logical verification extra lecture
2017 05 22
confluence
overview

confluence
Definition (Confluence)

every two coinital rewrite sequences can be joined

confluence yields uniqueness of normal forms and consistency
how do we prove confluence?

Definition (Diamond Property)

\[ \forall a, b, c \quad \exists d \]

Lemma

diamond property implies confluence
can we use this?

\[ \lambda \text{-calculus with } \beta \text{-reduction does not have the diamond property} \]
Definition (Weak Confluence)

\[ \forall a, b, c \exists d \text{ peak} \]
can we use weak confluence?

is $\beta$-reduction weakly confluent? (yes)

is Coq-reduction weakly confluent?

does weak confluence imply confluence? (no)
we can use weak confluence for terminating systems

Lemma (Newman’s Lemma)

termination and weak confluence $\Rightarrow$ confluence

a proof proceeds by well-founded induction
can we use Newman’s Lemma?

$\lambda$-calculus with $\beta$-reduction is not terminating, so no
typed $\lambda$-calculus with $\beta$-reduction is terminating, so yes?
however we also need confluence on pseudo-terms, so no
Example
in untyped $\lambda$-calculus:

$(\lambda x. (\lambda y. x) \cdot I) \cdot (I \cdot I) \leftrightarrow (\lambda y. I \cdot I) \cdot I$ and

$(\lambda x. (\lambda y. x) \cdot I) \cdot (I \cdot I) \leftrightarrow (\lambda x. x) \cdot I$
proving confluence for beta on pure untyped terms

• use finiteness of developments and the parallel moves lemma:
  the divergence $s \rightarrow s'$ and $s \rightarrow^* t$ is joinable

• use the method due to Tait and Martin–Löf:
  define relation $\rightsquigarrow$ with
    • $\rightsquigarrow$ has the diamond property
    • $\rightsquigarrow^* = \rightarrow^*$

the question is what to take for $\rightsquigarrow$
why does this work?

- suppose \( \rightarrow \) has the diamond property
- suppose \( \rightarrow \subseteq \rightarrow \subseteq \rightarrow^* \)
why does this work?

- suppose $\overrightarrow{\implies}$ has the diamond property
- suppose $\rightarrow \subseteq \overrightarrow{\implies} \subseteq \rightarrow^*$
why does this work?

- suppose $\supseteq$ has the diamond property
- suppose $\to \subseteq \supseteq \subseteq \to^*$
why does this work?

- Suppose $\supseteq$ has the diamond property
- Suppose $\rightarrow \subseteq \supseteq \subseteq \rightarrow^*$
why does this work?

- suppose $\vdash$ has the diamond property
- suppose $\rightarrow \subseteq \vdash \subseteq \rightarrow^*$
why does this work?

• suppose $\rightarrow$ has the diamond property
• suppose $\rightarrow \subseteq \rightarrow \subseteq \rightarrow^*$
definition simultaneous reduction

\[ x \Rightarrow x \]

\[ \lambda x. M \Rightarrow \lambda x. M' \text{ if } M \Rightarrow M' \]

\[ M N \Rightarrow M' N' \text{ if } M \Rightarrow M' \text{ and } N \Rightarrow N' \]

\[ (\lambda x. M) N \Rightarrow M'[x := N'] \text{ if } M \Rightarrow M' \text{ and } N \Rightarrow N' \]

exercise: consider (im)possible \( \Rightarrow \)-reducts of

\( (\lambda x. x)(\lambda y. I y)(I z), \)
\( (\lambda x. \lambda y. f x y)(I a)(I b), \)
\( (\lambda x. x I)(\lambda y. f y). \)
some properties

if $M \rightarrow_{\beta} M'$ then $M \Rightarrow M'$

if $M \Rightarrow M'$ then $M \rightarrow^{*} M'$

if $M \Rightarrow M'$ and $N \Rightarrow N'$ then $M[x := N] \Rightarrow M'[x := N']$

exercise: prove these properties
universal common reduct (Takahashi)

\[ x^* = x \]

\[ (\lambda x. M)^* = \lambda x. M^* \]

\[ (M N)^* = M^* N^* \text{ if } M \text{ is not an abstraction} \]

\[ ((\lambda x. M) N)^* = M^*[x := N^*] \]
if $M \Rightarrow N$ then $N \Rightarrow M^*$

so $*$ gives us indeed a universal common reduct
done!

but why?
alternative to simultaneous reduction

the following relation due to Aczel (1978) can also be used to prove confluence

\[ x \Rightarrow_A x \]

\[ \lambda x. M \Rightarrow_A \lambda x. M' \text{ if } M \Rightarrow_A M' \]

\[ MN \Rightarrow_A M'N' \text{ if } M \Rightarrow_A M' \text{ and } N \Rightarrow_A N' \]

\[ MN \Rightarrow_A M'[x := N'] \text{ if } M \Rightarrow_A \lambda x. M' \text{ and } N \Rightarrow_A N' \]

exercise: reconsider

\[ (\lambda x.x)(\lambda y. I y)(I z), \]
\[ (\lambda x. \lambda y. f x y)(I a)(I b), \]
\[ (\lambda x. x I)(\lambda y. f y). \]
critical pair for beta and eta

Example

\[ Y \cdot Z \xrightarrow{\eta} (\lambda x. Y \cdot x) \cdot Z \rightarrow_\beta Y \cdot Z \]

\[ \lambda u. Z(u) \xrightarrow{\beta} \lambda x. (\lambda u. Z(u)) \cdot x \rightarrow_\eta \lambda x. Z(x) \]
robust approach?

can we add $\eta$?

can we move to system $F$?

can we move to system $T$?
type checking for simply typed lambda-calculus

we consider $\Gamma \vdash M : A$?

we consider $M$ with all type information erased

we compute the principle type $C$ of $M$ if possible

we compute whether there exists $\sigma$ such that $C^\sigma = A$

if yes, then type checking succeeds, otherwise it fails
type checking for more general typed lambda-calculi

we prove type checking and type synthesis simultaneously

in the proof we use for example confluence