Higher Order Operator Overloading and Partial Evaluation

by Sven Nilsen, 2019

Higher Order Operator Overloading^[1] states that when you have functions of the same input, you can overload any operator on the return values of the functions:

$$g(f_0, f_1) := (a : A) = g(f_0(a), f_1(a))$$

Self Reference under Higher Order Operator Overloading^[2] means constructing functions like these:

$$x := \(x : real, y : real) = x$$

$$y := (x : real, y : real) = y$$

Partial Evaluation^[3] means calling some function `f` and interpreting it using Higher Order Operator Overloading, with one or more self-referential function:

$$f(x : real, y : real) = x^2 + y^2 \le 1$$

$$f(x, 0) \le \langle x : real, y : real \rangle = x^2 \le 1$$
 Partially evaluated function where $y = 0$

$$f(0, y) \le \langle x : real, y : real \rangle = y^2 \le 1$$
 Partially evaluated function where $x = 0$

To reduce the number of arguments, one can use the 'id' function:

$$f(x : real, y : real) = x^2 + 3 \cdot y$$

 $f(id, 0) <=> \setminus (x : real) = x^2$
 $f(0, id) <=> \setminus (y : real) = 3 \cdot y$

If there are more than one argument, one must use self-referential functions of the new context:

$$f(x : real, y : real, z : real) = x^3 + y^2 + z$$

 $x := \(x : real, z : real) = x$
 $z := \(x : real, z : real) = z$
 $f(x, 0, z) \le \(x : real, z : real) = x^3 + z$

One can also extend the context, e.g. to add infinitesimal changes:

```
x := (x : real, dx : real, y : real, dy : real, z : real, dz : real) = x + dx
```

Partial Evaluation^[3] is different from Function Currying^[4], although it is logically equivalent:

- Partial Evaluation reduces a program into basis function-calls
- All constant expressions are reduced

For example:

$$f(x) = x + (1 + 4)$$

When passing `x` as a self-refential function, `f` becomes:

$$f(x) = x + 5$$

Notice that this is a logically equivalent function, but it might run faster.

So, naturally, Higher Order Operator Overloading^[1] might be used to:

- Analyze the inner workings of programming languages
- Change performance characteristics of functions

References:

[1]	"Higher Order Operator Overloading"
	Sven Nilsen, 2018

 $\underline{https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/higher-order-operator-overloading.pdf}$

[2] "Higher Order Operator Overloading and Self Reference" Sven Nilsen, 2019

 $\underline{https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/higher-order-operator-overloading-and-self-reference.pdf}$

[3] "Partial evaluation"
Wikipedia
https://en.wikipedia.org/wiki/Partial_evaluation

[4] "Currying" Wikipedia

https://en.wikipedia.org/wiki/Currying