

### (High) School Timetabling

Daniel Gomes e Rui Andrade — { up201306839, ei12010 } @fe.up.pt Planning and Scheduling Methodologies — 2015/16





- 1. The (high) school timetabling problem
- 2. The XHSTT xml-standard
- 3. HSTTE an engine for XHSTT
- 4. Solving with genetic algorithms
- 5. Tests

### The (high) school timetabling problem



- The (high) school timetabling problem is a scheduling problem
- Consists of attributing a set of resources and a time slot to a set of events
- It may also include transforming an event into multiple of smaller durations

### Formalization – machine environment $(\alpha)$

- Machines are rooms and/or teachers
- Multiple machines in parallel
  - For a given lesson multiple rooms and teachers are available
  - For a given lesson both a room and a teacher may be scheduled

#### Formalization – job characteristics and scheduling constraints $(\beta)$

#### Jobs

- May have different durations
- Different jobs may run on different machines
- Preemptions may happen

#### Constraints:

- One machine must be allocated to one job at time
- Limit on the time between job group's lessons
- Machine or time preferences

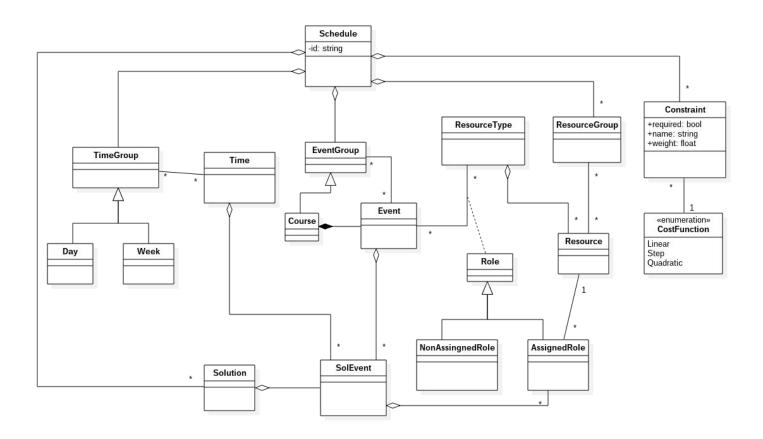
### Formalization – objective function $(\gamma)$

- Associated with the number of violated constraints.
  - ♦ A cost function that should be minimized

## The XHSTT xml-standard



- Developed to be a generic standard for the exchange of datasets
- Each file can store multiple specifications and multiple solutions for each
- Currently an archive of datasets XHSTT-2014 serves as benchmark for (high) school timetabling algorithms



XHSTT conceptual model UML representation





- XHSTT contemplates several constraints definitions:
  - ↓ E.g., AssignResource, AvoidClashes, AssignTime, PreferResources



- Cost function consists of a pair of two values
  - (Feasibility value, Objective value)
- Each constraint may be specified as required or not
  - If required, contributes to Feasibility value, otherwise to Objective value
- In one of three different modes:
  - Step, Linear or Quadratic

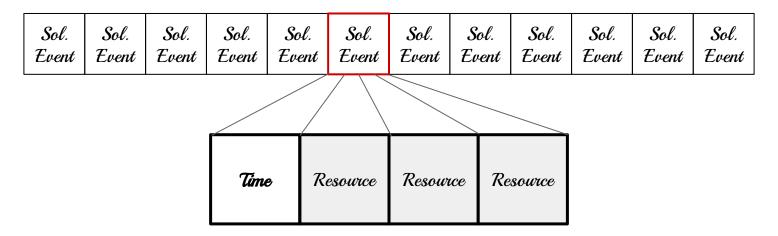
## $HSTTE-an\ engine\ for$ $\chi HSTT$

### An approach through genetic algorithms



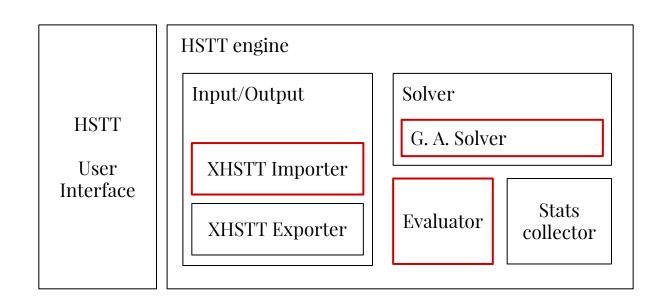
- A chromosome is scheduling solution
- A solution is a set Event-solution
- An Event-solution has: a time and the resources that are required to be assigned

#### Chromosome



Chromosome representation





Engine's modules





- For each event, a random Event-solution is generated.
  - Is assigned to a random Time
  - ♦ For each role required, a random Resource of that type is attributed
- N solutions are generated

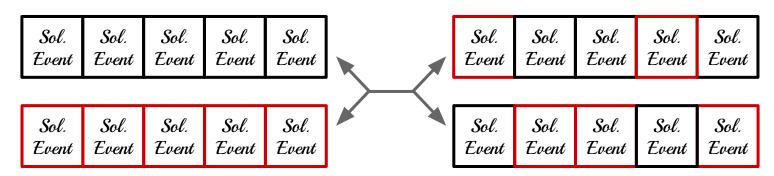


- Each solution is selected with a probability of
  - $\diamondsuit$  PS(Sol.) = Fitness(Sol.) /  $\Sigma$ (Fitness(Sol.<sub>i</sub>))
- XHSTT objective function is a cost. It must be transformed into a fitness function
  - ♦ Fitness(Sol.) = MaxCost(Spec.) Cost(Sol.)
- XHSTT cost-pair should be translated to a number:
  - $\Diamond$  C = Cost.Feasibility \*  $10^{\text{len(Max(Obj))}}$  + Cost.Obj

# Offspring – at the Solution-event level

- Crossover
  - Uniform-crossover is used
  - With a 0.5 probability each parent Sol-event is swapped
- Mutation
  - With a small probability, random Event-solutions are replaced by a random new one

#### Selection



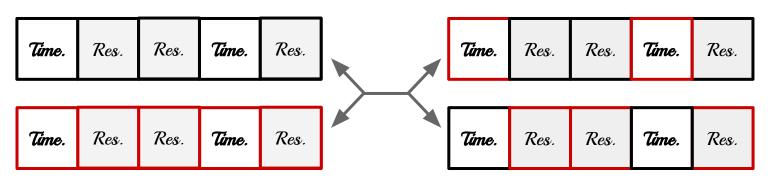
Offspring – at the Solution-event level representation



# Offspring – at the Time/Resource level

- Crossover
  - Uniform-crossover is used
  - With a 0.5 probability each parent Time/Resource is swapped
- Mutation
  - With a small probability, Times or Resources are replaced by a random new one

#### Selection



Offspring - at the Time/Resource level







- Run Genetic Solver with both offspring strategies
- Run both datasets:
  - ♦ Sudoku4x4
  - ♦ Abramson15



- 4 constraint types: AssignResource, AssignTime, PreferResources and AvoidClashes
- ♦ 4 Classes, 4 Teachers, 4 Times and 4 Rooms
- 16 Events



- 2 constraints types: AssignTime and AvoidClashes
- 15 Classes, 15 Teachers, 30 Times and 15 Rooms
- 450 Events



### Thank you.

Daniel Gomes e Rui Andrade { up201306839, ei12010 } @fe.up.pt