Modelling and Solving the Multi-Skill Project Scheduling Problem

Kenneth Young

23 February 2017

www.data61.csiro.au

Intro: The Problem

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers
- Skills

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers
- Skills

Aim: Find the fastest way to complete all the activities

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers
- Skills

Aim: Find the fastest way to complete all the activities

Constraints

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers
- Skills

Aim: Find the fastest way to complete all the activities

Constraints

- Activity constraint: Precedence relations between activities

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers
- Skills

Aim: Find the fastest way to complete all the activities

Constraints

- Activity constraint: Precedence relations between activities
- Skill constraint: Activities require skills

Intro: The Problem

What is the Multi-Skill Project Scheduling Problem (MSPSP)?

- Activites
- Workers
- Skills

Aim: Find the fastest way to complete all the activities

Constraints

- Activity constraint: Precedence relations between activities
- Skill constraint: Activities require skills
- Worker constraint: Workers each have a variety of skills

Intro: Example

Table : Workers' Skills

	Alice	Bob	Carl	Dora
Programmer	-	✓	✓	✓
DB Designer	✓	-	-	-
Webmaster	✓	✓	-	✓

Intro: Example

Table : Workers' Skills

	Alice	Bob	Carl	Dora
Programmer	-	✓	✓	✓
DB Designer	✓	-	-	-
Webmaster	✓	✓	-	✓

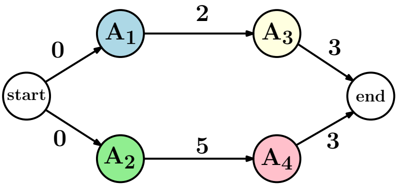


Figure : Precedence Graph

Intro: Example

Table : Workers' Skills

	Alice	Bob	Carl	Dora
Programmer	-	✓	✓	✓
DB Designer	✓	-	-	-
Webmaster	✓	✓	-	✓

Table : Skill Requirement

	A_1	A_2	A_3	A_4
Programmer	-	1	2	1
DB Designer	1	-	-	1
Webmaster	1	1	-	-

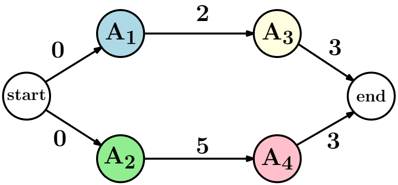


Figure : Precedence Graph

Intro: Example

Table : Workers' Skills

	Alice	Bob	Carl	Dora
Programmer	-	✓	✓	✓
DB Designer	✓	-	-	-
Webmaster	✓	✓	-	✓

Table : Skill Requirement

	A_1	A_2	A_3	A_4
Programmer	-	1	2	1
DB Designer	1	-	-	1
Webmaster	1	1	-	-

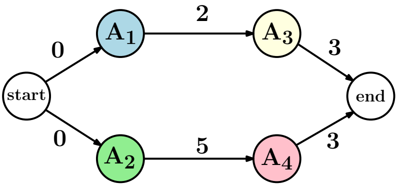


Figure : Precedence Graph

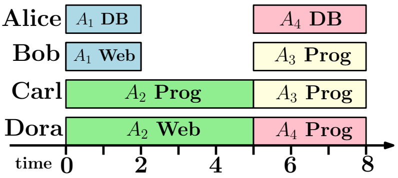


Figure : Schedule

Intro: Constraint Programming

Domain propagation

- Variables have domains of possible values
- Constraints reduce the size of these domains

Intro: Constraint Programming

Domain propagation

- Variables have domains of possible values
- Constraints reduce the size of these domains

Nogood learning

- Learn from failures
- Record these failures as constraints
- Use these constraints to make inferences

Intro: The Literature

- Portuguese research group
  - Principal researchers: Almeida, Saldanha-da-Gama, Correia
  - Constructive heuristics
  - Randomised search heuristics

Intro: The Literature

- Portuguese research group
  - Principal researchers: Almeida, Saldanha-da-Gama, Correia
  - Constructive heuristics
  - Randomised search heuristics
- French research group
  - Principal researchers: Belleguez-Morineau, Néron, Montoya
  - Exact branch and bound methods
  - Lower bounds

Intro: The Literature

- Portuguese research group
  - Principal researchers: Almeida, Saldanha-da-Gama, Correia
  - Constructive heuristics
  - Randomised search heuristics
- French research group
  - Principal researchers: Belleguez-Morineau, Néron, Montoya
  - Exact branch and bound methods
  - Lower bounds
- Polish research group
  - Principal researchers: Myzskowski, Skowronski
  - Randomised search heuristics

Model: Overview

Model: Overview

- Objective

Model: Overview

- Objective
  - Minimise the total project duration

Model: Overview

- Objective
  - Minimise the total project duration

- Two main decisions

Model: Overview

- Objective
  - Minimise the total project duration

- Two main decisions
  1. Scheduling decisions
     - Activity start times

Model: Overview

- Objective
  - Minimise the total project duration

- Two main decisions
  1. Scheduling decisions
     - Activity start times
  2. Assignment decisions
     - Workers to activities
     - Skill contribution of workers

Model: Constraints

Model: Constraints

- Precedence relations are respected

Model: Constraints

- Precedence relations are respected
- Workers perform only one activity at a time

Model: Constraints

- Precedence relations are respected
- Workers perform only one activity at a time
- Workers cannot multi-task

Model: Constraints

- Precedence relations are respected
- Workers perform only one activity at a time
- Workers cannot multi-task
- Skill requirement is satisfied
  - A worker for each skill must be present to perform the activity

Model: Constraints

- Precedence relations are respected
- Workers perform only one activity at a time
- Workers cannot multi-task
- Skill requirement is satisfied
    - A worker for each skill must be present to perform the activity
- Redundant constraints

Model: Choice of Constraints

Unary Resource Constraint

- Each worker only performs one activity at a time

Model: Choice of Constraints

Unary Resource Constraint

- Each worker only performs one activity at a time

Three equivalent ways of modelling

Model: Choice of Constraints

Unary Resource Constraint

- Each worker only performs one activity at a time

Three equivalent ways of modelling

1. Boolean satisfiability constraint using extra decision variable

Model: Choice of Constraints

Unary Resource Constraint

- Each worker only performs one activity at a time

Three equivalent ways of modelling

1. Boolean satisfiability constraint using extra decision variable
2. Disjunctive global constraint

Model: Choice of Constraints

Unary Resource Constraint

- Each worker only performs one activity at a time

Three equivalent ways of modelling

1. Boolean satisfiability constraint using extra decision variable
2. Disjunctive global constraint
3. Cumulative global constraint

Data: Overview

Data: Overview

- Generated our own data
  - equivalent to the Portuguese group's data

Data: Overview

- Generated our own data
  - equivalent to the Portuguese group's data
- Small dataset: 216 unique instances
  - 20 activities
  - 10-30 workers

Data: Overview

- Generated our own data
  - equivalent to the Portuguese group's data
- Small dataset: 216 unique instances
  - 20 activities
  - 10-30 workers
  - 13 unsolved

Data: Overview

- Generated our own data
  - equivalent to the Portuguese group's data
- Small dataset: 216 unique instances
  - 20 activities
  - 10-30 workers
  - 13 unsolved
- Large dataset: 216 unique instances
  - 40 activities
  - 20-60 workers

Data: Overview

- Generated our own data
  - equivalent to the Portuguese group's data
- Small dataset: 216 unique instances
  - 20 activities
  - 10-30 workers
  - 13 unsolved
- Large dataset: 216 unique instances
  - 40 activities
  - 20-60 workers
  - 211 unsolved

Data: Complexity Measures

1. Skill Factor
   - varied over 4 values

Data: Complexity Measures

1. Skill Factor
   - varied over 4 values
2. Network Complexity
   - varied over 3 values

Data: Complexity Measures

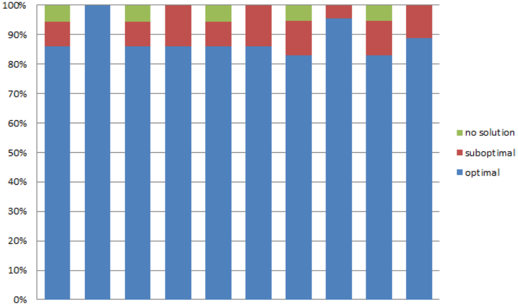
1. Skill Factor
   - varied over 4 values
2. Network Complexity
   - varied over 3 values
3. Modified Resource Strength
   - varied over 3 values

Experiments: Constraint Choice

Sample of 72 small instances

Experiments: Constraint Choice

Sample of 72 small instances



free search	0	1	0	1	0	1	0	1	0	1
edge finding	-	-	0	0	1	1	0	0	1	1
	bool		disjunctive				cumulative			

Experiments: Search Strategies

Experiments: Search Strategies

- Start time variables

Experiments: Search Strategies

- Start time variables
- Start time variables, then contrbution of each worker

Experiments: Search Strategies

- Start time variables
- Start time variables, then contrbution of each worker
- Activity-based

Experiments: Results

- Tested on all 216 small instances
- Time limit of 5 minutes

Experiments: Results

- Tested on all 216 small instances
- Time limit of 5 minutes

search strategy	#no soln	#sub-opt	%gap	#optimal	%optimal	avg. runtime
default	0	0	0.00	216	100.00	3.25s
start	0	1	2.50	215	99.54	1.26s
start then worker	0	0	0.00	216	100.00	2.89s
start then skill	**0**	**0**	**0.00**	**216**	**100.00**	**1.63s**
activity-based	0	1	2.50	215	99.54	0.82s

Summary

- Applied the constraint programming solver chuffed to the MSPSP

Summary

- Applied the constraint programming solver chuffed to the MSPSP
- Generated a set of benchmark instances

Summary

- Applied the constraint programming solver chuffed to the MSPSP
- Generated a set of benchmark instances
- Found an effective model formulation

Summary

- Applied the constraint programming solver chuffed to the MSPSP
- Generated a set of benchmark instances
- Found an effective model formulation
- Solved all small instances

Future Work

- Apply activity-based search to the large dataset

Future Work

- Apply activity-based search to the large dataset
- Create a more structured search procedure in the chuffed

Acknowledgements

- Dr. Andreas Schutt
- Dr. Thibaut Feydy
- Adrian Goldwaser

Thanks for listening!

Questions?