

**Generador d'Horaris d'Instituts**  
versió resumida

Cristòfor Nogueira Gascons

14 de juny de 2015

# Índex

1	Introducció . . . . .	2
2	Definició del problema . . . . .	2
3	Anàlisi de l'Estat de l'Art realitzat . . . . .	3
4	Cardinality Encodings . . . . .	4
5	Implementació . . . . .	5
6	Resultats . . . . .	5
7	Conclusions . . . . .	6

## 1 Introducció

Una de les majors complicacions que han de superar els instituts a l'hora d'organitzar el nou curs acadèmic és la confecció dels horaris. L'elevat nombre de combinacions entre assignatures dificulta moltíssim el procés de confeccionar un horari viable i equitatiu. Amb l'aparició dels computadors, molts instituts van relegar aquesta feina a eines informàtiques capaces d'explorar centenars de milers de combinacions per segon. Tot i així, aquelles primeres eines no semblaven tenir millor fortuna ja que el nombre de combinacions possibles rere un problema d'aquestes característiques és exponencial al nombre d'assignatures. Per tant, també es tenien moltíssimes dificultats per generar horaris viables. Aquests horaris generalment eren de baixa qualitat i requerien d'un refinament manual posterior.

Un altre aspecte que afegeix complexitat al problema és la gran varietat de requeriments dels horaris, que varien d'un país a un altre. En particular dificulta l'abstracció necessària a l'hora de modelar una eina capaç de generar horaris de manera satisfactòria per instituts d'arreu. Aquest fet fa difícil la creació de generadors capaços d'explotar les peculiaritats inherents a cada instància concreta.

Encara avui dia, no es coneix cap eina capaç de resoldre el problema de manera determinista i en un temps raonable. Però l'estat de l'art ha avançat moltíssim i ja són pocs els instituts que opten per atacar el problema de manera manual.

Els objectius d'aquest treball són per una banda, aprofundir sobre el problema de la confecció d'horaris, veure com de dur és el problema, què és el que el fa tan dur i aprofundir en les tècniques deterministes de més èxit a l'actualitat. I per altra banda, implementar un generador determinista capaç de generar horaris per a instàncies de formulació específica utilitzant dues de les tecnologies amb més èxit i futur en el sí de l'estat de l'art actual, com són tècniques basades en SAT (Satisfactibilitat proposicional booleana) i tècniques basades en SMT (Satisfiability Modulo Theories).

## 2 Definició del problema

Els instituts disposen d'una sèrie de recursos que al llarg del curs acadèmic hauran d'assistir a un conjunt d'assignatures. Exemples de recursos són els professors, les aules, els grups d'alumnes, etc. El problema consisteix a assignar un espai de temps a cada assignatura de manera que a tots els recursos els sigui possible assistir-hi. A més, com a mínim un dels recursos dels que considera el centre a l'hora de generar l'horari té disponibilitat limitada. Aquesta característica és una de les fonts principals de duresa del problema. És a dir, si tots els recursos tinguessin disponibilitat il·limitada el problema tindria solució trivial.

En aquest treball demostrarem que el problema és NP-Complet, i per tant, pertany a la classe de problemes NP-durs. Precisament la demostració es basa en la disponibilitat limitada dels recursos com a font independent d'NP-Completesa.

Per tal de considerar una formulació el més estàndard possible s'ha adoptat el format XHSTT, que és un format que pretén establir-se com a format estàndard per a problemes d'aquest tipus. A més, aquest format també és el format que s'ha utilitzat en les diverses edicions de la competició internacional de generadors d'horaris d'instituts. Per tant, a internet es disposa d'un petit arxiu amb nombroses instàncies disponibles al públic. L'arxiu disposa d'instàncies reals i d'artificials.

### 3 Anàlisi de l'Estat de l'Art realitzat

Aquest apartat és una de les seccions principals del treball ja que es correspon amb un dels objectius que s'havien marcat a l'inici. A més, considero que el fet d'efectuar aquest anàlisi també és molt beneficiós a l'hora de complir el segon objectiu del treball, que és la implementació del generador. En concret, en aquest anàlisi es realitza una revisió de diverses de les tècniques deterministes més efectives que hi ha en l'actualitat, fent especial èmfasi en SAT i SMT, que són les dues tècniques que s'han arribat a implementar de manera satisfactòria:

- Constraint Satisfaction Problems (CSP)

S'explica breument en què consisteix el paradigma així com el concepte d'arc-consistència, que és fonamental per entendre com funcionen els *solvers* de CSP.

- Mixed Integer Linear Programming (MILP)

Es presenta aquesta tècnica de resolució i optimització de problemes combinatoris. No obstant, aquesta tècnica només s'ha vist a nivell teòric i al llarg de la confecció del treball no s'ha posat en pràctica.

- Boolean Satisfiability Problem (SAT)

En aquest punt, s'ha destinat un gruix important de la dedicació del treball tal com es reflecteix a la memòria. Resultats empírics evidencien que els *SAT-solvers* moderns són capaços de resoldre instàncies de milers de variables i centenars de milers de clàusules en qüestió de segons. En un intent d'entendre el perquè són tan eficients aquest tipus de *solvers* es realitza un estudi de les tècniques que utilitzen. En concret, es revisen les tècniques i fonaments dels *solvers* de més èxit que són els que incorporen Conflict Driven Clause Learning (CDCL), que és una tècnica que apliquen els solvers basats en DPLL. S'ha realitzat un estudi sobre aquest tipus d'algorismes i tècniques. Tanmateix, s'ha intentat explicar de manera més distesa i a través d'exemples.

S'introdueixen temes com el de la *Propagació Unitària*, *Clause Learning*, *Unit Implication Point*, *Clause Cleaning*, *Restarts...*

- Satisfiability Modulo Theories (SMT)

S'explica la idea fonamental rere SMT, contestant a la pregunta “en què consisteix SMT?”. Aquest apartat s'ha decidit mantenir-lo breu perquè SMT es basa en hibridar SAT amb altres Teories i per tant, en lloc d'enumerar quines són i en què consisteixen aquestes teories ens hem centrat a explicar el concepte de *hibridatge*.

## 4 Cardinality Encodings

A banda d'estudiar els fonaments de les diverses tècniques de resolució, en el cas concret de SAT i SMT amb vectors de bits (BV) com a teoria de rerefons, també s'ha realitzat un estudi de la codificació d'un tipus de restriccions molt particular: les restriccions de cardinalitat.

En el cas de SMT amb BV es descriu la codificació de la restricció *at\_most\_k*, que imposa al vector d'entrada a tenir, com a molt,  $k$  bits activats.

En el cas de SAT es descriuen set codificacions per a la restricció *at\_most\_k*, que imposa que del conjunt de variables booleans d'entrada, com a molt conté  $k$  variables avaluades a *Cert*. Les codificacions que es revisen són:

- Direct Encoding
- Totalizer
- Simplified Totalizer
- Modulo Totalizer
- Sorting Networks
- K-Cardinality Networks
- Mixed Cardinality Network

També en el cas de SAT, es descriuen dos codificacions per al cas particular de la restricció *at\_most\_1* (AMO), que imposa que del conjunt de variables booleans d'entrada, com a molt una s'avalua a *Cert*.

- Ladder Encoding
- Commander Encoding

Per a cada codificació es dona un *upper-bound* del nombre de variables i clàusules que generen per donar una noció d'eficiència que ens permeti comparar codificacions.

## 5 Implementació

L'implementació del generador d'horaris consisteix en quatre blocs:

1. Parser de fitxers en format XHSTT
2. Bloc de codificació de la instància a SAT (format Dimacs)
3. Bloc de codificació de la instància a format SMT amb BV
4. Bloc d'optimització d'instàncies SMT

Essent els blocs de codificació els més importants ja que són els punts d'aplicació de tot el coneixement adquirit durant l'etapa d'anàlisi de l'estat de l'art. A més, en motiu de la codificació de les restriccions de cardinalitat s'han implementat els mètodes de codificació descrits en l'apartat anterior. Cal tenir en compte que la majoria són millores i/o refinaments d'altres encodings i per tant existia un alt reaprofitament del codi.

Al final, el generador és capaç de generar horaris utilitzant dues tècniques: SAT i SMT amb vectors de bits (BV). En cas d'utilitzar la tècnica SAT, existeixen dues versions: una que codifica les restriccions de cardinalitat utilitzant la codificació basada en *Modulo Totalizer* i l'altre que ho fa utilitzant la codificació *Mixed Cardinality Network*.

## 6 Resultats

El format XHSTT disposa d'un avaluador de solucions online capaç de determinar el valor d'optimalitat d'un horari i de mostrar-lo gràficament. Per tant, totes les solucions generades pels nostres generadors es passen per l'avaluador per validar-les. En la competició internacional s'admeten temps de còmput de fins a 24 hores per instància. En proves inicials, va quedar palès que el generador implementat en aquest treball no és capaç de superar els millors generadors i per això es va descartar la idea de generar proves amb temps de còmput tant elevats.

Les proves d'execució s'han realitzat amb un *timeout* de 1800 segons (mitja hora) que s'adapta prou al concepte de "temps acotat" al que es fa referència als objectius del projecte. A més, realitzar proves amb un *timeout* tan curt ens ha permès introduir millores (com nous encodings) en les etapes finals del treball. A continuació, es mostra una taula amb la comparativa entre les diferents versions del generador.

Generador	BrazilInstance1	BrazilInstance2	BrazilInstance3
SMT (BitVectors)	(0, 53)	(0, 88)	(0, 273)
MaxSAT (MCN)	(0, 86)	(0, 109)	(0, 245)
MaxSAT (MTO)	(0, 87)	(0, 103)	(0, 259)

Taula 1: Rendiment dels generadors basats en MaxSat, en les versions usant encodings de cardinalitat basats en *Mixed Cardinality Networks* i en la versió basada en *Modulo Totalizers* i SMT, basat en vectors de bits. Per cada solver i instància s'indiquen dos valors. El primer és el cost d'inviabilitat. És a dir, quin nombre de restriccions “inviolables” s'han violat. El segon representa el valor d'optimalitat de la solució obtinguda i es calcula a partir del nombre i pes de les restriccions “violables” que s'han violat.

De la taula 6 es destaca que el generador basat en MaxSAT guanya en quant a rendiment al generador basat en SMT en instàncies grans (la instància 1 i 2 són relativament petites). Mentre que la codificació basada en *Mixed Cardinality Network* per a restriccions de cardinalitat és més eficient que la homònima basada en *Modulo Totalizer*, que són les dues codificacions més sofisticades que s'han vist en aquest treball.

## 7 Conclusions

Al llarg d'aquest treball s'han vist i aplicat diverses tècniques per resoldre el problema de la generació d'horaris per a un institut (HSTT). El problema, ha permès posar a prova tots els coneixements que s'anaven adquirint al llarg del treball i si bé el producte final és millorable, compleix amb els requisits inicials ja que ha estat capaç de generar solucions “viables” per a totes les instàncies.

No obstant, el punt fort d'aquest treball ha estat l'estudi realitzat sobre varies de les tècniques de resolució deterministes de més èxit de l'actualitat. En especial l'estudi realitzat sobre SAT i també sobre part de SMT.

S'han aplicat els coneixements adquirits sobre la codificació de restriccions de cardinalitat per a realitzar implementacions alternatives de “collita pròpia” a l'hora de codificar restriccions de cardinalitat. A més s'ha intentat realitzar una síntesi amb paraules pròpies dels coneixements adquirits, per tal d'evidenciar l'assoliment del primer dels objectius del treball.

Aquest treball m'ha servit per consolidar els meus estudis en enginyeria informàtica; a entendre millor el món de la recerca, ajudant-me a comprendre millor la naturalesa dels problemes, especialment els NP-Complets.