Courses on Combinatorial Optimisation

Pierre Flener
http://user.it.uu.se/∼pierref
Pierre.Flener@it.uu.se

Computing Science Division
Department of Information Technology
Uppsala University
Sweden

Autumn semester

Example (Doctor Rostering)

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Doctor A							
Doctor B							
Doctor C							
Doctor D							
Doctor E							

Constraints to be **satisfied**:

1. #on-call doctors / day $= 1$
2. #operating drs / weekday ≤ 2
3. #operating drs / week ≥ 7
4. #appointed drs / week ≥ 4
5. day off after operating day
6. ...

Objective function to be **minimised**:

- Cost: ...

Example (Doctor Rostering)

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Doctor A	call	none	oper	none	oper	none	none
Doctor B	appt	call	none	oper	none	none	call
Doctor C	oper	none	call	appt	appt	call	none
Doctor D	appt	oper	none	call	oper	none	none
Doctor E	oper	none	oper	none	call	none	none

Constraints to be **satisfied**:

1. #on-call doctors / day $= 1$
2. #operating drs / weekday ≤ 2
3. #operating drs / week ≥ 7
4. #appointed drs / week ≥ 4
5. day off after operating day
6. ...

Objective function to be **minimised**:

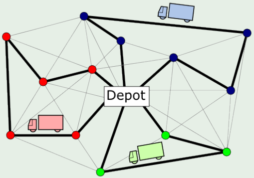
- Cost: ...

Example (Vehicle Routing: Parcel Delivery)

Given a depot with parcels for clients and a vehicle fleet, **find** which vehicle visits which client when.

Constraints to be **satisfied**:

1. All parcels are delivered on time.
2. No vehicle is overloaded.
3. Driver regulations are respected.
4. ...



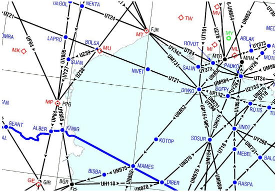
Objective function to be **minimised**:

- Cost: the total fuel consumption and driver salary.

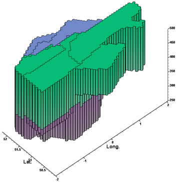
UPPSALA UNIVERSITET

Applications in Air Traffic Management

Demand vs capacity



Airspace sectorisation



Contingency planning

Flow	Time Span	Hourly Rate
From: Arlanda To: west, south	00:00 – 09:00 09:00 – 18:00 18:00 – 24:00	3 5 2
From: Arlanda To: east, north	00:00 – 12:00 12:00 – 24:00	4 3
...	...	...

Workload balancing

UPPSALA UNIVERSITET

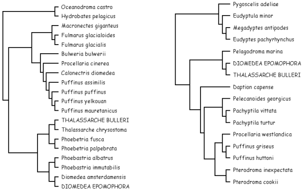
Combinatorial Optimisation

Modelling (course 1DL451)

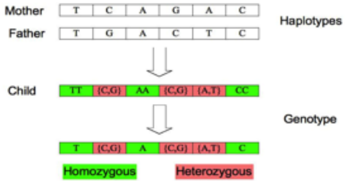
Constraint Programming (course 1DL442)

Applications in Biology and Medicine

Phylogenetic supertree



Haplotype inference



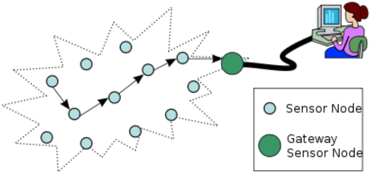
Medical image analysis

Doctor rostering

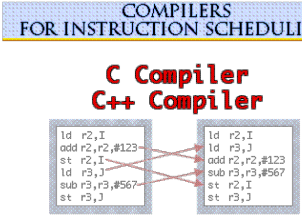
Applications in Programming and Testing

Robot programming

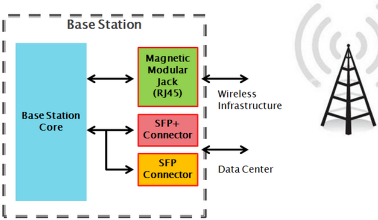
Sensor-net configuration



Compiler design



Base station testing



UPPSALA UNIVERSITET

Combinatorial Optimisation

Modelling (course 1DL451)

Constraint Programming (course 1DL442)

Other Application Areas

School timetabling

	Monday	Tuesday	Wednesday	Thursday	Friday
9.00	MT2202 *Ordinary Differential Equations* FTh1		LABCS2072 *Computer Graphics (G)* Dual	MT2282 *Numerical Analysis II* Williamson, G03	
10.00	XMT2202 *Ordinary Differential Equations* NG15 / Roscoe, 2.3		LABCS2072 *Computer Graphics (G)* Dual	XMT2202 *Ordinary Differential Equations* Simon Engineering, Basement Theatre 3A XMT2282 *Numerical Analysis II* LG20	XMT2202 *Ordinary Differential Equations* NG15
11.00	CS2012 *Algorithms and Data Structures* 1.1		XMT2212 *Further Linear Algebra* 1.5		MT2202 *Ordinary Differential Equations* Stopford, Theatre 1
12.00	MT2212 *Further Linear Algebra* Roscoe, Theatre A	MT2282 *Numerical Analysis II* Williamson, G03	CS2072 *Computer Graphics* 1.1		MT2212 *Further Linear Algebra* Stopford, Theatre 1
1.00			PASS *Peer Assisted Study* I407 / LF15 / LF17 / I406		XMT2212 *Further Linear Algebra* Simon Engineering, Basement Theatre 4A
2.00	CS2072 *Computer Graphics* 1.1			XMT2212 *Further Linear Algebra* NG17	
3.00		CSTUT *Tutorial*			
4.00		CS2012 *Algorithms and Data Structures* 1.1			

Sports tournament design

svensk handboll

Security: SQL injection?

Container packing

Combinatorial Optimisation

Many important real-life problems are NP-hard or worse: their real-life instances can only be solved optimally and fast enough by intelligent search, unless P $=$ NP.

Note that our small instance for Doctor Rostering already has $4^{5 \cdot 7} \approx 1.2 \cdot 10^{21}$ candidate solutions, but real-life instances have more than 4 activities and 5 doctors, and assign hourly instead of daily, over more than 7 days.

Combinatorial optimisation covers satisfaction problems *and* optimisation problems, for variables over discrete sets.

A solving technology offers methods and tools for:

what: **Modelling** constraint problems in declarative language.

how: **Solving** constraint problems intelligently.

A solver is an off-the-shelf problem-independent program that takes a model & data as input and tries to find optimal solutions to that problem instance as fast as possible.

Examples: CP, LS, MIP, SAT, and SMT solvers (see below).

UPPSALA UNIVERSITET

Combinatorial Optimisation

Modelling (course 1DL451)

Constraint Programming (course 1DL442)

Example (Doctor Rostering)

	Mon	Tue	Wed	Thu	Fri	Sat	Sun
Doctor A	call	none	oper	none	oper	none	none
Doctor B	appt	call	none	oper	none	none	call
Doctor C	oper	none	call	appt	appt	call	none
Doctor D	appt	oper	none	call	oper	none	none
Doctor E	oper	none	oper	none	call	none	none

Constraints to be **satisfied**:

1. #on-call doctors / day $= 1$
2. #operating drs / weekday ≤ 2
3. #operating drs / week ≥ 7
4. #appointed drs / week ≥ 4
5. day off after operating day
6. …

Objective function to be **minimised**:

- Cost: …

UPPSALA UNIVERSITET

Combinatorial Optimisation

Modelling (course 1DL451)

Constraint Programming (course 1DL442)

Example (MiniZinc Model for Doctor Rostering)

```
-4 set of int: Days;  % d mod 7 = 1 iff d is a Monday
-3 enum Doctors;
-2 enum ShiftTypes = {appt, call, oper, none};
-1 array[Doctors,Days] of var ShiftTypes: Roster;
 0 solve minimize ...;  % plug in an objective function
 1 constraint forall(d in Days)
     (count(Roster[..,d],call) = 1);
 2 constraint forall(d in Days where d mod 7 in 1..5)
     (count(Roster[..,d],oper) <= 2);
 3 constraint count(Roster,oper) >= 7;
 4 constraint count(Roster,appt) >= 4;
 5 constraint forall(d in Doctors)
     (regular(Roster[d,..], "((oper none)|appt|call|none)*"));
 6 ...  % other constraints
```

Example (Instance data for our small hospital unit)

```
-4 Days = 1..7;
-3 Doctors = {Dr_A, Dr_B, Dr_C, Dr_D, Dr_E};
```

Modelling for Combinatorial Optimisation (1DL451)

Race the *same* model under *several* solving technologies:

- **Constraint programming (CP):**
  any kinds of constraints on any kinds of variables
  ☞ full details in Part 2 of my course 1DL442 (below)
- **Mixed integer linear programming (MIP):**
  linear constraints & objective on int & float variables
- **Propositional satisfiability (SAT):**
  clausal constraints on Boolean variables
- **SAT modulo theories (SMT):**
  SAT + integer arithmetic, bit vectors, …
- **Local search (LS):**
  trade for speed all guarantees of provable optimality of solutions and provable unsatisfiability of problems

without knowing their languages and solving algorithms:

☞ Model once, solve everywhere!

Modelling for Combinatorial Optimisation (1DL451)

- Period 1: late August to late October
- 12 **lectures**, in English
- No textbook required: slides, documentation, Coursera
- **Modelling** problems using the MiniZinc.org toolchain:
  - 3 **assignments**, to be done in pairs (3 credits)
  - 1 **project**, to be done in pairs (2 credits)

  3 **help sessions** + 1 **solution session** per deliverable
- No exam
- **Prerequisites:** define or learn basic concepts in algebra, combinatorics, logic, graph theory, set theory
- http://user.it.uu.se/∼pierref/courses/M4CO/course.html

Combinatorial Optimisation and Constraint Programming (CP) (1DL442)

- Periods 1 & 2: late August to mid January(!)
- 24 **lectures**, in English
- No textbook required: slides and documentation
- Part 1: **Modelling** using MiniZinc.org: course 1DL451
- Part 2: **Programming** using MiniCP.org (Java):
  - 3 **assignments** towards understanding & extending a solver of CP technology; to be done in pairs (5 credits)

  3 **help sessions** + 1 **solution session** per assignment
- No exam
- **Prerequisites:** define or learn basic concepts in algebra, combinatorics, logic, graph theory, set theory; implement basic search algorithms
- http://user.it.uu.se/∼pierref/courses/COCP/course.html

Success Stories: Users and Contributors

AIMMS · AMPL · BOUYGUES
cadence · CISCO · FICO
Google · IBM · intel
JEPPESEN A BOEING COMPANY · NASA · ORACLE
QUINTIQ · RedPrairie · RENAULT
SAP · SIEMENS · TACTON
THALES · XEROX · ...

Success stories: CP = **technology of choice** in scheduling, configuration, personnel rostering, timetabling, ...

UPPSALA UNIVERSITET

Combinatorial Optimisation

Modelling (course 1DL451)

Constraint Programming (course 1DL442)

See you on Monday 29 August!?