Rewriting

Part 3.1 Equational Problems. Deciding $\approx_E$

Temur Kutsia

RISC, JKU Linz

Validity and Satisfiability

Notation: $s \approx_E t$ iff $s \approx t$ belongs to the equational theory generated by E.

Validity and Satisfiability

Notation: $s \approx_E t$ iff $s \approx t$ belongs to the equational theory generated by E.

Validity problem:

Given: A set of identities E and terms s and t.

Decide: $s \approx_E t$.

Validity and Satisfiability

Notation: $s \approx_E t$ iff $s \approx t$ belongs to the equational theory generated by E.

Validity problem:

Given: A set of identities E and terms s and t.
Decide: $s \approx_E t$.

Satisfiability problem:

Given: A set of identities E and terms s and t.
Find: A substitution σ such that $\sigma(s) \approx_E \sigma(t)$.

Equational Problems

The following methods solve special cases:

- Term rewriting decides $\approx_E$ if $\rightarrow_E$ is convergent.
- Congruence closure decided $\approx_E$ when E is variable-free.
- Syntactic unification computes σ such that $\sigma(s) = \sigma(t)$.

Equations Problems

Relating validity and satisfiability problems.

- Validity: $s \approx t$ is valid in E iff

  $$\forall \overline{x}.\ s \approx t$$

  holds in all models of E.
- Satisfiability: $s \approx t$ is satisfiable in E iff

  $$\exists \overline{x}.\ s \approx t$$

  holds in all nonempty models of E.

Deciding $\approx_E$

- By Birkhoffs theorem, $s \approx_E t$ iff $s \overset{*}{\leftrightarrow}_E r$.
- Hence, deciding $\approx_E$ is equivalent to deciding $\overset{*}{\leftrightarrow}_E$.

Deciding $\approx_E$

- By Birkhoffs theorem, $s \approx_E t$ iff $s \overset{*}{\longleftrightarrow}_E r$.
- Hence, deciding $\approx_E$ is equivalent to deciding $\overset{*}{\longleftrightarrow}_E$.
- Word problem:

  Given: A set of identities E and terms s and t.

  Decide: $s \overset{*}{\longleftrightarrow}_E t$.

Deciding $\approx_E$: Finite E, Convergent $\rightarrow_E$

Recall from abstract reduction systems:

If $\rightarrow$ is confluent and terminating, then

- every element x has a unique normal form $x\downarrow$,
- $x \overset{*}{\leftrightarrow} y$ iff $x\downarrow = y\downarrow$.

Deciding $\approx_E$: Finite E, Convergent $\to_E$

Recall from abstract reduction systems:

If $\to$ is confluent and terminating, then

- every element x has a unique normal form $x\downarrow$,
- $x \overset{*}{\leftrightarrow} y$ iff $x\downarrow = y\downarrow$.

- Hence, if $\to_E$ is convergent, we can decide $x \overset{*}{\leftrightarrow} y$.
- Provided that we are able to compute normal forms.

Deciding $\approx_E$: Finite E, Convergent $\to_E$

Recall from abstract reduction systems:

If $\to$ is confluent and terminating, then

- every element x has a unique normal form $x\downarrow$,
- $x \stackrel{*}{\leftrightarrow} y$ iff $x\downarrow = y\downarrow$.

- Hence, if $\to_E$ is convergent, we can decide $x \stackrel{*}{\leftrightarrow} y$.
- Provided that we are able to compute normal forms.
- This is possible if we can effectively
  - decide whether a term is in normal form wrt $\to_E$, and
  - compute some s' such that $s \to_E s'$ if s is not in normal form.

Deciding $\approx_E$: Finite E, Convergent $\rightarrow_E$

How to decide whether a term is in normal form wrt $\rightarrow_E$?

Deciding $\approx_E$: Finite E, Convergent $\to_E$

How to decide whether a term is in normal form wrt $\to_E$?

- Check whether it can be rewritten by $\to_E$.

Deciding $\approx_E$: Finite E, Convergent $\to_E$

How to decide whether a term is in normal form wrt $\to_E$?

- Check whether it can be rewritten by $\to_E$.
- That is, check whether any of its subterms is an instance of the lhs of a rule in $\to_E$.

Deciding $\approx_E$: Finite E, Convergent $\to_E$

How to decide whether a term is in normal form wrt $\to_E$?

- Check whether it can be rewritten by $\to_E$.
- That is, check whether any of its subterms is an instance of the lhs of a rule in $\to_E$.
- Matching problem:

  Given: Two terms s and t.
  Find: A substitution σ such that $\sigma(s) = t$.

Deciding $\approx_E$: Finite E, Convergent $\rightarrow_E$

How to decide whether a term is in normal form wrt $\rightarrow_E$?

- Check whether it can be rewritten by $\rightarrow_E$.
- That is, check whether any of its subterms is an instance of the lhs of a rule in $\rightarrow_E$.
- Matching problem:

  Given: Two terms s and t.
  Find: A substitution σ such that $\sigma(s) = t$.
- Matching is decidable. (Details later, with unification.)

Deciding $\approx_E$: Finite E, Convergent $\to_E$

Theorem 3.1
If E is finite and $\to_E$ is convergent, then $\approx_E$ is decidable.

Proof.

1. Decide whether a term s is in normal form wrt $\to_E$:
   Check all $l \approx r \in E$ and all positions $p \in \mathcal{P}os(s)$
   if there is σ such that $s|_p = \sigma(l)$.
2. Compute some s' such that $s \to_E s'$ if s is not in normal form:
   Reduce s to $s[\sigma(r)]_p$ if the test above is positive.

Iterate the process to compute a normal form.
The iteration stops because $\to_E$ is terminating.
The obtained normal form is unique because $\to_E$ is confluent.
To decide $s \approx_E t$, compute $s\downarrow_E$ and $t\downarrow_E$ and compare. □

Deciding $\approx_E$: Finite E, Convergent $\rightarrow_E$

- Convergence of $\rightarrow_E$ is important for decidability of $\approx_E$.
- There exist finite sets E for which $\approx_E$ is not decidable.
- Example: Combinatory logic.

Deciding $\approx_E$: Finite E, Convergent $\rightarrow_E$

Definition 3.1 (Term Rewriting System)

- Rewrite rule: An identity $l \approx r$ such that
  - l is not a variable,
  - $\mathcal{V}ar(l) \supseteq \mathcal{V}ar(r)$.
- Notation: $l \rightarrow r$ instead of $l \approx r$.
- A term rewriting system (TRS) is a set of rewrite rules.

Deciding $\approx_E$: Finite E, Convergent $\to_E$

Definition 3.1 (Term Rewriting System)

- Rewrite rule: An identity $l \approx r$ such that
  - l is not a variable,
  - $\mathcal{V}ar(l) \supseteq \mathcal{V}ar(r)$.
- Notation: $l \to r$ instead of $l \approx r$.
- A term rewriting system (TRS) is a set of rewrite rules.

By definition, a TRS R is a set of identities.

Hence, $\to_R$ and $\approx_R$ are well-defined.

We say that R is terminating, confluent, etc. if $\to_R$ is.

Deciding $\approx_E$: Finite E, Convergent $\to_E$

Definition 3.1 (Term Rewriting System)

- Rewrite rule: An identity $l \approx r$ such that
  - l is not a variable,
  - $\mathcal{V}ar(l) \supseteq \mathcal{V}ar(r)$.
- Notation: $l \to r$ instead of $l \approx r$.
- A term rewriting system (TRS) is a set of rewrite rules.

By definition, a TRS R is a set of identities.

Hence, $\to_R$ and $\approx_R$ are well-defined.

We say that R is terminating, confluent, etc. if $\to_R$ is.

Theorem 3.2

If R is a finite convergent TRS, then $\approx_R$ is decidable.

Deciding $\approx_E$: Finite Ground E

- An identity $l \approx r$ is a ground identity if $\mathcal{V}ar(l) = \mathcal{V}ar(r) = \emptyset$.
- Ground word problem for E: Word problem for ground terms over the signature of E.

Deciding $\approx_E$: Finite Ground E

- An identity $l \approx r$ is a ground identity if $\mathcal{V}ar(l) = \mathcal{V}ar(r) = \emptyset$.
- Ground word problem for E: Word problem for ground terms over the signature of E.
- G: A set of ground identities.
- Congruence on terms: Equivalence relation closed under operations.
- Congruence closure of G: smallest congruence on terms which contains G.

Deciding $\approx_E$: Finite Ground E

Relating $\approx_G$ and congruence closure of G:

- By Theorem 2.1, $\overset{*}{\longleftrightarrow}_G$ is the smallest equivalence relation closed under substitutions and operations.
- G is ground, substitutions are irrelevant.
- Hence, $\overset{*}{\longleftrightarrow}_G$ is the congruence closure of G.
- By Birkhoffs Theorem, $\approx_G$ is the congruence closure of G.

Deciding $\approx_E$: Finite Ground E

Operational description of congruence closure: A functional version of the rules of equational logic.

$$
\begin{aligned}
R(E) &:= \{(t,t) \mid t \in T(\mathcal{F},\mathcal{V})\}.\\
S(E) &:= \{(s,t) \mid (t,s) \in E\}.\\
T(E) &:= \{(s,r) \mid \text{for some } t,\ (s,t) \in E \text{ and } (t,r) \in E\}.\\
C(E) &:= \{(f(s_1,\ldots,s_n), f(t_1,\ldots,t_n)) \mid \\
&\qquad\qquad f \in \mathcal{F}^n, (s_i,t_i) \in E \text{ for all } 1 \le i \le n\}.
\end{aligned}
$$

$$
Cong(E) := E \cup R(E) \cup S(E) \cup T(E) \cup C(E)
$$

Deciding $\approx_E$: Finite Ground E

Operational description of congruence closure: A functional version of the rules of equational logic.

$$
\begin{aligned}
R(E) &:= \{(t,t) \mid t \in T(\mathcal{F},\mathcal{V})\}.\\
S(E) &:= \{(s,t) \mid (t,s) \in E\}.\\
T(E) &:= \{(s,r) \mid \text{for some } t,\ (s,t) \in E \text{ and } (t,r) \in E\}.\\
C(E) &:= \{(f(s_1,\ldots,s_n), f(t_1,\ldots,t_n)) \mid \\
&\qquad\qquad f \in \mathcal{F}^n, (s_i,t_i) \in E \text{ for all } 1 \le i \le n\}.
\end{aligned}
$$

$$
Cong(E) := E \cup R(E) \cup S(E) \cup T(E) \cup C(E)
$$

- E is congruence iff E is closed under $Cong$ (i.e., $Cong(E) \subseteq E$).

Deciding $\approx_E$: Finite Ground E

Operational description of congruence closure: A functional version of the rules of equational logic.

$$
\begin{aligned}
R(E) &:= \{(t,t) \mid t \in T(\mathcal{F},\mathcal{V})\}.\\
S(E) &:= \{(s,t) \mid (t,s) \in E\}.\\
T(E) &:= \{(s,r) \mid \text{for some } t,\ (s,t) \in E \text{ and } (t,r) \in E\}.\\
C(E) &:= \{(f(s_1,\dots,s_n), f(t_1,\dots,t_n)) \mid \\
&\qquad\qquad f \in \mathcal{F}^n, (s_i,t_i) \in E \text{ for all } 1 \le i \le n\}.
\end{aligned}
$$

$$
Cong(E) := E \cup R(E) \cup S(E) \cup T(E) \cup C(E)
$$

- E is congruence iff E is closed under $Cong$ (i.e., $Cong(E) \subseteq E$).
- E is congruence iff $Cong(E) = E$.

Deciding $\approx_E$: Finite Ground E

The process of closing G under $Cong$:

$$G_0 := G.$$
$$G_{i+1} := Cong(G_i).$$

$$CC(G) := \bigcup_{i \geq 0} G_i$$

Deciding $\approx_E$: Finite Ground E

Lemma 3.1

$CC(G) = \approx_G$.

Deciding $\approx_E$: Finite Ground E

Lemma 3.1
$CC(G) = \approx_G$.

Proof.

($\subseteq$) Use monotonicity of $Cong$: If $E_1 \subseteq E_2$, then $Cong(E_1) \subseteq Cong(E_2)$.
Proof by induction on i. $G_0 = G \subseteq \approx_G$. Assume $G_i \subseteq \approx_G$ and show $G_{i+1} \subseteq \approx_G$. $G_{i+1} = Cong(G_i) \subseteq Cong(\approx_G) = \approx_G$.

□

Deciding $\approx_E$: Finite Ground E

Lemma 3.1

$CC(G) = \approx_G$.

Proof.

- ($\subseteq$) Use monotonicity of $Cong$: If $E_1 \subseteq E_2$, then $Cong(E_1) \subseteq Cong(E_2)$.
  Proof by induction on i. $G_0 = G \subseteq \approx_G$. Assume $G_i \subseteq \approx_G$ and show $G_{i+1} \subseteq \approx_G$. $G_{i+1} = Cong(G_i) \subseteq Cong(\approx_G) = \approx_G$.
- ($\supseteq$) $CC(G)$ is a congruence containing G (because $CC(G)$ is closed under $Cong$. Check!). $\approx_G$ is the least congruence containing G. Hence, $\approx_G \subseteq CC(G)$.

□

Deciding $\approx_E$: Finite Ground E

- $CC(G)$ may be infinite. If the signature consists of a, b, and a unary function symbol f:

$$CC(\{a \approx b\}) \supseteq \{(f^i(a), f^i(b)) \mid i \geq 0\}$$

- Check whether $f^2(a) \approx_G f^2(b)$ is easy: $(f^2(a), f^2(b)) \in \approx_G$.
- But how to conclude that $f^3(a) \not\approx_G f^2(b)$?
- Shall we examine all G_i's?

Deciding $\approx_E$: Finite Ground E

- $CC(G)$ may be infinite. If the signature consists of a, b, and a unary function symbol f:

$$CC(\{a \approx b\}) \supseteq \{(f^i(a), f^i(b)) \mid i \geq 0\}$$

- Check whether $f^2(a) \approx_G f^2(b)$ is easy: $(f^2(a), f^2(b)) \in \approx_G$.
- But how to conclude that $f^3(a) \not\approx_G f^2(b)$?
- Shall we examine all G_i's?
- It turns out that since G is ground, the search space is finite.
- We need to test only terms occurring in G or in the input terms.

Deciding $\approx_E$: Finite Ground E

$$
\begin{aligned}
Subterms(t) &:= \{t|_p \mid p \in \mathcal{P}os(t)\} \\
Subterms(E) &:= \bigcup_{(l,r) \in E} \Big(Subterms(l) \cup Subterms(r) \Big)
\end{aligned}
$$

Deciding $\approx_E$: Finite Ground E

$$Subterms(t) := \{t|_p \mid p \in \mathcal{P}os(t)\}$$

$$Subterms(E) := \bigcup_{(l,r)\in E} \Big(Subterms(l) \cup Subterms(r)\Big)$$

Fix a finite set of ground identities G and two terms s and t.

$$S := Subterms(G) \cup Subterms(s) \cup Subterms(t)$$

Deciding $\approx_E$: Finite Ground E

$$\begin{aligned} Subterms(t) &:= \{t|_p \mid p \in \mathcal{P}os(t)\} \\ Subterms(E) &:= \bigcup_{(l,r)\in E} \Big(Subterms(l) \cup Subterms(r)\Big) \end{aligned}$$

Fix a finite set of ground identities G and two terms s and t.

$$S := Subterms(G) \cup Subterms(s) \cup Subterms(t)$$

S is finite. It will be used to decide $s \approx_G t$.

Deciding $\approx_E$: Finite Ground E

Define the sequence:

$$H_0 := G$$
$$H_{i+1} := Cong(H_i) \cap (S \times S)$$

Deciding $\approx_E$: Finite Ground E

Define the sequence:

$$H_0 := G$$
$$H_{i+1} := Cong(H_i) \cap (S \times S)$$

Lemma 3.2

There is some m such that $H_{m+1} = H_m$.

Proof.

By definition, $H_i \subseteq S \times S$. Moreover, $H_i \subseteq Cong(H_i)$. Hence, $H_i \subseteq H_{i+1}$. Therefore, $H_0 \subseteq H_1 \subseteq H_2 \subseteq \cdots \subseteq S \times S$ and S is finite. □

Deciding $\approx_E$: Finite Ground E

Define the sequence:

$$H_0 := G$$
$$H_{i+1} := Cong(H_i) \cap (S \times S)$$

Lemma 3.2

There is some m such that $H_{m+1} = H_m$.

Proof.

By definition, $H_i \subseteq S \times S$. Moreover, $H_i \subseteq Cong(H_i)$. Hence, $H_i \subseteq H_{i+1}$. Therefore, $H_0 \subseteq H_1 \subseteq H_2 \subseteq \cdots \subseteq S \times S$ and S is finite. □

The limit H_m is denoted by $CC_S(G)$.

$CC_S(G)$ Is Not a Congruence

- $CC_S(G)$ is not a congruence, in general.
- It is symmetric and transitive, not reflexive.
- It is reflexive only for terms from $S \times S$.

Example 1

Assume $G = \{a \approx b\}$, $s = f(a)$, $t = b$. Then $S = \{a, b, f(a)\}$.
We have:

$$
\begin{aligned}
H_0 &= G \\
H_1 &= G \cup \{a \approx a,\ b \approx b,\ f(a) \approx f(a),\ b \approx a\} \\
H_2 &= H_1 = CC_S(G)
\end{aligned}
$$

$CC_S(G)$ Is Not a Congruence

- $CC_S(G)$ is not a congruence, in general.
- It is symmetric and transitive, not reflexive.
- It is reflexive only for terms from $S \times S$.

Example 1

Assume $G = \{a \approx b\}$, $s = f(a)$, $t = b$. Then $S = \{a, b, f(a)\}$.
We have:

$$\begin{aligned}
H_0 &= G \\
H_1 &= G \cup \{a \approx a,\ b \approx b,\ f(a) \approx f(a),\ b \approx a\} \\
H_2 &= H_1 = CC_S(G)
\end{aligned}$$

Nevertheless, $CC_S(G)$ is what we need. See the next slide.

Deciding $\approx_E$: Finite Ground E

Theorem 3.3
$CC_S(G) = \approx_G \cap (S \times S)$.

Proof.

Deciding $\approx_E$: Finite Ground E

Theorem 3.3
$CC_S(G) = \approx_G \cap (S \times S)$.

Proof.

($\subseteq$) By definition, $H_i \subseteq G_i \cap (S \times S)$. Therefore, $CC_S(G) \subseteq CC(G) \cap (S \times S)$.

Deciding $\approx_E$: Finite Ground E

Theorem 3.3
$CC_S(G) = \approx_G \cap (S \times S)$.

Proof.

($\subseteq$) By definition, $H_i \subseteq G_i \cap (S \times S)$. Therefore, $CC_S(G) \subseteq CC(G) \cap (S \times S)$.

($\supseteq$) Let $u, v \in S$ and $u \leftrightarrow_G^n v$. Prove $(u, v) \in H_m$ (the limit of H_i) by well-founded induction on the lexicographically ordered pair $(n, |u|)$:

- $n = 0$. Then $u = v$. Hence, $(u, v) \in H_1 \subseteq H_m$.
- $u \leftrightarrow_G^{n+1} v$. Two cases:
  1. There is a rewrite step at the root.
  2. There is no rewrite step at the root.

Deciding $\approx_E$: Finite Ground E

Theorem 3.3

$CC_S(G) = \approx_G \cap (S \times S)$.

Proof (Cont.)

1. There is a rewrite step at the root.

$$u \leftrightarrow_G^{n_1} l \leftrightarrow_G r \leftrightarrow_G^{n_2} v$$

for some $l \approx r \in G \cup G^{-1}$. ($G$ is ground: No substitutions). $n_1, n_2 < n$. By induction hypothesis,

$$(u, l) \in H_m \text{ and } (r, v) \in H_m.$$

If $(l, r) \in G$, then $(l, r) \in H_0 \subseteq H_m$. If $(l, r) \in G^{-1}$, then $(l, r) \in H_1 \subseteq H_m$. By transitivity of H_m, $(u, v) \in H_m$.

□

Deciding $\approx_E$: Finite Ground E

Theorem 3.3
$CC_S(G) = \approx_G \cap (S \times S)$.

Proof (Cont.)

2. There is no rewrite step at the root.

$$u = f(u_1, \ldots, u_k),\ v = f(v_1, \ldots, v_k)$$

and $u_i \leftrightarrow_G^{n_i} v_i$ for all $1 \leq i \leq k$.
Since $n_i \leq n + 1$, $|u_i| < |u|$, and $u_i, v_i \in S$, by the induction hypothesis, $(u_i, v_i) \in H_m$ for all $1 \leq i \leq k$.
By congruence, $(u, v) \in H_{m+1} = H_m$.

□

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s = f(a)$, $t = a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s = f(a)$, $t = a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Constructing $CC_S(G)$:

$$S \times S :$$

$$\begin{array}{rrrr}
a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\
f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\
f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\
f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a)
\end{array}$$

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s := f(a)$, $t := a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Constructing $CC_S(G)$:

H_0 :

$$\begin{array}{rrrr} a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\ f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\ f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\ f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a) \end{array}$$

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s := f(a)$, $t := a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Constructing $CC_S(G)$:

$$H_1:$$

$$\begin{array}{rrrr} a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\ f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\ f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\ f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a) \end{array}$$

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s := f(a)$, $t := a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Constructing $CC_S(G)$:

H_2 :

$$\begin{array}{rrrr} a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\ f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\ f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\ f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a) \end{array}$$

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s := f(a)$, $t := a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Constructing $CC_S(G)$:

$$H_3 :$$

$$\begin{array}{rrrr}
a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\
f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\
f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\
f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a)
\end{array}$$

Deciding $\approx_E$: Finite Ground E

Example 3.1

Let $\mathcal{F} = \{a, f\}$, $G := \{f^2(a) \approx a,\ f^3(a) \approx a\}$, $s := f(a)$, $t := a$.

Then $S := \{a, f(a), f^2(a), f^3(a)\}$.

Constructing $CC_S(G)$:

H_3 :

$a \approx a$	$a \approx f(a)$	$a \approx f^2(a)$	$a \approx f^3(a)$
$f(a) \approx a$	$f(a) \approx f(a)$	$f(a) \approx f^2(a)$	$f(a) \approx f^3(a)$
$f^2(a) \approx a$	$f^2(a) \approx f(a)$	$f^2(a) \approx f^2(a)$	$f^2(a) \approx f^3(a)$
$f^3(a) \approx a$	$f^3(a) \approx f(a)$	$f^3(a) \approx f^2(a)$	$f^3(a) \approx f^3(a)$

Hence, $(f(a), a) \in CC_S(G)$, showing $f(a) \approx_G a$.

Deciding $\approx_E$: Finite Ground E

Example 3.1

$s := f(a)$, $t := a$.

$S := \{a, f(a), f^2(a), f^3(a)\}$.

$CC_S(G)$:

$a \approx a$	$a \approx f(a)$	$a \approx f^2(a)$	$a \approx f^3(a)$
$f(a) \approx a$	$f(a) \approx f(a)$	$f(a) \approx f^2(a)$	$f(a) \approx f^3(a)$
$f^2(a) \approx a$	$f^2(a) \approx f(a)$	$f^2(a) \approx f^2(a)$	$f^2(a) \approx f^3(a)$
$f^3(a) \approx a$	$f^3(a) \approx f(a)$	$f^3(a) \approx f^2(a)$	$f^3(a) \approx f^3(a)$

Deciding $\approx_E$: Finite Ground E

Example 3.1

$s := f(a)$, $t := a$.

$S := \{a, f(a), f^2(a), f^3(a)\}$.

$CC_S(G)$:

$$\begin{array}{rrrr}
a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\
f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\
f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\
f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a)
\end{array}$$

Hence, $(f(a), a) \in CC_S(G)$, showing $f(a) \approx_G a$.

Deciding $\approx_E$: Finite Ground E

Example 3.1

$s := f(a)$, $t := a$.

$S := \{a, f(a), f^2(a), f^3(a)\}$.

$CC_S(G)$:

$$\begin{array}{rrrr}
a \approx a & a \approx f(a) & a \approx f^2(a) & a \approx f^3(a) \\
f(a) \approx a & f(a) \approx f(a) & f(a) \approx f^2(a) & f(a) \approx f^3(a) \\
f^2(a) \approx a & f^2(a) \approx f(a) & f^2(a) \approx f^2(a) & f^2(a) \approx f^3(a) \\
f^3(a) \approx a & f^3(a) \approx f(a) & f^3(a) \approx f^2(a) & f^3(a) \approx f^3(a)
\end{array}$$

Hence, $(f(a), a) \in CC_S(G)$, showing $f(a) \approx_G a$.
Note that $CC_S(G) = S \times S$. In general the iteration may stop before $S \times S$ is reached.