Computational Logic CS157

Equality III

Selene Makarios
(*docet pro* Mike Genesereth)

Autumn 2005

Preprocessing - Review

Collect all the functions and predicates in our theory. Replace $=$ with E, and add the formulas

$$\forall(x)(E(x,x))$$
$$\forall(x,y)(E(x,y) \rightarrow E(y,x))$$
$$\forall(x,y,z)(E(x,y) \wedge E(y,z) \rightarrow E(x,z))$$

Also add the monotonicity formulas

$$\forall(x_1, ..., x_n, x, y)(E(x,y) \rightarrow E(f(x_1, ..., x, ..., x_n), f(x_1, ..., y, ..., x_n)))$$

and

$$\forall(x_1, ..., x_n, x, y)(E(x,y) \rightarrow (P(x_1, ..., x, ..., x_n) \leftrightarrow P(x_1, ..., y, ..., x_n)))$$

Sidebar

Instead of writing $E(x, y)$, let's replace $x = y$ using a special new symbol '$\risingdotseq$', that kind-of looks like '$=$', but $x \risingdotseq y$ really just means $E(x, y)$. (Now we can use infix.) Preprocessing, then, involves creating the formulas

$$\forall(x)(x \risingdotseq x)$$
$$\forall(x, y)((x \risingdotseq y) \rightarrow (y \risingdotseq x))$$
$$\forall(x, y, z)((x \risingdotseq y) \wedge (y \risingdotseq z) \rightarrow (x \risingdotseq z))$$

and the monotonicity formulas

$$\forall(x_1, ..., x_n, x, y)((x \risingdotseq y) \rightarrow (f(x_1, ..., x, ..., x_n) \risingdotseq f(x_1, ..., y, ..., x_n)))$$

and

$$\forall(x_1, ..., x_n, x, y)((x \risingdotseq y) \rightarrow (P(x_1, ..., x, ..., x_n) \leftrightarrow P(x_1, ..., y, ..., x_n)))$$

Sidebar – Help Name Our New Symbol!

In the literature of computational logic, the symbols '$\simeq$' and '$\approx$' (approximately equals), and sometimes '$\doteq$' (convergently equals), are often used to denote various sorts of equality-type relations other than '$=$'.

We chose to employ something without an existing usage, so we're using '$\substack{z\\z}$', but we don't know what to call it.

1. '$\substack{z\\z}$' is "*equalz*"?
2. '$\substack{z\\z}$' is "*zeequal*"?
3. '$\substack{z\\z}$' is "*double-zorro*"?
4. '$\substack{z\\z}$' is "____"?

Preprocessing - Review

$$\left\{ \begin{array}{c} \forall x(P(x) \rightarrow Q(h(x))) \\ f(a) = g(b,c) \\ \forall x(R(x) \rightarrow h(x) = g(x, f(x))) \end{array} \right\}$$

$$\Downarrow$$

$$\left\{ \begin{array}{cc} \forall x(P(x) \rightarrow Q(h(x))) & (f(a) \doteq g(b,c)) \\ \forall x(R(x) \rightarrow (h(x) \doteq g(x, f(x)))) & \forall x(x \doteq x) \\ \forall (x,y)((x \doteq y) \rightarrow (y \doteq x)) & \forall (x,y,z)((x \doteq y) \wedge (y \doteq z) \rightarrow (x \doteq z)) \\ \forall (x,x')((x \doteq x') \rightarrow (f(x) \doteq f(x'))) & \forall (x,x',y)((x \doteq x') \rightarrow (g(x,y) \doteq g(x',y))) \\ \forall (x,y,y')((y \doteq y') \rightarrow (g(x,y) \doteq g(x,y'))) & \forall (x,x')((x \doteq x') \rightarrow (h(x) \doteq h(x'))) \\ \forall (x,x')((x \doteq x') \rightarrow (P(x) \leftrightarrow P(x'))) & \forall (x,x')((x \doteq x') \rightarrow (Q(x) \leftrightarrow Q(x'))) \\ \forall (x,x')((x \doteq x') \rightarrow (R(x) \leftrightarrow R(x'))) & \end{array} \right\}$$

Theory becomes substantially larger.

Preprocessing - Review

Add a new equation to our theory for each function symbol and each predicate symbol, per each function or predicate argument.

That is, very roughly speaking, the size of the resulting theory is proportional to the size of the vocabulary.

Bad for small theories over large vocabularies.

Might be okay for large theory over small vocabulary (but in practice no-one handles equality this way).

Surprise: There are other ways of “eliminating equality” that can be very effective. (Competitive Model-Elimination systems... E-SETHEO?)

The 2005 CADE ATP System Competition
The World Championship for 1st Order Automated Theorem Proving

MIX	Vampire 6	E-SETHEO csp03	E 0.8	Vampire 5	EP 0.8	Gandalf c-2.6	Gandalf c-2.6-PRF	DCTP 10.2p
Attempted	140	140	140	140	140	140	140	140
Solved	120	119	113	113	113	102	79	72
Av. Time	65.64	34.65	20.91	23.08	25.35	67.75	30.94	43.28
Solutions	120	0	0	113	110	0	79	0

EPR	DCTP 1.3-EPR	Gandalf c-2.6-SAT	E-SETHEO csp02	E-SETHEO csp03	DCTP 10.2p	Paradox 1.0-casc	Vampire 6	E 0.8
Attempted	70	70	70	70	70	70	70	70
Solved	66	61	57	57	55	48	47	47
Av. Time	95.83	248.13	24.95	71.19	47.36	60.71	49.7	80.74
Solutions	0	0	0	0	0	26	32	0

UEQ	Waldmeister 702	Waldmeister 703	E 0.8	E-SETHEO csp03	Vampire 6	Gandalf c-2.6	CiME 2.01	Otter 3.2
Attempted	70	70	70	70	70	70	70	70
Solved	56	56	53	52	48	45	21	11
Av. Time	6.89	6.95	26.36	45.73	52.58	73.29	93.86	44.09
Solutions	56	56	0	0	48	0	0	11

Eliminating Equality

- In our original preprocessing procedure, we "eliminated equality" by accounting for each of the five properties of equality, strictly through the *addition of new formulas* that manifest those properties.
- A different preprocessing approach accounts for the various properties of equality by *modifying the existing equations*, rather than adding more equations.

Eliminating Equality – Flattening

We can account for monotonicity by "flattening":

A formula is "flat" when all proper subterms in the formula are *variables* (i.e. not constants, or more complex terms):

$$P(x,y) \vee (f(x) = y) \quad \textit{flat?}$$
$$P(x,y) \vee (w = a) \vee (f(x) = w) \quad \textit{flat?}$$
$$f(a) \neq g(a) \quad \textit{flat?}$$
$$(f(w) \neq g(w)) \vee (a \neq w) \quad \textit{flat?}$$

Eliminating Equality – Flattening

$$P(x, y) \vee (f(x) = y) \quad \checkmark$$
$$P(x, y) \vee (w = a) \vee (f(x) = w) \quad \checkmark$$
$$f(a) \neq g(a) \quad \times$$
$$(f(w) \neq g(w)) \vee (a \neq w) \quad \checkmark$$

We can "flatten" a formula by introducing new variables:

$$(f(a) \neq g(a)) \;\Rightarrow\; (z = a) \rightarrow (f(z) \neq g(z))$$

$$(i(x) * x = e) \;\Rightarrow\; (i(x) = z) \rightarrow (z * x = e)$$

$$(i(x) * i(x) = i(x * x)) \;\Rightarrow$$
$$(i(x) = z_1) \wedge (x * x = z_2) \rightarrow (z_1 * z_1 = i(z_2))$$

$$P(f(g(h(a)))) \;\Rightarrow$$
$$(g(z_2) = z_1) \wedge (h(z_3) = z_2) \wedge (z_3 = a) \rightarrow P(f(z_1))$$

Eliminating Equality – Flattening

Important: when we flatten, for example like this

$$P(f(a)) \;\Rrightarrow\; (a \neq z) \vee P(f(z)) \quad ,$$

note that the resulting formula is *not logically equivalent* to the original one, i.e.,

$$P(f(a)) \not\equiv (a \neq z) \vee P(f(z)) \; !$$

But that's okay, because it doesn't need to be. Since we (generally) are seeking proof by contradiction, all we need is for the original formula, and the flattened formula, to be co-consistent/co-satisfiable, i.e. each one is satisfiable if and only if the other one is.

Eliminating Symmetry

Each equality $s = t$ produces the two symmetric variants of $s \eqsim t$:

$$C \vee (s = t) \Rightarrow C \vee (s \eqsim t)$$
$$C \vee (s = t) \Rightarrow C \vee (t \eqsim s)$$

Example:

$$R(g(x, y)) \vee (x = y) \Rightarrow \begin{array}{l} R(g(x, y)) \vee (x \eqsim y) \\ R(g(x, y)) \vee (y \eqsim x) \end{array}$$

Eliminating Transitivity

Suppose we have a theory including

$$\{(s_1 \doteq s_2), (s_2 \doteq s_3), (s_3 \doteq s_4)\} \quad ,$$

and we wish to prove $(s_1 \doteq s_4)$. Transitivity would entail this for analogous formulas with $=$, but we need to somehow manifest the transitivity property for $\doteq$.

Idea: rewrite

$$C \vee (s \doteq t) \;\Rrightarrow\; C \vee ((t \doteq x) \rightarrow (s \doteq x))$$

and

$$C \vee (s \not\doteq t) \;\Rrightarrow\; C \vee ((t \doteq x) \rightarrow (s \not\doteq x))$$

if t is not already a variable. (Question: are the left and right sides of these transformation logically equivalent?)

Eliminating Transitivity

So, our three equalities

$$(s_1 \doteq s_2), (s_2 \doteq s_3), (s_3 \doteq s_4) \quad ,$$

become

$$\begin{aligned}(s_2 \doteq x) &\longrightarrow (s_1 \doteq x) \quad ,\\ (s_3 \doteq x) &\longrightarrow (s_2 \doteq x) \quad ,\\ (s_4 \doteq x) &\longrightarrow (s_3 \doteq x) \quad ,\end{aligned}$$

or rather

$$\begin{aligned}(s_2 \not\doteq x) &\vee (s_1 \doteq x) \quad ,\\ (s_3 \not\doteq x) &\vee (s_2 \doteq x) \quad ,\\ (s_4 \not\doteq x) &\vee (s_3 \doteq x) \quad .\end{aligned}$$

Eliminating Transitivity

Check: try to prove $(s_1 \doteq s_4)$. Negate the goal to get

$$(s_1 \not\doteq s_4) \quad .$$

applying the transform, it becomes

$$(s_1 \not\doteq z) \vee (s_4 \not\doteq z) \quad .$$

So, we resolve in order to "trade s_1 for s_2", like this:

$$\frac{(s_1 \not\doteq z) \vee (s_4 \not\doteq z) \qquad (s_2 \not\doteq x) \vee (s_1 \doteq x)}{(s_2 \not\doteq z) \vee (s_4 \not\doteq z)} \quad ,$$

then we trade s_2 for s_3, like this:

$$\frac{(s_2 \not\doteq z) \vee (s_4 \not\doteq z) \qquad (s_3 \not\doteq x) \vee (s_2 \doteq x)}{(s_3 \not\doteq z) \vee (s_4 \not\doteq z)} \quad .$$

Eliminating Transitivity

Finally, we trade s_3 for s_4, like this:

$$\frac{(s_3 \not\doteq z) \vee (s_4 \not\doteq z) \qquad (s_4 \not\doteq x) \vee (s_3 \doteq x)}{(s_4 \not\doteq z) \vee (s_4 \not\doteq z)} \quad .$$

Now we can resolve:

$$\frac{(s_4 \not\doteq z) \qquad (x \doteq x)}{\perp} \quad ,$$

where mgu of $(s_4 \not\doteq z)$ and $(x \doteq x)$ is $\{x \mapsto s_4, z \mapsto s_4\}$. Thus, our proof-by-contradiction of $(s_1 \doteq s_4)$ is complete.

Reflexivity

And finally, just toss the reflexivity postulate in to the theory:

$$(x \doteqdot x)$$

Equality Elimination

Returning to our example theory, clausified,

$$\left\{ \begin{array}{c} \neg P(x) \vee Q(h(x)) \\ f(a) = g(b, c) \\ \neg R(x) \vee (h(x) = g(x, f(x))) \end{array} \right\}$$

$$\Downarrow$$

$$\left\{ \begin{array}{c} \neg P(x) \vee Q(h(x)) \\ (a \not\doteq z_1) \vee (b \not\doteq z_2) \vee (c \not\doteq z_3) \vee (g(z_2, z_3) \not\doteq z_4) \vee (f(z_1) \doteq z_4) \\ (a \not\doteq z_1) \vee (b \not\doteq z_2) \vee (c \not\doteq z_3) \vee (f(z_1) \not\doteq z_4) \vee (g(z_2, z_3) \doteq z_4) \\ \neg R(x) \vee (f(x) \not\doteq z_1) \vee (g(x, z_1) \not\doteq z_2) \vee (h(x) \doteq z_2) \\ \neg R(x) \vee (f(x) \not\doteq z_1) \vee (h(x) \not\doteq z_2) \vee (g(x, z_1) \doteq z_2) \\ (x \doteq x) \end{array} \right\}$$

which is a more modest expansion then before.

AC-Unification

Suppose the function f is associative

$$f(f(x,y),z) = f(x,f(y,z)) \quad ,$$

and commutative

$$f(x,y) = f(y,x) \quad .$$

We want to unify

$$f(x_1,f(x_1,f(x_2,x_3))) \longleftrightarrow\!\!\!\!\!\!\circ\;\; f(f(y_1,y_1),y_2)$$

modulo associativity and commutativity. That is, terms that either are syntactically identical, or are equal due to the AC postulates, are allowed to unify.

AC-Unification – Variables-Only Case

Due to associativity and commutativity, we can write

$$f(x_1, f(f(x_1, x_2), x_3)) \leftarrow\!\circ\!\rightarrow f(f(y_1, y_1), y_2)$$

as

$$f(x_1 x_1 x_2 x_3) \leftarrow\!\circ\!\rightarrow f(y_1 y_1 y_2) \quad .$$

Here's one unifier, (a lucky guess), θ_G:

$$\{x_1 \mapsto b, x_2 \mapsto a, x_3 \mapsto f(aa), y_1 \mapsto a, y_2 \mapsto f(abb)\} \quad ,$$

yielding

$$f(bbaf(aa)) \leftarrow\!\bullet\!\rightarrow f(aaf(abb)) \quad ,$$

or rather

$$f(aaabb) \leftarrow\!\bullet\!\rightarrow f(aaabb) \quad .$$

AC-Unification – Variables-Only Case

Consider again

$$f(x_1 x_1 x_2 x_3) \leftarrow\!\circ\!\rightarrow f(y_1 y_1 y_2) \quad .$$

Condition for AC-unification: after substitution, left and right sides should have *same number* of 'a's. Likewise for 'b's. So, based on above, construct the *constraint equation*

$$2x_1 + x_2 + x_3 = 2y_1 + y_2 \quad .$$

The constraint is satisfied for all replacement terms iff

$$f(x_1 x_1 x_2 x_3) \leftarrow\!\bullet\!\rightarrow f(y_1 y_1 y_2) \quad .$$

AC-Unification – Variables-Only Case

Note that our constraint is satisfied for our θ_G,

$$\{x_1 \mapsto b, x_2 \mapsto a, x_3 \mapsto f(aa), y_1 \mapsto a, y_2 \mapsto f(abb)\} \quad ,$$

for both a,

$$\begin{array}{c} 2x_1 + x_2 + x_3 = 2y_1 + y_2 \\ 2(0) + (1) + (2) = 2(1) + (1) \quad \checkmark \end{array} \quad ,$$

and b,

$$\begin{array}{c} 2x_1 + x_2 + x_3 = 2y_1 + y_2 \\ 2(1) + (0) + (0) = 2(0) + (2) \quad \checkmark \end{array} \quad .$$

AC-Unification – Variables-Only Case

Finding unifiers: find all solutions for the *Diophantine equation* (our constraint equation)

$$2x_1 + x_2 + x_3 = 2y_1 + y_2 \quad .$$

It turns out that every solution-tuple $\langle x_1, x_2, x_3, y_1, y_2 \rangle$ is a weighted sum of the 5-tuples on the left side of the table:

	x_1	x_2	x_3	y_1	y_2	$2x_1{+}x_2{+}x_3$	$2y_1{+}y_2$
r_1	0	0	1	0	1	1	1
r_2	0	1	0	0	1	1	1
r_3	0	0	2	1	0	2	2
r_4	0	1	1	1	0	2	2
r_5	0	2	0	1	0	2	2
r_6	1	0	0	0	2	2	2
r_7	1	0	0	1	0	2	2

AC-Unification – Variables-Only Case

Weighted sum of 5-tuples from solution table

$$\begin{array}{r}
r_1\langle 0,\ 0,\ 1,\ 0,\ 1\rangle \\
r_2\langle 0,\ 1,\ 0,\ 0,\ 1\rangle \\
r_3\langle 0,\ 0,\ 2,\ 1,\ 0\rangle \\
r_4\langle 0,\ 1,\ 1,\ 1,\ 0\rangle \\
r_5\langle 0,\ 2,\ 0,\ 1,\ 0\rangle \\
r_6\langle 1,\ 0,\ 0,\ 0,\ 2\rangle \\
r_7\langle 1,\ 0,\ 0,\ 1,\ 0\rangle \\
\hline
\langle x_1, x_2, x_3, y_1, y_2\rangle
\end{array}$$

we can write x_i, y_i as sums of the tuple weights r_j, i.e.

$$x_1 = r_6+r_7,\quad x_2 = r_2+r_4+2r_5,\quad x_3 = r_1+2r_3+r_4,$$
$$y_1 = r_3+r_4+r_5+r_7,\quad y_2 = r_1 + r_2 + 2r_6 \quad .$$

So, we want to find r_j such that all x_i, y_i are positive integers.

AC-Unification – Variables-Only Case

Example – with

$$r_1=0, r_2=1, r_3=1, r_4=0, r_5=0, r_6=1, r_7=0 \quad ,$$

all the x_i, y_i are positive. That is, setting the r_j values in

$$\begin{array}{l} x_1 = r_6+r_7, \quad x_2 = r_2+r_4+2r_5, \quad x_3 = r_1+2r_3+r_4, \\ y_1 = r_3+r_4+r_5+r_7, \quad y_2 = r_1 + r_2 + 2r_6 \end{array} ,$$

we see that

$$x_1 = 1, x_2 = 1, x_3 = 2, y_1 = 1, y_2 = 3 \quad . \quad \checkmark$$

AC-Unification – Variables-Only Case

Now, since setting the values

$$r_1{=}0\,, r_2{=}1\,, r_3{=}1\,, r_4{=}0\,, r_5{=}0\,, r_6{=}1\,, r_7{=}0 \quad ,$$

of our weight-variables r_j produces a weighted sum of 5-tuples which not only solves our Diophantine equation above, but also causes all x_i, y_i to have positive values (meaning every x_i,y_i gets a term assigned to it), we can use the relations

$$x_1 = r_6{+}r_7, \quad x_2 = r_2{+}r_4{+}2r_5, \quad x_3 = r_1{+}2r_3{+}r_4,$$
$$y_1 = r_3{+}r_4{+}r_5{+}r_7, \quad y_2 = r_1 + r_2 + 2r_6 \quad .$$

to create a *unifier* for the variables-only case,

$$\theta_{V_0} = \{x_1 \mapsto r_6, x_2 \mapsto r_2, x_3 \mapsto f(r_3 r_3), y_1 \mapsto r_3, y_2 \mapsto f(r_2 r_6 r_6)\} \quad .$$

AC-Unification – Variables-Only Case

Verifying: instantiate

$$f(x_1 x_1 x_2 x_3) \longleftrightarrow\!\!\!\!\!\circ\;\; f(y_1 y_1 y_2)$$

using θ_{V_0},

$$\{x_1 \mapsto r_6, x_2 \mapsto r_2, x_3 \mapsto f(r_3 r_3), y_1 \mapsto r_3, y_2 \mapsto f(r_2 r_6 r_6)\} \quad ,$$

producing

$$f(r_6 r_6 r_2 f(r_3 r_3)) \longleftrightarrow\!\!\!\!\!\bullet\;\; f(r_3 r_3 f(r_2 r_6 r_6)) \quad .$$

or rather,

$$f(r_2 r_3 r_3 r_6 r_6) \longleftrightarrow\!\!\!\!\!\bullet\;\; f(r_2 r_3 r_3 r_6 r_6) \quad . \quad \checkmark$$

AC-Unification – Variables-Only Case

Some other variables-only unifiers:

$$\theta_{V_1} = \{x_1 \mapsto r_6, x_2 \mapsto r_4, x_3 \mapsto r_4, y_1 \mapsto r_4, y_2 \mapsto f(r_6 r_6)\}$$
$$\theta_{V_2} = \{x_1 \mapsto r_6, x_2 \mapsto f(r_2 r_4), x_3 \mapsto r_4, y_1 \mapsto r_4, y_2 \mapsto f(r_2 r_6 r_6)\}$$
$$\theta_{V_3} = \{x_1 \mapsto r_6, x_2 \mapsto f(r_5 r_5), x_3 \mapsto r_1, y_1 \mapsto r_5, y_2 \mapsto f(r_1 r_6 r_6)\}$$
$$\theta_{V_4} = \{x_1 \mapsto r_6, x_2 \mapsto f(r_2 r_5 r_5), x_3 \mapsto r_1, y_1 \mapsto r_5, y_2 \mapsto f(r_1 r_2 r_6 r_6)\}$$
$$\theta_{V_5} = \{x_1 \mapsto r_7, x_2 \mapsto r_2, x_3 \mapsto r_1, y_1 \mapsto r_7, y_2 \mapsto f(r_1 r_2)\}$$
$$\theta_{V_6} = \{x_1 \mapsto f(r_6 r_7), x_2 \mapsto r_2, x_3 \mapsto r_1, y_1 \mapsto r_7, y_2 \mapsto f(r_1 r_2 r_6 r_6)\}$$

AC-Unification – θ_G Redux

If we compose θ_{V_0},

$\{x_1 \mapsto r_6, x_2 \mapsto r_2, x_3 \mapsto f(r_3 r_3), y_1 \mapsto r_3, y_2 \mapsto f(r_2 r_6 r_6)\}$

with this *ground substitution* (that we just made up),

$$\{r_2 \mapsto a, r_3 \mapsto a, r_6 \mapsto b\} \ ,$$

we get

$$\{x_1 \mapsto b, x_2 \mapsto a, x_3 \mapsto f(aa), y_1 \mapsto a, y_2 \mapsto f(abb)\}$$

which, by the way, is the same as our substitution guess θ_G from slide 20.

AC-Unification – θ_G Redux

Checking again: instantiate

$$f(x_1 x_1 x_2 x_3) \leftarrow\!\circ\!\rightarrow f(y_1 y_1 y_2)$$

using our newly derived substitution θ_G,

$$\{x_1 \mapsto b, x_2 \mapsto a, x_3 \mapsto f(aa), y_1 \mapsto a, y_2 \mapsto f(abb)\} \ ,$$

producing

$$f(aaabb) \leftarrow\!\bullet\!\rightarrow f(aaabb) \ . \ \checkmark$$

AC-Unification – General Case

Some terms may be non-variables. Our example problem:

$$f(xxya) \leftarrow\!\circ\!\rightarrow f(bbz) \ .$$

First, create a *variable-abstracted* version,

$$f(x_1 x_1 x_2 x_3) \leftarrow\!\circ\!\rightarrow f(y_1 y_1 y_2) \ ,$$

from which we can recover the original version by applying the *recovery substitution*

$$\theta_R = \{x_1 \mapsto x, x_2 \mapsto y, x_3 \mapsto a, y \mapsto b, y_2 \mapsto z\} \ .$$

AC-Unification – General Case

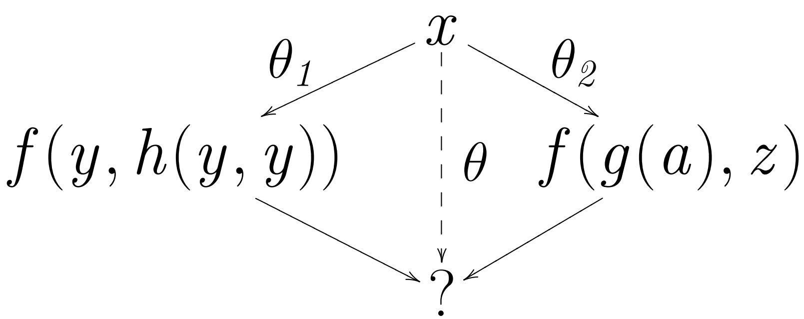
The story so far: we now have two different types of substitutions that can act upon the pair $(f(x_1 x_1 x_2 x_3), f(y_1 y_1 y_2))$ – our recovery substitution, and our variables-only unifiers.

$\theta_R =$	$\{x_1 \mapsto x,$	$x_2 \mapsto y,$	$x_3 \mapsto a,$	$y_1 \mapsto b,$	$y_2 \mapsto z\}$
$\theta_{V_0} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto r_2,$	$x_3 \mapsto f(r_3 r_3),$	$y_1 \mapsto r_3,$	$y_2 \mapsto f(r_2 r_6 r_6)\}$
$\theta_{V_1} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto r_4,$	$x_3 \mapsto r_4,$	$y_1 \mapsto r_4,$	$y_2 \mapsto f(r_6 r_6)\}$
$\theta_{V_2} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto f(r_2 r_4),$	$x_3 \mapsto r_4,$	$y_1 \mapsto r_4,$	$y_2 \mapsto f(r_2 r_6 r_6)\}$
$\theta_{V_3} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto f(r_5 r_5),$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_5,$	$y_2 \mapsto f(r_1 r_6 r_6)\}$
$\theta_{V_4} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto f(r_2 r_5 r_5),$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_5,$	$y_2 \mapsto f(r_1 r_2 r_6 r_6)\}$
$\theta_{V_5} =$	$\{x_1 \mapsto r_7,$	$x_2 \mapsto r_2,$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_7,$	$y_2 \mapsto f(r_1 r_2)\}$
$\theta_{V_6} =$	$\{x_1 \mapsto f(r_6 r_7),$	$x_2 \mapsto r_2,$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_7,$	$y_2 \mapsto f(r_1 r_2 r_6 r_6)\}$

Sidebar

Given two substitutions θ_1 and θ_2, devise a substitution that produces a term that could have resulted either from applying θ_1, or from applying θ_2, and then specializing the result in either case. Example:

$$\theta_1 = \{x \mapsto f(y, h(y, y))\} \quad \theta_2 = \{x \mapsto f(g(a), z)\}$$



We'll call the resulting substitution $\theta = \theta_1 \cap \theta_2$. [Hint: unify something.]

AC-Unification – General Case

Determine $\theta_R \cap \theta_{V_i}$ for each of the θ_{V_i}.

$\theta_R =$	$\{x_1 \mapsto x,$	$x_2 \mapsto y,$	$x_3 \mapsto a,$	$y_1 \mapsto b,$	$y_2 \mapsto z\}$
$\theta_{V_0} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto r_2,$	$x_3 \mapsto$ ~~$f(r_3 r_3)$~~,	$y_1 \mapsto r_3,$	$y_2 \mapsto f(r_2 r_6 r_6)\}$
$\theta_{V_1} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto r_4,$	$x_3 \mapsto$ ~~r_4~~,	$y_1 \mapsto$ ~~r_4~~,	$y_2 \mapsto f(r_6 r_6)\}$
$\theta_{V_2} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto f(r_2 r_4),$	$x_3 \mapsto$ ~~r_4~~,	$y_1 \mapsto$ ~~r_4~~,	$y_2 \mapsto f(r_2 r_6 r_6)\}$
$\theta_{V_3} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto f(r_5 r_5),$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_5,$	$y_2 \mapsto f(r_1 r_6 r_6)\}$
$\theta_{V_4} =$	$\{x_1 \mapsto r_6,$	$x_2 \mapsto f(r_2 r_5 r_5),$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_5,$	$y_2 \mapsto f(r_1 r_2 r_6 r_6)\}$
$\theta_{V_5} =$	$\{x_1 \mapsto r_7,$	$x_2 \mapsto r_2,$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_7,$	$y_2 \mapsto f(r_1 r_2)\}$
$\theta_{V_6} =$	$\{x_1 \mapsto f(r_6 r_7),$	$x_2 \mapsto r_2,$	$x_3 \mapsto r_1,$	$y_1 \mapsto r_7,$	$y_2 \mapsto f(r_1 r_2 r_6 r_6)\}$

Result is

$$\theta_{RV_3} = \{x_1 \mapsto x,\ x_2 \mapsto f(bb),\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(axx)\}$$
$$\theta_{RV_4} = \{x_1 \mapsto x,\ x_2 \mapsto f(r_2 bb),\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(ar_2 xx)\}$$
$$\theta_{RV_5} = \{x_1 \mapsto b,\ x_2 \mapsto r_2,\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(ar_2)\}$$
$$\theta_{RV_6} = \{x_1 \mapsto f(r_6 b),\ x_2 \mapsto y,\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(ayr_6 r_6)\}$$

AC-Unification – General Case

Verify – applying the various θ_{RV_i}

$$
\begin{aligned}
\theta_{RV_3} &= \{x_1 \mapsto x,\ x_2 \mapsto f(bb),\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(axx)\}\\
\theta_{RV_4} &= \{x_1 \mapsto x,\ x_2 \mapsto f(r_2bb),\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(ar_2xx)\}\\
\theta_{RV_5} &= \{x_1 \mapsto b,\ x_2 \mapsto r_2,\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(ar_2)\}\\
\theta_{RV_6} &= \{x_1 \mapsto f(r_6b),\ x_2 \mapsto y,\ x_3 \mapsto a,\ y_1 \mapsto b,\ y_2 \mapsto f(ayr_6r_6)\}
\end{aligned}
$$

we get

$$
\begin{aligned}
(f(x_1x_1x_2x_3) \leftarrow\!\circ\!\rightarrow f(y_1y_1y_2))\theta_{RV_3} &= f(xxf(bb)a) \leftarrow\!\bullet\!\rightarrow f(bbf(axx))\\
(f(x_1x_1x_2x_3) \leftarrow\!\circ\!\rightarrow f(y_1y_1y_2))\theta_{RV_4} &= f(xxf(r_2bb)a) \leftarrow\!\bullet\!\rightarrow f(bbf(ar_2xx))\\
(f(x_1x_1x_2x_3) \leftarrow\!\circ\!\rightarrow f(y_1y_1y_2))\theta_{RV_5} &= f(bbr_2a) \leftarrow\!\bullet\!\rightarrow f(bbf(ar_2))\\
(f(x_1x_1x_2x_3) \leftarrow\!\circ\!\rightarrow f(y_1y_1y_2))\theta_{RV_6} &= f(f(r_6b)f(r_6b)ya) \leftarrow\!\bullet\!\rightarrow f(bbf(ayr_6r_6))
\end{aligned}
$$

or rather,

$$
\begin{aligned}
f(xxbba) &\leftarrow\!\bullet\!\rightarrow f(xxbba)\ \checkmark\\
f(xxr_2bba) &\leftarrow\!\bullet\!\rightarrow f(xxr_2bba)\ \checkmark\\
f(bbr_2a) &\leftarrow\!\bullet\!\rightarrow f(bbr_2a)\ \checkmark\\
f(bbr_6r_6ya) &\leftarrow\!\bullet\!\rightarrow f(bbr_6r_6ya)\ \checkmark \quad \text{It works! ☺}
\end{aligned}
$$