Logical Methods in Computer Science
Volume 16, Issue 3, 2020, pp. 7:1–7:17
https://lmcs.episciences.org/

Submitted Oct. 05, 2018
Published Jul. 23, 2020

FIXED POINT COMBINATORS AS FIXED POINTS OF HIGHER-ORDER FIXED POINT GENERATORS

ANDREW POLONSKY

Appalachian State University, Boone NC 28608-2133, USA
e-mail address: andrew.polonsky@gmail.com

Abstract. Corrado Böhm once observed that if Y is any fixed point combinator (fpc), then $Y(\lambda yx.x(yx))$ is again fpc. He thus discovered the first "fpc generating scheme" – a generic way to build new fpcs from old. Continuing this idea, define an *fpc generator* to be any sequence of terms $G_1, \ldots, G_n$ such that

$$Y \text{ is fpc } \implies YG_1 \cdots G_n \text{ is fpc.}$$

In this contribution, we take first steps in studying the structure of (weak) fpc generators. We isolate several robust classes of such generators, by examining their elementary properties like injectivity and (weak) constancy. We provide sufficient conditions for existence of fixed points of a given generator $(G_1, \cdots, G_n)$: an fpc Y such that $Y = YG_1 \cdots G_n$. We conjecture that weak constancy is a necessary condition for existence of such (higher-order) fixed points. This statement generalizes Statman's conjecture on non-existence of "double fpcs": fixed points of the generator $(G) = (\lambda yx.x(yx))$ discovered by Böhm.

Finally, we define and make a few observations about the monoid of (weak) fpc generators. This enables us to formulate new conjectures about their structure.

Dedicated to Corrado Böhm, a pioneer of the Lambda Calculus

1. Introduction

Fixed point combinators (fpcs) are a fascinating class of lambda terms. Arising in the proof of the Fixed Point Theorem, their dynamical character affects the global structure of the Lambda Calculus in a fundamental way. Being a mechanism of unrestricted recursion, they are directly responsible for the Turing-completeness of the lambda calculus as a programming language.[1] And when lambda terms are used as the computational basis of a logical system — whether based on the Curry–Howard isomorphism or illative combinatory logic — fixed point combinators appear unexpectedly as the untyped skeletons of paradoxes, heralding inconsistency of the logic lying over the computational calculus. [2] [9] [4] [8] [5] [13]

It is an elementary fact that a term Y is a fixed point combinator if and only if Y is itself a fixed point of the combinator $\delta = \lambda yx.x(yx)$. [1, 6.5.3] This can even be taken as the definition of fpcs: $Y \in \Lambda$ is fpc iff $Y = \delta Y$. Corrado Böhm noticed that also $Y\delta$ is fpc

Key words and phrases: Fixed point combinator, Lambda Calculus, Bohm tree, FPC generator.

[1] In fact, the very notion of Turing-completeness traces back to Church's bold suggestion that lambda calculus can encode arbitrary computational processes. Yet the idea was only accepted after Kleene and Turing, using fixed point constructions, showed equivalence between Church's formalism and their own ones.

DOI:10.23638/LMCS-16(3:7)2020

whenever Y is. [1, 6.5.4] For example, if $Y = \mathtt{Y}$ is Curry's fpc, then $Y\delta = \Theta$ is Turing's fpc. A major open problem in the Lambda Calculus asks whether there exists a "double fpc" Y satisfying $\delta Y = Y = Y\delta$. Statman [15] conjectures that no such Y exists. An early attack on this problem was undertaken by Intrigila [10]. Unfortunately, Endrullis discovered a gap in the argument which seems difficult to overcome. As of this writing, the conjecture remains open. For recent developments, see [6], [12]. We will also discuss the conjecture in Section 4.

Böhm's observations revealed that fpcs themselves have a compositional structure, where one constructs new fpcs from old by applying them to δ. Since then, other "fpc generating schemes" have been discovered and investigated by several authors. [14] [6] These contributions have confirmed that fpcs have a rich mathematical structure indeed.

In this paper, we will explore such fpc generators "in the abstract", studying their general properties and providing a basic taxonomy. We formulate several new problems, including a significant strengthening of Statman's conjecture.

2. Notations and definitions

Notation 2.1. We assume the reader is familiar with the basic notions of lambda calculus: λ-terms, free variables, substitution, and beta-conversion. We refer to [1] for background on these matters. Here we shall employ the following symbols and notions.

- Λ is the set of λ-terms. $\Lambda^0 = \{M \in \Lambda \mid \mathsf{FV}(M) = \varnothing\}$ is the set of *closed* λ-terms.
- $\mathsf{FV}(M)$ is the set of free variables of $M \in \Lambda$.
- $M[x := N]$ is the result of capture-avoiding substitution of N for x in M.
- If $\vec{N} = (N_1, \dots, N_k)$ is a sequence of λ-terms, then $M\vec{N} = MN_1 \cdots N_k$.
- $F^k(z) := F(F(\cdots F(z)\cdots))$, with k Fs.
- $\mathtt{I} = \lambda x.x$, $\mathtt{K} = \lambda xy.x$, $\mathtt{c}_k = \lambda xy.x^k(y)$, $\delta = \lambda yx.x(yx)$.
- $M = N$ denotes beta conversion between M and N.
- $M \twoheadrightarrow N$ denotes beta reduction from M to N.
- M is *solvable*, if $M\vec{N} = \mathtt{I}$ for some $\vec{N}$. Otherwise, M is *unsolvable*.
- $M =^\infty N$ if M and N have the same Böhm tree.
  We note without proof that this relation can be defined using one axiom and one inference rule, the latter to be understood *coinductively* (see [11],[7], and especially [3, Def. 5.6]):

$$\frac{M, N \quad \text{unsolvable}}{M =^\infty N} \qquad \frac{M = \lambda\vec{x}.y\vec{M} \quad N = \lambda\vec{x}.y\vec{N} \qquad M_1 =^\infty N_1 \quad \cdots \quad M_k =^\infty N_k}{M =^\infty N}$$

- $z\#M$ means $z \notin \mathsf{FV}(M)$. For $S \subseteq \Lambda$, $z\#S$ means $z\#M$ for each $M \in S$.
- $z \notin M$ if there exists $N = M$ such that $z\#N$. $z \in M$ if $z \in \mathsf{FV}(N)$ for all $N = M$.
- $z \notin^\infty M$ if there exists $N =^\infty M$ such that $z\#N$. Otherwise, $z \in^\infty M$.

Definition 2.2. $Y \in \Lambda$ is a *fixed point combinator (fpc)* if $Yx = x(Yx)$ for $x\#Y$.

Definition 2.3. $Y \in \Lambda$ is a *weak fixed point combinator (wfpc)* if $Yx =^\infty x(Yx)$ for $x\#Y$.

Notice that every fpc is a wfpc. See Examples 3.1 for both types of terms.

All (w)fpcs have the same Böhm tree, so $Y \in \Lambda$ is wfpc iff $Y =^\infty Y_0$ for some fpc Y_0.

A wfpc Y can equivalently be given by a sequence of terms (Y_n) with $Y = Y_0$ and $Y_n x = x(Y_{n+1}x)$, with $x\#\{Y_n, Y_{n+1}\}$. [12, Prop. 3.9] If Y is fpc, then $Y_n = Y_0$ for all n.

Notation 2.4. We write FPC (WFPC) for the set of fpcs (weak fpcs).

Notation 2.5. Henceforth, we shall often write (W)FPC in a sentence that is meant to apply to both FPC and WFPC. Such a statement should always be read as a conjunction of two statements: one, in which parentheses are ignored together with their contents, and another, where parentheses are removed but their contents remain.

Definition 2.6. A *(weak) fpc generating vector*, or *(w)fgv*, is a sequence of terms $\vec{G}$ satisfying

$$Y \in \mathsf{(W)FPC} \implies Y\vec{G} \in \mathsf{(W)FPC}.$$

Proposition 2.7. *TFAE:*
(i) $\vec{G}$ *is wfgv.*
(ii) $Y \in \mathsf{FPC} \implies Y\vec{G} \in \mathsf{WFPC}.$
(iii) $Y\vec{G} \in \mathsf{WFPC}$ *for some* $Y \in \mathsf{FPC}$.

Proof. $(i) \Rightarrow (ii)$. Let $\vec{G}$ be wfgv, Y be fpc. Then Y is wfpc, and $Y\vec{G}$ is wfpc.
$(ii) \Rightarrow (iii)$. Trivial.
$(iii) \Rightarrow (i)$. Let Z be wfpc. Then $Z =^\infty Y$. Also $Z\vec{G} =^\infty Y\vec{G}$ is wfpc. □

Corollary 2.8. *Every fpc generator is wfpc generator.*

Proof. Let $\vec{G}$ be fpc generator. Pick $Y \in \mathsf{FPC}$. Then $Y\vec{G}$ is fpc, hence $\vec{G}$ is wfgv. □

Proposition 2.9. *Consider the following conditions on* $\vec{G}$.

$$\begin{array}{llrcll}
(i) & Y & \textit{fpc} & \implies & Y\vec{G} & \textit{fpc} \\
(ii) & Y & \textit{wfpc} & \implies & Y\vec{G} & \textit{wfpc} \\
(iii) & Y & \textit{fpc} & \implies & Y\vec{G} & \textit{wfpc} \\
(iv) & Y & \textit{wfpc} & \implies & Y\vec{G} & \textit{fpc}
\end{array}$$

The following relations are valid:

$$(iv) \implies (i) \implies (ii) \iff (iii)$$

Proof. These relations simply summarize the facts noted above. □

3. Examples and first observations

Examples 3.1.
- *Turing's fpc.* Let $\Theta x = VVx$, where $V = \lambda vx.x(vvx)$. Then $\Theta \in \mathsf{FPC}$.
- *Parametrized Turing's fpc.* For $M \in \Lambda$, let $\Theta_M x = VVMx$, where $V = \lambda vmx.x(vvmx)$. Then $\Theta_M \in \mathsf{FPC}$. (This example can be generalized to have multiple parameters.)
- Let z be a variable. Put $\Psi_z = W_z W_z \mathtt{I}$, where $W_z = \lambda wpx.x(ww(zp)x)$. Then $\Psi_z \in \mathsf{WFPC} \setminus \mathsf{FPC}$.
- A slight variant of the above will play a central role in the proof of our main result. Let c be a variable. Put $\Upsilon = \lambda x.V_x \mathtt{I} V_x$, where $V_x = \lambda pv.x(v(cp)v)$. Then $\Upsilon \in \mathsf{WFPC} \setminus \mathsf{FPC}$.

  A nice feature of this wfpc is that it has a very simple reduction graph.

Proposition 3.2. $\Theta_M = \Theta_N \implies M = N$.

Proof. This is manifest upon inspecting the reduction graph of Θ_z — the set of reducts of Θ_z. For a precise proof, see [12, Lemma 4.1]. □

Examples 3.3.

- Let $\vec{G} = ()$, the empty vector. Obviously, $Y \in \mathsf{(W)FPC} \implies Y\vec{G} = Y \in \mathsf{(W)FPC}$.
  We call this generator *trivial.* In subsequent sections, we will tacitly assume all generators to be non-trivial.
- Fix a (w)fpc Y, and let $\vec{G} = (\mathtt{K}Y)$. Then $(\mathtt{K}Y)$ yields the same (w)fpc on every input:
  $$Z = (Z_0, Z_1, \dots) \in \mathsf{WFPC} \implies Z_0(\mathtt{K}Y) = \mathtt{K}Y(Z_1(\mathtt{K}Y)) = Y \in \mathsf{(W)FPC}.$$
  We call such generators *constant.* Their only interesting feature is the fixed point $Y = Y\vec{G}$.
- Recall that $\delta yx = x(yx)$. It is easy to verify the following:
  - $\delta^k(z)x = x^k(zx)$.
  - If Y is fpc, then $Y = \delta Y = \delta^k(Y)$.
  - If $Y = (Y_n)$ is wfpc, then $Y_0 = \delta^k(Y_k)$.

  Let $\vec{G} = (\delta)$. Then $Y \in \mathsf{FPC} \implies Y\delta x = \delta(Y\delta)x = x(Y\delta x) \implies Y\delta \in \mathsf{FPC}$.
  As noted in the introduction, it is open whether there exists $Y \in \mathsf{(W)FPC}$ such that $Y = Y\delta$.
- Let $\vec{G} = (\lambda y.\Theta_y)$. Then
  $$Y \in \mathsf{FPC} \implies Y\vec{G} = Y(\lambda y.\Theta_y) = (\lambda y.\Theta_y)(Y(\lambda y.\Theta_y)) = \Theta_{Y\vec{G}} \in \mathsf{FPC}.$$
  Furthermore, there exists fpc Y such that $Y = Y\vec{G}$.
  Indeed, take $Y = \Theta(\lambda x.\Theta_{x(\lambda y.\Theta_y)}) = \Theta_{Y(\lambda y.\Theta_y)}$. Then $Y \in \mathsf{FPC}$, and
  $$Y(\lambda y.\Theta_y) = (\lambda y.\Theta_y)(Y(\lambda y.\Theta_y)) = \Theta_{Y(\lambda y.\Theta_y)} = Y.$$
- Yet another single-term fgv is given by $\vec{G} = (\lambda yx.x(y(\mathtt{K}[y,x])\mathtt{I}))$, where $[P,Q] = \lambda z.zPQ$:
  $$\begin{aligned} Y \in \mathsf{FPC} \implies YG_0x = G_0(YG_0)x &= x(YG_0(\mathtt{K}[YG_0,x])\mathtt{I}) \\ &= x(G_0(YG_0)(\mathtt{K}[YG_0,x])\mathtt{I}) \\ &= x(\mathtt{K}[YG_0,x](\cdots)\mathtt{I}) \\ &= x([YG_0,x]\mathtt{I}) = x(\mathtt{I}(YG_0)x) = x(YG_0x) \end{aligned}$$
- The set of (w)fgvs is closed under composition: if $\vec{G}$ and $\vec{G}'$ are fgvs, then
  $$Y \in \mathsf{FPC} \implies Y\vec{G} \in \mathsf{FPC} \implies Y\vec{G}\vec{G}' \in \mathsf{FPC}.$$
  Thus, $(\delta, \lambda y.\Theta_y)$ and $(\lambda y.\Theta_y, \delta)$ are both fgvs.
- Many other examples of fpcs and fgvs can be found in [6] and [12].

Definition 3.4. A (w)fgv $\vec{G}$ is *injective* if for all (w)fpcs Y, Y', $Y\vec{G} = Y'\vec{G}$ implies $Y = Y'$.

Proposition 3.5. *No non-trivial (w)fgv is injective.*

Proof. Suppose $\vec{G} = (G_0, \dots, G_n)$, for $n \geq 0$, is injective. Since $\Theta\vec{G} = G_0(\Theta G_0)G_1 \cdots G_n$ is (w)fpc, G_0 must be solvable. That is, $G_0\vec{P} = \mathtt{I}$ for some closed $\vec{P}$. Put $Y = \Theta_{x\vec{P}}$, $Y' = \Theta_{x\vec{P}\mathtt{I}}$. By Proposition 3.2, $Y \neq Y'$. Yet
$$Y\vec{G} = \Theta_{G_0\vec{P}}\vec{G} = \Theta_{\mathtt{I}}\vec{G} = \Theta_{G_0\vec{P}\mathtt{I}}\vec{G} = Y'\vec{G}.$$
It follows that $\vec{G}$ is not injective. ∎

(Notice that in the above proof both Y and Y' are closed, so even restricting injectivity hypothesis to closed terms, no non-trivial wfpc generator is injective.)

Corollary 3.6. *Suppose wfgv $\vec{G}$ fixes every fpc: $Y\vec{G} = Y$ for all fpc Y. Then $\vec{G}$ is trivial.*

An interesting consequence of these observations is that there is no uniform way to "Böhm out" an inner level of a wfpc.

Proposition 3.7. *For $m > 0$, it is not possible to find terms $(M_0, \dots, M_n)$ such that*

$$Z = (Z_n) \text{ wfpc} \quad \Longrightarrow \quad Z_0\vec{M} = Z_m. \tag{3.1}$$

Proof. Suppose such $\vec{M} = (M_0, \dots, M_n)$ exists. Then $\vec{M}$ is a wfgv. For every fpc Y, we have $Y\vec{M} = Y$, so every fpc is fixed by $\vec{M}$. (In particular, $\vec{M}$ is a fgv.) By Corollary 3.6, $\vec{M}$ is trivial: $\vec{M} = ()$. But then $\vec{M}$ fixes every wfpc as well, and thus cannot satisfy the hypothesis in (3.1). □

4. The four main classes of generators

Let us consider again Statman's conjecture on the non-existence of fixed points of the generator $\vec{G} = (\delta)$. This conjecture is intuitively compelling, because applying any fpc Y to δ leads to a slowdown of head reductions that seems impossible to remove:

$$\begin{aligned} Yx &\twoheadrightarrow_w x(Y') && =_\beta x(Yx) \\ Y\delta x &\twoheadrightarrow_w \delta(Y'[x := \delta])x \twoheadrightarrow_w x(Y'[x := \delta]x) && =_\beta x(Y\delta x) \end{aligned}$$

Upon closer inspection, the central property of the generator (δ) that this reasoning depends on is that $\delta(Y\delta)x$ *adds to* the reduction length needed for the Böhm tree to develop, while still using the given fpc Y in constructing this Böhm tree infinitely often. Since any conversion between the two will synchronize their Böhm reductions, no such conversion can be possible.

If this reasoning proves to be correct for δ, it should remain valid for any generator possessing the same property. This leads us to the following definition and conjecture.

Recall, for any wfpc (Y_n), $Y_0 = \lambda x.x^k(Y_k x) = \delta^k(Y_k) = \delta^k(z)[z := Y_k]$.

Definition 4.1. A generator $\vec{G}$ is *accretive* if, for each k, $z \in^\infty \delta^k(z)\vec{G}$.

That is, $\vec{G}$ is accretive if it actually uses every level of the input fpc in constructing the output fpc, so that replacing Y with any approximant will also cut the Böhm tree of $Y\vec{G}$.

Conjecture 4.2. If $\vec{G}$ is accretive, then there exists no $Y \in \mathsf{WFPC}$ such that $Y = Y\vec{G}$.

Remark 4.3. Conjecture 4.2 generalizes Statman's conjecture. Indeed,

$$\delta^k(z)\delta = (\lambda x.x^k(zx))\delta = \delta^k(z\delta) = \lambda x.x^k(z\delta x)$$

Thus, $z \in^\infty \delta^k(z)\delta$ for all k, and the fgv $\vec{G} = (\delta)$ is accretive.

Conjecture 4.2 is as sharp as possible: later in this section, we will show that every $\vec{G}$ which is *not* accretive possesses a fixed point among the wfpcs.

The non-accretive generators can be naturally divided into several classes given below. (We eschew the (W)FPC notation in the next definition to emphasize that there are actually four properties of generators that are being defined.)

From the earlier equality $Y_0 = \delta^k(Y_k)$, note that for each (w)fpc Y, $k \geq 0$, we can write

$$Y\vec{G} = \delta^k(Y')\vec{G} = G_0^k(Y'G_0)G_1 \cdots G_n.$$

Definition 4.4. Let $\vec{G}$ be a wfgv. Fix $z\#\vec{G}$.

- $\vec{G}$ is *constant* if there is a k such that $z \notin G_0^k(z)G_1\cdots G_n$.
- $\vec{G}$ is *weakly constant* if there is a k s.t. $z \notin^\infty G_0^k(z)G_1\cdots G_n$.
- $\vec{G}$ is *compact* if there is a k such that $G_0^k(z)G_1\cdots G_n \in \mathsf{FPC}$.
- $\vec{G}$ is *weakly compact* if there is a k s.t. $G_0^k(z)G_1\cdots G_n \in \mathsf{WFPC}$.

The least k satisfying one of these conditions is then called the *modulus of constancy*, or *modulus of compactness*, accordingly. Note that $\vec{G}$ is accretive iff $\vec{G}$ is not weakly compact.

From now on, let $\vec{G}$ be a possibly weak fgv. We will omit freshness conditions $x\#Y$, $z\#\vec{G}$ etc., as they will always be obvious from the context.

Proposition 4.5. *Let $\vec{G}$ be a constant (w)fgv. There is a term Z such that*

$$Y\vec{G} = Z$$

for all wfpc Y. Hence Z is (w)fpc.

Proof. Let $\vec{G}$ be constant, and let k be such that $z \notin G_0^k(z)G_1\cdots G_n$.

That is, $z \notin \mathsf{FV}(Z)$ for some $Z \in \Lambda$ convertible to $G_0^k(z)G_1\cdots G_n$.

Then for any wfpc $Y = (Y_0, Y_1, \dots)$, we have

$$\begin{aligned} Y\vec{G} = Y_0\vec{G} &= Y_0G_0\cdots G_n \\ &= G_0(Y_1G_0)G_1\cdots G_n \\ &= \quad \vdots \\ &= G_0^k(Y_kG_0)G_1\cdots G_n \\ &= G_0^k(z)G_1\cdots G_n[z := Y_kG_0] \\ &= Z[z := Y_kG_0] \\ &= Z \end{aligned}$$

□

Proposition 4.6. *The following observations are immediate.*

(1) *Every constant fgv is compact.*
(2) *Every constant (w)fgv is weakly constant.*
(3) *Every compact (w)fgv is weakly compact.*

Proposition 4.7. *$\vec{G}$ is weakly constant iff $\vec{G}$ is weakly compact.*

Proof. Let $\vec{G}$ be a wfgv. Then

$$z \notin^\infty G_0^k(z)G_1\cdots G_n \implies G_0^k(z)G_1\cdots G_n =^\infty G_0^k(\Theta G_0)G_1\cdots G_n = \Theta\vec{G} \in \mathsf{WFPC},$$

$$G_0^k(z)G_1\cdots G_n \in \mathsf{WFPC} \implies G_0^k(z)G_1\cdots G_n =^\infty \Theta \implies z \notin^\infty G_0^k(z)G_1\cdots G_n.$$

□

Proposition 4.8. *Every (weakly) compact generator has a fixed point:*

(1) *If $\vec{G}$ is compact fgv, there exists fpc Y with $Y\vec{G} = Y$.*
(2) *If $\vec{G}$ is weakly compact wfgv, there exists wfpc Y with $Y\vec{G} = Y$.*

Proof. The construction is the same for both claims. We will first treat the weak case, and then specialize the proof to the first claim as well.

Let $\vec{G}$ be a weakly compact wfgv. Let k be the modulus of weak compactness, so that $G_0^k(z)G_1\cdots G_n \in \mathsf{WFPC}$. Put $F_0[z] := G_0^k(z)G_1\cdots G_n$. Since $F_0[z]$ is wfpc, write

$$F_0[z]x = x(F_1[z]x) = x^2(F_2[z]x) = \cdots = x^k(F_k[z]x) = \cdots \tag{4.1}$$

Define $Y := \Theta(\lambda y.F_k[yG_0]) = F_k[YG_0]$, and $X := F_0[YG_0]$. Now compute

$$\begin{aligned} X\vec{G} &= XG_0 \cdots G_n \\ &= F_0[YG_0]G_0 \cdots G_n && \text{by definition of } X \\ &= G_0^k(F_k[YG_0]G_0)G_1 \cdots G_n && \text{by (4.1) with } [x := G_0, z := YG_0] \\ &= G_0^k(YG_0)G_1 \cdots G_n && \text{by definition of } Y \\ &= F_0[YG_0] && \text{by definition of } F_0 \\ &= X && \text{by definition of } X \end{aligned}$$

Since $F_0[z]$ is wfpc, so is $X = F_0[YG_0]$, proving the second claim.

Now suppose that $\vec{G}$ was actually compact fgv. Then $F_0[z]$ would be fpc, while all of the steps above would remain valid, with $F_0[z] = F_k[z]$ for each k. Then $X = F_0[YG_0]$ would be fpc as well, proving the first claim. □

Remark 4.9. The reader will recognize the converse to Proposition 4.8(2) as the contrapositive of Conjecture 4.2. The converse to Proposition 4.8(1) seems plausible, but we do not have sufficient evidence to assert it as a formal conjecture.

5. Rectifying generators

Definition 5.1. A vector $\vec{G}$ is *rectifying* if it satisfies condition (iv) of Proposition 2.9:

$$Y \in \mathsf{WFPC} \implies Y\vec{G} \in \mathsf{FPC}$$

Example 5.2. $\vec{G} = (\lambda y.\Theta_y)$ is rectifying:

$$(Y_n) \in \mathsf{WFPC} \implies Y_0(\lambda y.\Theta_y) = (\lambda y.\Theta_y)(Y_1(\lambda y.\Theta_y)) = \Theta_{Y_1(\lambda y.\Theta_y)} \in \mathsf{FPC}$$

In Example 3.3, we saw that $(\lambda y.\Theta_y)$ has a fpc fixed point. We shall presently see that so does every rectifying fgv.

Our original proof of this fact first showed that if $\vec{G}$ is rectifying, then $\vec{G}$ is weakly constant, and thus has a wfpc fixed point Y. But then $Y = Y\vec{G} \in \mathsf{FPC}$ because $\vec{G}$ is rectifying, hence Y is fpc.

Considering that compactness provides another sufficient condition for existence of fpc fixed points, it was natural to wonder whether rectifying and compact fgvs are related. This led us to the following result.

Theorem 5.3. *A fgv $\vec{G}$ is compact iff it is rectifying.*

Proof. **(⇒):** Suppose $G_0^k(z)G_1 \cdots G_n \in \mathsf{FPC}$. Then

$$\begin{aligned} & Y = (Y_0, Y_1, \dots) \in \mathsf{WFPC} \\ \implies\; & Y\vec{G} = Y_0G_0 \cdots G_n = G_0^k(Y_kG_0)G_1 \cdots G_n = G_0^k(z)G_1 \cdots G_n[z := Y_kG_0] \in \mathsf{FPC}. \end{aligned}$$

(⇐): The intuition for this direction is that, although the Böhm tree of a wfpc Y is infinite, only a finite part of it can be used in any conversion $\rho : Y\vec{G}x = x(Y\vec{G}x)$. Thus, writing $Yx = x^k(Y_kx) = \delta^k(Y_k)x$ for large enough k will ensure that Y_k is not touched by any redex contractions. Then the whole conversion ρ could be lifted to $\rho = \sigma[z := Y_k]$, where

$$\sigma : \delta^k(z)\vec{G}x = x(\delta^k(z)\vec{G}x).$$

To formalize this intuition, suppose $\vec{G}$ is rectifying. Fix $c\#\vec{G}$. Recall the wfpc Υ from Examples 3.3:

$$\begin{aligned} W_{x,p} &= \lambda v.x(v(cp)v) \\ V_x &= \lambda p.W_{x,p} \\ \Upsilon &= \lambda x.V_x \mathtt{I} V_x \end{aligned}$$

That is, $V_x = \lambda pv.x(v(cp)v)$. Note that $W_{x,p}$ and V_x are normal forms. Let $\Upsilon_x^k = V_x c^k(\mathtt{I})V_x$. The term Υx reduces as follows:

$$\begin{aligned} \Upsilon x \to \Upsilon_x^0 \equiv V_x \mathtt{I} V_x \to W_{x,\mathtt{I}} V_x &\to x(V_x(c\mathtt{I})V_x) \equiv x(\Upsilon_x^1) \\ &\to x(W_{x,c\mathtt{I}} V_x) \to x(x(V_x c^2(\mathtt{I})V_x)) \equiv x^2(\Upsilon_x^2) \\ &\to \cdots \\ &\to x^k(\Upsilon_x^k) \\ &\to \cdots \end{aligned}$$

Since each term appearing in the above reduction sequence has a unique redex, the reduction is completely deterministic. That is — the above sequence actually comprises the entire reduction graph of Υ_x^0. The sequence also shows that Υ is a wfpc. It is not a fpc however, since Υ_x^0 obviously has no reducts in common with Υ_x^1. But $\vec{G}$ is rectifying, so $\Upsilon\vec{G}$ is fpc. By the Church–Rosser theorem, let X be a common reduct

$$\Upsilon\vec{G}x \twoheadrightarrow X \twoheadleftarrow x(\Upsilon\vec{G}x). \tag{5.1}$$

We will use these reductions to show that $\delta^k(z)\vec{G} \in \mathsf{FPC}$ for large enough k.

The main idea behind the construction is as follows. Any finite reduction $\Upsilon M \twoheadrightarrow X$ can be continued until all the descendants of Υ project the same number of steps from Υ_M^0 in the sequence above. Afterwards, all descendants of M can be further synchronized by confluence.

For example, if $M = \lambda y.[y\mathsf{K}, y]$, then

$$\begin{aligned} \Upsilon(\mathtt{I}M) \twoheadrightarrow (\mathtt{I}M)(\Upsilon_{\mathtt{I}M}^1) &\twoheadrightarrow (\lambda y.[y\mathsf{K}, y])(V_{\mathtt{I}M} c^1(\mathtt{I})V_{\mathtt{I}M}) \\ &\twoheadrightarrow [V_{\mathtt{I}M} c^1(\mathtt{I})V_{\mathtt{I}M}\mathsf{K}, V_{\mathtt{I}M} c^1(\mathtt{I})V_{\mathtt{I}M}] \\ &\twoheadrightarrow [W_{\mathtt{I}M,c^1(\mathtt{I})} V_{\mathtt{I}M}\mathsf{K}, \mathtt{I}M(V_{\mathtt{I}M} c^2(\mathtt{I})V_M)] \end{aligned}$$

and this reduction can be further continued to

$$\begin{aligned} [W_{\mathtt{I}M,c^1(\mathtt{I})} V_{\mathtt{I}M}\mathsf{K}, \mathtt{I}M(V_{\mathtt{I}M} c^2(\mathtt{I})V_M)] &\twoheadrightarrow [M(V_M c^2(\mathtt{I})V_M)\mathsf{K},\ M(V_M c^2(\mathtt{I})V_M)] \\ &\equiv\ [M(\Upsilon_M^2), M(\Upsilon_M^2)]. \end{aligned}$$

We proceed with the following sequence of claims, which are hopefully sufficiently clear not to warrant additional elaboration.

(1) If $\Upsilon M \twoheadrightarrow X$, then $X \twoheadrightarrow X' \equiv C[V_{M_1} c^{k_1}(\mathtt{I})V_{M_1'}, \cdots, V_{M_k} c^{k_m}(\mathtt{I})V_{M_m'}]$, with $M \twoheadrightarrow M_i$, $M \twoheadrightarrow M_i'$, and every occurrence of c in X' being displayed in the subterm $c^{k_i}(\mathtt{I})$ in one of the holes in $C[]$.

(2) If $\Upsilon M \twoheadrightarrow X$, then $X \twoheadrightarrow X' \equiv C[\Upsilon_{M_1}^{k_1}, \ldots, \Upsilon_{M_m}^{k_m}]$, with $M \twoheadrightarrow M_i$ and every occurrence of c being uniquely determined by its occurrence in some $\Upsilon_{M_i}^{k_i}$. This is obtained from above by finding a common reduct for each M_i,M_i'.

(3) If $C[\Upsilon M] \twoheadrightarrow X$, then $X \twoheadrightarrow C'[\Upsilon^{k_1}_{M_1}, \cdots, \Upsilon^{k_m}_{M_m}]$, with the same conditions on M_i and occurrences of c as before. This is obtained from the previous point by factoring the reduction into a part that does not depend on ΥM, $C[x] \twoheadrightarrow D[x\vec{P}_1, \dots, x\vec{P}_k]$, followed by reductions inside $D[\cdots]$ which are treated separately via 2. (See "Barendregt's Lemma" in [1, Exercise 15.4.8])
(4) If $C[\Upsilon M] \twoheadrightarrow X$, then $X \twoheadrightarrow C'[\Upsilon^k_N, \cdots, \Upsilon^k_N]$, where $M \twoheadrightarrow N$ and each occurrence of c being uniquely determined by its occurrence in some Υ^k_N. This is obtained from the previous claim by "bumping all Υ^{k_i}s along" to stage $k \geq \max\{k_i\}$, and letting N be a common reduct of all the M_is.
(5) If the reduction $\rho : C[\Upsilon M] \twoheadrightarrow C'[\Upsilon^k_N, \cdots, \Upsilon^k_N]$ is obtained by the algorithm given in the previous steps, then ρ lifts to the instance $\rho = \sigma[ux := \Upsilon^k_x]$, where
$$\sigma : C[\delta^k(u)M] \twoheadrightarrow C'[uN, \cdots, uN].$$

And now we are done! The common reductions in (5.1) can be both continued to
$$\Upsilon\vec{G}x \twoheadrightarrow C'[\Upsilon^k_N, \cdots, \Upsilon^k_N] \twoheadleftarrow x(\Upsilon\vec{G}x)$$
so that all of the descendants of Υ (under *both* reductions) are displayed in the context. (This follows from the fact that every occurrence of c is witnessed in some Υ^k_N, and c was chosen to be fresh. The variable c acts as a "label" for the unfolding depth of Υ.)

The conclusion of the last step therefore holds for both of these reductions, and so
$$C[\delta^k(u)M] \coloneqq [\delta^k(u)G_0]G_1 \cdots G_n \twoheadrightarrow C'[uN, \dots, uN] \twoheadleftarrow x(\delta^k(u)G_0 \cdots G_n).$$
That is, $G^k_0(uG_0)G_1 \cdots G_n \in \mathsf{FPC}$. Applying the substitution $[u \coloneqq \mathtt{K}u]$, we find that $G^k_0(u)G_1 \cdots G_n \in \mathsf{FPC}$ as well. □

Corollary 5.4. *Every rectifying fgv has a fixed point in* FPC.

Remark 5.5. The proof of the nontrivial direction of Theorem 5.3 suggests a deeper connection between uniform properties (finite conversions) and terms obeying a coinductive pattern (such as wfpcs). In the next section, we will see a different application of the same type of argument. There seems to be a more general "continuity principle" at work here that could be worthwhile to isolate.

We finish this section with an example of a weakly constant fgv which is not rectifying. It follows that, even restricting to fgvs, compactness is indeed stronger than weak compactness.

This gives the full picture of the relationships between various classes of wfgvs we defined here. These relationships are summarized in Figure 1.

Proposition 5.6. *There exist weakly constant fpc generators which are not rectifying.*

Proof. Consider the following combinators:
$$\begin{aligned} Pxy &= yx \\ Qyz &= z(yQz) \\ Wwpz &= z(ww(zp)z) \\ Ryz &= WW(yQz)z \end{aligned}$$

First we observe that (P, Q) is an fgv: for Y fpc, we have
$$YPQx = P(YP)Qx = Q(YP)x = x(YPQx). \tag{5.2}$$

We claim that (P, Q) is not rectifying. If it was, then by the previous theorem, it would be compact, hence weakly compact. To the contrary, (5.2) shows that (P, Q) is accretive. So it cannot be rectifying.

Next, we verify that (P, R) is again fgv:

$$\begin{aligned} YPRx = P(YP)Rx = R(YP)x &= WW(YPQx)\,x \\ &= x(WW(x(YPQx))\,x) \\ &=_{(5.2)} x(WW(YPQx)\,x) = x(YPRx) \end{aligned}$$

At the same time, we claim (P, R) is weakly constant with modulus 1:

$$\begin{aligned} P^1(z)Rx = PzRx = Rzx &= WW(zQx)x \\ &= x(WW(\cdots)x) \\ &= x^2(WW(\cdots)x) \\ &= \cdots \\ &= x^n(\cdots) \end{aligned}$$

The variable z is being pushed to infinity, and does not appear in the Böhm tree of $PzRx$—nor in the Böhm tree of $PzR = \lambda x.PzRx$. That is, $z \notin^\infty P^1(z)R$. Indeed, $\vec{G} = (P, R)$ is weakly constant. We claim it is not rectifying.

For a wfpc Z, the term ZPR reduces as follows:

$$\begin{aligned} ZPRx \twoheadrightarrow P(ZP)Rx \to^2 R(ZP)x &\to^2 WW(ZPQx)x \\ &\to^3 x(WW(x(ZPQx))x) \\ &\to^3 x^2(WW(x^2(ZPQx))x) \\ &\to \quad \vdots \\ ZPRx \quad &\twoheadrightarrow^{z_0+2+2+3n} x^n(WW(x^n(ZPQx))x) \end{aligned}$$

From this analysis, it is manifest that any common reduction

$$ZPRx \quad \twoheadrightarrow \quad \cdot \quad \twoheadleftarrow \quad x(ZPRx)$$

must contain a common reduction between

$$x^n(ZPQx) \quad \twoheadrightarrow \quad \cdot \quad \twoheadleftarrow \quad x^{n+1}(ZPQx).$$

As we observed earlier, (P, Q) is not rectifying, so there exist wfpcs Z for which such conversion is not possible. Thus (P, R) is not rectifying either. This completes the proof.

(Note that the modulus of constancy can be adjusted to any $k > 0$ by passing the argument of the generator into the head position k times before pushing it to infinity.) □

6. The monoid of wfgvs

The wfpc and fpc generators have an obvious monoid structure:

$$(G_0, \ldots, G_n) \odot (G'_0, \ldots, G'_m) \quad := \quad (G_0, \ldots, G_n, G'_0, \ldots, G'_m)$$

The identity is the trivial generator $()$. The concatenation operation is associative, and satisfies the identity laws. We thus have a monoid $(\mathcal{G}, \odot, ())$ of wfgvs, containing a submonoid $(\mathcal{F}, \odot, ())$ of fgvs.

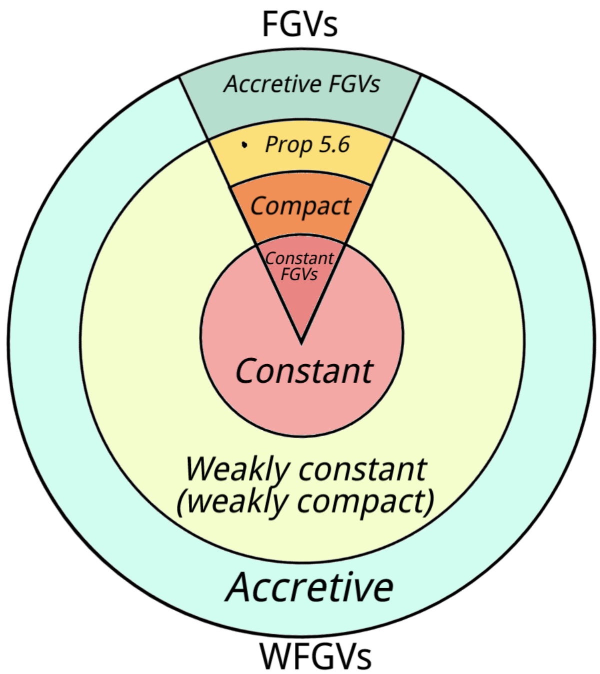

Figure 1: The hierarchy of (weak) fpc generators

Neither of these monoids is finitely generated, as there are infinitely many constant fgvs of the form $\mathtt{K}\Theta_M$ that cannot be obtained by composition of more elementary ones.

Extensional equality. Since the primary interest in (w)fgvs is in their ability to generate new (w)fpcs from old, it is natural to identify generators having the same functional behavior.

Definition 6.1. We say a fgv or wfgv $\vec{G}$ is *extensionally equal* to $\vec{G}'$, written $G \simeq G'$, if for every *fpc* Y, $Y\vec{G} = Y\vec{G}'$. (Note that this is an equivalence relation on $\mathcal{G}$, preserved by $\odot$.)

Examples 6.2. • If $\vec{G}$ is a constant generator, say, $Y\vec{G} = Z$ for all Y, then $\vec{G} \simeq (\mathtt{K}Z)$:

$$Y(\mathtt{K}Z) = \mathtt{K}Z(Y'(\mathtt{K}Z)) = Z = Y\vec{G}$$

• Recall the combinator $\mathtt{C} = \lambda fxy.fyx$. Note that $\mathtt{CK} = \mathtt{KI}$, and $\mathtt{C}(\mathtt{CK}) = \mathtt{C}(\mathtt{KI}) = \mathtt{K}$. Let $Gyz = z(y(\mathtt{C}z))(\delta(y(\mathtt{C}z)))$. Then $(G,\mathtt{K})$, and $(G,\mathtt{CK})$ are fgvs, and $(G,\mathtt{K}) \simeq (G,\mathtt{CK})$:

$$\begin{aligned} Y \in \mathsf{FPC} \quad \Longrightarrow \quad YG\mathtt{K} &= G(YG)\mathtt{K} = \mathtt{K}(YG(\mathtt{CK}))(\delta(YG(\mathtt{CK}))) = YG(\mathtt{CK}) \\ &= G(YG)(\mathtt{CK}) = (\mathtt{CK})(YG(\mathtt{C}(\mathtt{CK})))(\delta(YG(\mathtt{C}(\mathtt{CK})))) \\ &= (\mathtt{KI})(YG\mathtt{K})(\delta(YG\mathtt{K})) = \delta(YG\mathtt{K}) \in \mathsf{FPC} \end{aligned}$$

The reason that in the definition of $\simeq$ the quantifier ranges over fpcs both in the case of fgvs as well as wfgvs is that, when the quantifier is taken over all wfpcs, it makes the resulting notion of equality much more restrictive, as we shall now demonstrate.

Because we obviously want equal fgvs to remain equal as wfgvs, the definition of extensional equality is expressed in terms of behavior on fpcs in both contexts.

Proposition 6.3. *If $Y\vec{G} = Y\vec{G}'$ for every wfpc Y, then for some k, $\delta^k(z)\vec{G} = \delta^k(z)\vec{G}'$.*

Proof. This statement follows by the same reasoning as used in Theorem 5.3. Take $z\#\vec{G},\vec{G}'$, and let $\Upsilon = \Upsilon_z$ be the canonical wfpc defined there with a deterministic reduction graph that uses the variable z to track its unfolding history. The argument subsequently showed how every conversion $C[\Upsilon M] \twoheadrightarrow X \twoheadleftarrow C'[\Upsilon M]$ can be extended through $X \twoheadrightarrow X'$, such that $X' = D[\Upsilon^k_N,\cdots,\Upsilon^k_N]$, with $M \twoheadrightarrow N$ and every occurrence of z in X' to be found among the displayed Υ^k_N. We could then conclude that the common reduction may be lifted to a finite truncation of Υ. In the present case, our starting conversion has the form

$$C[\Upsilon G_0] \coloneqq [\Upsilon G_0]G_1\cdots G_n = [\Upsilon G'_0]G'_1\cdots G'_{n'} \eqqcolon C'[\Upsilon G'_0] \tag{6.1}$$

To justify application of the same argument, we should thus argue why $G_0 = G'_0$. Let X be a reduct of $C[\Upsilon G_0]$. By recalling the reduction graph of Υ, it is evident that every innermost occurrence of z in X is applied to a reduct of G_0. If X is also a reduct of $C'[\Upsilon G'_0]$, then the same conclusion will hold, with G'_0 in place of G_0. Thus, the very fact of occurrence of z in X forces G_0 and G'_0 to be convertible. Of course, if z does not occur in X at all, that only means that all descendants of Υ have already been erased, in which case there is nothing left to prove. So $G_0 = G'_0$. We can thus adjust conversion in (6.1) to

$$C[\Upsilon G_0] \equiv [\Upsilon G_0]G_1\cdots G_n = [\Upsilon G_0]G'_1\cdots G'_{n'} \equiv C'[\Upsilon G_0] = C'[\Upsilon G'_0]$$

where the conversion on the right takes place inside the subterm that Υ is applied to.

Now we extend the other conversion to a common reduct

$$[\Upsilon G_0]G_1\cdots G_n \twoheadrightarrow D[\Upsilon^k_N,\cdots,\Upsilon^k_N] \twoheadleftarrow [\Upsilon G_0]G'_1\cdots G'_{n'}$$

and proceed, as in the proof of Theorem 5.3, to lift these reductions to

$$[\delta^k(u)G_0]G_1\cdots G_n \twoheadrightarrow D[uN,\ldots,uN] \twoheadleftarrow [\delta^k(u)G_0]G'_1\cdots G'_{n'}.$$

Converting G_0 in the right term to G'_0, we obtain the desired result. □

From now on, we will consider the monoid $\mathcal{G}$ up to extensional equality. We will also write concatenation of vectors by juxtaposition: $\vec{F}\vec{G} = \vec{F} \odot \vec{G}$.

Ideals.

Definition 6.4. A *two-sided ideal* in a monoid $(M,\cdot)$ is a set $I \subseteq M$ such that

$$i \in I,\, m \in M \quad\Longrightarrow\quad i\cdot m,\, m\cdot i \,\in I.$$

Proposition 6.5.

(1) *The constant generators form the minimal ideal in both monoids.*
(2) *The weakly constant/weakly compact wfgvs form a two-sided ideal in $\mathcal{G}$.*
(3) *The compact fgvs form a two-sided ideal in $\mathcal{F}$ (and a right ideal in $\mathcal{G}$).*

Proof. (1) Let $\vec{G}$ be constant, so that $Y\vec{G} = Z$ for all (w)fpc Y. Let $\vec{G}'$ be arbitrary. Then $Y\vec{G}\vec{G}' = Z\vec{G}'$ for all Y, and $Y\vec{G}'\vec{G} = Z$ for all Y. Thus $\vec{G}\vec{G}'$ and $\vec{G}'\vec{G}$ are constant. Moreover, since any ideal includes the constant generators by composition on the right, these generators together constitute the minimal ideal.

(2) Let $\vec{G} \in \mathcal{G}$ be weakly constant, so that $z \notin^\infty G^k_0(z)G_1\cdots G_n$. Let $\vec{G}' \in \mathcal{G}$ be arbitrary. Clearly, $z \notin^\infty G^k_0(z)G_1\cdots G_nG'_0\cdots G'_m$. That is, $\vec{G}\vec{G}'$ is weakly constant. On the other hand, we know that $\vec{G}'$ maps wfpcs to themselves:

$$(\lambda y.y\vec{G}') : \mathsf{WFPC} \to \mathsf{WFPC}$$

All wfpcs have the same Böhm tree, and in the tree topology, its neighborhood basis consists of the set $\{\lambda x.x^n(\Omega) \mid n \geq 0\}$. By Continuity Theorem [1, 14.3.22], there exists $l \geq 0$ such that

$$(\lambda x.x^l(z))\vec{G}' = (\lambda y.y\vec{G}')(\lambda x.x^l(z)) = \lambda x.x^k(X)$$

for some X, possibly containing z. And yet,

$$z \notin^\infty G_0^k(X)G_1\cdots G_n = (\lambda x.x^k(X))\vec{G} = (\lambda x.x^l(z))\vec{G}'\vec{G} = (G_0')^l(z)G_1'\cdots G_m'\vec{G}.$$

Indeed, $\vec{G}'\vec{G}$ is weakly constant.

(3) It is immediate that the rectifying fgvs form a two-sided ideal in $\mathcal{F}$. By Theorem 5.3, so do the compact ones. Also, for $\vec{G}, \vec{G}' \in \mathcal{G}$, $\vec{G}'$ rectifying clealry implies $\vec{G}\vec{G}'$ rectifying. □

Green's relations. The structure of many monoids can be characterized in terms of Green's relations. Here we record several observations about these relations in $\mathcal{G}$, which could be useful for future study of this monoid.

Definition 6.6. For $\vec{G}, \vec{G}' \in \mathcal{G}$, put

$$\begin{array}{rcl}
\mathcal{L}(\vec{G}) & = & \{\vec{H}\vec{G} \mid \vec{H} \in \mathcal{G}\} \\
\mathcal{R}(\vec{G}) & = & \{\vec{G}\vec{H} \mid \vec{H} \in \mathcal{G}\} \\
\vec{G} \preccurlyeq_{\mathcal{L}} \vec{G}' & \iff & \mathcal{L}(\vec{G}) \subseteq \mathcal{L}(\vec{G}') \iff \vec{G} \in \mathcal{L}(\vec{G}') \\
\vec{G} \preccurlyeq_{\mathcal{R}} \vec{G}' & \iff & \mathcal{R}(\vec{G}) \subseteq \mathcal{R}(\vec{G}') \iff \vec{G} \in \mathcal{R}(\vec{G}') \\
\vec{G} \sim_{\mathcal{L}} \vec{G}' & \iff & \mathcal{L}(\vec{G}) = \mathcal{L}(\vec{G}') \\
\vec{G} \sim_{\mathcal{R}} \vec{G}' & \iff & \mathcal{R}(\vec{G}) = \mathcal{R}(\vec{G}').
\end{array}$$

(1) If $\vec{G} \simeq (\mathtt{K}Z)$ is a constant generator, then $\mathcal{L}(\vec{G}) = \{\mathtt{K}Z\}$, so all constant generators are each in their own left class. That is, $\vec{G} \in \mathcal{L}(\mathtt{K}Z)$ implies $\vec{G} \simeq (\mathtt{K}Z)$.
(2) On the other hand, $(\mathtt{K}Z, \mathtt{K}Z') \simeq (\mathtt{K}Z')$, thus $\mathtt{K}Z' \in \mathcal{R}(\mathtt{K}Z)$. Since the choice of Z, Z' was arbitrary, $\mathtt{K}Z \sim_{\mathcal{R}} \mathtt{K}Z'$ for all Z and Z'. That is, constant generators are all in the same right class. Since constant generators form an ideal, $\vec{G} \sim_{\mathcal{R}} (\mathtt{K}Z)$ or $\vec{G} \preccurlyeq_{\mathcal{R}} (\mathtt{K}Z)$ imply $\vec{G} \simeq (\mathtt{K}Z')$ for some Z'. So $\mathcal{R}(\mathtt{K}Z) = \{(\mathtt{K}Z') \mid Z' \in \mathsf{WFPC}\}$ is the ideal of all constant generators.
(3) Similarly, if $\vec{G}$ is weakly constant, then so is every element of $\mathcal{L}(\vec{G})$ and $\mathcal{R}(\vec{G})$. That is, the only (w)fgvs that can be congruent to $\vec{G}$ modulo $\sim_{\mathcal{L}}$ or $\sim_{\mathcal{R}}$ are again weakly constant.
(4) If $\vec{G}$ is compact, then so is every element of $\mathcal{L}(\vec{G})$. When restricted to $\mathcal{F}$, both $\mathcal{L}(\vec{G})$ and $\mathcal{R}(\vec{G})$ consist of compact generators.
(5) Suppose $\vec{G} \sim_{\mathcal{R}} \vec{G}'$. Then we can find $\vec{F}, \vec{F}' \in \mathcal{G}$ such that $\vec{G} \simeq \vec{G}'\vec{F}$, and $\vec{G}' \simeq \vec{G}\vec{F}'$. But then $\vec{G} \simeq \vec{G}\vec{F}'\vec{F}$, and $\vec{G}' \simeq \vec{G}'\vec{F}\vec{F}'$. If $\vec{G} \simeq \vec{G}\vec{F}'\vec{F}$, then for every Y, $Y\vec{G} = Y\vec{G}\vec{F}'\vec{F}$ is a fixed point of $\vec{F}'\vec{F}$. By Conjecture 4.2, $\vec{F}'\vec{F}$ is weakly constant. By Proposition 6.5, so is $\vec{G}\vec{F}'\vec{F}$. But $\vec{G}\vec{F}'\vec{F} \simeq \vec{G}$. So $\vec{G}$ is weakly constant. Of course, everything we just said applies to $\vec{G}'$ as well. We conclude that, modulo Conjecture 4.2, nontrivial $\sim_{\mathcal{R}}$-relations can only exist between weakly constant wfgvs.

In fact, we believe that for accretive $\vec{G}, \vec{G}'$, either $\vec{G} \sim_{\mathcal{R}} \vec{G}'$ or $\vec{G} \sim_{\mathcal{L}} \vec{G}'$ implies $\vec{G} \simeq \vec{G}'$. Indeed, this is a consequence of the "freeness" conjecture we formulate next.

Unique factorization of accretive generators.

Definition 6.7. An accretive generator $\vec{G}$ is *prime* if $\vec{G} = \vec{G}_1\vec{G}_2$ implies $\{\vec{G}_1, \vec{G}_2\} = \{\vec{G}, ()\}$.

The following "unique factorization conjecture" states that the accretive generators are freely generated by the prime ones.

Conjecture 6.8. If $\vec{G} \in \mathcal{G}$ is accretive, then $\vec{G} = \vec{G}_1 \cdots \vec{G}_k$, where each $\vec{G}_i$ is prime.

Furthermore, this decomposition is unique up to extensional equality. That is, for all prime $\vec{G}'_1, \dots, \vec{G}'_{k'}$, if $\vec{G} = \vec{G}'_1 \cdots \vec{G}'_{k'}$, then $k = k'$ and $\vec{G}_i \simeq \vec{G}'_i$ for all $i \in \{1, \dots, k\}$.

The conjecture implies that any relations in the monoid of wfpc generators under composition can only arise between non-accretive i.e., weakly compact generators. In particular, the equations

$$\vec{F}\vec{G} = \vec{F} \tag{6.2}$$

$$\vec{F}\vec{G} = \vec{G} \tag{6.3}$$

admit no (non-trivial) solutions among the accretive generators. As a result, the left and right classes of accretive generators would indeed all be distinct if Conjecture 6.8 was to be validated.

And what about the solutions to (6.2) or (6.3) among the rest of $\mathcal{G}$? The following examples show that there indeed exist solutions to these equations under certain conditions. In all cases, (weak) compactness plays an essential role.

Proposition 6.9. *For $\vec{G}$ weakly constant, there exists non-constant $\vec{F}$ with $\vec{F}\vec{G} \simeq \vec{F}$. (In particular, $\vec{F} \preccurlyeq_{\mathcal{L}} \vec{G}$.)*

Proof. The idea is to make $\vec{F}$ generate the fixed points of $\vec{G}$ according to the scheme in Proposition 4.8. Let k, F_0, F_k be chosen as in the proof of that proposition. Put $A = \lambda yb.b(y\delta)$, $B = \lambda y.F_0[y(\lambda u.F_k[uG_0])G_0]$, and $\vec{F} = (A, B)$. Observe that

$$\begin{aligned} Y \in \mathsf{FPC} \quad \Longrightarrow \quad & YA\delta = A(YA)\delta = \delta(YA\delta) \\ & YAB = A(YA)B = B(YA\delta) = F_0[(YA\delta)(\lambda u.F_k[uG_0])G_0]. \end{aligned}$$

Since $YA\delta$ is thereby forced to be fpc, it follows that $YAB = F_0[UG_0]$, where $U = F_k[UG_0]$. This allows us to calculate as in the proof of Proposition 4.8 that YAB is a fixed point of $\vec{G}$:

$$Y\vec{F}\vec{G} = YAB\vec{G} = YAB$$

Note however, that $\vec{F}$ will not be constant in general, because it uses its fpc argument to define U. $\square$

Proposition 6.10. *Let $\vec{F} = (F_0, \cdots, F_n)$ be wfgv with $n \geq 1$. There exists a compact fgv $\vec{G}$ such that $\vec{F}\vec{G} \simeq \vec{G}$. (In particular, $\vec{G} \preccurlyeq_{\mathcal{R}} \vec{F}$.)*

Proof. First, recall that $F_0 = \lambda v_0..v_l.v_m\vec{P}$ is solvable. Since $Y\vec{F} = F_0(Y'F_0)F_1 \cdots F_n$, we also know that the head variable v_m cannot be v_0, for otherwise the result would be unsolvable, while it must be a wfpc.

We let $\vec{G} = (F_0, G_1, \cdots, G_{n+1})$. We will only need to specify a couple of G_is. Set $G_m = \lambda\vec{p}.\lambda g_{l+1}, ..., g_{n+1}.\Theta_{g_{n+1}(F_0 v_0 F_1 \cdots F_n)}$, $G_{n+1} y = \Theta(\lambda gy.g(y\vec{F})) = G_{n+1}(y\vec{F})$.

$$
\begin{aligned}
Y\vec{G} &= G_0(Y'G_0)G_1 \cdots G_{n+1} \\
&= F_0(Y'F_0)G_1 \cdots G_{n+1} \\
&= G_m\vec{P}[v_0 := Y'F_0][v_i := G_i]_{1\le i\le l} G_{l+1} \cdots G_{n+1} \\
&= \Theta_{G_{n+1}(F_0(Y'F_0)F_1 \cdots F_n)} \\
&= \Theta_{G_{n+1}(YF_0 \cdots F_n)} \\
&= \Theta_{G_{n+1}(Y\vec{F})} \qquad (\star) \\
&= \Theta_{G_{n+1}(Y\vec{F}\vec{F})} \\
&= \Theta_{G_{n+1}((Y\vec{F})\vec{F})} \\
&= (Y\vec{F})\vec{G} \quad \text{by } (\star), \text{ with } Y := Y\vec{F} \qquad \square
\end{aligned}
$$

Our final observation is a corollary to one of the first ones.

Proposition 6.11. *The monoid $\mathcal{G}$ is* zerosum-free*: If $\vec{F}\vec{G} \simeq ()$, then $\vec{F} \simeq () \simeq \vec{G}$.*

Proof. Suppose $\vec{F}\vec{G} \simeq ()$. Then, considered as endofunctions on $\mathsf{WFPC}/{=_\beta}$, $\vec{G}$ acts as a left inverse of $\vec{F}$, making $\vec{F}$ a split mono (modulo beta). But we have seen in Proposition 3.5 that no wfgv is injective, so no wfgv can be monic. Specifically, take $Y \neq Y'$ such that $Y\vec{F} = Y'\vec{F}$. Since $() \simeq \vec{F}\vec{G}$, we have $Y = Y\vec{F}\vec{G} = Y'\vec{F}\vec{G} = Y'$, a contradiction. $\square$

7. Concluding remarks

In this paper,we have broached the topic of abstract fpc generators. Our first investigations revealed that these operators naturally fall into a few robust classes. We established elementary relationships between these classes.

What becomes clear from our investigations is that there is yet much to be uncovered about the structure of fixed point combinators. Some of the possible future research directions include the following.

(1) The most pressing issue is the status of Conjecture 4.2. All the evidence available points to this conjecture being true, yet current techniques in untyped lambda calculus decidedly come up short in settling the question. However it will be decided, the insights to be gathered from the new approaches will greatly deepen our understanding of lambda terms.
(2) Of course, one could take the next step and ask whether the converse to the first claim in Proposition 4.8 is also valid. Considering how difficult the former question is, this one will likely remain out of reach for the foreseeable future.
(3) What else can be said about the structure of the monoid $\mathcal{G}$? Is the submonoid of accretive generators freely generated by the prime generators, as Conjecture 6.8 asserts? What about the (weakly) compact generators? How can their compositional structure be characterized?
(4) Since the monoid of (w)fgvs naturally acts on the set of (w)fpcs, how much of the structure of fpcs is captured by this monoidal action? Does every fpc have a representation in terms of the prime elements of the monoid — again, modulo extensional equality, and the ideal of compact generators?

(5) Finally, while not directly relevant to the earier discussion, an answer to the following question could also shed light on recursion-theoretic properties of FPCs:

Let $\mathtt{Y}$ be Curry's simplest fpc. Is $\{\#M \mid M = \mathtt{Y}\}$ a decidable subset of FPC? Specifically, does there exist a term $\Delta_{\mathtt{Y}}$ satisfying, for all *closed* $Y \in \mathsf{FPC}$, the following:

$$\Delta_{\mathtt{Y}} \ulcorner Y \urcorner xy = \begin{cases} x & \mathtt{Y} =_\beta Y \\ y & \mathtt{Y} \neq_\beta Y \end{cases}$$

Notice that Scott's theorem does not apply here because FPC is not all of Λ, but is only a computably enumerable subset of it. $\Delta_{\mathtt{Y}}$ is allowed to diverge outside of this set.

The recent paper [12] proposes another approach to Statman's conjecture based on simple types. We note that the generalization of the conjecture stated there is consistent with ours, since every simply-typed generator is accretive thanks to strong normalization of ($\mathtt{Y}$-free) typed terms.

I would like to thank Jan Willem Klop, Joerg Endrullis, Dimitri Hendriks, Giulio Manzonetto, and Stefano Guerrini for wonderful discussions about fpc generators.

I would also like to extend my gratitude to the anonymous referees whose many suggestions have significantly improved this paper.

References

[1] H. P. Barendregt. *The lambda calculus, its syntax and semantics*. Number 103 in Studies in Logic and the Foundations of Mathematics. North-Holland, second edition, 1984.

[2] H. P. Barendregt. Lambda calculi with types. In S. Abramsky, Dov M. Gabbay, and S. E. Maibaum, editors, *Handbook of Logic in Computer Science (Vol. 2)*, pages 117–309. Oxford University Press, Inc., New York, NY, USA, 1992.

[3] L. Czajka. A new coinductive confluence proof for infinitary lambda calculus. *Logical Methods in Computer Science*, 16(1), 2020.

[4] Thierry Coquand and Hugo Herbelin. A-translation and looping combinators in pure type systems. *Journal of Functional Programming*, 4(1):77–88, 1994.

[5] Haskell B. Curry. The inconsistency of certain formal logics. *Journal of Symbolic Logic*, 7(3):115–117, 1942.

[6] Jörg Endrullis, Dimitri Hendriks, Jan Willem Klop, and Andrew Polonsky. Discriminating lambda-terms using clocked Böhm trees. *Logical Methods in Computer Science*, 10(2), 2014.

[7] Jörg Endrullis and Andrew Polonsky. Infinitary rewriting coinductively. In Nils Anders Danielsson and Bengt Nordström, editors, *18th International Workshop on Types for Proofs and Programs, TYPES 2011, September 8-11, 2011, Bergen, Norway*, volume 19 of *LIPIcs*, pages 16–27. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2011.

[8] Herman Geuvers and Joep Verkoelen. On fixed point and looping combinators in type theory. `http://www.cs.ru.nl/~herman/PUBS/TLCApaper.pdf`.

[9] Douglas J. Howe. The computational behaviour of Girard's paradox. In *Proceedings of the Symposium on Logic in Computer Science (LICS '87), Ithaca, New York, USA, June 22-25, 1987*, pages 205–214. IEEE Computer Society, 1987.

[10] Benedetto Intrigila. Non-existent Statman's double fixedpoint combinator does not exist, indeed. *Inf. Comput.*, 137(1):35–40, 1997.

[11] Richard Kennaway, Jan Willem Klop, M. Ronan Sleep, and Fer-Jan de Vries. Infinitary lambda calculus. *Theor. Comput. Sci.*, 175(1):93–125, 1997.

[12] Giulio Manzonetto, Andrew Polonsky, Alexis Saurin, and Jakob Grue Simonsen. The Fixed Point Property and a Technique to Harness Double Fixed Point Combinators. *Journal of Language and Computation*, 29(5):831–880, 2019.

[13] Albert R. Meyer and Mark B. Reinhold. "type" is not a type. In *Conference Record of the Thirteenth Annual ACM Symposium on Principles of Programming Languages, St. Petersburg Beach, Florida, USA, January 1986*, pages 287–295. ACM Press, 1986.

[14] Dana S. Scott. Some philosophical issues concerning theories of combinators. In Corrado Böhm, editor, *Lambda-Calculus and Computer Science Theory, Proceedings of the Symposium Held in Rome, Italy, March 25-27, 1975*, volume 37 of *Lecture Notes in Computer Science*, pages 346–366. Springer, 1975.

[15] Richard Statman. Some examples of non-existent combinators. *Theoretical Computer Science*, 121(1&2):441–448, 1993.