Sertifikovano svođenje CSP problema zapisanih na jeziku FlatZinc na SAT

Matija Lojović

Katedra za računarstvo i informatiku
Matematički fakultet, Univerzitet u Beogradu

April 26, 2026

Pregled

1. Uvod i motivacija
2. FlatZinc-SAT kodiranje
3. Teorijski okvir
4. SMT kodiranje
5. Zaključak

CSP

Definicija

Problem zadovoljenja ograničenja (CSP) podrazumeva dodelu vrednosti svakoj od *promenljivih* koje učestvuju u problemu. Svaka od promenljivih uzima vrednost iz svog fiksiranog *domena*, dok *ograničenja* koja figurišu u problemu određuju koje vrednosti promenljivih ne mogu ići zajedno.

CSP rešavači

- Ukoliko postoji dodela koja zadovoljava sva ograničenja, problem ima *rešenje*
- Alati za rešavanje CSP - *CSP rešavači*
- Zasnovani na *propagaciji* i *pretrazi*
- Zahtevaju standardizovan ulazni jezik (npr. *MiniZinc*)

Primer

Modelovanje problema 8 dama u vidu CSP-a:

- **Promenljive:** x_i za $i \in \{1, \ldots, 8\}$, gde svako x_i predstavlja broj reda u kojem će biti postavljena dama iz i-te kolone.
- **Domeni:** $D_i = \{1, \ldots, 8\}$ za svaku promenljivu x_i.
- **Ograničenja:**
  1. **Ograničenja redova:** Svaki red mora sadržati tačno jednu damu:

$$\forall i, j \in \{1, \ldots, 8\} \quad i < j \implies x_i \neq x_j$$

  2. **Ograničenja dijagonala:** Dve dame ne smeju biti postavljene na istoj dijagonali:

$$\forall i, j \in \{1, \ldots, 8\} \quad i < j \implies |x_i - x_j| \neq |i - j|$$

SAT problem

Definicija

SAT problem predstavlja problem ispitivanja zadovoljivosti iskazne formule u proizvoljnom obliku

Osobine SAT problema

- Pripada klasi *NP-kompletnih* problema
- Postoje efikasni alati koji rešavaju SAT problem - *SAT rešavači*
- Najčešće kao ulaz dobijaju formulu u *KNF obliku* (DIMACS format)
- Brojne praktične primene - mnogi problemi se mogu modelovati kao SAT problem; mnogi problemi se mogu svesti na SAT problem

Primer KNF formule

$(p \vee \neg q) \wedge (r \vee q \vee \neg p) \wedge (\neg r \vee \neg q)$

Svođenje CSP na SAT

Pristupi

- *Nestrpljivo (eager) generisanje klauza* - Kompletan problem se kodira u vidu KNF formule
- Bira se vrsta *kodiranja* (*uređeno*, *direktno*, *log*...)
- Ukoliko je dobijeni SAT problem nezadovoljiv, polazni CSP problem nema rešenja
- U suprotnom, potrebno je *dekodirati* zadovoljavajuću valuaciju SAT problema kako bi se dobilo rešenje polaznog CSP problema.

Korektnost kodiranja

Pitanje korektnosti

- Dosadašnji pristupi svođenju su mahom podrazumevali *korektnost* kodiranja, bez formalnih garancija
- Za primene gde je cena greške velika, ovo je često neprihvatljivo
- Kod *optimizacionih problema sa ograničenjima* želeli bismo i garanciju optimalnosti rešenja

Pristupi verifikaciji

Formalizacija

- Formalizacija pomoću *interaktivnih dokazivača teorema*
- Daje formalne garancije, ali može iziskivati značajne napore
- Dokazuje se korektnost samo jednog konkretnog kodiranja

Generisanje obligacija

- *Obligacije* - tvrđenja čijim dokazivanjem se dokazuje korektnost prevođenja za konkretne instance problema
- Ta tvrđenja se dalje mogu dokazati korišćenjem *SMT rešavača*

Tok rešavanja

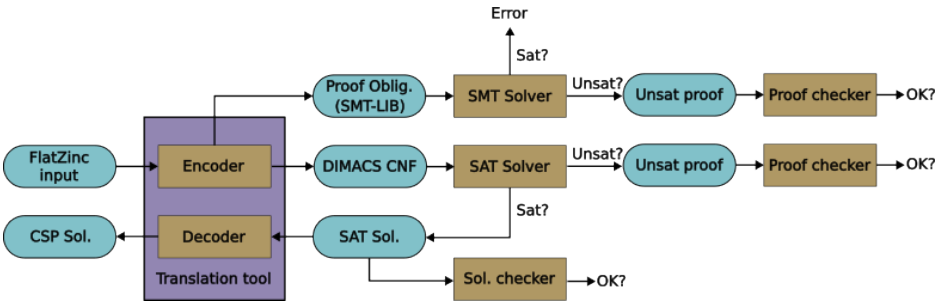

Figure: Tok rešavanja korišćenjem našeg alata

FlatZinc-SAT kodiranje

Kodiranje domena promenljivih

- Pre kodiranja ograničenja, moraju se zadati domeni promenljivih
- Tri tipa promenljivih: *celobrojne*, *logičke (bulovske)*, *skupovne*
- Dva smera - *kodiranje* domena KNF formulom; *dekodiranje* zadovoljavajuće valuacije
- *Konzistentnost dodele* - svaka promenljiva uzima tačno jednu vrednost iz domena

Celobrojne promenljive

Uređeno kodiranje

- *Kodiranje* - za svaku vrednost a iz domena promenljive x postoji iskazna promenljiva koja je tačna akko $x \leq a$

$$\forall a \in \{l(x) - 1, \ldots, u(x)\}, \quad p_{x,a} \Leftrightarrow x \leq a$$

- *Dekodiranje* - Pronaći vrednost i za koju je $p_{x,i}$ tačno, a $p_{x,i-1}$ netačno
- Kako bi se obezbedila *konzistentnost dodele* dodaju se sledeće klauze u KNF formulu:

$$\forall i \in \{l(x), \ldots, u(x)\}, \quad \neg p_{x,i-1} \vee p_{x,i}$$

$$\neg p_{x,l(x)-1}$$

$$p_{x,u(x)}$$

Pomoćne promenljive

Korišćenje pomoćnih promenljivih

U situaciji kada je iskaznu formulu u DNF-u potrebno prebaciti u KNF, kako bi se smanjio rezultujući broj klauza, moguće je iskoristiti *pomoćne promenljive*. Ovo dolazi uz cenu održavanja ekvizadovoljivosti formule, ali ne i ekvivalentnosti. Konverzija se vrši na sledeći način:

$$\bigvee_{i=1}^{n}\bigwedge_{j=1}^{m} p_{ij} \mapsto \bigwedge_{i=1}^{n}\bigwedge_{j=1}^{m}(p_{ij} \vee \neg h_i)$$

pri čemu je potrebno dodati i disjunkciju pomoćnih promenljivih:

$$\bigvee_{i=1}^{n} h_i$$

Primer kodiranja

Primer kodiranja jednostavnog ograničenja

Kodiranje ograničenja $x + y \neq 4$ za $D_x, D_y = \{1, 2, 3\}$:

$$
\begin{aligned}
& \neg x_0 \wedge x_3 \wedge (\neg x_0 \vee x_1) \wedge (\neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_3) \\
\wedge\ & \neg y_0 \wedge y_3 \wedge (\neg y_0 \vee y_1) \wedge (\neg y_1 \vee y_2) \wedge (\neg y_2 \vee y_3) \\
\wedge\ & \neg h_1 \vee x_1 \vee y_1 \\
\wedge\ & \neg h_1 \vee x_0 \vee y_2 \\
\wedge\ & \neg h_1 \vee x_2 \vee y_0 \\
\wedge\ & \neg h_2 \vee \neg x_1 \vee \neg y_3 \\
\wedge\ & \neg h_2 \vee \neg x_2 \vee \neg y_2 \\
\wedge\ & \neg h_2 \vee \neg x_3 \vee \neg y_1 \\
\wedge\ & h_1 \vee h_2
\end{aligned}
$$

Logički okvir

Osnovni pojmovi

- Za formalno rezonovanje o CSP problemima koristimo *višesortnu logiku prvog reda*
- *Domen* promenljive x celobrojne sorte, u oznaci $D(x)$, je podskup skupa celih brojeva, određen *ograničenjima domena*. Na primer, za $D(x) = \{1, 2, 3, 4, 5\}$ ograničenja domena su $x \geq 1$ i $x \leq 5$
- Neka je P CSP problem nad promenljivama $\{x_1, \ldots, x_n\}$. Tada je *prostor pretrage* problema P, u oznaci $Dom(P)$, skup $D(x_1) \times \cdots \times D(x_n)$
- Kažemo da je S_P *potprostor* problema P, ako važi $S_P \subseteq Dom(P)$. Potprostor S_P je najčešće zadat pomoću ograničenja

Postavka problema svođenja

- Neka su dati CSP problemi F i G (u našem kontekstu F će biti *FlatZinc* model, a G SAT problem). Želimo da pronađemo *relaciju svođenja* R koja će uspostaviti preslikavanje između rešenja problema F i problema G

Primer

Za ranije navedeni primer ograničenja $x + y \neq 4$, problem F je predstavljen sledećom logičkom formulom:

$$F(x, y) = x \geq 1 \wedge x \leq 3 \wedge y \geq 1 \wedge y \leq 3 \wedge (x + y \leq 3 \vee x + y \geq 5)$$

Problem G je predstavljen ranije navedenom SAT formulom.
Relacija R je data pravilima uređenog kodiranja:

$$\begin{aligned} x_0 &\Leftrightarrow x \leq 0 \wedge \\ x_1 &\Leftrightarrow x \leq 1 \wedge \\ &\vdots \\ y_3 &\Leftrightarrow y \leq 3 \end{aligned}$$

Definicije svojstava

Sadržavanje

Kažemo da je potprostor S_P *sadržavajući* ukoliko sadrži sva rešenja polaznog problema P

Totalnost

- Kažemo da je relacija R *levo-totalna* u odnosu na potprostore S_F i S_G ukoliko za svaku torku $u \in S_F$ postoji torka $v \in S_G$ takva da važi $R(u, v)$
- Kažemo da je relacija R *desno-totalna* u odnosu na potprostore S_F i S_G ukoliko za svaku torku $v \in S_G$ postoji torka $u \in S_F$ takva da važi $R(u, v)$

Saglasnost

Kažemo da je relacija R *saglasna* ukoliko za svako $u \in S_F$ i svako $v \in S_G$ za koje važi $R(u, v)$, važi i da je u rešenje problema F akko je v rešenje problema G

Zašto se ograničavamo na potprostore?

Narušena saglasnost

- U ranije navedenom primeru, postoje rešenja problema F koja su u relaciji sa valuacijama koje nisu rešenje problema G (dobijenim postavljanjem h_1 i h_2 na *false*)
- Posledično, relacija R nije saglasna.
- Razlog leži u *pomoćnim promenljivama*

Rešenje

Ograničavamo se na potprostor S_G:

$$\begin{aligned} &h_1 \Rightarrow ((x_1 \vee y_1) \wedge (x_0 \vee y_2) \wedge (x_2 \vee y_0)) && \wedge \\ &h_2 \Rightarrow ((\neg x_1 \vee \neg y_3) \wedge (\neg x_2 \vee \neg y_2) \wedge (\neg x_3 \vee \neg y_1)) && \wedge \\ &h_1 \vee h_2 \end{aligned}$$

Zašto se ograničavamo na potprostore?

Narušena leva-totalnost

- Sada postoje torke u $Dom(F)$ koje nisu u relaciji ni sa jednom torkom iz S_G (dobijene za dodele kod kojih je i $x + y \leq 3$ i $x + y \geq 5$ netačno).
- Posledično, relacija R nije levo-totalna

Rešenje

Ograničavamo se na potprostor S_F:

$$x + y \leq 3 \vee x + y \geq 5$$

Relacija svođenja

Relacija R je sada *relacija svođenja* nad S_F i S_G, a posledično se može koristiti za svođenje problema F na problem G.

Dokazivaje svojstava

Sadržavanje

$\forall x_1 \dots x_n . P(x_1, \dots, x_n) \Rightarrow S_P(x_1, \dots, x_n)$

Leva-totalnost i desna-totalnost

$\forall x_1 \dots x_n . S_F(x_1, \dots, x_n) \Rightarrow (\exists y_1 \dots y_m . S_G(y_1, \dots, y_m) \wedge R(x_1, \dots, x_n, y_1, \dots, y_m))$
$\forall y_1 \dots y_m . S_G(y_1, \dots, y_m) \Rightarrow (\exists x_1 \dots x_n . S_F(x_1, \dots, x_n) \wedge R(x_1, \dots, x_n, y_1, \dots, y_m))$

Saglasnost

$\forall x_1 \dots x_n y_1 \dots y_m .$
$S_F(x_1, \dots, x_n) \wedge S_G(y_1, \dots, y_m) \wedge R(x_1, \dots, x_n, y_1, \dots, y_m) \Rightarrow$
$(F(x_1, \dots, x_n) \Leftrightarrow G(y_1, \dots, y_m))$

SMT kodiranje

Svrha SMT kodiranja

- Želimo da predstavimo oba problema na *SMT-LIB* jeziku, kako bismo za dokazivanje tvrđenja iskoristili *SMT rešavač*
- SAT problem se predstavlja trivijalno
- Većina FlatZinc ograničenja se predstavlja trivijalno

Primer

- $int\_plus(x, y, z)$ se kodira kao $(= \ (+ \ x \ y) \ z)$ u *LIA* teoriji
- $int\_max(x, y, z)$ se kodira kao $(= z \ (ite \ (> \ x \ y) \ x \ y))$ u *LIA* teoriji
- $int\_times(x, y, z)$ se kodira kao $(= \ (* \ x \ y) \ z)$ u *NIA* teoriji

SMT kodiranje

Netrivijalna ograničenja

- Ipak, postoje ograničenja koja nije trivijlano kodirati
- Ovde leži najveća "rupa" našeg pristupa - SMT kodiranju moramo verovati

SMT kodiranje

Primer netrivijalnog kodiranja

```
(define-fun mzn_mod_f ((x Int) (y Int)) Int
   (ite (and (< x 0) (distinct (mod x y) 0))
       (- (mod x y) (abs y))
       (mod x y)
   )
)

(define-fun mzn_mod ((x Int) (y Int) (z Int)) Bool
   (and
       (distinct y 0)
       (= z (mzn_mod_f x y))
   )
)

(mzn_mod x y z)  ; encodes int_mod(x,y,z)
```

Zaključak

Završni komentari

- *Evaluacija*
- Demonstracija alata
- Pitanja, komentari i ideje

HVALA VAM NA PAŽNJI