Theorems for Free: an Introduction

José N. Oliveira
Dept. Informatica
Universidade do Minho, 4700 Braga, Portugal
jno@di.uminho.pt

September 16, 2004

Parametric polymorphism: why?

- Less code (**specific** solution = **generic** solution + **customization**)
- Intellectual reward
- Last but not least, quotation (from Theorems for free!, by Philip Wadler [?]):

  > *From the type of a polymorphic function we can derive a theorem that is satisfies.* (...) *How useful are the theorems so generated? Only time and experience will tell* (...)

- No doubt: free theorems are **very** useful!

Polymorphic type signatures

Polymorphic function signature:

$$f \quad : \quad t$$

where t is a functional type, according to the following "grammar" of types:

$$\begin{array}{lll} t & ::= & t' \leftarrow t'' \\ t & ::= & \mathsf{F}(t_1, \ldots, t_n) \\ t & ::= & v \qquad \text{type variables } v\text{, cf. polymorphism} \end{array}$$

What does it mean that f is **parametrically** polymorphic?

Free theorem of type t

Let

- V be the set of type variables involved in type t
- $\{R_v\}_{v \in V}$ be a V-indexed family of relations (f_v in case all such R_v are functions).
- R_t be a relation defined inductively as follows:

$$\begin{array}{rcl} R_{t=\mathsf{F}(t_1,\ldots,t_n)} & = & \mathsf{F}(R_{t_1}, \ldots, R_{t_n}) \\ R_{t=v} & = & R_v \\ R_{t=t' \leftarrow t''} & = & R_{t'} \leftarrow R_{t''} \end{array}$$

- What kind of relation is $R_{t'} \leftarrow R_{t''}$?

Reynolds arrow operator

$$f(R \leftarrow S)g \quad \equiv \quad f \cdot S \subseteq R \cdot g$$

$$\begin{tikzcd} A \arrow[d, "f"'] & B \arrow[l, "S"'] \arrow[d, "g"] \\ C & D \arrow[l, "R"] \end{tikzcd}$$

That is to say,

$$\frac{A \xleftarrow{S} B \qquad C \xleftarrow{R} D}{C^A \xleftarrow{R \leftarrow S} D^B}$$

For instance, $f(id \leftarrow id)g \quad \equiv \quad f = g$ that is, $id \leftarrow id \quad = \quad id$

Free theorem (FT) of type t

The following (remarkable) theorem — due to J. Reynolds and advertised by P. Wadler — holds:

> *Given any function $\theta : t$, and V as above, then $\theta \; R_t \; \theta$ holds, for any relational instantiation of type variables in V.*

Note that this theorem

- is a result about t
- holds **independently** of the actual definition of θ.
- holds about any function of type t

First example

- The target function: $\theta = invl : a^\star \leftarrow a^\star$.
- Calculation of $R_{t=a^\star \leftarrow a^\star}$:

$$
\begin{array}{ll}
 & R_{a^\star \leftarrow a^\star} \\
\equiv & \qquad \{ \text{ rule } \ R_{t=t' \leftarrow t''} \ = \ R_{t'} \leftarrow R_{t''} \ \} \\
 & R_{a^\star} \leftarrow R_{a^\star} \\
\equiv & \qquad \{ \text{ rule } \ R_{t=\mathsf{F}(t_1,\ldots,t_n)} \ = \ \mathsf{F}(R_{t_1},\ldots,R_{t_n}) \ \} \\
 & {R_a}^\star \leftarrow {R_a}^\star
\end{array}
$$

- Calculation of FT $invl\ (R_{a^\star \leftarrow a^\star})\ invl$ follows

$invl$ FT calculation

The FT itself will predict (R_a abbreviated to R):

$$
\begin{array}{ll}
 & invl(R^\star \leftarrow R^\star)invl \\
\equiv & \qquad \{ \text{ definition } \ f(R \leftarrow S)g \ \equiv \ f \cdot S \subseteq R \cdot g \ \} \\
 & invl \cdot R^\star \subseteq (R^\star) \cdot invl
\end{array}
$$

In case R is a function r, the FT theorem boils down to $invl$'s **natural** property:

$$invl \cdot r^\star \quad = \quad r^\star \cdot invl$$

Further calculation (back to R):

Pointwise version of FT

$$
\begin{array}{ll}
 & invl \cdot R^\star \subseteq (R^\star) \cdot invl \\
\equiv & \quad \{ \text{ shunting rule } \} \\
 & R^\star \subseteq invl^\circ \cdot (R^\star) \cdot invl \\
\equiv & \quad \{ \text{ going pointwise } \} \\
 & l(R^\star)r \quad \Rightarrow \quad (invl\ l)(R^\star)(invl\ r) \\
\equiv & \quad \{ \text{ pointwise definition of } R^\star \text{ } \} \\
 & \forall i \in inds\ l.(l\ i)R(r\ i) \quad \Rightarrow \quad (invl\ l)(R^\star)(invl\ r)
\end{array}
$$

Pointwise version of FT

For example, $invl$ will respect orderings:

$$
\begin{array}{ll}
 & \forall i \in inds\ l.(l\ i) < (r\ i) \\
\Rightarrow & \forall j \in inds(invl\ l).(invl\ l)j < (invl\ r)j
\end{array}
$$

Exercise: calculate the FT of

$$sort : a^\star \leftarrow a^\star \leftarrow (2 \leftarrow (a \times a))$$

(the first parameter stands for the ordering relation.)

Second example: FT of $(\!|\_|\!)$

- $(\!|\_|\!)$ has generic type

  $$(\!|\_|\!) : b \leftarrow \mathsf{F}\, a \leftarrow (b \leftarrow \mathsf{B}\,(a, b))$$

  where $\mathsf{F}\, a \cong \mathsf{B}\,(a, \mathsf{F}\, a)$.

- FT-$(\!|\_|\!)$:

  $$(\!|\_|\!) \cdot (R_b \leftarrow \mathsf{B}\,(R_a, R_b)) \quad \subseteq \quad (R_b \leftarrow \mathsf{F}\, R_a) \cdot (\!|\_|\!)$$

- FT-$(\!|\_|\!)$ calculation follows (R_a, R_b abbreviated to R, S):

FT-$(\!|\_|\!)$ corollaries

$$
\begin{array}{ll}
& (\!|\_|\!) \cdot (S \leftarrow \mathsf{B}\,(R,S)) \quad \subseteq \quad (S \leftarrow \mathsf{F}\,R) \cdot (\!|\_|\!) \\
\equiv & \qquad \{ \text{ definition } \; f(R \leftarrow S)g \;\; \equiv \;\; f \cdot S \subseteq R \cdot g \;\; \} \\
& f(S \leftarrow \mathsf{B}\,(R,S))g \quad \Rightarrow \quad (\!|f|\!)(S \leftarrow \mathsf{F}\,R)(\!|g|\!) \\
\equiv & \qquad \{ \text{ idem } \} \\
& f \cdot \mathsf{B}\,(R,S) \subseteq S \cdot g \quad \Rightarrow \quad (\!|f|\!) \cdot \mathsf{F}\,R \subseteq S \cdot (\!|g|\!)
\end{array}
$$

At this point, we can infer ...

FT-$(\!|\_|\!)$ corollaries

From this, infer

- $(\!|\_|\!)$-fusion $(R, S := id, s)$:

$$
(f \cdot \mathsf{B}\,(id, s) = s \cdot g) \quad \Rightarrow \quad (\!|f|\!) = s \cdot (\!|g|\!)
$$

- $(\!|\_|\!)$-absorption $(R, S := r, id)$:

$$
\begin{array}{ll}
& (f \cdot \mathsf{B}\,(r, id) = g) \quad \Rightarrow \quad (\!|f|\!) \cdot \mathsf{F}\,r = (\!|g|\!) \\
\equiv & \qquad \{ \text{ replacement of } g \; \} \\
& (\!|f|\!) \cdot \mathsf{F}\,r = (\!|f \cdot \mathsf{B}\,(r, id)|\!)
\end{array}
$$

Bibliography

[Wad89] P. Wadler. Theorems for free! In *4th International Symposium on Functional Programming Languages and Computer Architecture*, London, Sep. 1989. ACM.