Term unification

Assume a Signature Σ of constructor symbols with $\alpha : \Sigma \to \mathbb{N}$ (arity function)

Eg: $\Sigma = \{\to, num, bool\}$ $\quad \alpha(\to) = 2$, $\alpha(num) = \alpha(bool) = 0$

Assume a denumerable set of variables X disjoint from Σ.

The set of <u>terms</u> $\boxed{T(\Sigma, X)}$ is the least set closed under the following rules:

① $$\frac{x \in X}{x \in T(\Sigma, X)}$$

② $$\frac{\tau_1 \in T(\Sigma, X) \;, \dots \; \tau_n(\Sigma, X) \;, f \in \Sigma \;, \alpha(f) = n}{f(\tau_1, \dots \tau_n) \in T(\Sigma, X)}$$

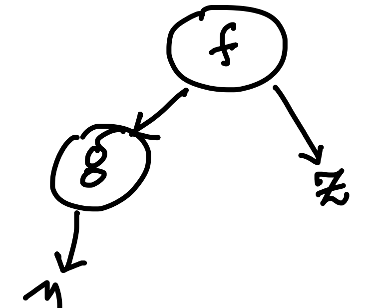
a term $f()$ is written f.

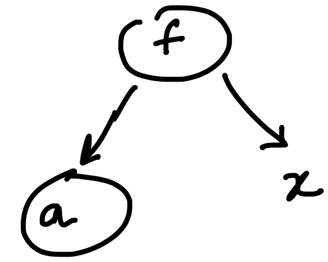
Examples of terms for $\Sigma = \{f, g, a, b\}$ $\quad \alpha(f) = 2$, $\alpha(g) = 1$, $\alpha(b) = \alpha(a) = 0$

and $X = \{x, y, z, \dots\}$

$x, y, z, a, b, g(x), f(a, x), f(g(y), z)$, etc.

Representation of Terms as Trees. (Term Trees).





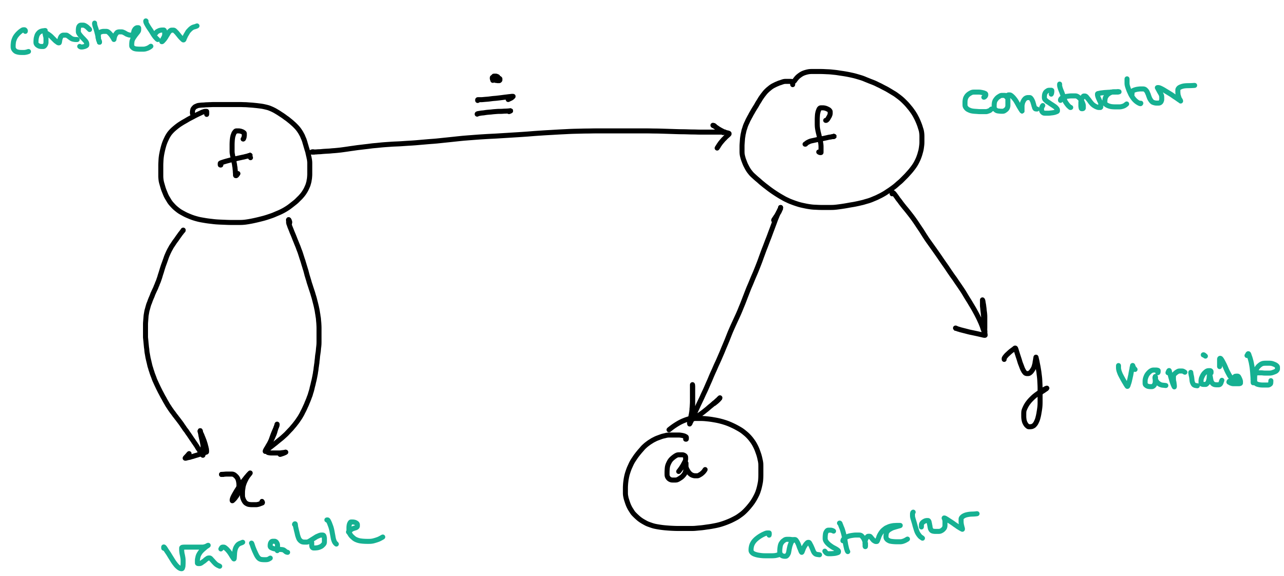
constructor symbols are in circles, variables are not, in order to distinguish variables from nullary constructors (constants)

Term Equation

$$\boxed{T \doteq T'}$$

Representation : 1. equational edge between roots of terms.
2. variable nodes are shared.

Eg: $f(x, x) \doteq f(a, y)$



Substitutions

$\sigma : X \xrightarrow{fin} T(\Sigma, X)$

apply(σ, T) also written as $\hat{\sigma}(T)$ or just $\sigma(T)$

apply$(\sigma, x) = \sigma(x)$ if $x \in$ dod(σ)
apply$(\sigma, x) = x$ " $x \notin$ dod(σ)
apply$(\sigma, f(T_1 \ldots T_n)) = f($apply$(\sigma, T_1), \ldots$ apply$(\sigma, T_n))$

Composition of Substitutions :

Let σ_1 and σ_2 be two substitution
The substitution $\sigma = \sigma_1 \cdot \sigma_2$ is defined as follows:

dod$(\sigma) =$ dod$(\sigma_1) \cup$ dod(σ_2)

$\sigma(x) =$ apply$(\sigma_1,$ apply$(\sigma_2, x))$ if $x \in$ dod(σ)
$=$ undefined o.w.

RESTRICTION :

Let $D \subseteq$ dod(σ).

Then $\sigma|_D$ is the restriction of σ to D.

dod$(\sigma|D) = D$

if $x \in D$, $\sigma|_D(x) =df= \sigma(x)$

SUBSTITUTION ORDERING

Defn $\sigma_1 \sqsubseteq \sigma_2$ iff $\exists$ a substitution α s.t.

$$\sigma_2 = \alpha \cdot \sigma_1 \,\big|\, dod(\sigma_2)$$

We say σ_1 is <u>less specific</u> (or more general) than σ_2.

<u>Examples:</u>

Let $\sigma_1 = \{x \mapsto u\}$

$\sigma_2 = \{x \mapsto \underline{bool}\}$

constructor

Then, $\sigma_1 \sqsubseteq \sigma_2$

σ_1 constrains x to be another variable u

while σ_2 " " " a constructor

Let $\alpha = \{u \mapsto \underline{bool}\}$

$dod(\sigma_2) = \{x\}$

$\sigma_2(x) = bool$

and $(\alpha \cdot \sigma_1)(x) = \hat{\alpha}(\hat{\sigma_1}(x))$

$= \hat{\alpha}(u)$

$= bool$

But $\sigma_2 \not\sqsubseteq \sigma_1$. For, suppose it is then we need to show for each $x \in dod(\sigma_1)$

$$\sigma_1(x) = \hat{\beta}(\hat{\sigma_2}(x)).$$

Now $dod(\sigma_1) = x$.

So $\sigma_1(x) = u$

$\hat{\beta}(\hat{\sigma_2}(x)) = \hat{\beta}(\underline{bool}) = \underline{bool}$

No substitution β when applied to bool can make it equal to u.

Is $\sqsubseteq$ reflexive? Yes, trivially (take $\alpha = \{\}$)

Is $\sqsubseteq$ transitive? Yes

Let $\sigma_1 \sqsubseteq \sigma_2$

$\sigma_2 \sqsubseteq \sigma_3$

Now $\sigma_2 = \alpha \cdot \sigma_1 \mid dod(\sigma_2)$ for some α.

and $\sigma_3 = \beta \cdot \sigma_2 \mid dod(\sigma_3)$

let $x \in dod(\sigma_3)$:

$$\begin{aligned} \sigma_3(x) &= \hat{\beta}(\hat{\sigma_2}(x)) \\ &= \hat{\beta}(\hat{\alpha}(\hat{\sigma_1}(x))) \\ &= (\beta \cdot \alpha)(\hat{\sigma_1}(x)) \end{aligned}$$

$\therefore \sigma_1 \sqsubseteq \sigma_3$

Example: does $\sigma_1 \sqsubseteq \sigma_2 \wedge \sigma_2 \sqsubseteq \sigma_1 \Rightarrow \sigma_1 = \sigma_2$?

$\sigma_1 \quad x \mapsto y$

$\sigma_2 \quad y \mapsto x$

is $\sigma_1 \sqsubseteq \sigma_2$:

$\sigma_1(x) = y = \sigma_1(\sigma_2(x))$

$\therefore \sigma_1 \sqsubseteq \sigma_2$

$||^{ly} \quad \sigma_2 \sqsubseteq \sigma_1$

but $\sigma_1 \neq \sigma_2$!

But note that variable may be <u>renamed</u>.

A renaming substitution is one that maps variables to variables.

Thus let $\sigma_z = [x \mapsto z, y \mapsto z]$

then $\sigma_z \cdot \sigma_1 = [x \mapsto z, y \mapsto z] = \sigma_z$

& $\sigma_z \cdot \sigma_2 = \sigma_z$

$\therefore \sigma_1 = \sigma_2$ (modulo σ_z)

Another example:

Let $\sigma_1 = \{x \mapsto u\}$

$\sigma_2 = \{x \mapsto v\}$

Is $\sigma_1 = \sigma_2$?

Let $\sigma_3 = \{u \mapsto v\}$

Then $\sigma_2 = \sigma_3 \circ \sigma_1$

Since σ_3 is a renaming substitution

$\sigma_1 = \sigma_2$ (modulo σ_3)

$\therefore$ $\sqsubseteq$ is a partial order over substitutions (when equality is considered modulo renaming)

A minimal element of the ordering is called a PRINCIPAL (or most general or least specific) substitution.

System of Term Equations:

$$E = \begin{cases} T_1 \doteq T_1' \\ T_2 \doteq T_2' \\ \vdots \\ T_m \doteq T_m' \end{cases}$$

A substitution σ is a SOLUTION of E iff

$$\sigma(T_i) = \sigma(T_i') \quad \forall 1 \le i \le m$$

σ is also called a UNIFIER for E

Intuition: Terms have variables
Can we find a substitution that when applied to terms, makes them equal?

Eg. $E = \boxed{f(x, a) \doteq f(g(y), z)}$

$\sigma_0 = \{x \mapsto g(y),\ z \mapsto a\}$ is a solution to the above term equation $\because$

$$\sigma_0(f(x,a)) = f(g(y),z) = f(g(y), a)$$

$\sigma_1 = \{x \mapsto g(b),\ z \mapsto a\}$ is also a solution (unifier) of E.

And $\sigma_0 \sqsubseteq \sigma_1$

$$\because \quad \sigma_1 = \{y \mapsto b\}\, \sigma_0 \mid dod(\sigma_1)$$

PRINCIPAL (or most general) unifier for E:

1) σ is a unifier for E.
2) if σ' is a unifier for E, then $\sigma \sqsubseteq \sigma'$

PRINCIPAL UNIFIERS are unique (modulo renaming subst.)

Computing Principal unifiers

1. Represent system of Equations as a graph G.
2. Two types of Edges: Equational & Branch edges.
3. Compute Downward Equivalence closure. $\sim$ on vertices of G.
4. if the quotient graph $G/\sim$ is consistent (no cycles or clashes) then, mgu for E exists.
5. Construct the mgu from $G/\sim$.

Intuition: Unification involves equating subterms when terms are equal.

E.g. suppose we need to solve

$$\underline{f}(\underline{a}, x) = \underline{f}(y, \underline{b})$$

From this we infer

$$\underline{a} = y$$

$$\& \quad x = \underline{b}$$

which gives the soln $\sigma = \{x \mapsto b, y \mapsto a\}$

Let G be a graph representing a system of Equations E.

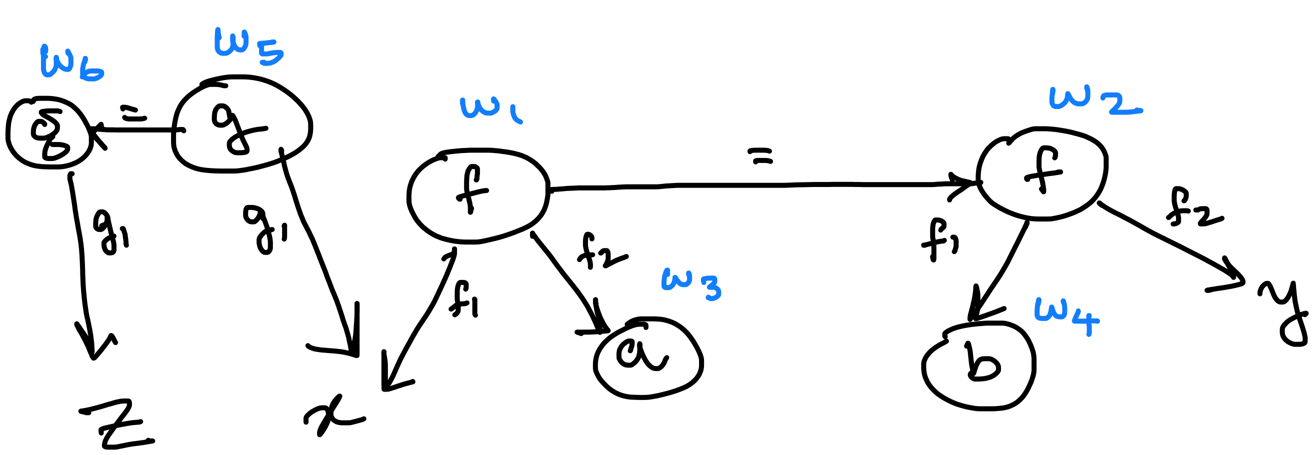
Let w denote constructor vertices,
x, y etc denote variable vertices.
let u, v denote either type of vertices:

Let $w \xrightarrow{f_i} u$ mean that u is the ith child of the constructor vertex w.

Label $(w) = \{f\}$
Label $(x) = \{\ \}$

Example:

$$\underline{f}(x, \underline{a}) \doteq \underline{f}(\underline{b}, y)$$
$$\underline{g}(x) \doteq \underline{g}(z)$$



Unification closure $G/\!\sim$.

$$\frac{u \xrightarrow{\doteq} v \in G}{G \vdash u \sim v}\ \text{EQ}$$

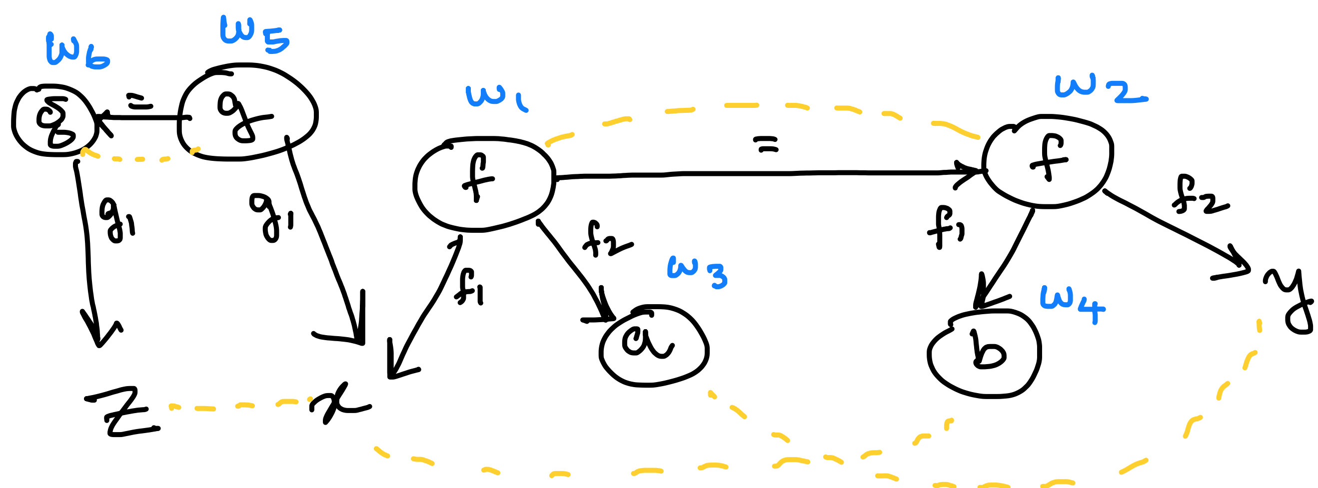
$$\frac{u \in G}{G \vdash u \sim u}\ \text{REFL}$$

$$\frac{G \vdash u \sim v}{G \vdash v \sim u}\ \text{SYM}$$

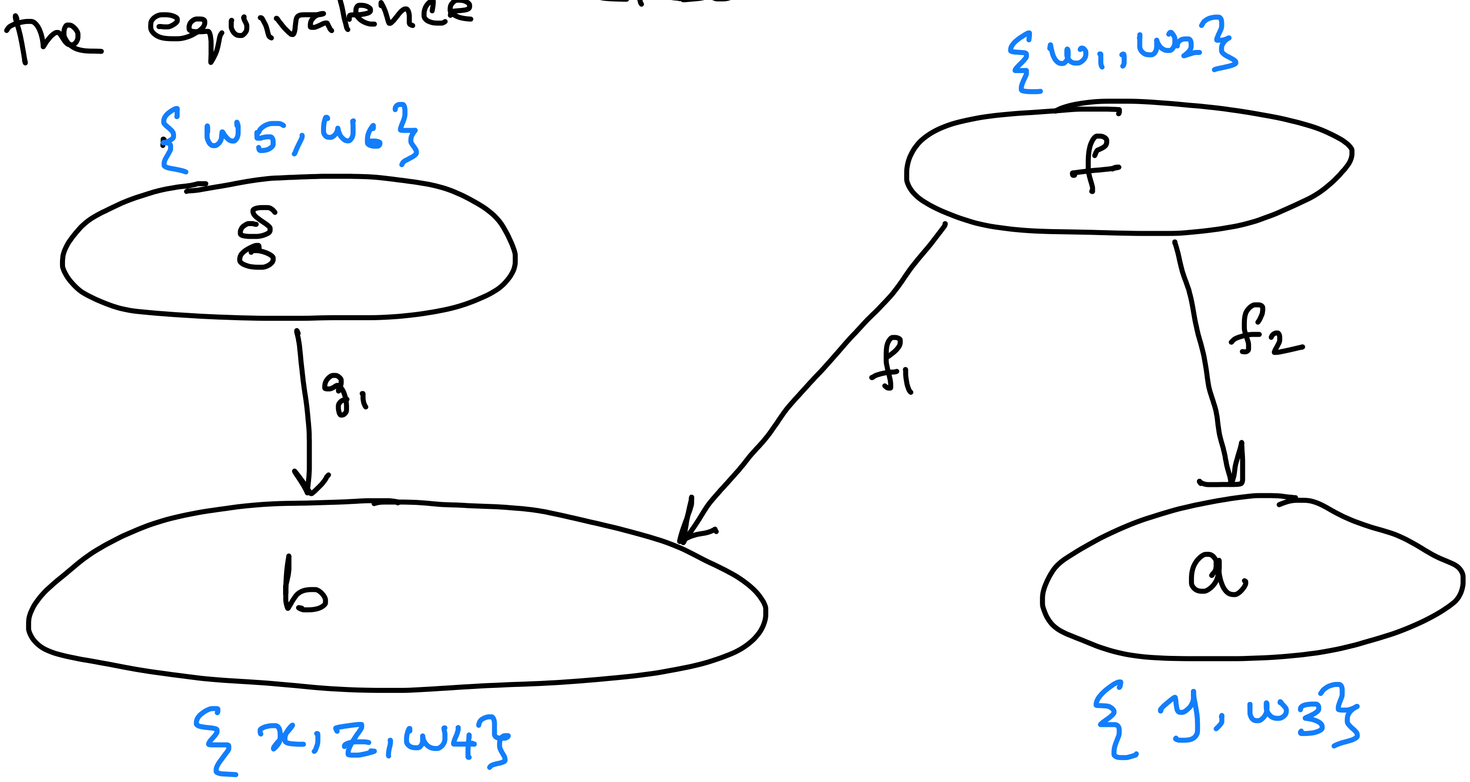
$$\frac{G \vdash u \sim v \quad G \vdash v \sim v'}{G \vdash u \sim v'}\ \text{TRANS}$$

$$\text{DN}\ \frac{G \vdash w \xrightarrow{f_i} u \quad G \vdash w' \xrightarrow{f_i} v}{G \vdash u \sim v}$$

The DN propagates the 'equality' to subterms



$G/\sim$ is a graph whose vertices are the equivalence classes under $\sim$.



$$\text{Label}(q) = \bigcup_{u \in q} \text{label}(u)$$ for each equiv class q.

$q \xrightarrow{f_i} q' \in G/\sim$ iff $\exists\, w \in q \wedge u \in q'$ s.t.

$$w \xrightarrow{f_i} u \in G$$

Consistency of $G/\sim$:

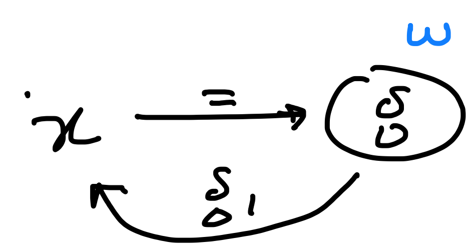
$G/\sim$ is <u>consistent</u> iff

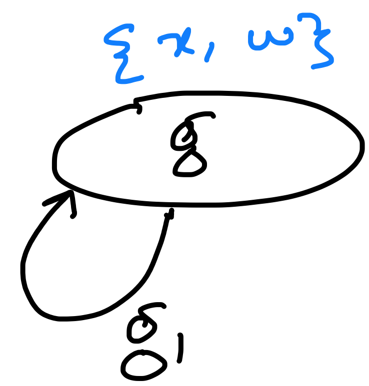
1) $G/\sim$ has no cycles

2) $\forall q \in G/\sim$, $|Label(q)| \leq 1$
(No clashes).

Example of cycle:

$$x \doteq g(x)$$

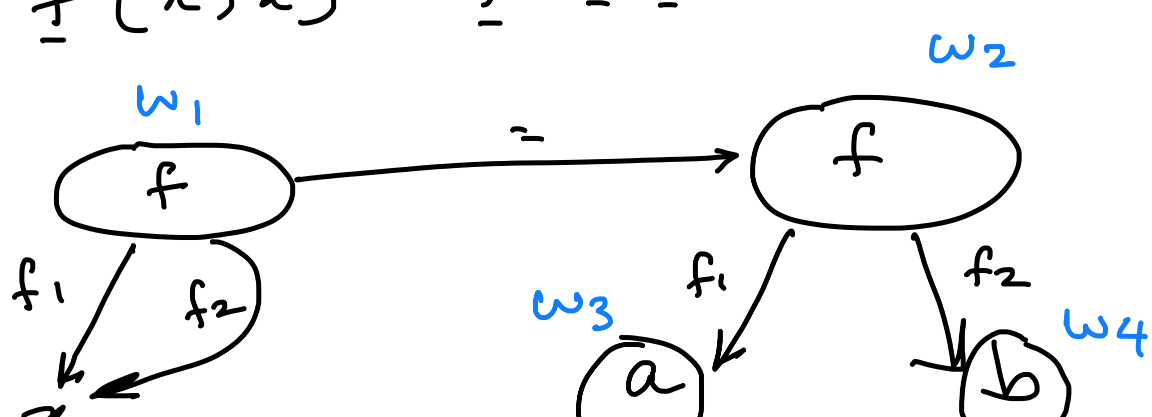




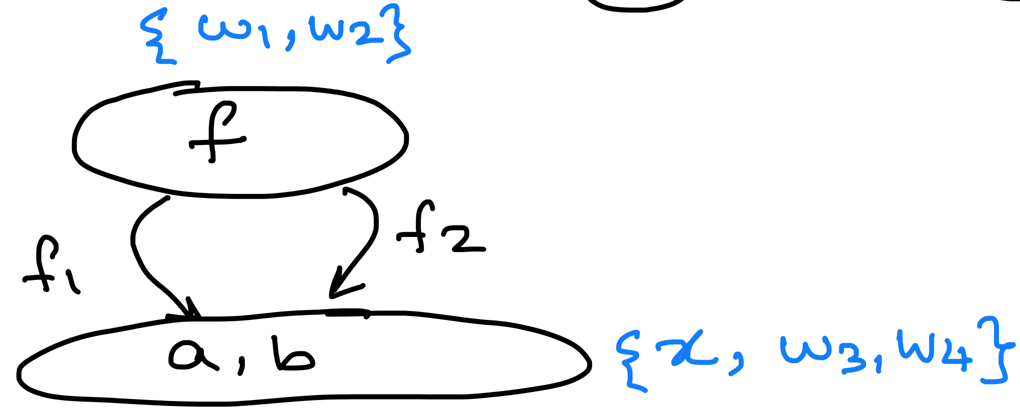
Example of clash.

$$\underline{f}(x, x) = \underline{f}(\underline{a}, \underline{b})$$

G :



$G/\sim$:



If $G/\sim$ is consistent then it is a term graph.

q is a 'variable' node if $L(q) = \emptyset$
q is a constructor node with label l if $Label(q) = \{l\}$.

Let $rep(q)$ be a variable that is a 'class' representative of q, if q is a variable node.

Define $Term(q) =_{df}$

$Term(q) = rep(q)$ if $Label(q) = \emptyset$
$Term(q) = f(Term(q_1) \ldots Term(q_n))$, if
$Label(q) = \{f\}$,
$\alpha(f) = n$
and $q \xrightarrow{f_i} q_i$

Define

σ as:

$dod(\sigma) = vars(G)$
$\sigma(x) = Term(q_x)$ where $x \in q_x$
q_x is the equivalence class containing x.