Benchmarks from Reduction Finding

Skip Jordan and Łukasz Kaiser

ERATO Group, JST & Hokkaido University
LIAFA, CNRS & Université Paris Diderot – Paris 7

QBF 2013

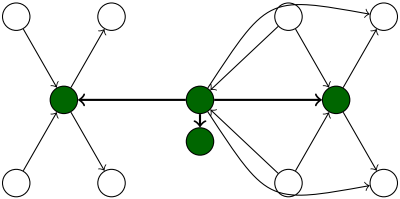
Structures and Second-Order Logic

Motivation: Higher Abstraction Level for QBF

Structures and Second-Order Logic

Motivation: Higher Abstraction Level for QBF

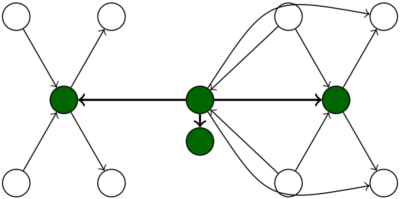
Relational Structures $\mathfrak{A} = (A, R_1^{\mathfrak{A}}, R_2^{\mathfrak{A}}, \ldots, R_k^{\mathfrak{A}}, c_1^{\mathfrak{A}}, \ldots, c_l^{\mathfrak{A}})$



Structures and Second-Order Logic

Motivation: Higher Abstraction Level for QBF

Relational Structures $\mathfrak{A} = (A, R_1^{\mathfrak{A}}, R_2^{\mathfrak{A}}, \ldots, R_k^{\mathfrak{A}}, c_1^{\mathfrak{A}}, \ldots, c_l^{\mathfrak{A}})$



First-Order and Second-Order Logic over $\sigma = \{\mathbf{E}\}$.

The graph is a clique (FO): $\forall x, y(x = y \vee \mathbf{E}(x, y))$

The graph is 3-colorable (SO):

$$\exists R, G, B\Big(\forall x, y\,(R(x) \vee G(x) \vee B(x)) \wedge \big(\mathbf{E}(x, y) \rightarrow \neg\,((R(x) \wedge R(y)) \vee (G(x) \wedge G(y)) \vee (B(x) \wedge B(y)))\big)\Big)$$

Model-Checking SO using QBF Solvers

Transformation: SO $\ni \varphi$, $\mathfrak{A} \rightsquigarrow \psi$ **QBF**

$$\mathbf{Rel}(a_1, \ldots, a_k) \rightsquigarrow \top /_{\mathfrak{A}} \bot \qquad Var(a_1, \ldots, a_k) \rightsquigarrow X_{Var,a_1,\ldots,a_k}$$

$$\varphi_1 \wedge \varphi_2 \rightsquigarrow \hat{\varphi}_1 \wedge \hat{\varphi}_2 \qquad \exists x \varphi \rightsquigarrow \bigvee_{a \in \mathfrak{A}} \hat{\varphi}(a) \qquad \forall x \varphi \rightsquigarrow \bigwedge_{a \in \mathfrak{A}} \hat{\varphi}(a)$$

$$\exists V \varphi \rightsquigarrow \exists X_{V,1,1} \ldots X_{V,n,n} \hat{\varphi} \qquad \forall V \varphi \rightsquigarrow \forall X_{V,1,1} \ldots X_{V,n,n} \hat{\varphi}$$

Model-Checking SO using QBF Solvers

Transformation: $\text{SO} \ni \varphi, \mathfrak{A} \rightsquigarrow \psi$ **QBF**

$$\mathbf{Rel}(a_1, \ldots, a_k) \rightsquigarrow \top /_{\mathfrak{A}} \bot \qquad Var(a_1, \ldots, a_k) \rightsquigarrow X_{Var, a_1, \ldots, a_k}$$

$$\varphi_1 \wedge \varphi_2 \rightsquigarrow \hat{\varphi}_1 \wedge \hat{\varphi}_2 \qquad \exists x \varphi \rightsquigarrow \bigvee_{a \in \mathfrak{A}} \hat{\varphi}(a) \qquad \forall x \varphi \rightsquigarrow \bigwedge_{a \in \mathfrak{A}} \hat{\varphi}(a)$$

$$\exists V \varphi \rightsquigarrow \exists X_{V,1,1} \ldots X_{V,n,n} \hat{\varphi} \qquad \forall V \varphi \rightsquigarrow \forall X_{V,1,1} \ldots X_{V,n,n} \hat{\varphi}$$

Steps: (1) construct the QBF (2) use a QBF solver (3) read the answer

Model-Checking SO using QBF Solvers

Transformation: SO $\ni \varphi, \mathfrak{A} \rightsquigarrow \psi$ **QBF**

$$\mathbf{Rel}(a_1, \ldots, a_k) \rightsquigarrow \top /_{\mathfrak{A}} \bot \qquad Var(a_1, \ldots, a_k) \rightsquigarrow X_{Var, a_1, \ldots, a_k}$$

$$\varphi_1 \wedge \varphi_2 \rightsquigarrow \hat{\varphi}_1 \wedge \hat{\varphi}_2 \qquad \exists x \varphi \rightsquigarrow \bigvee_{a \in \mathfrak{A}} \hat{\varphi}(a) \qquad \forall x \varphi \rightsquigarrow \bigwedge_{a \in \mathfrak{A}} \hat{\varphi}(a)$$

$$\exists V \varphi \rightsquigarrow \exists X_{V,1,1} \ldots X_{V,n,n} \hat{\varphi} \qquad \forall V \varphi \rightsquigarrow \forall X_{V,1,1} \ldots X_{V,n,n} \hat{\varphi}$$

Steps: (1) construct the QBF (2) use a QBF solver (3) read the answer

Example: 3-colouring a graph

`http://toss.sf.net/eval.html`

Two Basic Applications

Example (and counter-example) finding

$$\varphi \in \text{FO},\ n \in \mathbb{N} \quad \rightsquigarrow \quad \mathfrak{A} \mid |\mathfrak{A}| = n \text{ and } \mathfrak{A} \models \varphi \quad (\text{or } \mathfrak{A} \models \neg\varphi)$$

Reduction: change all relations in φ to SO variables

Two Basic Applications

Example (and counter-example) finding

$$\varphi \in \mathrm{FO},\ n \in \mathbb{N} \quad \rightsquigarrow \quad \mathfrak{A} \mid |\mathfrak{A}| = n \text{ and } \mathfrak{A} \models \varphi \quad (\text{or } \mathfrak{A} \models \neg\varphi)$$

Reduction: change all relations in φ to SO variables

Formula finding

$$\text{outline of } \varphi,\ \mathfrak{A} \quad \rightsquigarrow \quad \varphi \mid \mathfrak{A} \models \varphi$$

Outline: formula with Boolean atom guards. Example:

$$X_1\mathbf{E}(x_1, x_1) \wedge X_2\mathbf{E}(x_1, x_2) \wedge X_3\mathbf{E}(x_2, x_1) \wedge X_4\mathbf{E}(x_2, x_2) \wedge$$
$$X_5\neg\mathbf{E}(x_1, x_1) \wedge X_6\neg\mathbf{E}(x_1, x_2) \wedge X_7\neg\mathbf{E}(x_2, x_1) \wedge X_8\neg\mathbf{E}(x_2, x_2)$$

Automatic Reduction Finding

FO Interpretations (Queries): $\theta = (k, \varphi_0, \psi_1, \ldots, \psi_r)$

- k is the dimension
- $\varphi_0(x_1, \ldots, x_k)$ defines the new universe
- $\psi_i(x_1, \ldots, x_{kr_i})$ define the new relations

Automatic Reduction Finding

FO Interpretations (Queries): $\theta = (k, \varphi_0, \psi_1, \ldots, \psi_r)$

- k is the dimension
- $\varphi_0(x_1, \ldots, x_k)$ defines the new universe
- $\psi_i(x_1, \ldots, x_{kr_i})$ define the new relations

Complexity classes under interpretations (Immerman)

- **quantifier-free interpretations** are weaker than PTime
- **still** P=NP iff SAT $\leq_{\text{qfp}}$CVP
- **and** NL=NP iff SAT $\leq_{\text{qfp}}$REACH,
- **and** coNL=NL (true) iff $\neg$REACH $\leq_{\text{qfp}}$REACH

Automatic Reduction Finding

FO Interpretations (Queries): $\theta = (k, \varphi_0, \psi_1, \ldots, \psi_r)$

- k is the dimension
- $\varphi_0(x_1, \ldots, x_k)$ defines the new universe
- $\psi_i(x_1, \ldots, x_{kr_i})$ define the new relations

Complexity classes under interpretations (Immerman)

- **quantifier-free interpretations** are weaker than PTime
- **still** P=NP iff SAT $\leq_{\mathrm{qfp}}$CVP
- **and** NL=NP iff SAT $\leq_{\mathrm{qfp}}$REACH,
- **and** coNL=NL (true) iff $\neg$REACH $\leq_{\mathrm{qfp}}$REACH

Finding reductions **QBF:** $\exists\theta\ \forall\mathfrak{A}\ (\mathfrak{A} \models \varphi_P \ \leftrightarrow\ \theta(\mathfrak{A}) \models \varphi_Q)$ **CEGAR:**

- Find a l-DNF reduction θ_i good on counter-examples $\mathfrak{E}_0, \ldots, \mathfrak{E}_i$
- Find a counter-example $\mathfrak{E}_{i+1}$ to θ_i, iterate

Reduction Finding Results

Easy example: s-t reachability to strongly connected (both NL-complete)

$$\text{Reach} = [\text{tc}_{x,y}\, \mathbf{E}(x,y)](.s,.t) \qquad \text{SC} := \forall x, y(\text{tc}_{x,y}\, \mathbf{E}(x,y))$$

Reduction Finding Results

Easy example: s-t reachability to strongly connected (both NL-complete)

$$\text{Reach} = [\text{tc}_{x,y}\, \mathbf{E}(x,y)](.s,.t) \qquad \text{SC} := \forall x, y (\text{tc}_{x,y}\, \mathbf{E}(x,y))$$

http://toss.sf.net/reduct.html

Reduction Finding Results

Easy example: s-t reachability to strongly connected (both NL-complete)

$$\text{Reach} = [\text{tc}_{x,y}\, \mathbf{E}(x,y)](.s,.t) \qquad \text{SC} := \forall x,y(\text{tc}_{x,y}\, \mathbf{E}(x,y))$$

$$(k=1,\ \varphi_0 = \top,\ \psi_1 = x_1 = .s \lor x_2 = .t \lor \mathbf{E}(x_2,x_1))$$

Reduction Finding Results

Easy example: s-t reachability to strongly connected (both NL-complete)

$$\text{Reach} = [\text{tc}_{x,y}\, \mathbf{E}(x,y)](.s,.t) \qquad \text{SC} := \forall x,y(\text{tc}_{x,y}\, \mathbf{E}(x,y))$$

$$(k = 1,\ \varphi_0 = \top,\ \psi_1 = x_1 = .s \vee x_2 = .t \vee \mathbf{E}(x_2, x_1))$$

 # Unsolved Cases (out of 2304)

(c,n)	$(1,3)$	$(2,3)$	$(3,3)$	$(1,4)$	$(2,4)$	$(3,4)$
rareqs	0	0	16	19	65	204
depqbf	0	142	547	16	297	711
qube	10	536	949	82	760	1082
cirqit	58	673	1138	511	1092	1357
skizzo	522	1058	1156	975	1327	1434

Outlook

Experiences with Reduction Finding

- 2QBF in principle, 3QBF after qdimacs conversion
- Better to use **3QBF**, **qpro** helps when using **cirqit**
- Best performing QBF solver: **rareqs**
- Missing part against hand-made CEGAR: **incremental solving**
- Generator at `http://toss.sf.net/reductGen.html`

Outlook

Experiences with Reduction Finding

- 2QBF in principle, 3QBF after qdimacs conversion
- Better to use **3QBF**, **qpro** helps when using **cirqit**
- Best performing QBF solver: **rareqs**
- Missing part against hand-made CEGAR: **incremental solving**
- Generator at `http://toss.sf.net/reductGen.html`

Other Applications

- Finding LFP formulas for **NP $\cap$ coNP** properties
- Solving ***n*-step winning** in board games
  - the same generator plugged into a **GGP system**
  - easy to generate QBFs for many games and board sizes
  - best performing QBF solver (only a few tests): **depqbf**
  - `http://toss.sf.net/gameGen.html`
- **QBF is great for many applications** (but needs effort)

Outlook

Experiences with Reduction Finding

- 2QBF in principle, 3QBF after qdimacs conversion
- Better to use **3QBF**, **qpro** helps when using **cirqit**
- Best performing QBF solver: **rareqs**
- Missing part against hand-made CEGAR: **incremental solving**
- Generator at `http://toss.sf.net/reductGen.html`

Other Applications

- Finding LFP formulas for **NP $\cap$ coNP** properties
- Solving ***n*-step winning** in board games
  - the same generator plugged into a **GGP system**
  - easy to generate QBFs for many games and board sizes
  - best performing QBF solver (only a few tests): **depqbf**
  - `http://toss.sf.net/gameGen.html`
- **QBF is great for many applications** (but needs effort)

Thank You