

Benjamin De Leeuw

Departement Zuivere Wiskunde en Computer Algebra
Galglaan 2, B-9000 Gent
bdeleeuw@cage.ugent.be
Gent University, Belgium

November 2, 2004

1 Inleiding

1.1 Doelstelling

In deze praktische lessen voor het vak Herschrijfsystemen, zal een inleiding gegeven worden op een belangrijke software-tool voor de specificatie en uitvoering van herschrijfsystemen, die gebaseerd zijn op congruentierelaties. OBJ3 implementeert de theorie uit hoofdstukken 4 tem. 6 van de cursus bij dit vak.

OBJ3 is een interactief systeem dat toelaat om een systeem van gelijkheden E in te geven, dat bij toepassing van het *reduce* commando geïnterpreteerd wordt als herschrijfsysteem, door de regels van links naar rechts te ordenen. De specificatie van E gebeurt in een teksteditor. Bestandsnamen hebben extensie *.obj* en kunnen bewerkt worden in een teksteditor naar keuze (vi, pico, ...). We beschouwen de syntaxis en semantiek van zulke tekstbestanden in deze lessen.

Een systeem van gelijkheden, geïnterpreteerd als herschrijfsysteem kan worden ingeladen in OBJ3, en herschrijvingen (*reductions*) kunnen uitgevoerd worden, naargelang de wensen en de doelstelling van de gebruiker. Herschrijfsystemen hebben een belangrijke toepassing in de automatische bewijsvoering. Een gebruiker van OBJ3 zal meestal als belangrijkste doel hebben om een systeem van gelijkheden op te stellen dat zekere voorkeurseigenschappen bezit (door vergelijkingen toe te voegen of weg te laten uit het oorspronkelijke systeem, zie theorie). We zullen voornamelijk toepassingen van deze aard bestuderen.

1.2 Opstarten en Afsluiten

Nadat een systeem van gelijkheden werd ingegeven met de juiste syntaxis, kan dit bestand worden ingelezen. Een tekstbestand voor OBJ3 heeft volgende algemene structuur:

```
obj OBJECTNAAM is
  protecting BOOL .
  sorts element Bool .
  op o : - > element .
  ...
  op - _ : element - > element .
  ...
  op _ + _ : element element - > element .
  ...
  var x : element .
  ...
  [e1] eq x + x = x .
  [e2] eq x + - x = o .
  ...
endo
```

en wordt opgeslagen als *.obj* bestand. Men kan OBJ3 starten via de console:

```
[prompt]$ obj3
```

Om uitvoer naar het scherm te bewaren in een logfile voor later gebruik en nazicht gebruikt men beter:

```
[prompt]$ obj3 | tee logbestandsnaam.log
```

Indien men voor elke projectuitvoering een aparte logfile wil voorzien, zal men steeds opnieuw OBJ3 moeten opstarten met bovenstaand commando, telkens met een andere bestandsnaam. Bewaren van uitvoeringen (*traces*) is immers niet mogelijk op een andere manier in OBJ3. Met het volgende commando wordt een systeem van gelijkheden ingeladen:

```
OBJ > in bestandsnaam
```

zonder de extensie *.obj*. Wanneer er geen foutmeldingen verschijnen, kon OBJ3 het systeem *E*, zoals opgegeven in *bestandsnaam.obj* naar behoren parsen (syntactisch verstaan).

Verlaat OBJ3 met:

```
OBJ > q
```

Nadien kan de uitvoering nog eens bekeken worden (indien de logging gebruikt werd) door met een teksteditor het opgegeven logbestand te openen (*logbestandsnaam.log*).

1.3 Commando's

We bespreken hier kort enkele belangrijke commando's van OBJ3. Voor meer uitleg en voorbeelden verwijzen we naar verschillende beschikbare basiswerken omtrent OBJ3 (zie website prof. Hoogewijs). We vatten aan met de *show* commando's:

show all . print alle bekende regels, termen en objectinformatie op het scherm, ongeveer zoals deze vermeld staat in de inputfile (*.obj*).

show rules . print alle bekende regels op het scherm.

show term . print de huidige *term* ter verwerking of reductie af.

show OBJECTNAAM . print informatie over het huidige object af op het scherm, ongeveer zoals deze vermeld staat in de inputfile.

Voor de oefeningen zal steeds een *trace* gebruikt worden:

set trace on . print alle herschrijvingen die gebruikt werden in de reductie op het scherm.

set trace off . zet deze functionaliteit af.

De uitvoering en herschrijving van het ingeladen systeem van gelijkheden wordt met de volgende commando's gecontroleerd:

reduce a + b == c + d . reduceert een uitdrukking, zoals deze die wordt weergegeven, op basis van het opgegeven systeem van gelijkheden *E*. Dit is het belangrijkste commando van OBJ3. Er wordt zo ver mogelijk gereduceerd om tot de equivalentie te komen. Het al of niet bekomen van een equivalentie wordt aangegeven met *true* of *false* respectievelijk.

Omdat het niet steeds wenselijk is het programma vrij te laten herschrijven, zonder eigen inbreng, zijn er ook een aantal commando's om een term in stappen of in subtermen te herschrijven:

start a + b + c . geeft op wat we bedoelen met *term* tot een nieuwe term wordt opgegeven, of een herschrijving wordt uitgevoerd. Deze term wordt dus getransformeerd door onderstaande commando's en blijft niet dezelfde. Vraag de term regelmatig op het met het *show term* commando.

apply reduction at term . voert een vrije en zo ver mogelijke reductie uit op de term, op basis van het ingeladen systeem van gelijkheden.

apply reduction at (i) of term . voert een vrije en zo ver mogelijke reductie uit op subterm *i* van de term.

apply .e1 at term . voert een bepaalde herschrijfgregel van het huidige object, aangeduid met een label (hier *e1*) uit op de term.

apply OBJECTNAAM.e1 at term . idem, maar van een willekeurig object.

apply .e1 at (i) of term . voert een regel van het huidige object uit op subterm *i* van de term.

apply OBJECTNAAM.e1 at (i) of term . idem, maar van een willekeurig object.

2 Opgave 1

2.1

Beschouw volgend *eensoortig systeem van gelijkheden* E met vier *constanten* a, b, c, d :

$$\begin{aligned}b &= a \\c &= a \\d &= c\end{aligned}$$

Voer dit systeem in OBJ3 in en *bewijs* gebruikmakend van het *reduce* commando dat

$$E \models b = d$$

Gebruik *traces* om de *herschrijvingen* na te gaan. Bijvraag: waarom staat er “empty substitution” bij de applicatie van elke regel?

2.2

Doe hetzelfde voor het *eensoortig systeem van gelijkheden* E' met vier *constanten* a, b, c, d :

$$\begin{aligned}a &= b \\a &= c \\c &= d\end{aligned}$$

met

$$E' \models b = d$$

Lukt dit? Wat gebeurt er?

3 Opgave 2

Beschouw volgend *eensoortig systeem van gelijkheden* F met *constanten* a, b, c, d, e, f, x, y :

$$\begin{aligned}c &= d \\b &= f \\y &= f \\c &= a \\f &= c \\d &= e \\b &= a \\a &= e \\x &= b\end{aligned}$$

Voer dit systeem in OBJ3 in en *bewijs* gebruikmakend van het *reduce* commando dat

$$F \models x = y$$

Gebruik *traces* om ook het *aantal herschrijvingen* na te gaan. Probeer het minimale systeem F' te vinden zodanig dat geldt dat

$$F' \models x = y \quad \wedge \quad F' \subseteq F \quad \wedge \quad \nexists F'' . F'' \subset F', F'' \models x = y$$

Ga terug naar het oorspronkelijke systeem F . We noteren het *aantal herschrijvingen* als n . Probeer het maximale systeem F' te vinden zodanig dat geldt dat

$$F^8 \models x = y \quad \wedge \quad F^8 \subseteq F \quad \wedge \quad \nexists F' . F^8 \subset F', F' \models x = y \quad \wedge \quad n = 8$$

Kan je ook het maximale systeem $F^{10} \subseteq F$ construeren, waarbij $x = y$ in 10 stappen wordt gereduceerd? Probeer ook F^6 en F^4 te construeren, met respectievelijk 6 en 4 herschrijvingen. Bijvraag: Kan je nagaan (en eventueel voorspellen) naar welke elementen uit de *constantenverzameling* van F het systeem steeds zal evolueren? Kan dit tot een efficiëntere implementatie van de herschrijfroutines van OBJ3 leiden?

4 Opgave 3

4.1

Beschouw de *signatuur* $\Sigma \equiv \{\Sigma_{nn,n}, \Sigma_{n,n}, \Sigma_{\phi,n}\}$ met

$$\begin{aligned}\Sigma_{nn,n} &\equiv \{- \wedge -, - \vee -\} \\ \Sigma_{n,n} &\equiv \{\neg -\} \\ \Sigma_{\phi,n} &\equiv \{a, b, c, d\}\end{aligned}$$

Beschouw volgend *herschrijfsysteem* G over Σ :

$$\begin{aligned}(eq1) \quad & \neg(\neg x) \rightarrow x \\ (eq2) \quad & \neg(x \vee y) \rightarrow \neg x \wedge \neg y \\ (eq3) \quad & \neg(x \wedge y) \rightarrow \neg x \vee \neg y \\ (eq4) \quad & x \wedge (y \vee z) \rightarrow (x \wedge y) \vee (x \wedge z) \\ (eq5) \quad & (x \vee y) \wedge z \rightarrow (x \wedge z) \vee (y \wedge z)\end{aligned}$$

Ga volgende *herschrijvingen* na in G , gebruikmakend van OBJ3 :

$$\begin{aligned}(a \vee b) \wedge (c \vee d) &\rightarrow_{eq4} \dots \rightarrow_{eq5} \dots \\ (a \vee b) \wedge (c \vee d) &\rightarrow_{eq5} \dots \rightarrow_{eq4} \dots\end{aligned}$$

Gebruik variabelen in de specificatie zoals in de opgave. Gebruik *traces*, *termen* en *subtermen* om de herschrijvingen stap voor stap uit te voeren. Vul de puntjes in en verklaar.

4.2

Beschouw de *signatuur* $\Sigma \equiv \{\Sigma_{nn,n}, \Sigma_{n,n}, \Sigma_{\phi,n}\}$ met

$$\begin{aligned}\Sigma_{nn,n} &\equiv \{- + -\} \\ \Sigma_{n,n} &\equiv \{-'\} \\ \Sigma_{\phi,n} &\equiv \{o\}\end{aligned}$$

Beschouw volgend *eensoortig systeem van gelijkheden* G over Σ :

$$\begin{aligned}x + 0 &= x \\ x + y' &= (x + y)'\end{aligned}$$

Ga na of volgende *herschrijvingen* mogelijk zijn in G :

$$\begin{aligned} o'' + o' &\rightarrow_G o' + o'' \\ o''' + o'' &\rightarrow_G o'' + o''' \\ o' + o''' &\rightarrow_G o''' + o' \end{aligned}$$

Gebruik variabelen in de specificatie zoals in de opgave. Gebruik *traces*, *termen* en *subtermen* om de herschrijvingen stap voor stap uit te voeren. Selecteer bij elke stap ook manueel de gebruikte vergelijking $\in G$.

5 Opgave 4

In de theoriecursus, 6.6 *Kompletering van een TRS, voorbeeld 6.6.3* wordt het herschrijfsysteem **GA** geïntroduceerd:

$$\begin{aligned} (eq1) \quad & o + x \rightarrow x \\ (eq2) \quad & (-x) + x \rightarrow o \\ (eq3) \quad & (x + y) + z \rightarrow x + (y + z) \end{aligned}$$

Volg *voorbeeld 6.6.3* verder in de completering van **GA**, maak een *trace* met OBJ3. Gebruik hiervoor alle OBJ3 commando's die hoger werden geïntroduceerd. Maak gebruik van constanten of variabelen in de herschrijvingen naar eigen smaak.

6 Opgave 5

Pas het Knuth-Bendix algoritme stapsgewijs en manueel toe op het volgende *TRS* :

$$\begin{aligned} (eq1) \quad & ab \rightarrow b \\ (eq2) \quad & bc \rightarrow c \\ (eq3) \quad & cb \rightarrow a \end{aligned}$$

Stel hiervoor telkens de verzameling kritische paren op. Maak hiervan een *trace* in OBJ3 en becommentarieer de logfile.