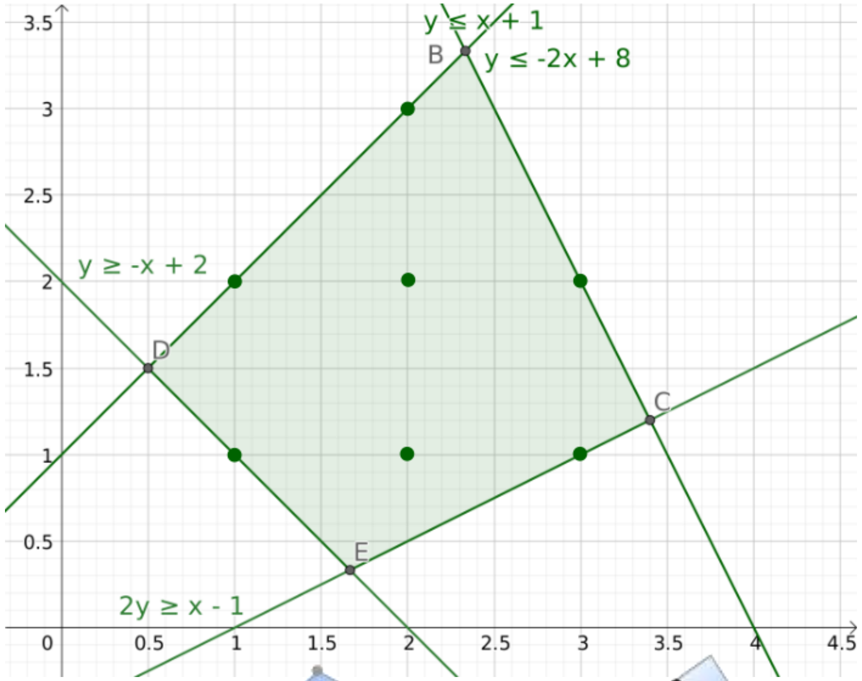
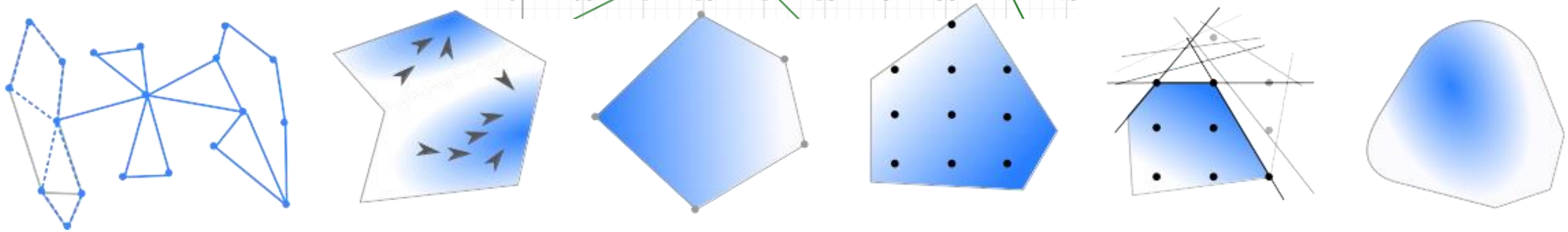
Decomposition 2

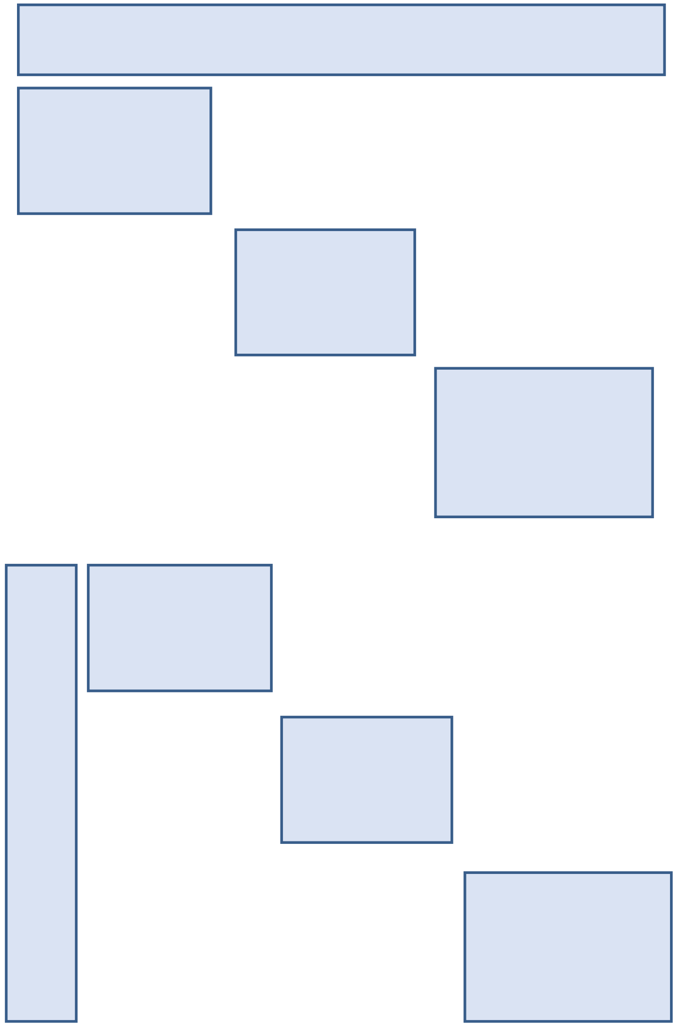
COMP4691/8691





Decomposition Topic Outline

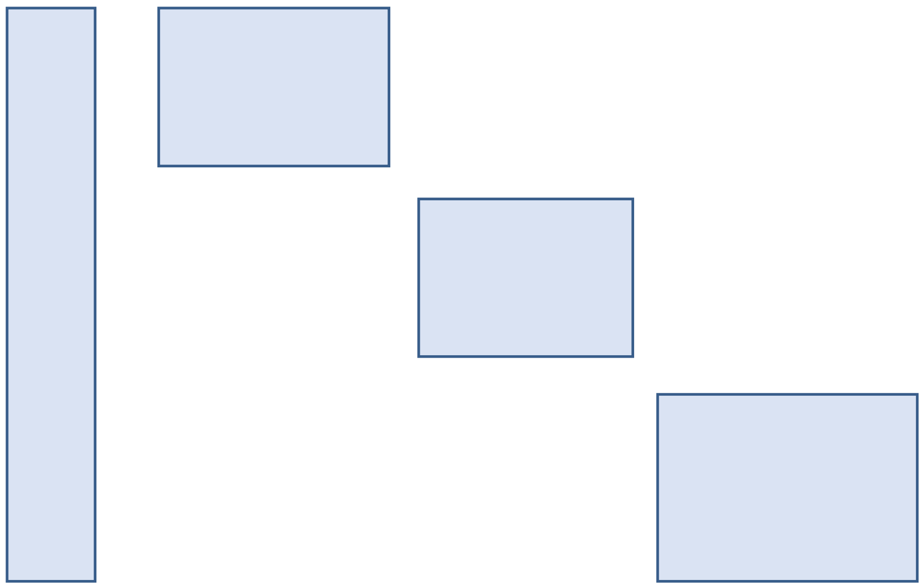
- Column Generation
- **Bender's Decomposition**
  - Facility Location problem
  - Derivation
  - Numerical Example
  - Connection with Column Generation
  - Real World Application: BusPlus



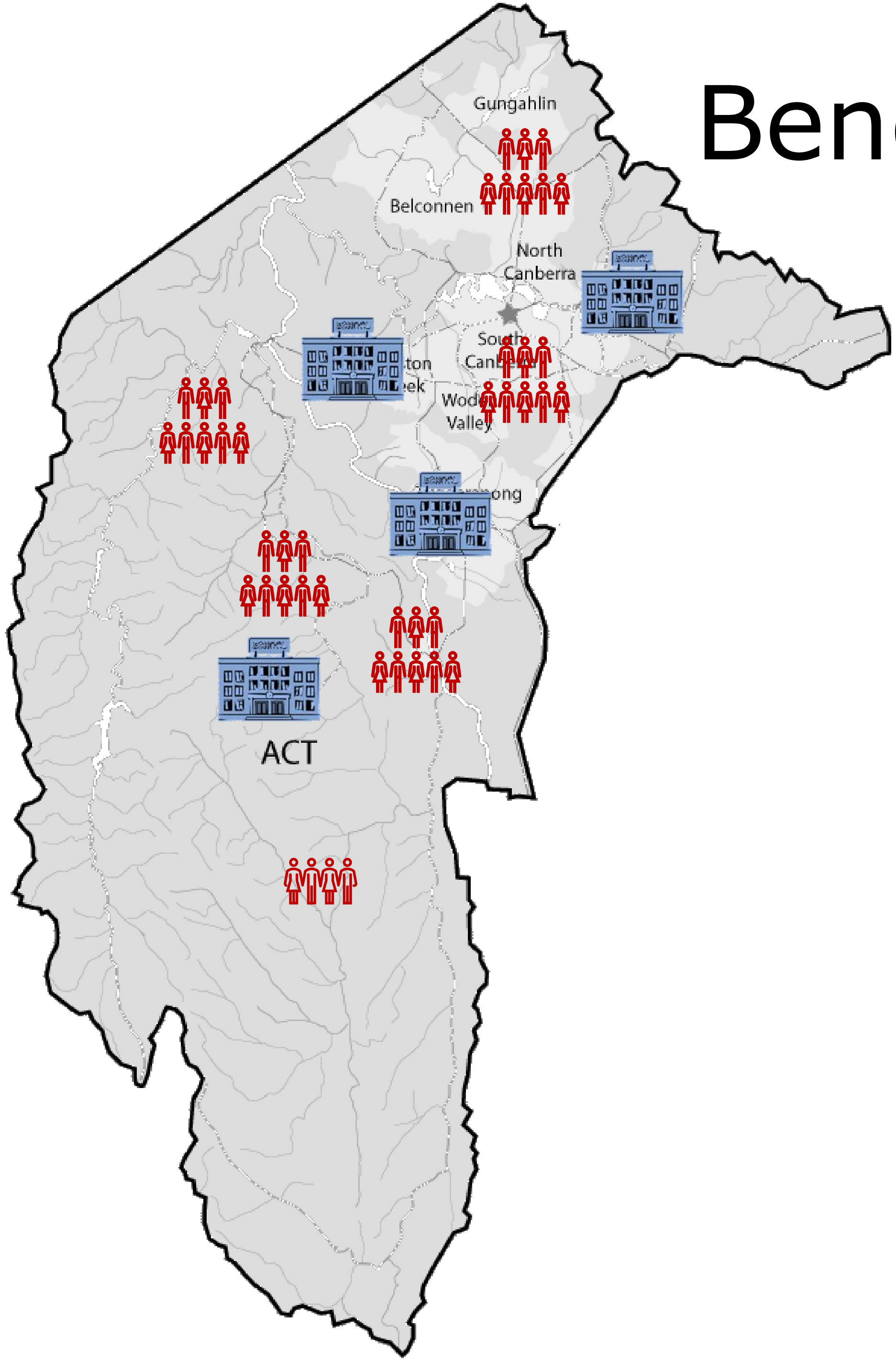
Benders Decomposition

- Benders is aimed at problems with *complicating variables*
- It works by generating new constraints!
- It applies to many MILP problems, but like Dantzig Wolfe/Column Generation, is very effective for block structured problems
- It has been used very effectively in Stochastic Optimisation, where it is known as “the L-shaped method”
- First described by Jacques Benders in 1962

[1] Benders, 1962 J.F. Benders **Partitioning procedures for solving mixed-variables programming problems** Numerische Mathematik, 4 (1) (1962), pp. 238-252



Benders: Intuition



Some problems have 2 parts – easy and hard.

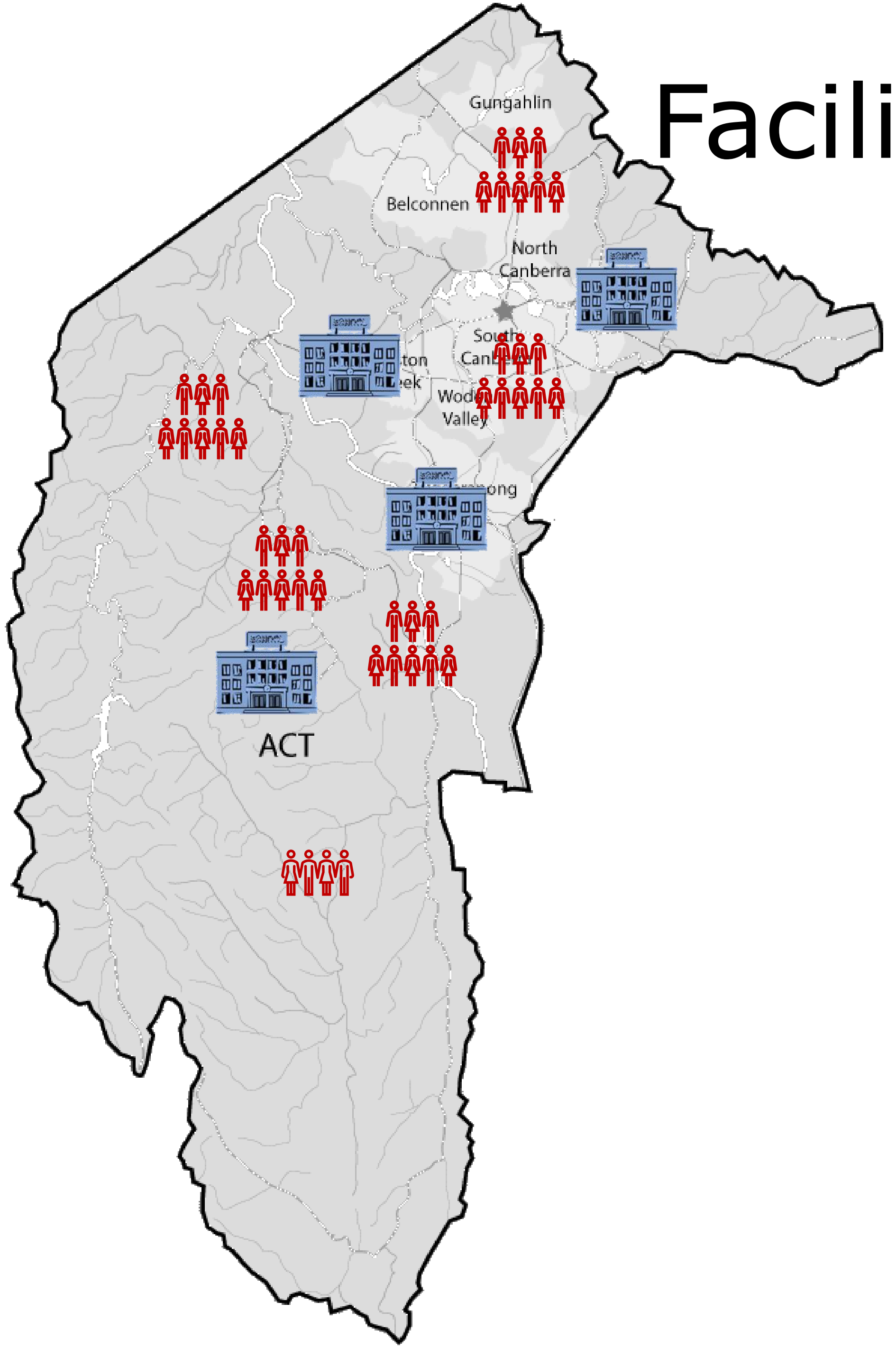
- If I knew the answer to the *hard* part, then the rest is easy.

Example: Facility Location

- Place *m* schools to serve *n* families at minimum cost

Let's think about a location k at which we want to build a facility. **What are the costs**?

- Cost of building a facility at location k: b_k
- Cost of transport from customer i to facility k is c_{ik}



Facility Location (1)

Place *m* schools to serve *n* families at minimum cost

What are the costs?

- Cost of building a facility at location k: b_k
- Cost of transport from customer i to facility k is c_{ik}

Variables:

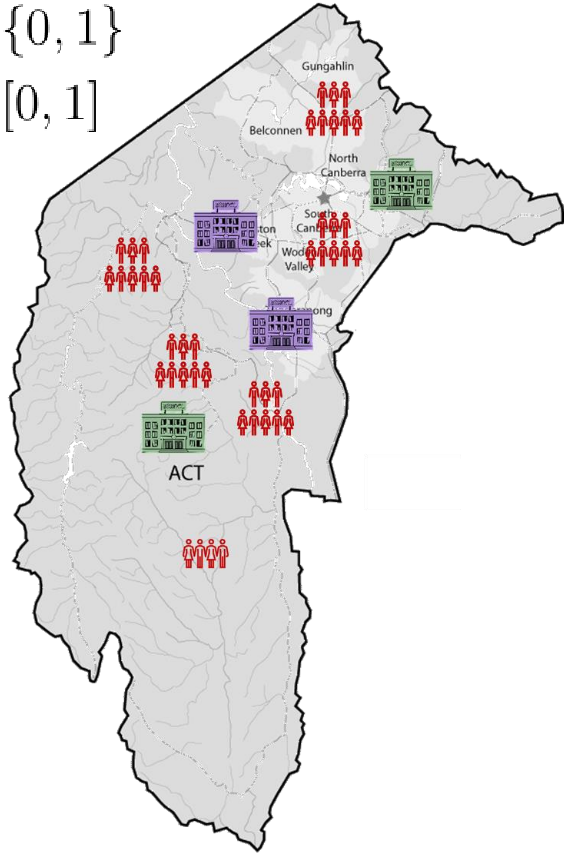
- y_k = 1 if we build a facility at location k, 0 otherwise
- x_{ik} = proportion of demand of customer i served by facility at k.

$$
\begin{array}{rcl}
\min & & \sum_{i} b_k y_k + \sum_{ik} c_{ik} x_{ik} \\
\text{subject to} & & \\
\sum_{k} x_{ik} & = & 1 \quad \forall i \\
x_{ik} & \leq & y_k \quad \forall i, k \\
y_k & \in & \{0, 1\} \\
x_{ik} & \in & [0, 1]
\end{array}
$$

Facility Location (2)

- $\mathcal{NP}$ hard problem BUT
- If I knew which locations I was building at, then the x_{ik} problem is really easy:
  - **Use the closest facility!**
- So the *y* variables are the “hard” variables
- Benders proceeds by “*guessing*” a *y* solution, and then generating constraints to force **feasibility** and **optimality**
- The pool of potential constraints is exponential, so convergence can be slow
- … but there is a long list of problems where it works well

$$
\begin{array}{rcl}
\min & & \sum_i b_k y_k + \sum_{ik} c_{ik} x_{ik} \\
\text{subject to} & & \\
\sum_k x_{ik} & = & 1 \quad \forall i \\
x_{ik} & \leq & y_k \quad \forall i, k \\
y_k & \in & \{0, 1\} \\
x_{ik} & \in & [0, 1]
\end{array}
$$



Benders: Intuition

So for the *original problem* (**OP**)

OP:

$$\begin{aligned} \text{minimise} \quad & f^\intercal y + c^\intercal x \\ \text{subject to} \quad & Ay = b \\ & By + Dx = d \\ & x \geq 0 \\ & y \in \mathbb{Z}^+ \end{aligned}$$

Benders solves the problem in *x* for a fixed value of *y*

Benders: Derivation

OP:

$$\begin{aligned} \text{minimise} \quad & f^\intercal y + c^\intercal x \\ \text{subject to} \quad & Ay = b \\ & By + Dx = d \\ & x \geq 0 \\ & y \in \mathbb{Z}^+ \end{aligned}$$

$$Y = \{y \mid Ay = b, y \in \mathbb{Z}^+\}$$

OP can be rewritten as

$$\min_{\hat{y} \in Y}\{f^\intercal \hat{y} + \min_{x \geq 0}\{c^\intercal x : Dx = d - B\hat{y}\}\}$$

Benders: Derivation

$$\min_{\hat{y}\in Y}\{f^{\intercal}\hat{y}+\min_{x\geq 0}\{c^{\intercal}x : Dx = d-B\hat{y}\}\}$$

This gives us our subproblem (SP) **for a given choice of $\hat{y}$**

SP: $\left\{ \begin{array}{l} \min\limits_{x\geq 0} c^{\mathrm{T}}x \\ \text{s.t. } Dx = d - B\hat{y} \quad (\pi) \end{array} \right.$ $\xrightarrow{\text{dual}}$ **DSP:** $\left\{ \begin{array}{l} \max\limits_{\pi} \pi^{\mathrm{T}}(d - B\hat{y}) \\ \text{s.t. } \pi^{\mathrm{T}}D \leq c \end{array} \right.$

DSP: $\max\limits_{\pi}\{\pi^{\intercal}(d-B\hat{y}) : \pi^{\intercal}D \leq c\}$

Benders: Derivation

SP: $\min_{x \geq 0} \{c^\intercal x : Dx = d - B\hat{y}\}$

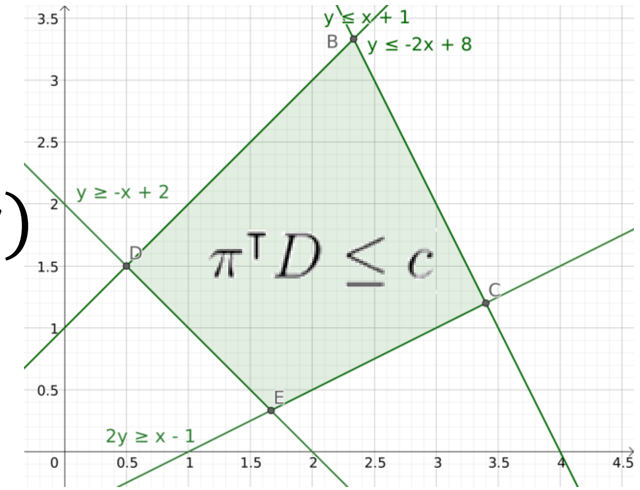
DSP: $\max_{\pi} \{\pi^\intercal (d - B\hat{y}) : \pi^\intercal D \leq c\}$

- We like dual subproblem (**DSP**) because $\hat{y}$ **only appears in the objective**, not the constraints
- For a given $\hat{y}$ there are two options:
  - **DSP is feasible**
  - **DSP is unbounded → SP is infeasible**

Benders: Derivation – DSP is feasible

DSP: $$\max_{\pi}\{\pi^\intercal(d-B\hat{y}) : \pi^\intercal D \leq c\}$$

- If feasible, π_e that achieves the maximum is a potential solution
- The maximum of the DSP **for any y** is **at least** $\pi_e(d-By)$
  - Why? Because π_e is a feasible point and **the constraints never change**
- We can theoretically enumerate every extreme point (vertex) $\pi_e \in E$



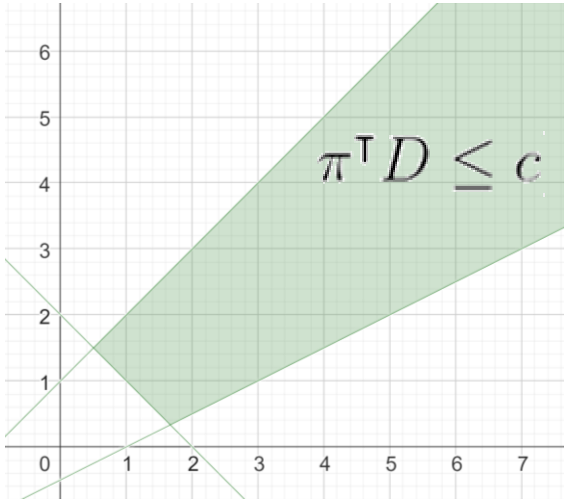
$$\min_{\hat{y}\in Y}\{f^\intercal\hat{y}+\min_{x\geq 0}\{c^\intercal x : Dx = d-B\hat{y}\}\}$$

$$\left\{\begin{array}{l} \min_{y,\eta} f^T y + \eta \\ \eta \geq \pi_e(d-By) \quad \forall \pi_e \in E \end{array}\right.$$

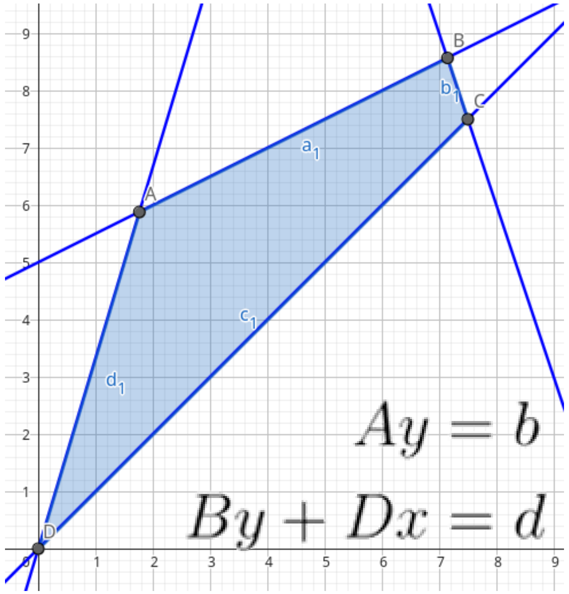
Optimality cuts

Benders: Derivation – DSP is Unbounded

DSP: $\max_{\pi}\{\pi^\intercal(d - B\hat{y}) : \pi^\intercal D \leq c\}$



- If unbounded, this means $\hat{y}$ is infeasible.
- It means there is an *extreme ray* r_q such that $r_q(d - B\hat{y})$ is unbounded.
- We can add the cut $r_q(d - By) \leq 0$ to remove $\hat{y}$
  - **Called a *feasibility cut***
- Theoretically, can add one cut for every $r_q \in Q$ for every infeasible $\hat{y}$.



Benders: Derivation

Benders Master Problem

BMP:

$$
\begin{aligned}
\min_{y,\eta} \quad & f^{\intercal} y + \eta \\
\text{subject to} \quad & Ay = b \\
& \eta \geq \pi_e^{\intercal}(d - By) \quad \forall \pi_e \in E && \text{Optimality cuts} \\
& 0 \geq r_q^{\intercal}(d - By) \quad \forall r_q \in Q && \text{Feasibility cuts} \\
& y \in \mathbb{Z}^+
\end{aligned}
$$

Benders: Derivation

- Just as for Column Generation, we do not try to enumerate all of set *E* and *Q*
- Instead, we solve a *Restricted Master Problem* (RMP)
- We solve with the optimality and feasibility cuts generated so far, $\bar{E}$ and $\bar{Q}$

RMP:

$$
\begin{aligned}
\min_{y,\eta} \quad & f^\intercal y + \eta \\
\text{subject to} \quad & Ay = b \\
& \eta \geq \pi_e^\intercal (d - By) \quad \forall \pi_e \in \bar{E} && \text{Optimality cuts} \\
& 0 \geq r_q^\intercal (d - By) \quad \forall r_q \in \bar{Q} && \text{Feasibility cuts} \\
& y \in \mathbb{Z}^+
\end{aligned}
$$

Benders: Derivation

Input: y_0 initial solution, ϵ tolerance
Output: Solution x^*, z^* within ϵ of optimal

UB $= \infty$, LB $= -\infty$
$\bar{E} = \emptyset$, $\bar{Q} = \emptyset$
$\hat{y} = y_0$
while UB - LB $\geq \epsilon$ **do**
  solve $\max_\pi \{\pi^\intercal (d - B\hat{y}) : \pi^\intercal D \leq c\}$ ▷ Solve subproblem
  if Unbounded **then**
    Get extreme ray r_q
    $\bar{Q} = \bar{Q} \cup \{r_q\}$
  else
    Get extreme point π_e
    $\bar{E} = \bar{E} \cup \{\pi_e\}$
    UB $=$ min (UB, $f^\intercal \hat{y} + \pi_e^\intercal (d - B\hat{y})$)
  end if
  solve $z = \min_{y \in Y}(f^\intercal y + \eta : \bar{E}$ and $\bar{Q}$ cuts) ▷ Solve reduced master problem
  $\hat{y} = \mathrm{argmin}(y)$
  LB $= z^*$
end while

Example

$$\begin{aligned} \min \quad & 2y + 2x_1 + 3x_2 \\ \text{subject to} \quad & y_1 + x_1 + 2x_2 \geq 3 \\ & 3y_1 + 2x_1 - x_2 \geq 4 \\ & x \geq 0, y \geq 0 \end{aligned}$$

$$\begin{aligned} \text{minimise} \quad & f^\intercal y + c^\intercal x \\ \text{subject to} \quad & Ay = b \\ & By + Dx = d \\ & x \geq 0 \\ & y \in \mathbb{Z}^+ \end{aligned}$$

f =[2] c^T = [2 3] $B = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$ $D = \begin{pmatrix} 1 & 2 \\ 2 & -1 \end{pmatrix}$ $d = \begin{pmatrix} 3 \\ 4 \end{pmatrix}$

Iteration 1a

- Let $\hat{y}$ = 0
- Solve subproblem

$$\max\ 2(0) + \pi^\intercal \left(\begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 1 \\ 3 \end{bmatrix} (0) \right)$$

$$\text{subject to } \pi^\intercal \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix} \leq \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$\pi \geq 0$$

- The optimal solution is $\pi^\intercal_e$ = [1.6, 0.2], giving UB 5.6
- Add cut $\eta \geq \pi_e^\intercal (d - By)$

$$\eta \geq [1.6\ 0.2] \left(\begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 1 \\ 3 \end{bmatrix} y \right)$$

$$\eta \geq 5.6 - 2.2y$$

Input: y_0 initial solution, ϵ tolerance
Output: Solution x^*, z^* within ϵ of optimal

UB $:= \infty$, LB $:= -\infty$
$\bar{E} := \emptyset$, $\bar{Q} := \emptyset$
$\hat{y} := y_0$
while UB - LB $\geq \epsilon$ **do**
  solve $\max_\pi \{\pi^\intercal (d - B\hat{y}) : \pi^\intercal D \leq c\}$
  if Unbounded **then**
    Get unbounded ray r_q
    $\bar{Q} := \bar{Q} \cup \{r_q\}$
  else
    Get extreme point π_e
    $\bar{E} := \bar{E} \cup \{\pi_e\}$
    UB $:= \min$ (UB, $f^\intercal \hat{y} + \pi_e^\intercal (d - B\hat{y})$)
  end if
  solve $z = \min_{y \in Y} (f^\intercal y + \eta :$ cuts)
  $\hat{y} = \text{argmin}(y)$
  LB $:= z^*$
end while

$$f = [2]$$
$$c^\intercal = [2,\ 3]$$
$$B = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$
$$D = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}$$
$$d^\intercal = [3,\ 4]$$

Iteration 1b

- Solve the reduced master problem

$$\begin{aligned} \min\ & 2y + \eta \\ \text{s.t.}\ & \eta \geq 5.6 - 2.2y \\ & y \geq 0 \end{aligned}$$

- Optimal solution $(\hat{y}, \eta) = (2.545, 0), LB = 5.091$

Input: y_0 initial solution, ϵ tolerance
Output: Solution x^*, z^* within ϵ of optimal

UB $:= \infty$, LB $:= -\infty$
$\bar{E} := \emptyset$, $\bar{Q} := \emptyset$
$\hat{y} := y_0$
while UB - LB $\geq \epsilon$ **do**
    solve $\max_\pi \{\pi^\intercal (d - B\hat{y}) : \pi^\intercal D \leq c\}$
    if Unbounded **then**
        Get unbounded ray r_q
        $\bar{Q} := \bar{Q} \cup \{r_q\}$
    else
        Get extreme point π_e
        $\bar{E} := \bar{E} \cup \{\pi_e\}$
        UB $:= \min$ (UB, $f^\intercal \hat{y} + \pi_e^\intercal (d - B\hat{y})$)
    end if
    solve $z = \min_{y \in Y} (f^\intercal y + \eta :$ cuts $)$
    $\hat{y} = \text{argmin}(y)$
    LB $:= z^*$
end while

$$\begin{aligned} f &= [2] \\ c^\intercal &= [2,\ 3] \\ B &= \begin{bmatrix} 1 \\ 3 \end{bmatrix} \\ D &= \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix} \\ d^\intercal &= [3,\ 4] \end{aligned}$$

Iteration 2a

- Solve the sub problem

$$\max\ 2(2.545) + \pi^\intercal \left(\begin{bmatrix} 3 \\ 4 \end{bmatrix} - \begin{bmatrix} 1 \\ 3 \end{bmatrix} (2.545) \right)$$

$$\text{subject to } \pi^\intercal \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix} \leq \begin{bmatrix} 2 \\ 3 \end{bmatrix}$$

$$\pi \geq 0$$

- Optimal solution is $\pi^\intercal_e$ = [1.5, 0], giving UB 5.772

- Add cut $\eta \geq 4.5 - 1.5y$

Input: y_0 initial solution, ϵ tolerance
Output: Solution x^*, z^* within ϵ of optimal

UB := ∞, LB := $-\infty$
$\bar{E} := \emptyset$, $\bar{Q} := \emptyset$
$\hat{y} := y_0$
while UB - LB $\geq \epsilon$ **do**
  solve $\max_\pi \{\pi^\intercal (d - B\hat{y}) : \pi^\intercal D \leq c\}$
  if Unbounded **then**
    Get unbounded ray r_q
    $\bar{Q} := \bar{Q} \cup \{r_q\}$
  else
    Get extreme point π_e
    $\bar{E} := \bar{E} \cup \{\pi_e\}$
    UB := min (UB, $f^\intercal \hat{y} + \pi_e^\intercal (d - B\hat{y})$)
  end if
  solve $z = \min_{y \in Y} (f^\intercal y + \eta :$ cuts)
  $\hat{y} = \text{argmin}(y)$
  LB := z^*
end while

$$f = [2]$$
$$c^\intercal = [2,\ 3]$$
$$B = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$
$$D = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}$$
$$d^\intercal = [3,\ 4]$$

Iteration 2b

- Solve the reduced master problem

$$\begin{aligned} \min\ & 2y + \eta \\ \text{s.t.}\ & \eta \geq 5.6 - 2.2y \\ & \eta \geq 4.5 - 1.5y \\ & y \geq 0 \end{aligned}$$

- Optimal solution $(\hat{y}, \eta) = (1.571, 2.143), LB = 5.286$

Input: y_0 initial solution, ϵ tolerance
Output: Solution x^*, z^* within ϵ of optimal

```
UB := ∞, LB := −∞
Ē := ∅, Q̄ := ∅
ŷ := y₀
while UB - LB ≥ ϵ do
    solve max_π {πᵀ(d − Bŷ) : πᵀD ≤ c}
    if Unbounded then
        Get unbounded ray r_q
        Q̄ := Q̄ ∪ {r_q}
    else
        Get extreme point π_e
        Ē := Ē ∪ {π_e}
        UB := min (UB, fᵀŷ + π_eᵀ(d − Bŷ))
    end if
    solve z = min_{y∈Y}(fᵀy + η : cuts )
    ŷ = argmin(y)
    LB := z*
end while
```

$$f = [2]$$
$$c^\intercal = [2,\ 3]$$
$$B = \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$
$$D = \begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix}$$
$$d^\intercal = [3,\ 4]$$

Iteration 3a

- Solve the sub problem

$$\max\ 2(1.571) + \pi^{\intercal}\left(\begin{bmatrix}3\\4\end{bmatrix} - \begin{bmatrix}1\\3\end{bmatrix}(1.571)\right)$$

$$\text{subject to } \pi^{\intercal}\begin{bmatrix}1 & 2\\2 & -1\end{bmatrix} \leq \begin{bmatrix}2\\3\end{bmatrix}$$

$$\pi \geq 0$$

- Optimal solution is $\pi^{\intercal}_e$ = [1.6, 0.2], giving UB 5.286
- LB == UB, so we can stop.
- Solution is (y=1.571, $\underbrace{x_1=0,\ x_2=0.714}$), with value 5.286

Obtained using $Dx = d - B\hat{y}$

Input: y_0 initial solution, ϵ tolerance
Output: Solution x^*, z^* within ϵ of optimal

```
UB := ∞, LB := −∞
Ē := ∅, Q̄ := ∅
ŷ := y_0
while UB - LB ≥ ϵ do
    solve max_π{π^T(d − Bŷ) : π^T D ≤ c}
    if Unbounded then
        Get unbounded ray r_q
        Q̄ := Q̄ ∪ {r_q}
    else
        Get extreme point π_e
        Ē := Ē ∪ {π_e}
        UB := min (UB, f^T ŷ + π_e^T(d − Bŷ))
    end if
    solve z = min_{y∈Y}(f^T y + η : cuts )
    ŷ = argmin(y)
    LB := z*
end while
```

$$f = [2]$$

$$c^{\intercal} = [2,\ 3]$$

$$B = \begin{bmatrix}1\\3\end{bmatrix}$$

$$D = \begin{bmatrix}1 & 2\\2 & -1\end{bmatrix}$$

$$d^{\intercal} = [3,\ 4]$$

Benders Decomposition

- No integer variables in the subproblem
  - We use the dual to solve the subproblem
  - The value of dual variables is not well-defined if we have integer constraints
  - The algorithm as presented cannot be used
- Subject to slow convergence,
- Many improvements, tricks and add-ons developed.
- For a recent and excellent review, see:

Rahmaniani, Ragheb & Crainic, Teodor Gabriel & Gendreau, Michel & Rei, Walter. (2017). **The Benders Decomposition Algorithm: A Literature Review**. *European Journal of Operational Research* (259) pp 801-817. doi:10.1016/j.ejor.2016.12.005

Relationship between CG and BD

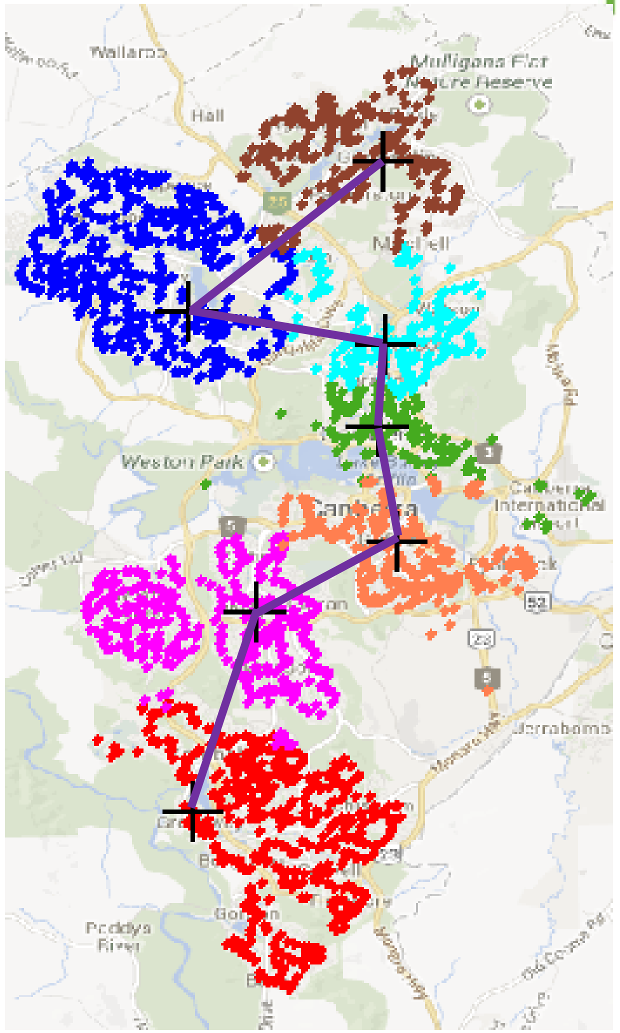
- If both problems are all-continuous (no integer variables) then CG and BD are indeed duals
  - The complicating variables in the CG become the complicating constraints in BD
  - The subproblems are equivalent.

- However, in the presence of integer constraints, the relationship is complex.
  - BD directly converges to a solution of the MIP, not a relaxation
    - No additional B&B tree
  - BD can’t handle integer constraints in the subproblem
    - CG is happy with that

Application: BusPlus (1)

Design a hybrid, on-demand public transport service for off-peak hours

- Buses run trunk routes between *hubs* (e.g. between town centres in Canberra)
- Multi-hire taxis provide transport from bus stops to/from hubs
- Use app to book travel, as close as 10 mins before you want to leave
- Developed by Dr. Phil Kilby and his team at NICTA in 2014



Application: BusPlus (1)

Design problem: Which Hubs? Which Routes?

- Objective: Minimise combination of operating cost + customer travel time
- Decision vars:
  - $z_i = 1 \rightarrow$ bus stop *i* is a hub
  - $y_{ij} = 1 \rightarrow$ open route between hubs *i* and *j*.
  - $x_{pijq} = 1 \rightarrow$ Customer trip from bus stop *p* to bus stop *q* uses route between *i* and *j*

Benders:

- Once you set *y* and *z* vars (choose hub and legs), *x* vars are easy (shortest path problem)
- Shortest path subproblems tell RMP the "value" of legs.
- Solved using BD for ≈3000 bus stops in about 100 iterations (15 mins) and 1% optimality gap