Evolution of Algebraic Terms 3

David M. Clark

Mathematics Department: SUNY, New Paltz, NY, USA

May, 2018

Algebras and Lattices in Hawaii
Honolulu, HI

Problem: *Given a finite algebra* $\mathbf{G}$ *and a* k*-ary operation* $t : G^k \to G$ *that is known to be a term operation of* $\mathbf{G}$, *can we find a computationally efficient method to produce a term whose term operation is* t*?*

Problem: *Given a finite algebra* **G** *and a* k*-ary operation* $t : G^k \to G$ *that is known to be a term operation of* **G**, *can we find a computationally efficient method to produce a term whose term operation is* t*?*

$\mathbf{G} = \langle G; * \rangle$ – a finite groupoid

Problem: *Given a finite algebra* **G** *and a k-ary operation* $t : G^k \to G$ *that is known to be a term operation of* **G**, *can we find a computationally efficient method to produce a term whose term operation is* t*?*

$\mathbf{G} = \langle G; * \rangle$ – a finite groupoid

G is **term continuous** TC if its term to term operation is (relative to appropriate metrics) continuous.

1. Evolution of algebraic terms 1: term to term operation continuity, *International Journal of Algebra and Computation*, Vol. 23, No. 5 (2013) 1175-1205.

1. Evolution of algebraic terms 1: term to term operation continuity, *International Journal of Algebra and Computation*, Vol. 23, No. 5 (2013) 1175-1205.

2. Evolution of algebraic terms 2: the deep drilling algorithm, (with L. Spector and M. Keijzer), *International Journal of Algebra and Computation*, Vol. 26, No. 6 (2016) 1141- 1176.

1. Evolution of algebraic terms 1: term to term operation continuity, *International Journal of Algebra and Computation*, Vol. 23, No. 5 (2013) 1175-1205.

2. Evolution of algebraic terms 2: the deep drilling algorithm, (with L. Spector and M. Keijzer), *International Journal of Algebra and Computation*, Vol. 26, No. 6 (2016) 1141- 1176.

3. Evolution of algebraic terms 3: beam algorithms and term continuity, (with L. Spector), *International Journal of Algebra and Computation*, 32pp (in press).

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4 x_1)\Diamond)x_2^2)x_3).$$

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4 x_1)\Diamond)x_2^2)x_3).$$

If $t : G^k \to G$,

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4 x_1)\Diamond)x_2^2)x_3).$$

If $t : G^k \to G$, we seek a term $u(\vec{x})$ such that $f(\vec{x}, u(\vec{x}))^{\mathbf{G}} = t$,

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4 x_1)\Diamond)x_2^2)x_3).$$

If $t : G^k \to G$, we seek a term $u(\vec{x})$ such that $f(\vec{x}, u(\vec{x}))^{\mathbf{G}} = t$, i.e.,

$$f(\vec{d}, u(\vec{d})) = t(\vec{d}) \ \text{ for all } \vec{d} \in G^k.$$

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4 x_1)\Diamond)x_2^2)x_3).$$

If $t : G^k \to G$, we seek a term $u(\vec{x})$ such that $f(\vec{x}, u(\vec{x}))^{\mathbf{G}} = t$, i.e.,

$$f(\vec{d}, u(\vec{d})) = t(\vec{d}) \ \text{ for all } \vec{d} \in G^k.$$

$f(\vec{x}, \Diamond)$ is **valid with respect to** t if there is an operation $h : G^k \to G$ such that

$$f(\vec{d}, h(\vec{d})) = t(\vec{d}) \ \text{ for all } \vec{d} \in G^k.$$

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4x_1)\Diamond)x_2^2)x_3).$$

If $t : G^k \to G$, we seek a term $u(\vec{x})$ such that $f(\vec{x}, u(\vec{x}))^{\mathbf{G}} = t$, i.e.,

$$f(\vec{d}, u(\vec{d})) = t(\vec{d}) \text{ for all } \vec{d} \in G^k.$$

$f(\vec{x}, \Diamond)$ is **valid with respect to** t if there is an operation $h : G^k \to G$ such that

$$f(\vec{d}, h(\vec{d})) = t(\vec{d}) \text{ for all } \vec{d} \in G^k.$$

In this case we say that h is a **witness** to the validity of $f(\vec{x}, \Diamond)$.

A k-ary **partial term** is a term $f(\vec{x}, \Diamond)$ in variables $x_0, x_1, \ldots, x_{k-1}$ that contains exactly one occurrence of a new variable $\Diamond$, e.g.,

$$f(\vec{x}, \Diamond) = x_0((((x_4 x_1)\Diamond)x_2^2)x_3).$$

If $t : G^k \to G$, we seek a term $u(\vec{x})$ such that $f(\vec{x}, u(\vec{x}))^{\mathbf{G}} = t$, i.e.,

$$f(\vec{d}, u(\vec{d})) = t(\vec{d}) \ \text{ for all } \vec{d} \in G^k.$$

$f(\vec{x}, \Diamond)$ is **valid with respect to** t if there is an operation $h : G^k \to G$ such that

$$f(\vec{d}, h(\vec{d})) = t(\vec{d}) \ \text{ for all } \vec{d} \in G^k.$$

In this case we say that h is a **witness** to the validity of $f(\vec{x}, \Diamond)$.

Theorem. *If* $\mathbf{G}$ *is idemprimal, then there is a term* $u(\vec{x})$ *such that* $f(\vec{x}, u(\vec{x}))^{\mathbf{G}} = t$ *if and only if* $f(\vec{x}, \Diamond)$ *is valid with respect to* t.

A Beam Enumeration Algorithm

A Beam Enumeration Algorithm

G - idemprimal, $t : G^k \to G$, finite set M of test terms

A Beam Enumeration Algorithm

G - idemprimal, $t : G^k \to G$, finite set M of test terms

$\lozenge$ t

A Beam Enumeration Algorithm

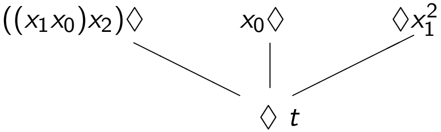
G - idemprimal, $t : G^k \to G$, finite set M of test terms

$\Diamond \; t$

Generate $\Diamond u(\vec{x})$ or $u(\vec{x})\Diamond$ valid w.r.t. t

A Beam Enumeration Algorithm

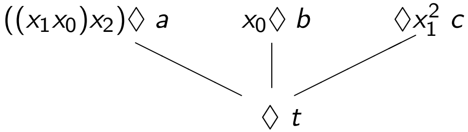
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Generate $\Diamond u(\vec{x})$ or $u(\vec{x})\Diamond$ valid w.r.t. t

A Beam Enumeration Algorithm

G - idemprimal, $t : G^k \to G$, finite set M of test terms

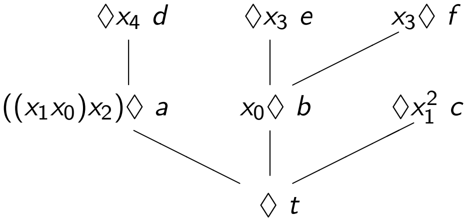


Generate $\Diamond u(\vec{x})$ or $u(\vec{x})\Diamond$ valid w.r.t. t

and validity witnesses $a, b, c, \cdots : G^k \to G$.

A Beam Enumeration Algorithm

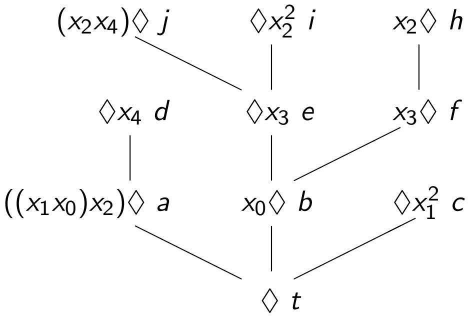
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Generate $\Diamond u(\vec{x})$ or $u(\vec{x})\Diamond$ valid w.r.t. parent operation and validity witnesses $a, b, c, \cdots : G^k \to G$.

A Beam Enumeration Algorithm

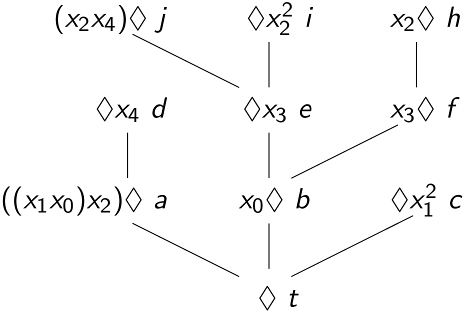
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Generate $\Diamond u(\vec{x})$ or $u(\vec{x})\Diamond$ valid w.r.t. parent operation and validity witnesses $a, b, c, \cdots : G^k \to G$.

A Beam Enumeration Algorithm

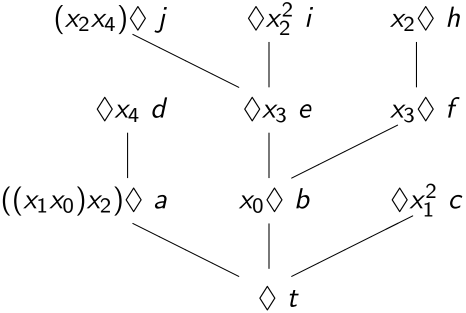
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Assume $u(\vec{x}) \in M$ and $u(\vec{x})^{\mathbf{G}} = j$.

A Beam Enumeration Algorithm

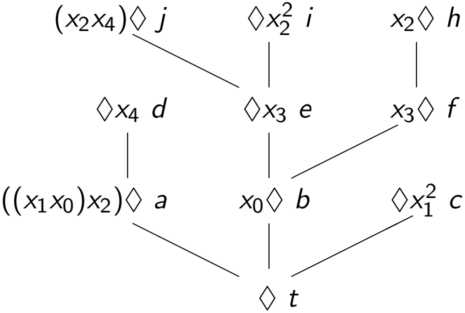
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Assume $u(\vec{x}) \in M$ and $u(\vec{x})^{\mathbf{G}} = j$. Then $(x_2x_4)u(\vec{x})^{\mathbf{G}} = e$

A Beam Enumeration Algorithm

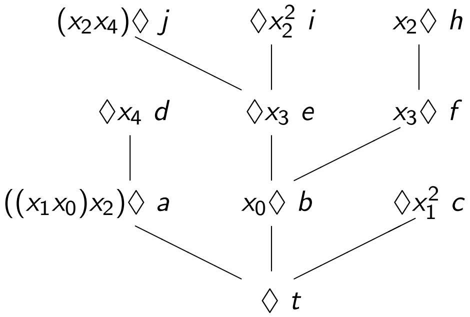
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Assume $u(\vec{x}) \in M$ and $u(\vec{x})^{\mathbf{G}} = j$. Then $(x_2x_4)u(\vec{x})^{\mathbf{G}} = e$
$\Rightarrow \quad ((x_2x_4)u(\vec{x}))x_3^{\mathbf{G}} = b$

A Beam Enumeration Algorithm

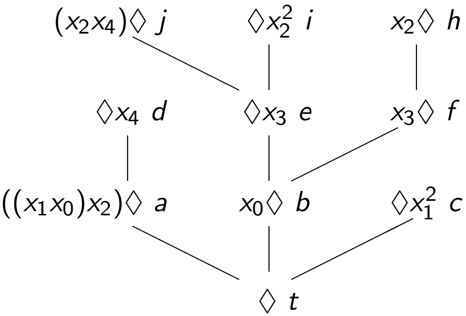
G - idemprimal, $t : G^k \to G$, finite set M of test terms



Assume $u(\vec{x}) \in M$ and $u(\vec{x})^{\mathbf{G}} = j$. Then $(x_2x_4)u(\vec{x})^{\mathbf{G}} = e$
$\Rightarrow \quad ((x_2x_4)u(\vec{x}))x_3^{\mathbf{G}} = b \quad \Rightarrow \quad x_0(((x_2x_4)u(\vec{x}))x_3)^{\mathbf{G}} = t.$

A Beam Enumeration Algorithm

G - idemprimal, $t : G^k \to G$, finite set M of test terms

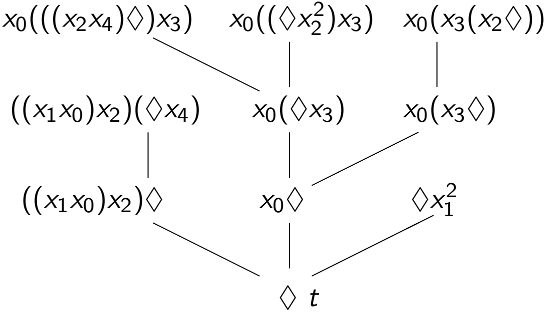


Assume $u(\vec{x}) \in M$ and $u(\vec{x})^{\mathbf{G}} = j$. Then $(x_2x_4)u(\vec{x})^{\mathbf{G}} = e$
$\Rightarrow \quad ((x_2x_4)u(\vec{x}))x_3^{\mathbf{G}} = b \quad \Rightarrow \; x_0(((x_2x_4)u(\vec{x}))x_3)^{\mathbf{G}} = t.$

$x_0(((x_2x_4)\Diamond)x_3)$ is valid w.r.t. t.

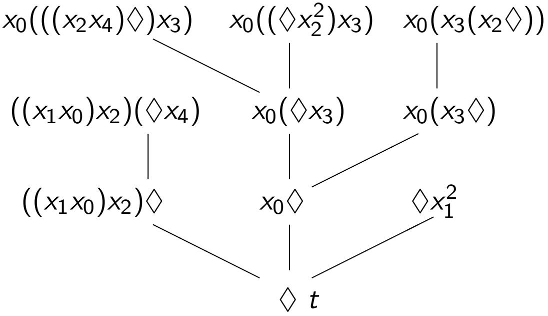
A Beam Enumeration Algorithm

G - idemprimal, $t : G^k \to G$, finite set M of test terms



A Beam Enumeration Algorithm

G - idemprimal, $t : G^k \to G$, finite set M of test terms



A partial term $f(\vec{x}, \Diamond)$ is **constant** on **G** if all terms $u(\vec{x})$ produce the same term operation $f(\vec{x}, u(\vec{x}))^{\mathbf{G}}$.

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Term Continuity Theorem.

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Term Continuity Theorem. *If* **G** *is* k-**term continuous** (TC) *for some* $k > 1$,

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Term Continuity Theorem. *If* **G** *is* k-**term continuous** (TC) *for some* $k > 1$, *then there is a number* λ *such that*

$$\frac{k}{k+1} \leq \lambda \leq 1$$

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Term Continuity Theorem. *If* **G** *is* k-**term continuous** (TC) *for some* $k > 1$*, then there is a number* λ *such that*

$$\frac{2}{3} \leq \frac{k}{k+1} \leq \lambda \leq 1$$

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Term Continuity Theorem. *If* **G** *is* k-**term continuous** (TC) *for some* $k > 1$*, then there is a number* λ *such that*

$$\frac{2}{3} \leq \frac{k}{k+1} \leq \lambda \leq 1$$

and the proportion of partial terms of height at most H *that are constant*

Idemprimality Theorem. *If* **G** *is* **idemprimal** (IPr) *and a BEA produces a constant partial term, then the BEA will terminate with a representation of the target operation.*

Term Continuity Theorem. *If* **G** *is* k-**term continuous** (TC) *for some* $k > 1$*, then there is a number* λ *such that*

$$\frac{2}{3} \leq \frac{k}{k+1} \leq \lambda \leq 1$$

and the proportion of partial terms of height at most H *that are constant converges to* λ *as* $H \to \infty$*.*

$\mathbf{G} := \langle\{0, 1, 2, 3\}, *\rangle$

$*$	0	1	2	3
0	1	3	1	2
1	1	2	2	3
2	2	0	0	3
3	2	2	1	2

$\mathbf{G} := \langle \{0, 1, 2, 3\}, * \rangle$

$*$	0	1	2	3
0	1	3	1	2
1	1	2	2	3
2	2	0	0	3
3	2	2	1	2

G is primal by Rouseau's Theorem, so it is idemprimal (IPr).

$\mathbf{G} := \langle \{0, 1, 2, 3\}, * \rangle$

$*$	0	1	2	3
0	1	3	1	2
1	1	2	2	3
2	2	0	0	3
3	2	2	1	2

G is primal by Rouseau's Theorem, so it is idemprimal (IPr).
Tests from Clark [1] shows that **G** is term continuous (TC).

$\mathbf{G} := \langle \{0,1,2,3\}, *\rangle$

$*$	0	1	2	3
0	1	3	1	2
1	1	2	2	3
2	2	0	0	3
3	2	2	1	2

G is primal by Rouseau's Theorem, so it is idemprimal (IPr).
Tests from Clark [1] shows that **G** is term continuous (TC).

Example: Find a discriminator term for **G**.

A Beam Enumeration Algorithm

A Beam Enumeration Algorithm

$d(x, y, z) =$

$$
(((((xy)x)x)(((xy)x)((zx)(yx))))(((xy)(((xx)x)x))(((((yx)(yx))x)((yx)y))(((yx)((xy)(y(yx))))((z(z((xz)
z)))(((x(zx))((x(zx))x))(((zx)(((zx)x)(x(zz))))(((zx)(zx))((((xz)z)((zz)z))(((((yz)z)x)(((zx)(yy))x))
(((z(zx))((z(zx))x))((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)(((x(x(yy)))(xx))((xx)((xx)
((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)(((zz)(((yx)y)z))((y((zy)(yy)))((y(y((zz)
(yy))))((z(z((zx)(yx))))(((((xz)(zy))y)((y(zy))((zy)(yy))))((z(z((xx)z)))((yy)((yy)((yy)((yy)((yy)((y(x((yx)
(yy))))((xx)((xx)((xx)((xx)((x(xx))(((x((xz)y))(x((xz)(zx))))(((x(z(xz)))(z(x(xz))))((zz)((zz)((zz)((zz)
((zz)((xx)((xx)((xx)((xx)((xx)((xx)((y(yy))(((xy)((yx)(x(xy))))((yy)((z(zz))((y(yy))((yy)((y(yy))(((yz)
(y((xz)(yz))))((zz)(((xx)(xz))(((((xy)y)x)((z(yz))(y(xx))))((z(z((xx)z)))(((yy)(((zx)y)(y(yz))))((((zy)x)
((xy)((yx)(zy))))((y(((xy)z)(zy)))(((yy)((yy)(yy)))((yy)((yy)((yy)(((yx)(x(z(zy))))((xx)((xx)(((y((yy)
(yy)))(y(xx)))((yy)((xx)((xx)((x(x((xz)z)))(((yx)x)((xx)((xx)((((y(xx))(x(yx)))(z((yz)(yx))))((yx)((xx)
((xx)((xx)((x(x((xx)(zz))))((zx)((xx)(((((y((yy)y))y)((y(y(xz)))((((yz)((zz)z))((z(xy))z))(((zz)(z((yx)
(zy))))(((x(y(zx)))(((yx)(zx))(zx)))(((((zx)y)x)(yz))(((zz)(((((yz)x)((zy)x))(((yx)(yx))((zz)x)))
((x((y(yy))x))(((xx)(((xy)y)(zx)))((((((xz)y)z)(yy))(yz))(y(x(xy))))((y((zy)(yz)))(((xz)(zx))
(y(yx)))))))))(xy)))))))(yy(y((y(yy))y)))
)))))))))))))))))))))))))))))))))))))
$$

A Beam Enumeration Algorithm

$d(x, y, z) =$

$(((((xy)x)x)(((xy)x)((zx)(yx))))(((xy)(((xx)x)x))(((((yx)(yx))x)((yx)y))(((yx)((xy)(y(yx))))((z(z((xz)$
$z)))(((x(zx))((x(zx))x))(((zx)(((zx)x)(x(zz))))(((zx)(zx))((((xz)z)((zz)z))(((((yz)z)x)(((zx)(yy))x))$
$(((z(zx))((z(zx))x))((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)(((x(x(yy)))(xx))((xx)((xx)$
$((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)(((zz)(((yx)y)z))((y((zy)(yy)))((y(y((zz)$
$(yy))))((z(z((zx)(yx))))(((((xz)(zy))y)((y(zy))((zy)(yy))))((z(z((xx)z)))((yy)((yy)((yy)((yy)((yy)((y(x((yx)$
$(yy))))((xx)((xx)((xx)((xx)((x(xx))(((x((xz)y))(x((xz)(zx))))(((x(z(xz)))(z(x(xz))))((zz)((zz)((zz)((zz)$
$((zz)((xx)((xx)((xx)((xx)((xx)((xx)((y(yy))(((xy)((yx)(x(xy))))((yy)((z(zz))((y(yy))((yy)((y(yy))(((yz)$
$(y((xz)(yz))))((zz)(((xx)(xz))(((((xy)y)x)((z(yz))(y(xx))))((z(z((xx)z)))(((yy)(((zx)y)(y(yz))))((((zy)x)$
$((xy)((yx)(zy))))((y(((xy)z)(zy)))(((yy)((yy)(yy)))((yy)((yy)((yy)(((yx)(x(z(zy))))((xx)((xx)(((y((yy)$
$(yy)))(y(xx)))((yy)((xx)((xx)((x(x((xz)z)))(((yx)x)((xx)((xx)((((y(xx))(x(yx)))(z((yz)(yx))))((yx)((xx)$
$((xx)((xx)((x(x((xx)(zz))))((zx)((xx)(((((y((yy)y))y)((y(y(xz)))((((yz)((zz)z))((z(xy))z))(((zz)(z((yx)$
$(zy))))(((x(y(zx)))(((yx)(zx))(zx)))(((((zx)y)x)(yz))(((zz)(((((yz)x)((zy)x))(((yx)(yx))((zz)x)))$
$((x((y(yy))x))(((xx)(((xy)y)(zx)))((((((xz)y)z)(yy))(yz))(y(x(xy))))((y((zy)(yz)))(((xz)(zx))$
$(y(yx))))))))(xy)))))))(yy(y((y(yy))y))$
$))))))))))))))))))))))))))))))))))$

492 variable occurrences,

A Beam Enumeration Algorithm

$d(x, y, z) =$

$$
\begin{aligned}
&(((((xy)x)x)(((xy)x)((zx)(yx))))(((xy)(((xx)x)x))(((((yx)(yx))x)((yx)y))(((yx)((xy)(y(yx))))((z(z((xz)\\
&z)))(((x(zx))((x(zx))x))(((zx)(((zx)x)(x(zz))))(((zx)(zx))((((xz)z)((zz)z))(((((yz)z)x)(((zx)(yy))x))\\
&(((z(zx))((z(zx))x))((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)(((x(x(yy)))(xx))((xx)((xx)\\
&((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)((xx)(((zz)(((yx)y)z))((y((zy)(yy)))((y(y((zz)\\
&(yy))))((z(z((zx)(yx))))(((((xz)(zy))y)((y(zy))((zy)(yy))))((z(z((xx)z)))((yy)((yy)((yy)((yy)((yy)((y(x((yx)\\
&(yy))))((xx)((xx)((xx)((xx)((x(xx))(((x((xz)y))(x((xz)(zx))))(((x(z(xz)))(z(x(xz))))((zz)((zz)((zz)((zz)\\
&((zz)((xx)((xx)((xx)((xx)((xx)((xx)((y(yy))(((xy)((yx)(x(xy))))((yy)((z(zz))((y(yy))((yy)((y(yy))(((yz)\\
&(y((xz)(yz))))((zz)(((xx)(xz))(((((xy)y)x)((z(yz))(y(xx))))((z(z((xx)z)))(((yy)(((zx)y)(y(yz))))((((zy)x)\\
&((xy)((yx)(zy))))((y(((xy)z)(zy)))(((yy)((yy)(yy)))((yy)((yy)((yy)(((yx)(x(z(zy))))((xx)((xx)(((y((yy)\\
&(yy)))(y(xx)))((yy)((xx)((xx)((x(x((xz)z)))(((yx)x)((xx)((xx)((((y(xx))(x(yx)))(z((yz)(yx))))((yx)((xx)\\
&((xx)((xx)((x(x((xx)(zz))))((zx)((xx)(((((y((yy)y))y)((y(y(xz)))((((yz)((zz)z))((z(xy))z))(((zz)(z((yx)\\
&(zy))))(((x(y(zx)))(((yx)(zx))(zx)))(((((zx)y)x)(yz))(((zz)(((((yz)x)((zy)x))(((yx)(yx))((zz)x)))\\
&((x((y(yy))x))(((xx)(((xy)y)(zx)))((((((xz)y)z)(yy))(yz))(y(x(xy))))((y((zy)(yz)))(((xz)(zx))\\
&(y(yx)))))))))(xy))))))))(yy(y((y(yy))y))\\
&)))))))))))))))))))))))))))))))))))))
\end{aligned}
$$

492 variable occurrences, 11.3 seconds.

Thanks for listening!

— DC