Computability Closure: Ten Years Later

Frédéric Blanqui

INRIA & LORIA

21 juin 2007

Outline

Computability

computability has been introduced for proving termination of β

a *computability predicate* is a set of terms P such that:

- $P \subseteq \mathrm{SN}(\rightarrow)$ where $\rightarrow = \rightarrow_\beta$
- $\rightarrow(P) \subseteq P$
- if t is *neutral* ($x\vec{v}$ or $(\lambda x t)u\vec{v}$) and $\rightarrow(t) \subseteq P$ then $t \in P$

if P and Q are computability predicates then the set
$P \rightarrow Q = \{t \mid \forall u \in P, tu \in Q\}$ is a computability predicate

Extension to function symbols defined by rewrite rules

take $\rightarrow = \rightarrow_\beta \cup \rightarrow_R$ where R is a set of rules

- every term is computable if every symbol f is computable

Extension to function symbols defined by rewrite rules

take $\rightarrow = \rightarrow_\beta \cup \rightarrow_R$ where R is a set of rules

- every term is computable if every symbol f is computable
- a symbol f is computable if $f\vec{t}$ is computable whenever $\vec{t}$ are computable

Extension to function symbols defined by rewrite rules

take $\rightarrow = \rightarrow_\beta \cup \rightarrow_R$ where R is a set of rules

- every term is computable if every symbol f is computable
- a symbol f is computable if $f\vec{t}$ is computable whenever $\vec{t}$ are computable
- a symbol f is computable if every head-reduct of $f\vec{t}$ is computable whenever $\vec{t}$ are computable

Extension to function symbols defined by rewrite rules

take $\rightarrow = \rightarrow_\beta \cup \rightarrow_R$ where R is a set of rules

- every term is computable if every symbol f is computable
- a symbol f is computable if $f\vec{t}$ is computable whenever $\vec{t}$ are computable
- a symbol f is computable if every head-reduct of $f\vec{t}$ is computable whenever $\vec{t}$ are computable
- a symbol f is computable if, for every rule $f\vec{l} \rightarrow r$ and substitution σ, $r\sigma$ is computable whenever $\vec{l}\sigma$ are computable

Outline

Computability closure

introduced by Jean-Pierre and Mitsu in a 1997 draft

definition: a function CC mapping every symbol f and terms $\vec{l}$ to a set of terms $\mathrm{CC}^f(\vec{l})$ is a *computability closure* if, for all $r \in \mathrm{CC}^f(\vec{l})$ and substitution θ, $r\theta$ is computable whenever $\vec{l}\theta$ are computable and θ is computable on $\mathcal{X} \setminus \mathrm{FV}(\vec{l})$

theorem: if CC is a computability closure and, for all rule $f\vec{l} \to r$, $r \in \mathrm{CC}^f(\vec{l})$, then every symbol is computable and $\to_\beta \cup \to_R$ is SN

Example of computability closure

assuming a precedence $>_{\mathcal{F}}$, statuses (lex or mul) and an ordering $>$ for comparing function arguments:

(arg) $l_i \in \mathrm{CC}^f_>(\vec{l})$

(decomp-symb) $$\frac{g\vec{u} \in \mathrm{CC}^f_>(\vec{l})}{u_i \in \mathrm{CC}^f_>(\vec{l})}$$

(prec) $$\frac{f >_{\mathcal{F}} g}{g \in \mathrm{CC}^f_>(\vec{l})}$$

(app) $$\frac{u \in \mathrm{CC}^f_>(\vec{l}) \quad v \in \mathrm{CC}^f_>(\vec{l})}{uv \in \mathrm{CC}^f_>(\vec{l})}$$

(call) $$\frac{f \simeq_{\mathcal{F}} g \quad \vec{u} \in \mathrm{CC}^f_>(\vec{l}) \quad \vec{l} >_{\mathrm{stat}_f} \vec{u}}{g\vec{u} \in \mathrm{CC}^f_>(\vec{l})}$$

(var) $$\frac{x \notin \mathrm{FV}(\vec{l})}{x \in \mathrm{CC}^f_>(\vec{l})}$$

(lam) $$\frac{u \in \mathrm{CC}^f_>(\vec{l}) \quad x \notin \mathrm{FV}(\vec{l})}{\lambda x u \in \mathrm{CC}^f_>(\vec{l})}$$

Outline

Rewriting with matching modulo some equational theory E

assuming that E is a symmetric set of rules $l \to r$ such that $l = f\vec{l}$ and $\mathrm{FV}(r) \subseteq \mathrm{FV}(l)$ (excludes $x \cdot 0 = 0$ and $x + 0 = x$)

<u>definition:</u> $t \to_{R,E} u$ if there are $p \in \mathrm{Pos}(t)$, $l \to r \in R$ and σ such that $t|_p \to_E^* l\sigma$ and $u = t[r\sigma]_p$

<u>theorem:</u> $\to_\beta \cup \to_{R,E}$ is terminating if:

- for all rule $f\vec{l} \to g\vec{r} \in E$, $\vec{r} \in \mathrm{CC}^f_>(\vec{l})$
- for all rule $f\vec{l} \to r \in R$, $r \in \mathrm{CC}^f_>(\vec{l})$

<u>example:</u> associativity and commutativity

β-normalized rewriting with matching modulo $\beta\eta$

assuming that rule left-hand sides are *patterns à la Miller*:

definition: $t \rightarrow_{R,\beta\eta} u$ if there are $p \in \mathrm{Pos}(t)$, $l \rightarrow r \in R$ and σ β-normal such that $t|_p$ is β-normal, $t|_p =_{\beta\eta} l\sigma$ and $u = t[r\sigma]_p$

theorem: a symbol f is computable if, for every rule $f\vec{l} \rightarrow r \in R$ and substitution σ, $r\sigma$ is computable whenever $\vec{l}\sigma$ are computable

uses the following facts:
– if l is a pattern, σ and t are β-normal and $t =_{\beta\eta} l\sigma$, then $t =_\eta \leftarrow^*_{\beta_0} l\sigma$ where $(\lambda x t)x \rightarrow_{\beta_0} t$
– computability is preserved by η-equivalence and β_0-expansion

remark: implies termination of CRS and HRS additional decomposition rules for patterns à la Miller:

(decomp-lam) $$\frac{\lambda y u \in \mathrm{CC}^f_>(\vec{l}) \quad y \notin \mathrm{FV}(\vec{l})}{u \in \mathrm{CC}^f_>(\vec{l})}$$

(decomp-app-left) $$\frac{u y \in \mathrm{CC}^f_>(\vec{l}) \quad y \notin \mathrm{FV}(\vec{l}) \cup \mathrm{FV}(u)}{u \in \mathrm{CC}^f_>(\vec{l})}$$

The Higher-Order Recursive Computability Ordering

the function CR mapping $>$ to the relation
$\{(f\vec{l}, r) \mid r \in \mathrm{CC}^f_>(\vec{l}), \tau(r) = \tau(f\vec{l}), \mathrm{FV}(r) \subseteq \mathrm{FV}(\vec{l})\}$
is monotone !

let $>_{\mathrm{whorco}}$ (weak HORCO) be the least fixpoint of CR
let $>_{\mathrm{horco}}$ be the closure by context of $>_{\mathrm{whorco}}$

properties:

- $>_{\mathrm{whorco}}$ is transitive

the function CR mapping $>$ to the relation
$\{(f\vec{l}, r) \mid r \in \mathrm{CC}^f_>(\vec{l}), \tau(r) = \tau(f\vec{l}), \mathrm{FV}(r) \subseteq \mathrm{FV}(\vec{l})\}$
is monotone !

let $>_{\mathrm{whorco}}$ (weak HORCO) be the least fixpoint of CR
let $>_{\mathrm{horco}}$ be the closure by context of $>_{\mathrm{whorco}}$

properties:

- $>_{\mathrm{whorco}}$ is transitive
- $>_{\mathrm{horpo}} \subseteq >^+_{\mathrm{horco}}$

The Higher-Order Recursive Computability Ordering

the function CR mapping $>$ to the relation
$\{(f\vec{l}, r) \mid r \in \mathrm{CC}^f_>(\vec{l}), \tau(r) = \tau(f\vec{l}), \mathrm{FV}(r) \subseteq \mathrm{FV}(\vec{l})\}$
is monotone !

let $>_{\mathrm{whorco}}$ (weak HORCO) be the least fixpoint of CR
let $>_{\mathrm{horco}}$ be the closure by context of $>_{\mathrm{whorco}}$

properties:

- $>_{\mathrm{whorco}}$ is transitive
- $>_{\mathrm{horpo}} \subseteq >^+_{\mathrm{horco}}$
- on first-order terms, $>_{\mathrm{horco}} = >_{\mathrm{whorco}} = >_{\mathrm{rpo}}$!

Inductive definition of HORCO

(rule) $$\frac{t > u \quad \mathrm{FV}(u) \subseteq \mathrm{FV}(t) \quad \tau(u) = \tau(t)}{t >_{\mathrm{whorco}} u}$$

(arg) $f\vec{l} > l_i$ (decomp-symb) $$\frac{f\vec{l} > g\vec{u}}{f\vec{l} > u_i}$$

(prec) $$\frac{f >_{\mathcal{F}} g}{f\vec{l} > g}$$ (app) $$\frac{f\vec{l} > u \quad f\vec{l} > v}{f\vec{l} > uv}$$

(call) $$\frac{f \simeq_{\mathcal{F}} g \quad f\vec{l} > \vec{u} \quad \vec{l}\ (>_{\mathrm{horco}})_{\mathrm{stat}_f}\ \vec{u}}{f\vec{l} > g\vec{u}}$$

(var) $$\frac{x \notin \mathrm{FV}(\vec{l})}{f\vec{l} > x}$$ (lam) $$\frac{f\vec{l} > u \quad x \notin \mathrm{FV}(\vec{l})}{f\vec{l} > \lambda xu}$$

Conclusion

- integration of HORPO and HORCO ?

Conclusion

- integration of HORPO and HORCO ?
- see "HORPO with computability closure: a reconstruction" with Jean-Pierre and Albert

  on `http://www.loria.fr/~blanqui/` ! :-)