

# Odabir CSP rešavača zasnovan na treniranju sa kratkim vremenskim ograničenjem.

Mirko Stojadinović i Filip Marić

Matematički fakultet  
Univerzitet u Beogradu

ARGO seminar 02.07.2014.

# Pregled

- 1 Uvod
- 2 Pozadina
- 3 ArgoCSPkNN pristup i eksperimentalni rezultati
- 4 Zaključci i dalji rad

# Motivacija

- Postoje mnogi metodi (i rešavači koji ih implementiraju) za rešavanje problema zadovoljenja ograničenja (Constraint Satisfaction Problems - CSP) kao i srodne optimizacione verzije ovog problema (COP).
- Ne postoji rešavač koji je efikasan u rešavanju svih tipova ovih problema.
- Postoje mnogi pristupi za odabir pogodnog rešavača iz skupa rešavača koji su na raspolaganju. Za pripremu (treniranje), ovi pristupi koriste puno vremena i zahtevaju jake računare sa velikim brojem jezgara.
- Cilj je razvoj novog pristupa koji ima znatno bržu pripremu koja se može obaviti na PC računaru.

# Konačni linearni CSP problemi

- Linearni izrazi nad konačnim skupom celobrojnih promenljivih  $V$  su algebarski izrazi oblika  $\sum_{i=0}^{m-1} a_i x_i$  gde  $\forall i \in \{0, \dots, m-1\} x_i \in V, a_i \in \mathbb{Z}$ .
- $B$  je skup Bulovskih promenljivih.
- Klauze se formiraju kao disjunkcije literala pri čemu su literali elementi unije skupova  $B, \{\neg p \mid p \in B\}$  i  $\{e \leq c \mid e \text{ je linearan izraz na } V, c \in \mathbb{Z}\}$ .
- $S$  je konačan skup klauza (nad  $V$  i  $B$ ).
- U praksi, Sintaksa se obično proširuje da podržava i razna druga ograničenja, npr. *globalna ograničenja*.

# Primene i primeri

## Primene CSP rešavača

- Pravljenje rasporeda časova, rasporeda smena.
- Igre i zagonetke: Sudoku, magični kvadrat, 8 dama, Golumbov lenjir.

## Jednostavan primer

```
(int x1 1 2)
```

```
(int x2 1 4)
```

```
(int x3 2 3)
```

```
(and (alldifferent(x1 x2 x3) ( $\leq$  x3 (+ x1 x2)))
```

Jedno od rešenja ovog problema je dodela  $x_1 = 1, x_2 = 2, x_3 = 3$ .

# Neke metode rešavanja CSP

## Svođenje na SAT

- Postoje razna kodiranja za svođenje na SAT: direktno, logaritamsko, kodiranje uređenja (order encoding), itd.
- Postoji veliki broj alata koji koriste svođenje na SAT: Spec2SAT, Sugar, Azucar, bee, ursa, *meSAT*.

## Svođenje na SMT

- Razvijeno više sistema: fzn2smt, ursa familija alata.

## Lenjo generisanje klauza

- U ovom pristupu su tesno povezani SAT rešavač i mehanizam propagacije, i jedno drugom prosleđuju informacije do kojih su došli. Ovaj pristup koriste rešavači mzn-g12lazy i mzn-g12cpx.

# Odabir solvera za SAT i CSP instance

- Različite metode su pogodne za različite tipove problema. Zato i ne postoji rešavač koji je bolji od ostalih na svim tipovima problema.
- *Portfelj* pristupi imaju za cilj da odaberu najbolji metod/rešavač za nerešenu CSP instancu.
- Mnogi portfelj pristupi su razvijeni za odabir SAT rešavača za SAT instancu (SATZilla, ISAC, ArgoSmArT) kao i za odabir CSP rešavača za CSP instancu (CPhydra, Proteus). Ovi pristupi koriste tehnike mašinskog učenja.

## Način funkcionisanja portfelj pristupa

- Prvo je potrebno odrediti skup *training instanci*. Svi dostupni rešavači pokreću se na ovim instancama i sakupljaju se karakteristike instanci (*training faza*).
- Korišćenjem određene metode mašinskog učenja na osnovu rezultata training faze određuje se rešavač koji treba pokrenuti na nekoj novoj instanci koju treba rešiti (*test faza*).



# ArgoCSPknn pristup

## Poreklo pristupa

- Zasnovan je na ArgoSmArT pristupu koji koristi algoritam  $k$  najbližih komšija.

## Parametri

- Koriste se 70 karakteristika CSP instanci (broj pojavljivanja svake vrste ograničenja, zbir domena operanada, itd.).
- Udaljenost između dve instance  $P$  i  $Q$  se izračunava pomoću neke od 4 funkcija udaljenosti. Jedna od njih je
$$d(P, Q) = \sqrt{\sum_{i=1}^{70} (P_i - Q_i)^2}.$$
- Koristi se različit broj komšija  $k$ , gde ovaj broj varira između 1 i 20.

## Trening faza ArgoCSPknn pristupa

- Pušta se svaki rešavač na svakoj trening instanci sa vremenskim ograničenjem  $t$  i prikupljaju se karakteristike ovih instanci. *PAR10 skor* za rešavač i instancu je vreme utrošeno za rešavanje instance ako je rešena, a  $10 * t$  ako nije rešena.
- Fiksirajmo neko  $d$  i  $k$ . Za svaku instancu trening skupa:
  - Izdvaja se ta instanca i nalaze se  $k$  najbliže instance među ostalim instancama koristeći funkciju udaljenosti  $d$ .
  - *PAR10 skor* za rešavač je zbir vrednosti *PAR10 skorova* za  $k$  najbližih instanci. Rešavač sa najmanjim *PAR10 skorom* je odabran za tu instancu. *PAR10 skor* rešavača za izdvojenu trening instancu dodaje se na ukupan skor.
- Ukupni skorovi se izračunavaju za sve kombinacije  $d$  i  $k$  (80 njih). Kombinacija sa najmanjim ukupnim skorom ( $D$  i  $K$ ) je ona koja će se koristiti u test fazi.

## Test faza ArgoCSPknn pristupa

- Prikupljaju se karakteristike nove instance koju treba rešiti.
- Nalaze se  $K$  najbliže instance među trening instancama koristeći funkciju udaljenosti  $D$ .
- Za svaki rešavač, sabiraju se vrednosti PAR10 skorova za  $K$  najbližih instanci.
- Rešavač sa najmanjim zbirom bira se kao rešavač nove instance.

# Eksperimentalni rezultati

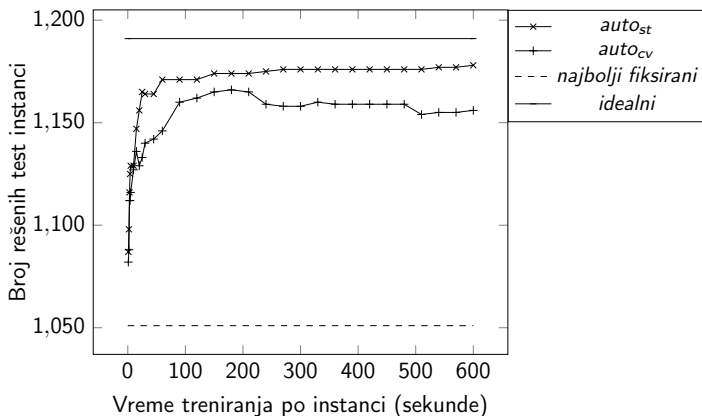
- Korišćen klaster Jason: AMD Opteron(tm) CPU 6168 na 1.9Ghz sa 2GB RAM po procesoru.
- Instance sa CSP takmičenja, iz Minizinc korpusa i instance problema Dominirajuće kraljice. Korišćene samo one instance koje imaju globalna ograničenja.
- Vremenskog ograničenje za treniranje varira, za testiranje 600s.
- Korišćeni rešavači:
  - Svođenje na SAT: *meSAT*, Sugar i Azucar. Kao SAT rešavač korišćen Minisat.
  - Svođenje na SMT pomoću sistema *meSAT*. Kao SMT rešavač korišćen Yices.
  - Lenjo generisanje klauza. Korišćeni su rešavači *mzn-g12lazy* i *mzn-g12cpx*.

# Rezultati za korišćene rešavače

Metod rešavanja	Rešavač	# (od 1379)	Vreme
Svođenje na SAT	<i>meSAT</i> (direktno)	945	4755
	Sugar	1051	3862
	Azucar	959	4713
	Azucar-log	916	5278
Svođenje na SMT	Yices	771	6456
Lenjo generisanje klauza	mzn-g12cpx	707	7002
	mzn-g12lazy	845	5824
	<i>najbolji fiksirani</i>	1051	3862
	<i>idealni</i>	1191	2200

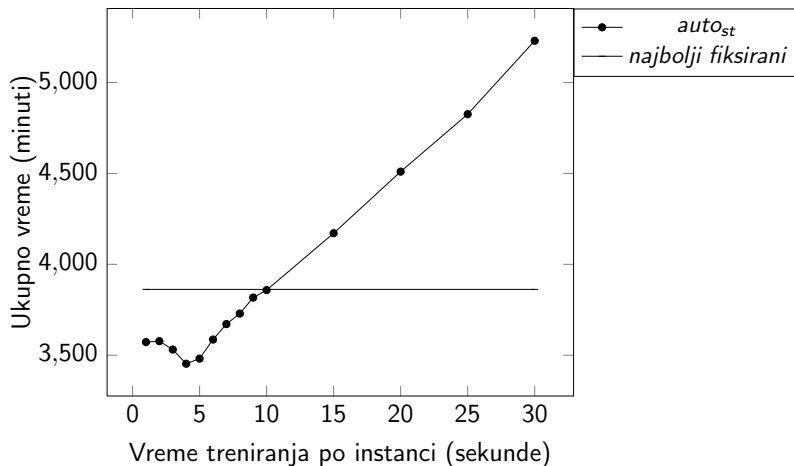
Tabela 1: Rezultati na svim instancama; # je broj rešenih instanci; vreme je dato u minutima; *najbolji fiksirani* – najefikasniji rešavač (Sugar u ovom slučaju), *idealni* – najbolji rešavač za svaku instancu.

# Poređenje različitih pristupa



Slika 1: *auto<sub>cv</sub>* – odabir korišćenjem unakrsne validacije, *auto<sub>st</sub>* – odabir korišćenjem kratkog treninga na svim instancama.

# Ukupno utrošeno vreme



Slika 2: Rezultati *auto<sub>st</sub>* i metoda *najbolji fiksirani*.

# Zaključci i dalji rad

## Zaključci

- Metod zasnovan na unakrsnoj validaciji predstavlja dobru generalizaciju i sa vrlo kratkim treniranjem može se značajno unaprediti efikasnost u odnosu na pojedinačne rešavače.
- Metod koji koristi kratko treniranje može predstavljati dobro rešenje u slučaju kada je na nekom korpusu potrebno rešiti što veći broj instanci za neko ograničeno vreme.

## Dalji rad

- Korišćenje šireg skupa rešavača i instanci.
- Testiranje hipoteze da li se slična metodologija može primeniti i na SAT rešavače.