École des Ponts ParisTech

Optimizing crew rotations for an airline

CERMICS

Axel Parmentier

May 24th, 2018

Airline operations problems

Schedule planning

Select flight legs operated

↓ Legs operated

Fleet assignment

Choose fleet covering each leg

↓ Legs operated by a fleet

Aircraft routing

Choose Aircraft rotations

↓ Airplane rotations

Crew pairing

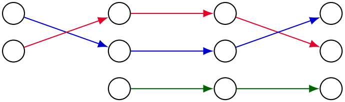
Choose Crew rotations

(ℓ_1) Paris 10:15 – New York

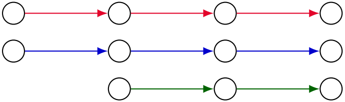
(ℓ_2) Paris 10:25 – Montreal

(ℓ_1) A380

(ℓ_2) A330



Rotation = sequence of flight legs

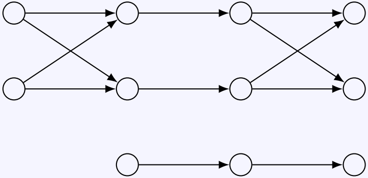


Aircraft Routing and Crew Pairing problems

Aircraft Routing Problem

○ Flight $v \in V$

⟶ Aircraft Connection



Feasible routes $r \in \mathcal{R}$

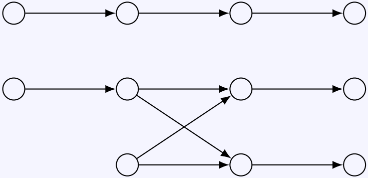
$$\min \sum_{r \in \mathcal{R}} c_r x_r$$

$$\text{s.t.} \quad \left| \begin{array}{ll} \sum_{r \ni v} x_r = 1 & \forall v \in V \\ x_r \in \{0,1\} & \forall r \in \mathcal{R} \end{array} \right.$$

Crew Pairing Problem

○ Flight $v \in V$

⟶ Crew Connection



Feasible pairings $p \in \mathcal{P}$

$$\min \sum_{p \in \mathcal{P}} c_p y_p$$

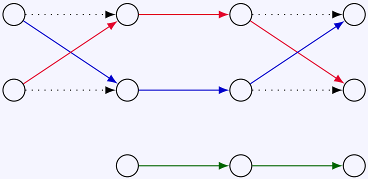
$$\text{s.t.} \quad \left| \begin{array}{ll} \sum_{p \ni v} y_p = 1 & \forall v \in V \\ y_p \in \{0,1\} & \forall p \in \mathcal{P} \end{array} \right.$$

Aircraft Routing and Crew Pairing problems

Aircraft Routing Problem

○ Flight $v \in V$

⟶ Aircraft Connection



Feasible routes $r \in \mathcal{R}$

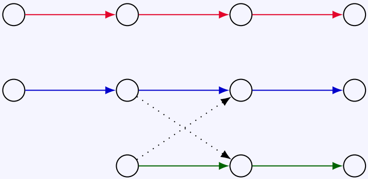
$$\min \sum_{r \in \mathcal{R}} c_r x_r$$

$$\text{s.t.} \quad \left| \begin{array}{ll} \sum_{r \ni v} x_r = 1 & \forall v \in V \\ x_r \in \{0,1\} & \forall r \in \mathcal{R} \end{array} \right.$$

Crew Pairing Problem

○ Flight $v \in V$

⟶ Crew Connection

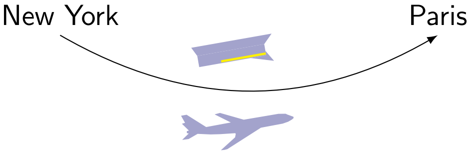


Feasible pairings $p \in \mathcal{P}$

$$\min \sum_{p \in \mathcal{P}} c_p y_p$$

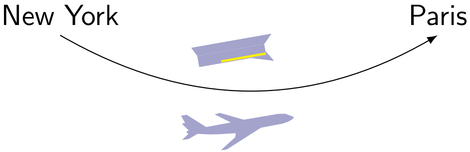
$$\text{s.t.} \quad \left| \begin{array}{ll} \sum_{p \ni v} y_p = 1 & \forall v \in V \\ y_p \in \{0,1\} & \forall p \in \mathcal{P} \end{array} \right.$$

Delay Propagation



Each flight must be operated by an airplane and a crew

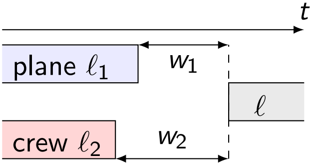
Delay Propagation



Each flight must be operated by an airplane and a crew

Delay propagation model

$$\xi_\ell = \max(\xi_{\ell_1} - w_1, \xi_{\ell_2} - w_2, 0) + \xi_\ell^{\mathrm{int}}$$



Probabilistic constraints on delay propagation

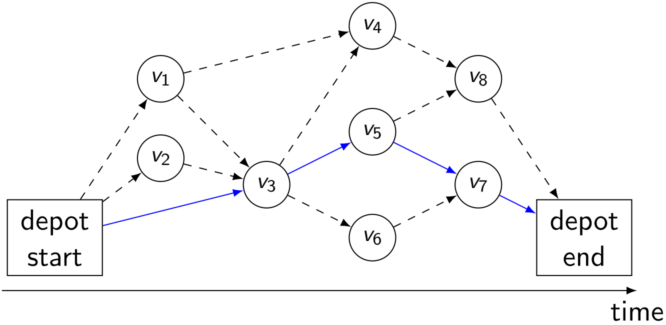
$$\mathbb{P}(\xi_\ell > \tau) \leq \alpha \quad \text{for all } \ell$$

Part content

1. Column generation for rotation problems
1.1 General method
1.2 What delay changes

2. An algebraic path problem framework

3. Stochastic paths problems and delay in rotation problems



$$\min \sum_{P \in \mathcal{P}} c_P x_P$$
$$\sum_{P \ni v} x_P = 1 \qquad \forall v$$
$$x_P \in \{0, 1\}$$

- Path cost not linear in arc costs
- Path must satisfy constraints

Constraint example

Limited number of arcs in P

Column generation primer

Restricted master problem $\mathcal{P}' \subset \mathcal{P}$, with $|\mathcal{P}'| \ll |\mathcal{P}|$

$$\begin{aligned} \min_{x} \quad & \sum_{P \in \mathcal{P}} c_P x_P \\ \mathrm{st} \quad & \sum_{P \ni v} x_v = 1 \quad \forall \ell \in \mathcal{L} \\ & x_P \geq 0 \end{aligned}$$

Column generation primer

Restricted master problem $\mathcal{P}' \subset \mathcal{P}$, with $|\mathcal{P}'| \ll |\mathcal{P}|$

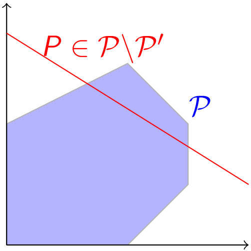
$$\begin{aligned} \min_x \quad & \sum_{P \in \mathcal{P}} c_P x_P \\ \text{st} \quad & \sum_{P \ni v} x_v = 1 \quad \forall \ell \in \mathcal{L} \\ & x_P \geq 0 \end{aligned}$$

Restricted dual problem

$$\begin{aligned} \max \quad & \sum_{v \in V} y_v \\ \text{s.t.} \quad & \sum_{v \in P} y_v \leq c_P \quad \forall P \in \mathcal{P}' \end{aligned}$$

Pricing subproblem

$$\min_{P \in \mathcal{P}} c_P - \sum_{v \in P} y_P$$

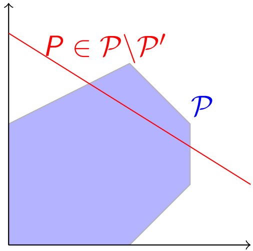


Algorithm:

- solve on $\mathcal{P}'$
- solve pricing subproblem
- add violated dual constraint to $\mathcal{P}'$

Restricted master problem $\mathcal{P}' \subset \mathcal{P}$, with $|\mathcal{P}'| \ll |\mathcal{P}|$

$$\min_x \sum_{P \in \mathcal{P}} c_P x_P$$
$$\text{st} \sum_{P \ni v} x_v = 1 \quad \forall \ell \in \mathcal{L}$$
$$x_P \geq 0$$



Restricted dual problem

$$\max \sum_{v \in V} y_v$$
$$\text{s.t.} \sum_{v \in P} y_v \leq c_P \quad \forall P \in \mathcal{P}'$$

Pricing subproblem

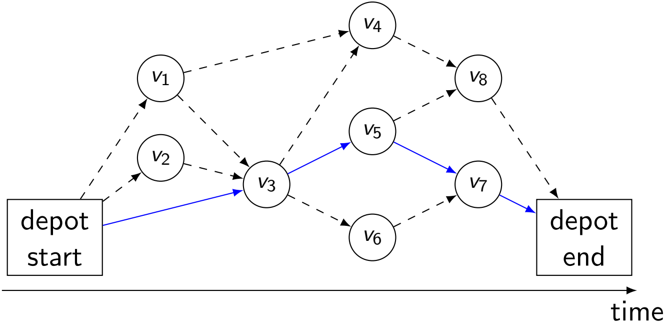
$$\min_{P \in \mathcal{P}} c_P - \sum_{v \in P} y_P$$

Algorithm:

- solve on $\mathcal{P}'$
- solve pricing subproblem
- add violated dual constraint to $\mathcal{P}'$

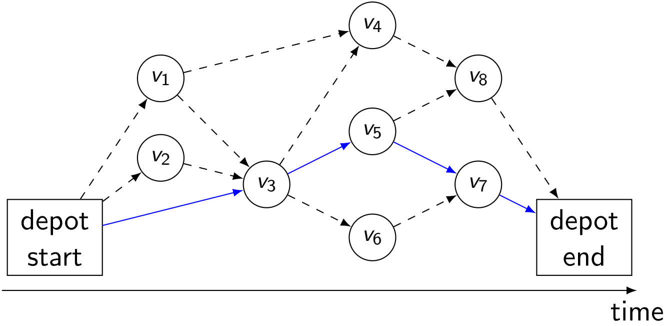
Key element in the performance: pricing subproblem algorithm

$$\min_{P \in \mathcal{P}} c_P - \sum_{v \in P} y_P$$



Resource constrained shortest path algorithm

$$\min_{P \in \mathcal{P}} c_P - \sum_{v \in P} y_P$$



Pricing subproblem is a resource constrained shortest path algorithm

What a good pricing algorithms changes – Airline crew pairing

Instance	$\|V\|$	Alg	RCSP time av (mm:ss)	Pricing time	Total time (hh:mm:ss)
CP50	290	LS	00:00.560	97.55%	00:04:37.5
		LC	00:01.275	97.38%	00:11:36.9
		Our A*	00:00.016	59.87%	00:00:17.2
CP70	408	LS	00:11.489	99.52%	05:07:05.0
		LC	00:17.157	99.56%	07:28:22.2
		Our A*	00:00.039	58.48%	00:01:12.1
CP90	516	LS	00:40.707	Stopped after 48h	
		LC	01:42.864	Stopped after 48h	
		Our A*	00:00.340	81.86%	00:12:36.3
A318	669	LS	00:53.009	Stopped after 48h	
		LC	01:36.035	Stopped after 48h	
		Our A*	00:01.651	86.97%	01:32:49.6

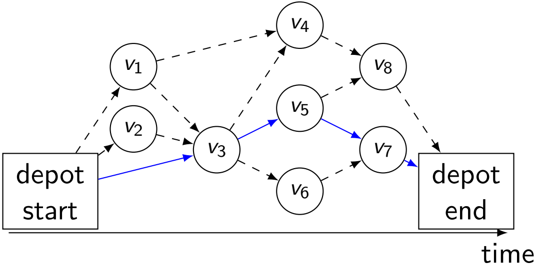
What delay changes for aircraft routing or crew pairing

Considering only aircraft or crews

$$\xi_\ell = \max(\xi_{\ell_2} - w_2, 0) + \xi_\ell^{\text{int}}$$

Monotonicity of $\cdot \mapsto \max(\cdot, 0)$ enables to use our framework

Delay is a pairing property: dealt with in the subproblem: *a stochastic resource constrained shortest path problem*



$$\min \sum_{P \in \mathcal{P}} c_P x_P$$

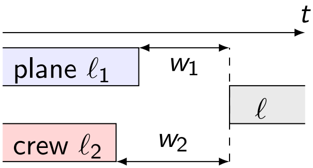
$$\sum_{P \ni v} x_P = 1 \qquad \forall v$$

$$\mathbb{P}(\xi_v > \tau) \leq \alpha \qquad \forall v \in V$$

$$x_P \in \{0, 1\}$$

Delay propagation model

$$\xi_\ell = \max(\xi_{\ell_1} - w_1, \xi_{\ell_2} - w_2, 0) + \xi_\ell^{\mathrm{int}}$$



When considering airplane and crews delay, we cannot hide delay anymore in the set of rotations $\mathcal{P}$.

Sequential resolution

Two solution schemes

Aircraft Routing

Column generation

Master problem

Solve LP on $\mathcal{R}' \subseteq \mathcal{R}$

cost ↓ ↑ rot.

Pricing subproblem

Update $\mathcal{R}'$ (path pb)

Short connect. → used

Crew Pairing

Column generation

Master problem

Solve LP on $\mathcal{P}' \subseteq \mathcal{P}$

cost ↓ ↑ rot.

Pricing subproblem

Update $\mathcal{P}'$ (path pb)

- Feasible solutions of the Crew Pairing depend on the solutions of the Aircraft Routing: sequential resolution is not optimal

Part content

1. Column generation for rotation problems

2. An algebraic path problem framework

2.1 Computing bounds

3. Stochastic paths problems and delay in rotation problems

Usual shortest path problem

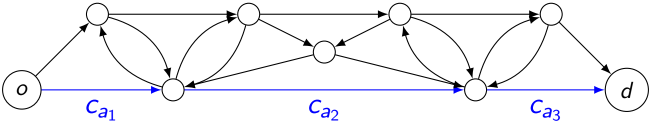
Input:

- Digraph $D = (V, A)$
- Origin o
- Destination d
- Cost $c_a \in \mathbb{R}$ for all $a \in A$

Output :

- An o-d path P minimizing

$$c_P = \sum_{a \in P} c_a$$



Ford-Bellman

Polynomial ($c_{\text{cyc.}} \geq 0$)
Dyn. Programming

Dijkstra

Polynomial
$c_a \geq 0$

A*

Non polynomial
Branch & Bound

Generalize the shortest path problem

A framework that enables to model

- many constraints,

$$\sum_{a \in P} q_a^i \leq Q^i, \text{ for } i \in [n],$$

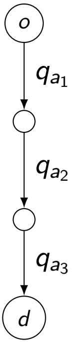
- non linear cost / constraints,

$$q_P \neq \sum_{a \in P} q_a \quad \left| \begin{array}{l} \text{Cost } c(q_P), \\ \text{Constraint } \rho(q_P) = 0, \end{array} \right.$$

- stochastic cost / constraints,

$$\min \mathbb{E}\left(c(\xi_P)\right) \quad / \quad \mathbb{P}(\xi_P \leq M) \leq \varepsilon.$$

q : resource



Constrained Shortest Path Problem : $\mathcal{NP}$-complete.

Ford-Bellman	Dijkstra	A*	Label cor.
Polynomial $\Rightarrow$ bounds	Polynomial $\Rightarrow$ bounds	Non polynomial $\Rightarrow$ solve	Non polynomial $\Rightarrow$ solve

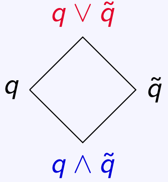
Ordered sets and lattices

Definition: *lattice*

A partially ordered set $(\mathcal{R}, \preceq)$ is a lattice if any pair $(q, \tilde{q})$ admits:

A greatest lower bound or **meet** denoted $q \wedge \tilde{q}$

$$\left.\begin{array}{l} b \preceq q \\ b \preceq \tilde{q} \end{array}\right\} \Leftrightarrow b \preceq q \wedge \tilde{q}$$



A least upper bound or **join** denoted $q \vee \tilde{q}$

$$\left.\begin{array}{l} b \succeq q \\ b \succeq \tilde{q} \end{array}\right\} \Leftrightarrow b \succeq q \wedge \tilde{q}$$

Ex: Natural numbers

$(\mathbb{N}, |)$

- $q \wedge \tilde{q} = \mathrm{GCD}(q, \tilde{q})$
- $q \vee \tilde{q} = \mathrm{LCM}(q, \tilde{q})$

Ex: Paris – Toulouse by car

$(\mathbb{R}^2, \leq)$ with the product order

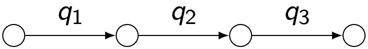
$$q = (d, t) = (\mathrm{distance}, \mathrm{time})$$

- $q \wedge \tilde{q} = \Big(\min(d, \tilde{d}), \min(t, \tilde{t})\Big)$
- $q \vee \tilde{q} = \Big(\max(d, \tilde{d}), \max(t, \tilde{t})\Big)$

Shortest Path in an ordered monoid

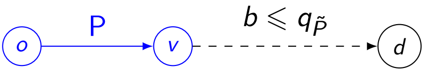
Arc resources q_a in a lattice ordered monoid $(\mathcal{R}, \oplus, \leqslant)$

- Associative $\oplus$: path resources



$$q_P = q_1 \oplus q_2 \oplus q_3$$

- Neutral element 0: empty path
- An order $\leqslant$ compatible with $\oplus$:



$$b \leqslant q_{\tilde{P}} \Rightarrow q_P \oplus b \leqslant q_P \oplus q_{\tilde{P}} = q_{P+\tilde{P}}$$

$(\mathcal{R}, \oplus, \leqslant)$ is a lattice ordered monoid if

$(\mathcal{R}, \oplus)$ is a *monoid*:

- $\oplus$ is associative,
- $\oplus$ has a neutral element 0

$\leqslant$ is *compatible* with $\oplus$:

$$q \leqslant \tilde{q} \Rightarrow \begin{cases} r \oplus q \leqslant r \oplus \tilde{q} \\ q \oplus r \leqslant \tilde{q} \oplus r \end{cases}$$

$(\mathcal{R}, \leqslant)$ is a *lattice*

Ex: Paris - Toulouse by car

$$q \oplus \tilde{q} = (d, t) \oplus (\tilde{d}, \tilde{t}) = (d + \tilde{d}, t + \tilde{t})$$

Shortest path with resources in an ordered monoid

Given a lattice ordered monoid $(\mathcal{R}, \oplus, \leqslant)$

Input:

- Digraph $D = (V, A)$
- Two vertices $o, d \in V$
- Resources $q_a \in \mathcal{R}$
- Two non-decreasing oracles
  $c : \mathcal{R} \to \mathbb{R}$
  $\rho : \mathcal{R} \to \{0, 1\}$

Output:

- An o-d path P such that

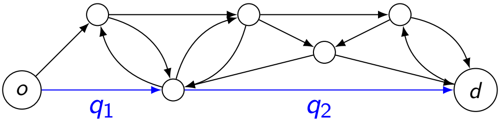
$$\rho\left(\bigoplus_{a \in P} q_a\right) = 0$$

which minimizes

$$c\left(\bigoplus_{a \in P} q_a\right)$$

Ex: Paris-Toulouse by car

- $q = (d, t)$
- Cost: $c(q) = \lambda_1 d + \lambda_2 t$
- On time arrival:
  $\rho(q) = \mathbb{1}_{(\tau, +\infty)}(t)$

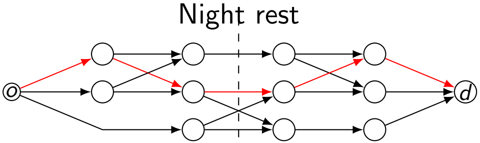


$q_P = q_1 \oplus q_2$ $\quad$ $\rho(q_P) = 0$

Example: Crew Pairing Pricing Subproblem

Find a pairing p of minimum reduced cost.

- p is an o-d path in the connection graph



- max. 4 legs per day.
- max. 3 legs if previous rest is reduced.

$$(\ell^t, \ell^d, \text{“p”}) \oplus (\tilde{\ell}^t, \tilde{\ell}^d, \tilde{\nu}) = \begin{cases} \infty & \text{if } \ell^d > 4 \\ (\ell^t + \tilde{\ell}^t, \tilde{\ell}^d, \text{“p”}) & \text{if } \nu = \text{“n”} \\ (\ell^t + \tilde{\ell}^t, \ell^d + \tilde{\ell}^d, \text{“p”}) & \text{otherwise} \end{cases}$$

$$q = (\ell^t, \ell^d, \nu)$$

ℓ^t: total nb legs

ℓ^d: daily nb legs

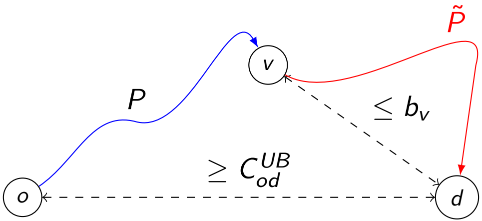
$$\nu = \begin{cases} \text{“n”} & \text{if rest} \\ \text{“p”} & \text{otherwise} \end{cases}$$

$$\text{Arc } \ell^d_a = \begin{cases} 2 & \text{if reduced rest} \\ 1 & \text{otherwise} \end{cases}$$

$$c(q) = \max(c_0, \lambda \ell^t)$$
$$\rho(q) = \mathbb{1}_{(4,+\infty)}(\ell^d)$$

$(\ell^t, \ell^d, \text{“p”}) \leqslant (\tilde{\ell}^t, \tilde{\ell}^d, \text{“p”})$ if

$$\ell^t \leq \tilde{\ell}^t \quad \text{and} \quad \ell^d \leq \tilde{\ell}^d$$



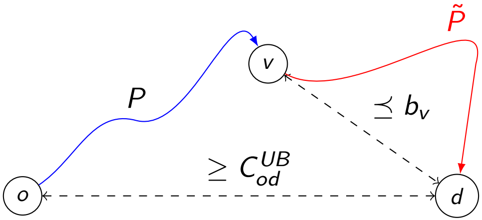
- $q_P \in \mathbb{R}$
- $C_{od}^{UB} \geq \min_{P \in \mathcal{P}_{o,d}} q_P$
- $b_v \leq q_P, \forall P \in \mathcal{P}_{vd}$

A path $P \in \mathcal{P}_{ov}$ satisfying $q_P + b_v > C_{od}^{UB}$ is not the subpath of an optimal path.

- Generate all the paths satisfying

$$q_P + b_v \leq C_{od}^{UB}$$

- Update C_{od}^{UB}



- $q_P \in \mathcal{R}$
- $C_{od}^{UB} \geq \min_{P \mid \rho(P)=0} c(q_P)$
- $b_v \preceq q_{\tilde{P}}, \forall \tilde{P} \in \mathcal{P}_{vd}$

A path $P \in \mathcal{P}_{ov}$ satisfying $c(q_P \oplus b_v) > C_{od}^{UB}$ or $\rho(q_P \oplus b_v) = 1$ is not the subpath of an optimal path.

Generalized A* Algorithm: a Branch & Bound

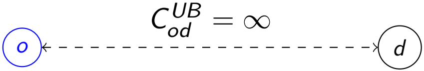
- Generate all the paths satisfying

$$c(q_P \oplus b_v) \leq C_{od}^{UB} \quad \text{and} \quad \rho(q_P \oplus b_v) = 0 \qquad \text{(Low)}$$

- Update C_{od}^{UB}

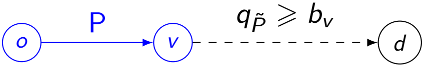
Generalized A* algorithm (2/2)

Initially: $L \leftarrow$ empty path in o

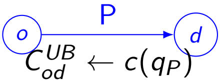


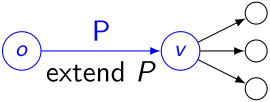
While L is not empty:

- extract $\min_{P \in L} c\left(q_P \oplus b_v\right)$



- If (Low) is satisfied, $\begin{cases} \rho(q_P \oplus b_v) = 0 \\ c(q_P \oplus b_v) < C_{od}^{UB} \end{cases}$





L: list of paths to be considered
C_{od}^{UB}: upper bound on optimal solution cost

Preprocessing: b_v lower bound on v-d paths resources

Key: $c(q_P \oplus b_v)$
Test: (Low)

Theorem

Under general assumptions (corresponding to the absence of negative cycles), A* *converges* after a finite number of iterations and

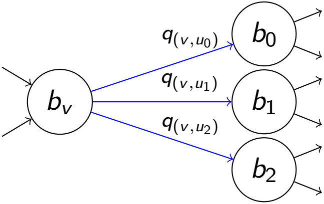
- if $C_{od}^{UB} = \infty$, then there is no feasible *o*-*d* paths,
- otherwise, C_{od}^{UB} is the cost of an optimal solution.

Instance	$\|V\|$	Alg	RCSP iter av. nb.	Cut Dom.	RCSP time av (mm:ss)
CP50	290	LS	1.020e+04	–	00:00.560
		LC	1.308e+04	–	00:01.275
		Our A*	4.914e+02	4.01%	00:00.016
CP70	408	LS	5.644e+04	–	00:11.489
		LC	7.730e+04	–	00:17.157
		Our A*	1.994e+03	4.28%	00:00.039
CP90	516	LS	9.779e+04	–	00:40.707
		LC	2.007e+05	–	01:42.864
		Our A*	9.966e+03	5.88%	00:00.340
A318	669	LS	1.319e+05	–	00:53.009
		LC	3.802e+05	–	01:36.035
		Our A*	2.549e+04	3.72%	00:01.651

Ford-Bellman algorithm for usual shortest path problem

Minimum costs b_v of v-d paths satisfy the dynamic programming equation:

$$\begin{cases} b_d = 0, \\ b_{v \neq d} = \min\left(b_v, \min_{u \in N^+(v)} \left(q_{(v,u)} + b_u\right)\right) \end{cases}$$



(b_v) is a fixed point of:

$$F : (b_v)_v \mapsto (b'_v)_v \text{ s.t.: } \begin{cases} b'_d = 0 \\ b'_{v \neq d} = \min\left(b_v, \min_{u \in N^+(v)} \left(q_{(v,u)} + b_u\right)\right) \end{cases}$$

Usual Ford-Bellman algorithm

$(b_v^k) = F^k(\infty)$ is the cost of a shortest v-d path with at most k arcs.

If there is no cycles of negative costs, $(b_v) = F^n(\infty)$ satisfies the dynamic programming equation. $n = |V|$.

Generalized dynamic programming (1/2)

Generalized dynamic programming equation

$$\begin{cases} b_d = 0, \\ b_{v \neq d} = \bigwedge \left(q_v, \bigwedge_{u \in N^+(v)} \left(q_{(v,u)} \oplus b_u \right) \right) \end{cases}$$

Admits a greatest solution $b_v^\dagger$ (Knaster-Tarski fixed-point theorem)

$$F : (b_v)_v \mapsto (b'_v)_v \text{ st: } \begin{cases} b'_d = 0 \\ b'_{v \neq o} = \bigwedge \left(b_v, \bigwedge_{u \in N^+(v)} \left(q_{(v,u)} \oplus b_u \right) \right) \end{cases}$$

Generalized Ford-Bellman algorithm

$(b_v^k) = F^k(\infty) \leqslant q_P$ for of any v-d path P with at most k arcs.

Generalized dynamic programming (2/2)

$$F : (b_v)_v \mapsto (b'_v)_v \text{ st: } \begin{cases} b'_d = 0 \\ b'_{v \neq o} = \bigwedge \left(b_v, \bigwedge_{u \in N^+(v)} \left(q_{(v,u)} \oplus b_u \right) \right) \end{cases}$$

- $b_v^k = F^k(b_v)$
- $b_v^\infty = \bigwedge_{k \in \mathbb{Z}_+} b_v^k$
- $b_v^\dagger = F(b_v^\dagger)$
- $b_v^{\text{opt}} = \bigwedge_{p \in \mathcal{P}_{vd}} q_P$
- ℓ^*: nb arcs in longest elem. path

Theorem

$$b_v^\dagger \leqslant b_v^\infty \leqslant b_v^{\ell^*} \leqslant b_v^{\text{opt}} \leqslant q_P \quad \text{for all } P \text{ in } \mathcal{P}_{vd}.$$

Part content

1. Column generation for rotation problems
2. An algebraic path problem framework
3. Stochastic paths problems and delay in rotation problems

Stochastic path problems

A slightly simpler problem

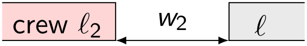
$$\min_{P} \quad c(q_P)$$
$$\text{s.t.} \quad \mathbb{P}\left(\sum_{a \in P} \xi_a > \tau\right) \leq 5\%$$

- lattice ordered monoid ?

Slacks makes things more complicated

$$\xi_\ell = \max(\xi_{\ell_2} - w_2, 0) + \xi_\ell^{\text{int}}$$

but the same ideas can be used



Monotonicity of $\cdot \mapsto \max(\cdot, 0)$ enables to use our framework

General case: almost sure order and addition

Almost sure order on random variables on $(\Omega, \mathcal{A}, \mathbb{P})$:

$$\xi \leq \tilde{\xi} \quad \text{if} \quad \xi(\omega) \leq \tilde{\xi}(\omega) \quad \text{a.s.}$$

Meet of two random variables:

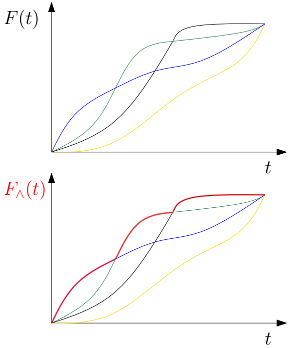
$$(\xi \wedge \tilde{\xi})(\omega) = \min(\xi(\omega), \tilde{\xi}(\omega)).$$

Compatible with *addition* $\xi + \tilde{\xi}$.

Sampling

- Any random variable can be approximated by a random variable on a finite probability space $\Omega = \{\omega_1, \ldots, \omega_n\}$
- Bounds on the error

Problem with n scenarios can be computationally difficult



Usual stochastic order $\leq_{\mathrm{st}}$

$$\xi \leq_{\mathrm{st}} \tilde{\xi} \quad \text{if} \quad \mathbb{P}(\xi \leq t) \geq \mathbb{P}(\tilde{\xi} \leq t) \quad \text{for all } t$$

Meet of two random variables

$$F_{\xi \wedge \tilde{\xi}} = \max(F_{\xi}, F_{\tilde{\xi}})$$

compatible with *convolution product* $*$

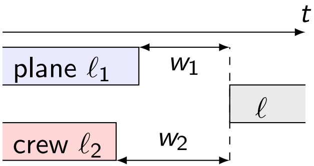
- $\leq_{\mathrm{st}}$ is *coarser* than almost sure order: $\xi \leq \tilde{\xi} \Rightarrow \xi \leq_{\mathrm{st}} \tilde{\xi}$.
- Better bounds: $b_v^{\mathrm{opt,as}} \leq_{\mathrm{st}} b_v^{\mathrm{opt,st}}$

Numerical results on Stochastic Crew Pairing

Instance	α (min)	Alg.	κ	CG iter.	Pricing time	Avg. paths	Cut Dom.	MIP time	Add. Cost	Total time (hh:mm:ss)
CP50	5%	A*	10	67	93.23%	1.730e+03	–	0.253%	138.01%	00:00:55.5
CP50	10%	A*	10	78	92.34%	1.741e+03	–	0.161%	72.65%	00:01:02.0
CP50	15%	A*	10	94	93.34%	3.029e+03	–	0.243%	0.00%	00:02:34.1
CP50	5%	cor.	10	54	94.53%	1.903e+03	0.33%	0.232%	138.01%	00:01:02.2
CP50	10%	cor.	10	62	94.75%	1.846e+03	0.30%	0.146%	72.65%	00:01:07.9
CP50	15%	cor.	10	97	95.53%	2.976e+03	0.27%	0.083%	0.00%	00:03:32.8
CP70	5%	A*	10	125	95.62%	1.172e+04	–	0.118%	57.64%	00:06:10.2
CP70	10%	A*	10	150	95.11%	1.059e+04	–	0.756%	53.04%	00:06:28.1
CP70	15%	A*	10	150	95.56%	1.822e+04	–	0.066%	0.00%	00:10:33.0
CP70	5%	cor.	10	121	97.20%	1.150e+04	0.49%	0.069%	57.64%	00:09:37.6
CP70	10%	cor.	10	140	97.46%	1.123e+04	0.45%	0.686%	53.07%	00:12:12.3
CP70	15%	cor.	10	145	98.30%	1.562e+04	0.31%	0.026%	0.00%	00:23:01.4
CP90	5%	A*	30	218	98.66%	7.928e+04	–	0.016%	45.57%	01:53:20.0
CP90	10%	A*	30	236	98.93%	8.701e+04	–	0.053%	41.22%	02:23:22.6
CP90	15%	A*	30	295	98.95%	1.324e+05	–	0.009%	0.00%	04:16:44.0
A318	5%	A*	150	341	99.76%	3.888e+05	–	0.002%	8.30%	57:08:59.0
A318	10%	A*	150	381	99.74%	4.342e+05	–	0.001%	7.32%	70:00:06.8
A318	15%	A*	150	395	99.77%	4.783e+05	–	0.001%	0.00%	94:32:26.5

- $\mathbb{P}(\xi_\ell > \tau) \leq \alpha$ for all ℓ with $\tau = 30$ minutes

Even if routing and pairings solution x_r and y_p are known, computing the distribution of ξ_ℓ is difficult (inference problem in a probabilistic graphical model with large treewidth)



Use a scenario approach.

- Delay cannot be dealt with in the subproblem
- Poorer relaxation