

Weihrauch Reducibility as a Lens

Ian Price (countingishard.org)

Computational problem: partial multi-valued function $f: \subseteq \mathbb{N}^k \rightrightarrows \mathbb{N}^l$

input: any $x \in \text{dom}(f)$

output: any $y \in f(x)$

$g \leq_W f \iff$ there are $\Phi, \Psi: \subseteq \mathbb{N}^k \rightarrow \mathbb{N}^l$ computable such that

- Given $p \in \text{dom}(g)$, $\Phi(p) \in \text{dom}(f)$
- Given $q \in f(\Phi(p))$, $\Psi(p, q) \in g(p)$

$G: \begin{array}{ccccc} I & \xleftarrow{u} & D & \xrightarrow{g} & C & \xrightarrow{v} & J \\ \parallel & & \uparrow & & \parallel & & \parallel \\ & & B' & \xrightarrow{\lambda} & C & & \\ & & \downarrow & & \downarrow & & \\ F: & I & \xleftarrow{s} & B & \xrightarrow{f} & A & \xrightarrow{t} & J \end{array}$

Corporate needs you to find the differences between this picture and this picture.

Reducibility

Definition (Weihrauch Reducible)

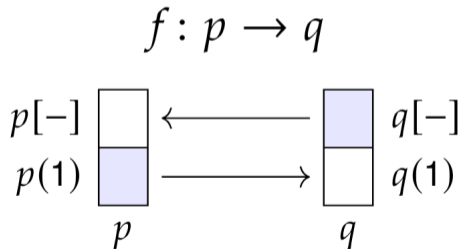
Let X, Y, Z, W be represented spaces and let $f : \subseteq X \rightrightarrows Y$, $g : \subseteq Z \rightrightarrows W$ be partial multi-valued functions.

Then f is *Weihrauch reducible* to g , if there are computable partial functions $\Phi, \Psi : \subseteq \mathbb{N}^{\mathbb{N}} \rightarrow \mathbb{N}^{\mathbb{N}}$ such that for all p with $\delta_X(p) \in \text{dom}(f)$, $\delta_Z(\Phi(p)) \in \text{dom}(g)$ and for all q with $\delta_W(q) \in g(\delta_Z(\Phi(p)))$, $\delta_Y(\Psi(p, q)) \in f(\delta_X(p))$.

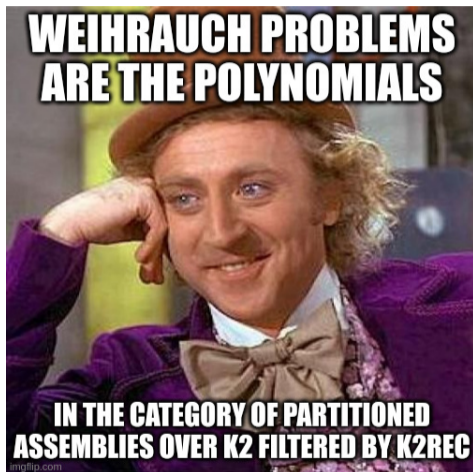
$$\begin{array}{ccc} p & \xrightarrow{\Phi} & \Phi(p) \\ \downarrow f & & \downarrow g \\ g(p) & \xleftarrow{\Psi(p, \cdot)} & q \end{array}$$

Theorem (Niu, Spivak)

Given polynomial functors p and q , a natural transformation (lens) $f : p \rightarrow q$ can be identified with a pair $(f_1, f^\#)$ where $f_1 : p(1) \rightarrow q(1)$ is a function and $f^\# : q[f_1(-)] \rightarrow p[-]$ is a natural transformation (a family of functions $f_i^\# : q[f_1(i)] \rightarrow p[i]$ for $i \in p(1)$).



My Advisor Weighs In



Long Story Short

- Sets Bad, Type 2 computability Good
- $\text{Pasm}(\mathcal{K}_2^{\text{rec}}, \mathcal{K}_2)$ is not an LCCC, so can't define Polynomial Functors
- but you can mimic *vertical-cartesian* factorisation
- this is enough

$$\begin{array}{ccc} Y & \xrightarrow{f} & X \\ \Psi \uparrow & & \parallel \\ W \times_Z X & \longrightarrow & X \\ \downarrow & \lrcorner & \downarrow \Phi \\ W & \xrightarrow{g} & Z \end{array}$$

Vertical-Cartesian factorisation

Thank You

Details at countingishard.org

