

GENERACIÓ D'HORARIS D'INSTITUT AMB OPERACIONS LÒGIQUES

Ismael El Habri
Tutors: Dr. Josep Suy i Dr. Jordi Coll

Universitat de Girona

10 de setembre del 2019

Table of Contents

- 1 Introducció
 - Marc de treball
 - Objectius
 - Metodologia
- 2 Implementació
 - Parser
 - Model
 - Restriccions
- 3 Conclusions i Resultats
 - Resultats

Introducció

Confecció d'horaris, problema recurrent amb el que es troben els instituts \Rightarrow High School TimeTabling problem (HSTT)

- Alta combinatòria i complexitat, és un problema NP-Complel.
- Repartir events i recursos de manera viable i tenint en compte preferències del professorat.
- Diferents països, diferents necessitats \Rightarrow més complexitat!

Marc de treball

- Grup de recerca de Lògica i Programació
- API SMT desenvolupada pel Dr. Jordi Coll. API per a la codificació de problemes SAT, SMT o MaxSAT, actuant com a interfície per a diferents solvers. En aquest treball s'utilitzarà el Yices 2. També té implementades les diferents implementacions de múltiples restriccions globals.
- Com a punt de partida s'ha utilitzat el treball realitzat el 2015 per en Cristòfor Nogueira. Mentre ell ha utilitzat BitVectors i MaxSAT, en aquest treball s'utilitzarà SMT. Així s'han aconseguit uns resultats superiors pel que fa el temps d'execució.

Objectius

- Aprofundir sobre el tema.
 - Problema de generació d'horaris d'institut.
 - Problemes de satisfacció de restriccions(CSP).
 - Tècniques per resoldre problemes CSP com ara SAT i extensions.
- Crear un generador.

Metodologia

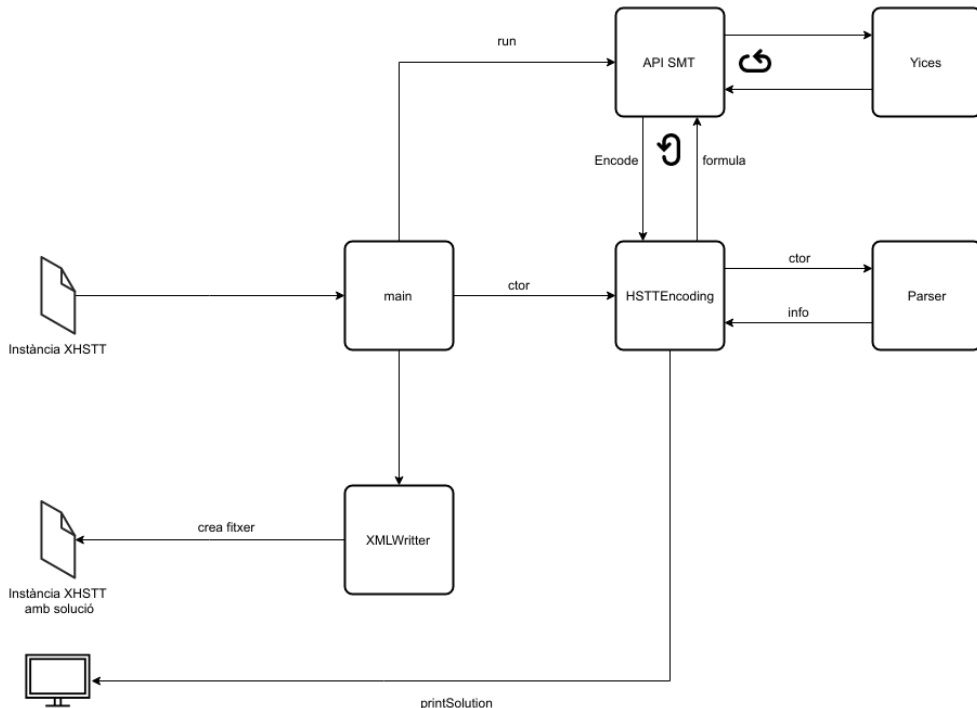
- Estudi del treball prèvi i estat de l'art
- Entregues periòdiques
- Prototipatge

Estudi de viabilitat

Table of Contents

- 1 Introducció
 - Marc de treball
 - Objectius
 - Metodologia
- 2 **Implementació**
 - Parser
 - Model
 - Restriccions
- 3 Conclusions i Resultats
 - Resultats

Implementació



Parser

Instància XHSTT \Rightarrow dades + restriccions

Model

- $X_{t_{0,0} \dots t_{|Events|-1, |Times|-1}}$
- $X_{s_{0,0} \dots s_{|Events|-1, |Times|-1}}$
- $X_{d_{0,1,0} \dots d_{|Events|-1, event.duration, |Times|-1}}$

Clàusules de Channeling

- Si un event comença a una hora determinada, llavors té una duració:

$$\forall e \in 0 \dots |Events| - 1 \quad \forall i \in 0 \dots |Times| - 1 \\ \text{exactly_one}(\neg X_{s_{e,i}} \vee \{\forall j \in 1 \dots e.duration \quad X_{d_{e,j,i}}\})$$

- Si un event té lloc a t però no a t-1, és que comença:

$$\forall e \in 0 \dots |Events| - 1 \quad \forall i \in 0 \dots |Times| - 1 \\ \neg X_{t_{e,i}} \vee X_{t_{i-1}} \vee X_{s_i}$$

- Si un event comença amb duració d, llavors té lloc en d hores consecutives:

$$\forall e \in 0 \dots |Events| - 1 \quad \forall d \in 1 \dots e.duration \quad \forall i \in 0 \dots |Times| - 1 \\ \forall j \in i \dots i + d - 1 \\ \neg X_{d_{e,d,i}} \vee X_{t_{e,j}}$$

Restriccions

- Assign Times Constraint

$$\forall e \in Events \text{ exactly}_k(\{X_{t_e,0} \dots X_{t_e,|Times|-1}\}, e.duration)$$

- Split Events Constraint

$$\forall e \in Events \text{ at_most_}k(\{X_{s_e,0} \dots X_{s_e,|Times|-1}\}, MaximumAmount)$$

$$\forall e \in Events \text{ at_least_}k(\{X_{s_e,0} \dots X_{s_e,|Times|-1}\}, MinimumAmount)$$

$$\forall e \in Events \forall d \notin MinimumDuration \dots MaximumDuration \wedge d \in 1 \dots e.duration$$

$$\forall t \in 0 \dots |Times| - 1 \quad \neg X_{d,e,d,t}$$

- Distribute Split Constraint

$$\forall e \in Events \text{ at_most_}k(\{X_{d_e,d,0} \dots X_{d_e,d,|Times|-1}\}, max) \quad \text{si } max < \frac{e.duration}{d}$$

$$\forall e \in Events \text{ at_most}_k(\{X_{d_e,d,0} \dots X_{d_e,d,|Times|-1}\}, min) \quad \text{si } min > 0$$

Restriccions

- Prefer Times Constraint

$$\forall e \in Events \forall t \in Times \wedge t \notin Ta \\ (\neg Xd_{e,d,t})$$

- Spread Events Constraint

$$\forall e \in Events \forall g \in Tg \\ at_most_k(\{Xs_{e,t} | t \leftarrow g\}, max) \\ \forall e \in Events \forall g \in Tg \\ at_least_k(\{Xs_{e,t} | t \leftarrow g\}, max)$$

- Avoid Clashes Constraint

$$\forall r \in Resources \forall t \in Times \\ at_most_one(\{Xt_{e,t} | e \leftarrow E_r\})$$

- Avoid Unavailable Times Constraint

$$\forall r \in Resources \forall t \in T \forall e \in E_r \quad (\neg Xt_{e,t})$$

Restriccions

- Limit Idle Times Constraint

Per a cada recurs r es fan les clàusules següents:

$$\forall g \in Tg \ \forall t \in g \ \forall e \in E_r \quad (\neg Idle_t \vee \neg Xt_{e,t}) \quad \text{si } (B_t \neq \emptyset)$$

$$\forall g \in Tg \ \forall t \in g \quad (\neg Idle_t \vee \{\forall b \in B_t \ \forall e \in E_r \ Xt_{e,b}\}) \quad \text{si } (B_t \neq \emptyset)$$

$$\forall g \in Tg \ \forall t \in g \quad (\neg Idle_t \vee \{\forall a \in A_t \ \forall e \in E_r \ Xt_{e,a}\}) \quad \text{si } (B_t \neq \emptyset)$$

$$\begin{aligned} &\forall g \in Tg \ \forall t \in g \ \forall b \in B_t \ \forall a \in A_t \\ &\quad \forall e_1 \in E_r \ \forall e_2 \in E_r \ \forall e_3 \in E_r \\ &\quad (Xt_{e_1,t} \vee \neg Xt_{e_2,b} \vee Xt_{e_3,a} \vee Idle_t) \quad \text{si } (B_t \neq \emptyset) \end{aligned}$$

Un cop definides i lligades les variables auxiliars l'únic que queda és, per cada recurs, imposar les restriccions de cardinalitat:

$$\forall r \in Resources$$

Restriccions

- Cluster Busy Times Constraint

Per a cada recurs r es fan les clàusules següents:

$$\begin{aligned} & \forall g \in Tg \\ & (\neg Busy_g \vee (\forall e \in E_r \forall t \in g \quad Xt_{e,t})) \\ \forall g \in Tg \forall t \in g \forall e \in E_r \\ & (\neg Xt_{e,t} \vee Busy_g) \end{aligned}$$

Un cop definides i lligades les variables auxiliars l'únic que queda és, per cada recurs, imposar les restriccions de cardinalitat:

$$\begin{aligned} & \forall r \in Resources \\ & at_most_k(\{Busy_g | g \leftarrow Tg\}, max) \\ & at_least_k(\{Busy_g | g \leftarrow Tg\}, min) \end{aligned}$$

Table of Contents

- 1 Introducció
 - Marc de treball
 - Objectius
 - Metodologia
- 2 Implementació
 - Parser
 - Model
 - Restriccions
- 3 Conclusions i Resultats
 - Resultats

