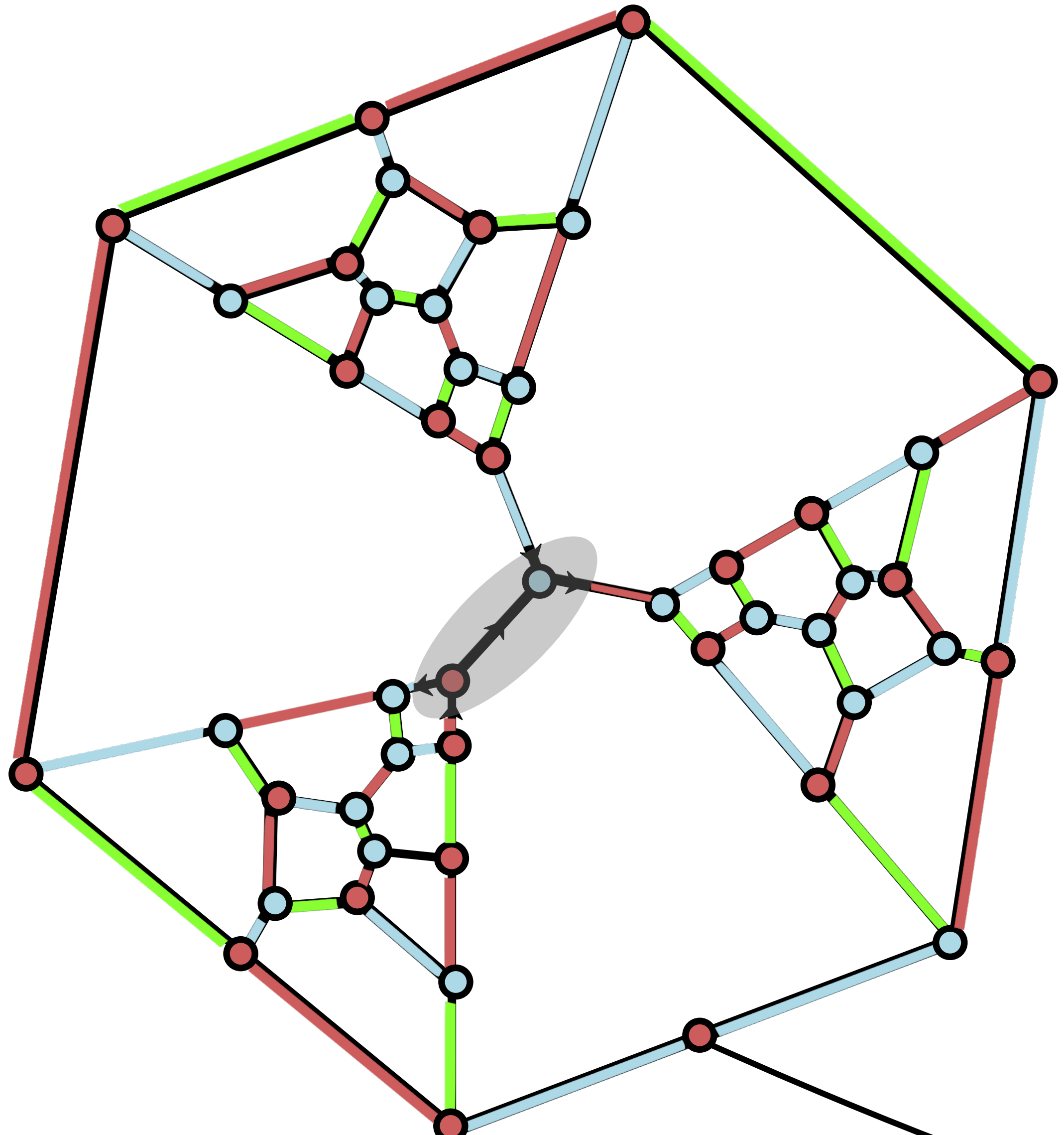
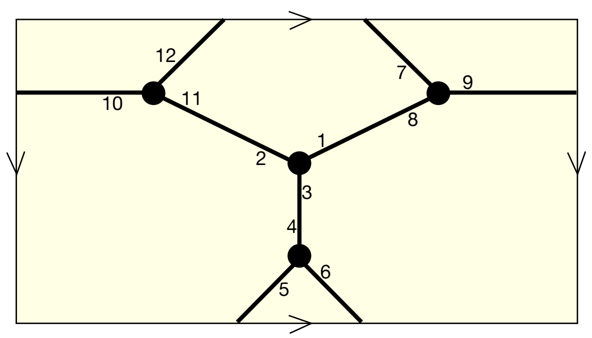
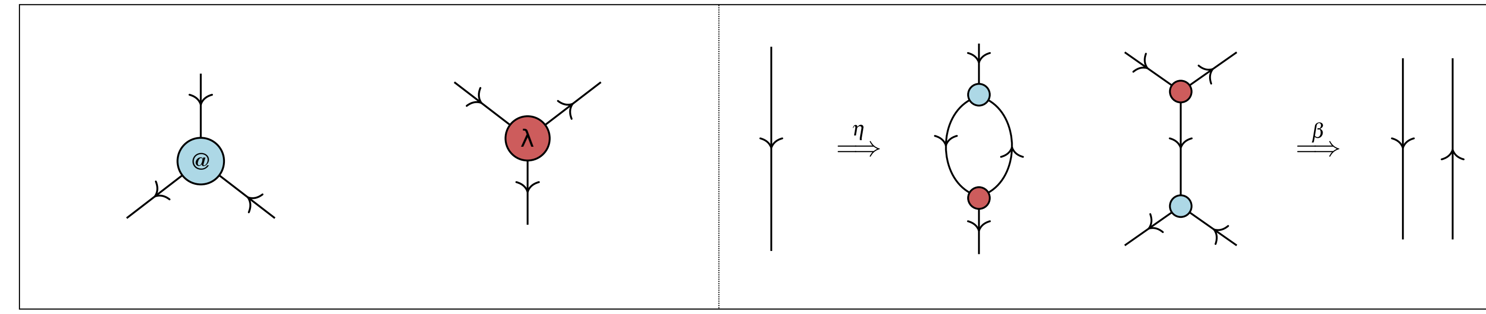


A theory of linear typings as flows on 3-valent graphs

Noam Zeilberger
School of Computer Science
University of Birmingham

LICS 2018
9 July
Oxford, UK





Part One: Background and Motivation

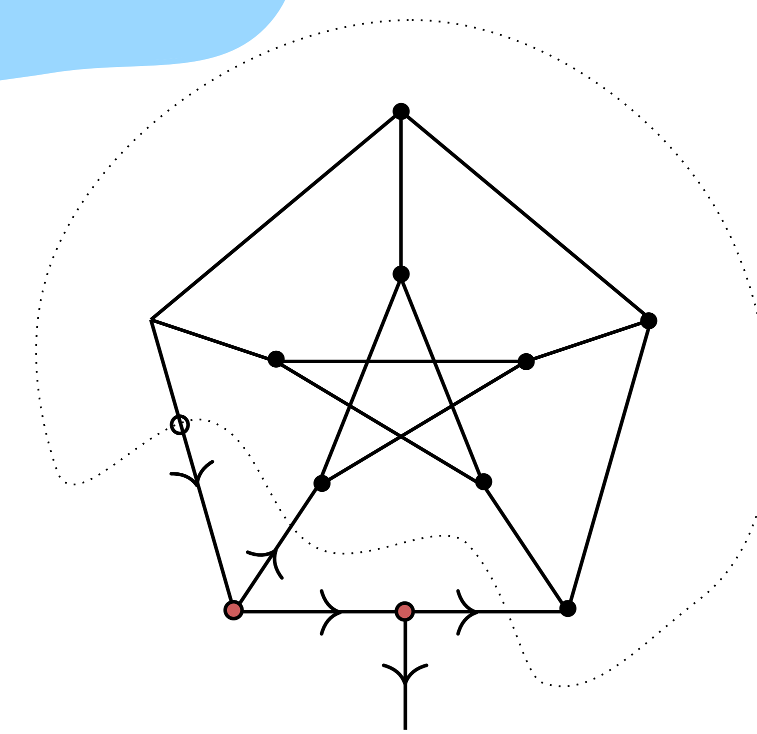
(5.1) *The number a_n of rooted maps with n edges is*

$$\frac{2(2n)!\,3^n}{n!\,(n+2)!}.$$

We write

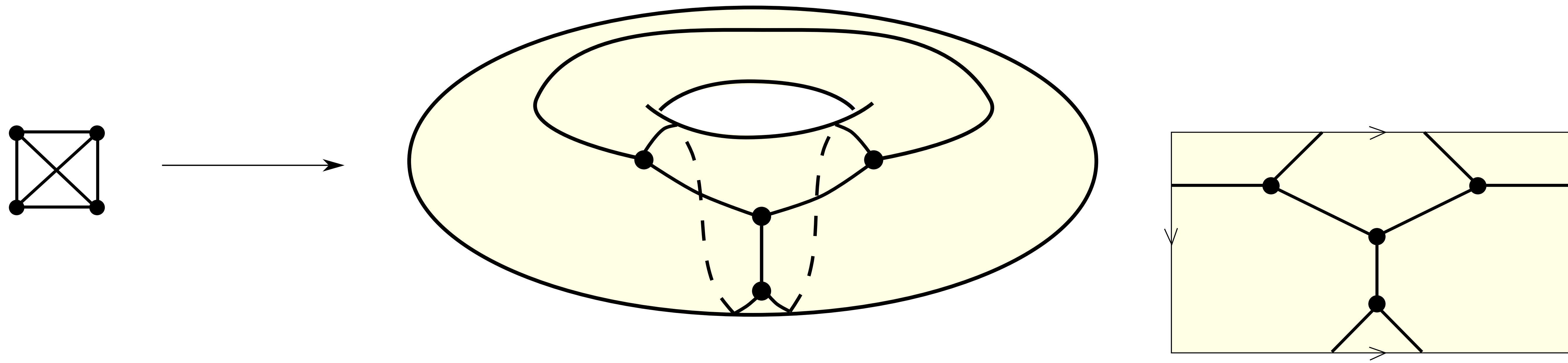
$$A(x) = \sum_{n=1}^{\infty} a_n x^n.$$

Thus $A(x) = 2x + 9x^2 + 54x^3 + 378x^4 + \ldots$. Figure 2 shows the 2 rooted maps with 1 edge, and Figure 3 the 9 rooted maps with 2 edges.



Topological definition

map = 2-cell embedding of a graph into a surface*



considered up to deformation of the underlying surface.

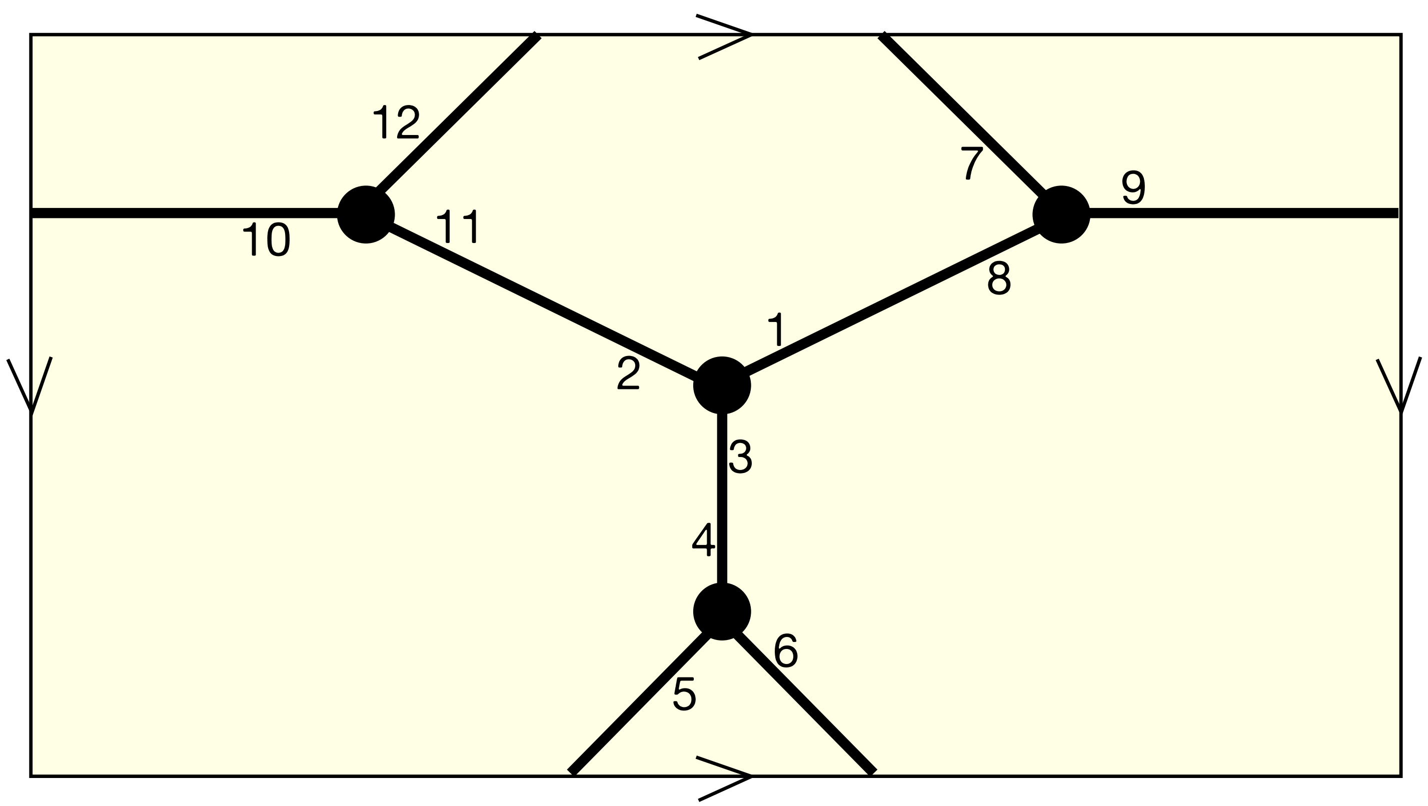
*All surfaces are assumed to be connected and oriented throughout this talk

Algebraic definition

map = transitive permutation representation of the group

G = $\langle v, e, f \mid e^2 = vef = 1 \rangle$

considered up to G-equivariant isomorphism.



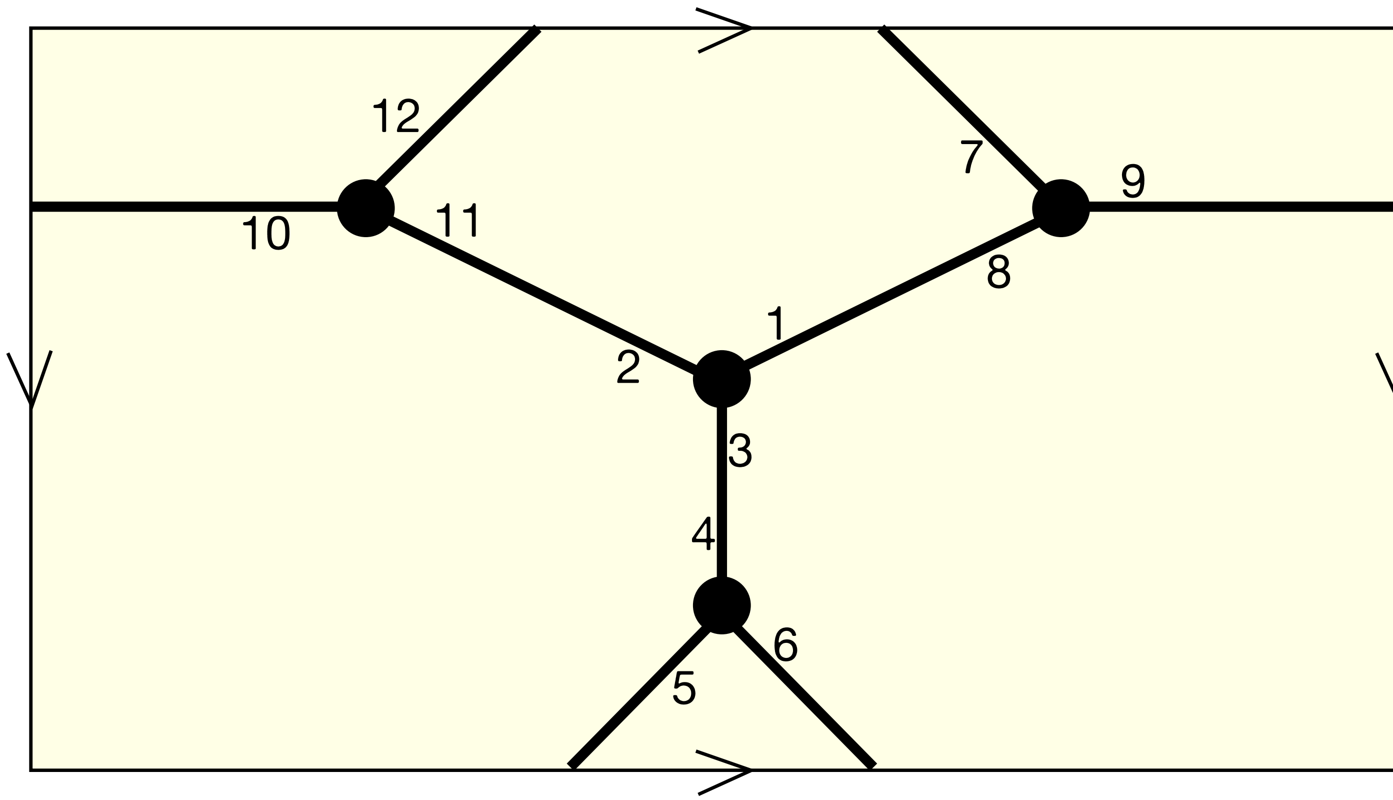
$$v = (1\ 2\ 3)(4\ 5\ 6)(7\ 8\ 9)(10\ 11\ 12)$$
$$e = (1\ 8)(2\ 11)(3\ 4)(5\ 12)(6\ 7)(9\ 10)$$
$$f = (1\ 7\ 5\ 11)(2\ 10\ 8\ 3\ 6\ 9\ 12\ 4)$$

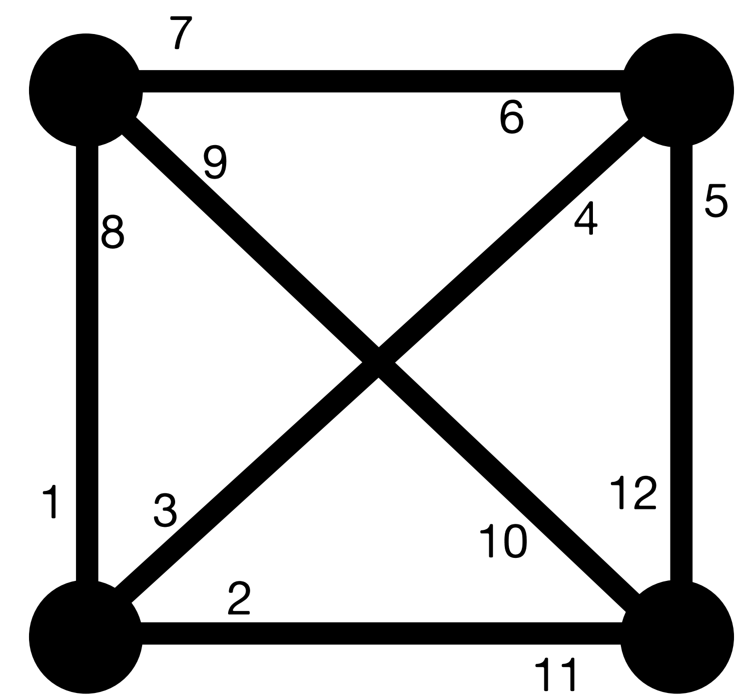
Note: can compute genus from Euler characteristic

$$c(v) - c(e) + c(f) = 2 - 2g$$

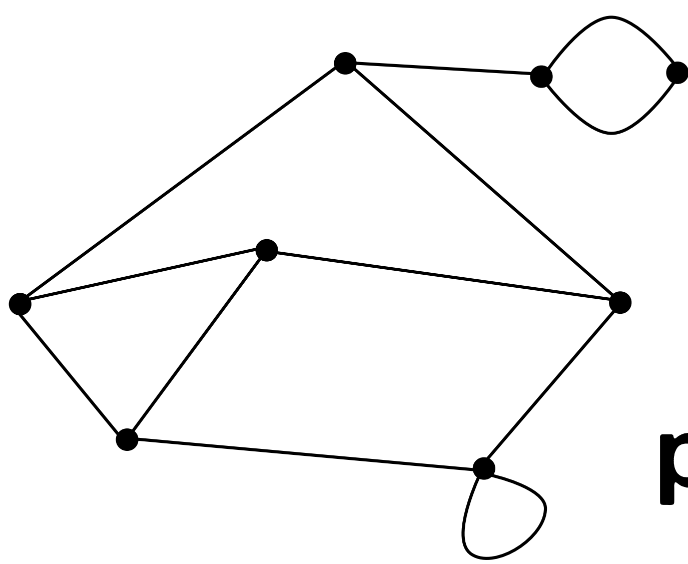
Combinatorial definition

map = connected graph + cyclic ordering of the half-edges around each vertex (say, as given by a planar drawing with "virtual crossings").



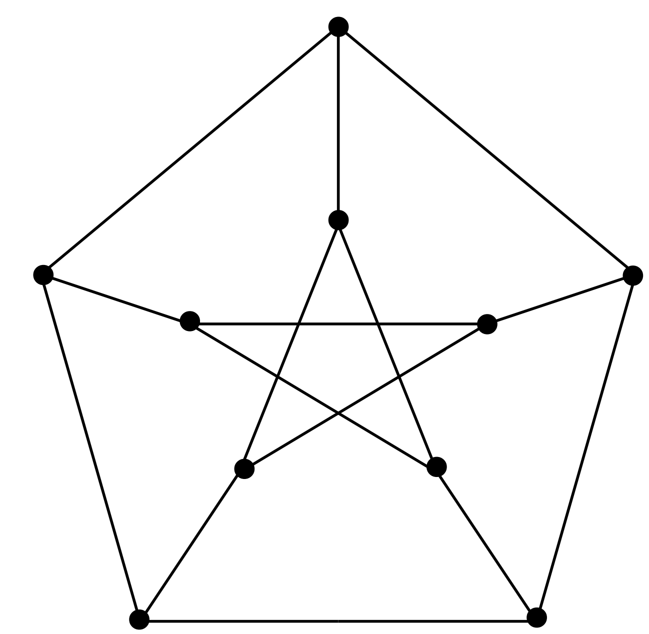


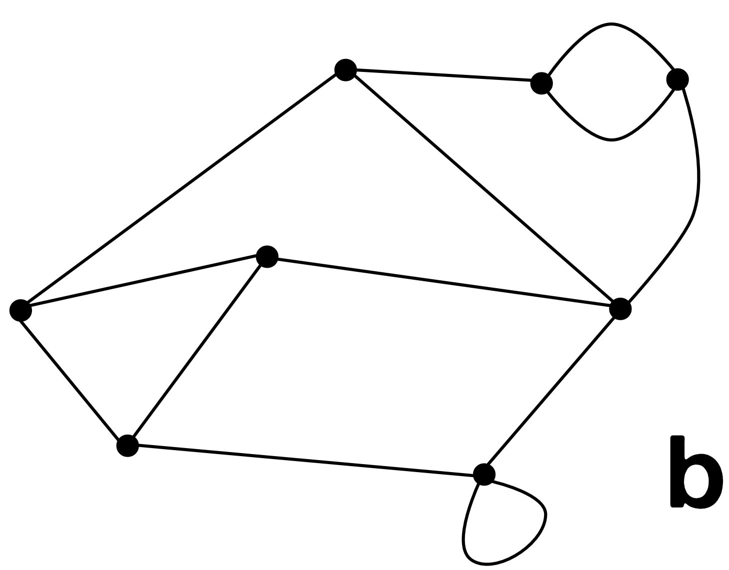
Some special kinds of maps



planar

3-valent

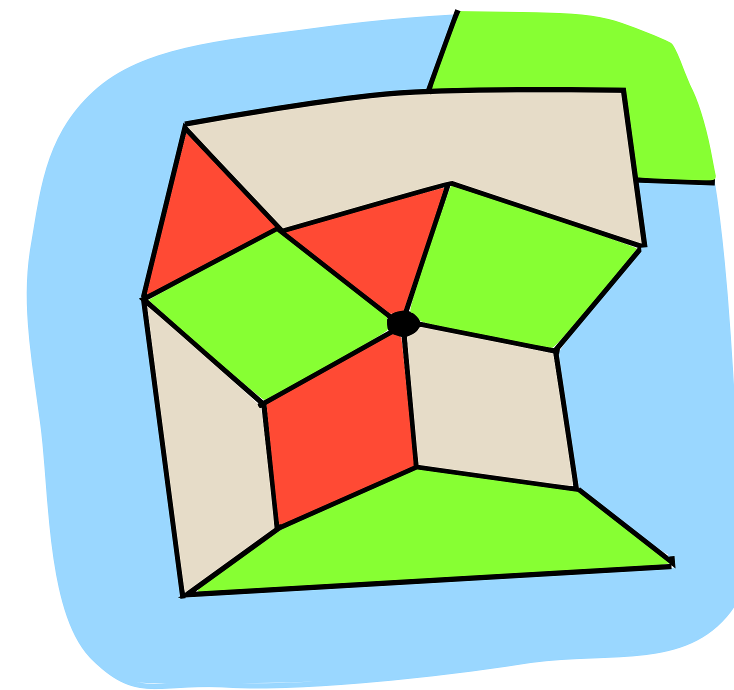


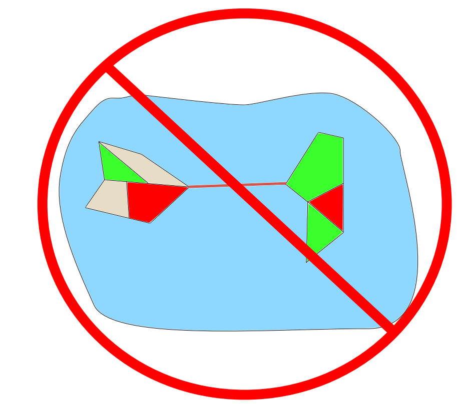


bridgeless

Four Colour Theorem

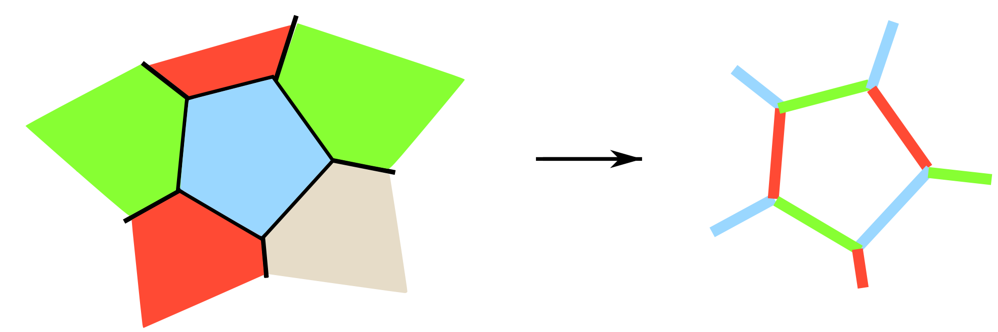
The 4CT is a statement about maps.



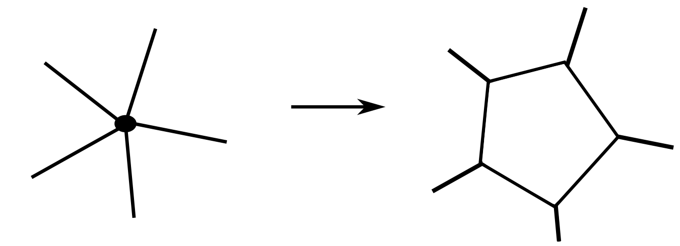


every bridgeless planar map
has a proper face 4-coloring

By a well-known reduction (Tait 1880), 4CT is equivalent
to a statement about 3-valent maps



every bridgeless planar 3-valent map
has a proper edge 3-coloring



Map enumeration

From time to time in a graph-theoretical career one's thoughts turn to the Four Colour Problem. It occurred to me once that it might be possible to get results of interest in the theory of map-colourings without actually solving the Problem. For example, it might be possible to find the average number of colourings on vertices, for planar triangulations of a given size.

One would determine the number of triangulations of 2n faces, and then the number of 4-coloured triangulations of 2n faces. Then one would divide the second number by the first to get the required average. I gathered that this sort of retreat from a difficult problem to a related average was not unknown in other branches of Mathematics, and that it was particularly common in Number Theory.

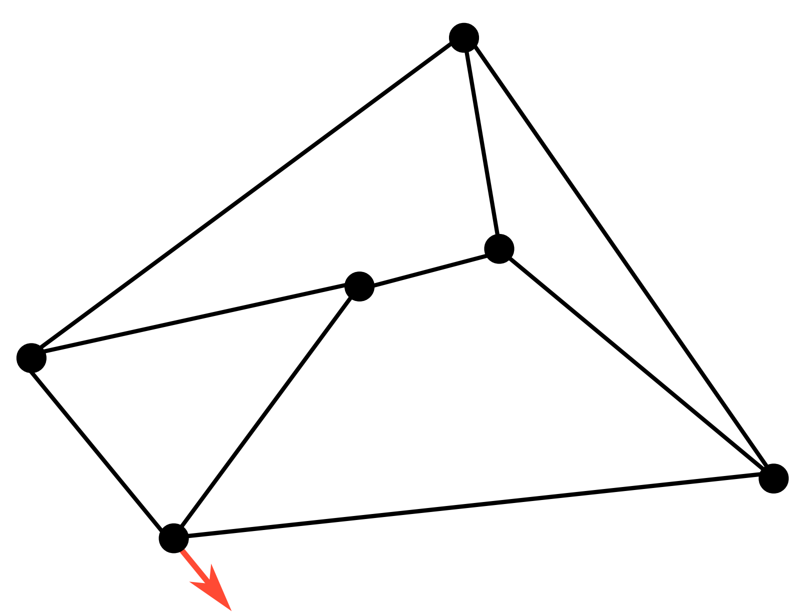
W. T. Tutte, Graph Theory as I Have Known It

Map enumeration

Tutte wrote a pioneering series of papers (1962-1969)

W. T. Tutte (1962), A census of planar triangulations. Canadian Journal of Mathematics 14:21–38
W. T. Tutte (1962), A census of Hamiltonian polygons. Can. J. Math. 14:402–417
W. T. Tutte (1962), A census of slicings. Can. J. Math. 14:708–722
W. T. Tutte (1963), A census of planar maps. Can. J. Math. 15:249–271
W. T. Tutte (1968), On the enumeration of planar maps. Bulletin of the American Mathematical Society 74:64–74
W. T. Tutte (1969), On the enumeration of four-colored maps. SIAM Journal on Applied Mathematics 17:454–460

One of his insights was to consider **rooted** maps



Key property: rooted maps have no non-trivial automorphisms

Some enumerative connections

family of rooted maps	family of lambda terms	sequence	OEIS
trivalent maps (genus g≥0)	linear terms	1,5,60,1105,27120,...	A062980

1. O. Bodini, D. Gardy, A. Jacquot (2013), Asymptotics and random sampling for BCI and BCK lambda terms, TCS 502: 227-238

OEIS = Online Encyclopedia of Integer Sequences (oeis.org)

Some enumerative connections

family of rooted maps	family of lambda terms	sequence	OEIS
trivalent maps (genus g≥0)	linear terms	1,5,60,1105,27120,...	A062980
planar maps	normal planar terms	1,2,9,54,378,2916,...	A000168

1. O. Bodini, D. Gardy, A. Jacquot (2013), Asymptotics and random sampling for BCI and BCK lambda terms, TCS 502: 227-238
2. Z, A. Giorgetti (2015), A correspondence between rooted planar maps and normal planar lambda terms, LMCS 11(3:22): 1-39

OEIS = Online Encyclopedia of Integer Sequences (oeis.org)

Some enumerative connections

family of rooted maps	family of lambda terms	sequence	OEIS
trivalent maps (genus g≥0)	linear terms	1,5,60,1105,27120,...	A062980
planar trivalent maps	planar terms	1,4,32,336,4096,...	A002005
bridgeless trivalent maps	unitless linear terms	1,2,20,352,8624,...	A267827
bridgeless planar trivalent maps	unitless planar terms	1,1,4,24,176,1456,...	A000309
maps (genus g≥0)	normal linear terms (mod ~)	1,2,10,74,706,8162,...	A000698
planar maps	normal planar terms	1,2,9,54,378,2916,...	A000168
bridgeless maps	normal unitless linear terms (mod ~)	1,1,4,27,248,2830,...	A000699
bridgeless planar maps	normal unitless planar terms	1,1,3,13,68,399,...	A000260

1. O. Bodini, D. Gardy, A. Jacquot (2013), Asymptotics and random sampling for BCI and BCK lambda terms, TCS 502: 227-238
2. Z, A. Giorgetti (2015), A correspondence between rooted planar maps and normal planar lambda terms, LMCS 11(3:22): 1-39
3. Z (2015), Counting isomorphism classes of beta-normal linear lambda terms, arXiv:1509.07596
4. Z (2016), Linear lambda terms as invariants of rooted trivalent maps, J. Functional Programming 26(e21)
5. J. Courtiel, K. Yeats, Z (2016), Connected chord diagrams and bridgeless maps, arXiv:1611.04611
6. Z (2017), A sequent calculus for a semi-associative law, FSCD

OEIS = Online Encyclopedia of Integer Sequences (oeis.org)

Some enumerative connections

(conceptual background for LICS paper)

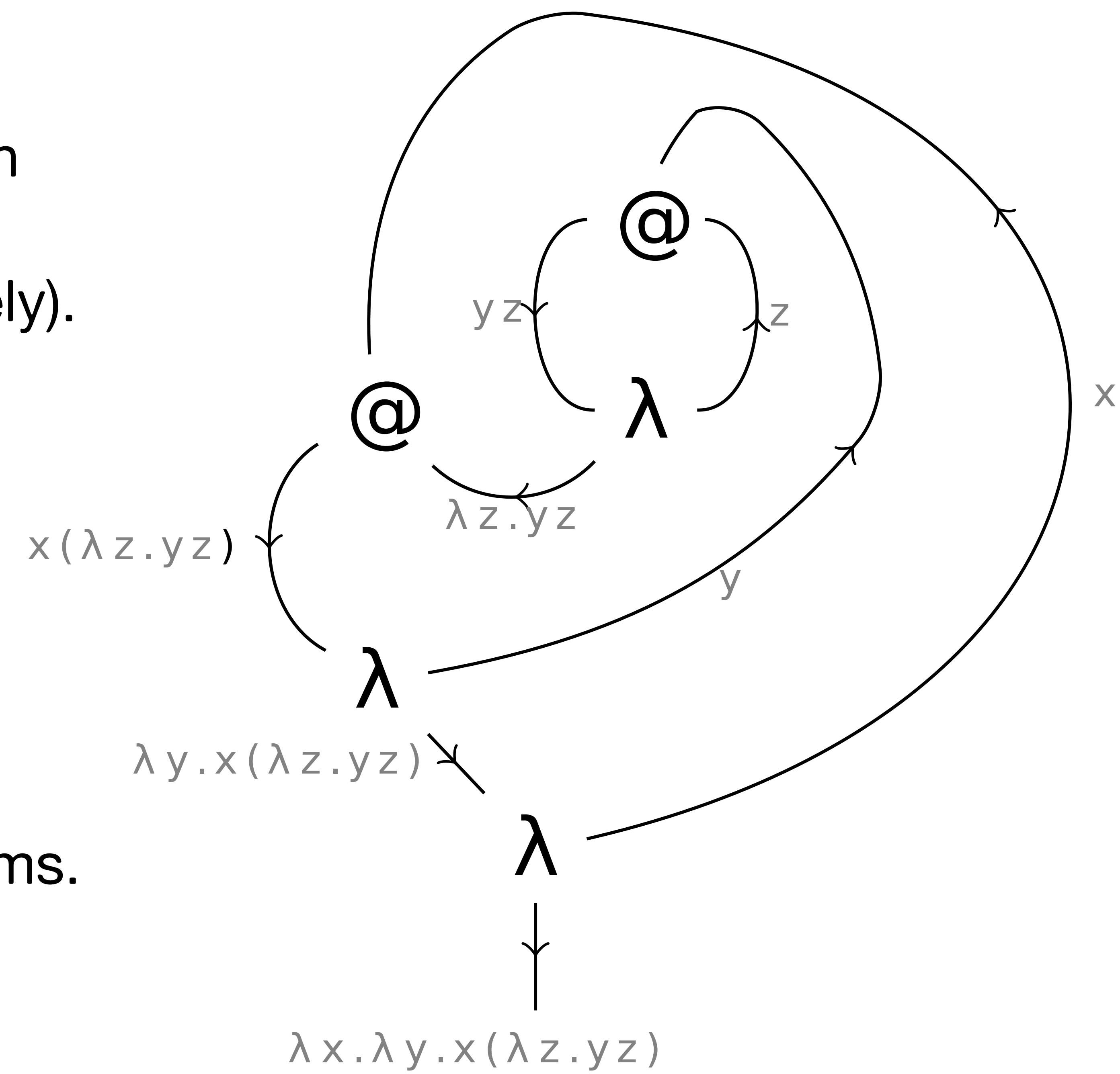
family of rooted maps	family of lambda terms	sequence	OEIS
trivalent maps (genus g≥0)	linear terms	1,5,60,1105,27120,...	A062980
planar trivalent maps	planar terms	1,4,32,336,4096,...	A002005
bridgeless trivalent maps	unitless linear terms	1,2,20,352,8624,...	A267827
bridgeless planar trivalent maps	unitless planar terms	1,1,4,24,176,1456,...	A000309
maps (genus g≥0)	normal linear terms (mod ~)	1,2,10,74,706,8162,...	A000698
planar maps	normal planar terms	1,2,9,54,378,2916,...	A000168
bridgeless maps	normal unitless linear terms (mod ~)	1,1,4,27,248,2830,...	A000699
bridgeless planar maps	normal unitless planar terms	1,1,3,13,68,399,...	A000260

1. O. Bodini, D. Gardy, A. Jacquot (2013), Asymptotics and random sampling for BCI and BCK lambda terms, TCS 502: 227-238
2. Z, A. Giorgetti (2015), A correspondence between rooted planar maps and normal planar lambda terms, LMCS 11(3:22): 1-39
3. Z (2015), Counting isomorphism classes of beta-normal linear lambda terms, arXiv:1509.07596
4. Z (2016), Linear lambda terms as invariants of rooted trivalent maps, J. Functional Programming 26(e21)
5. J. Courtiel, K. Yeats, Z (2016), Connected chord diagrams and bridgeless maps, arXiv:1611.04611
6. Z (2017), A sequent calculus for a semi-associative law, FSCD

OEIS = Online Encyclopedia of Integer Sequences (oeis.org)

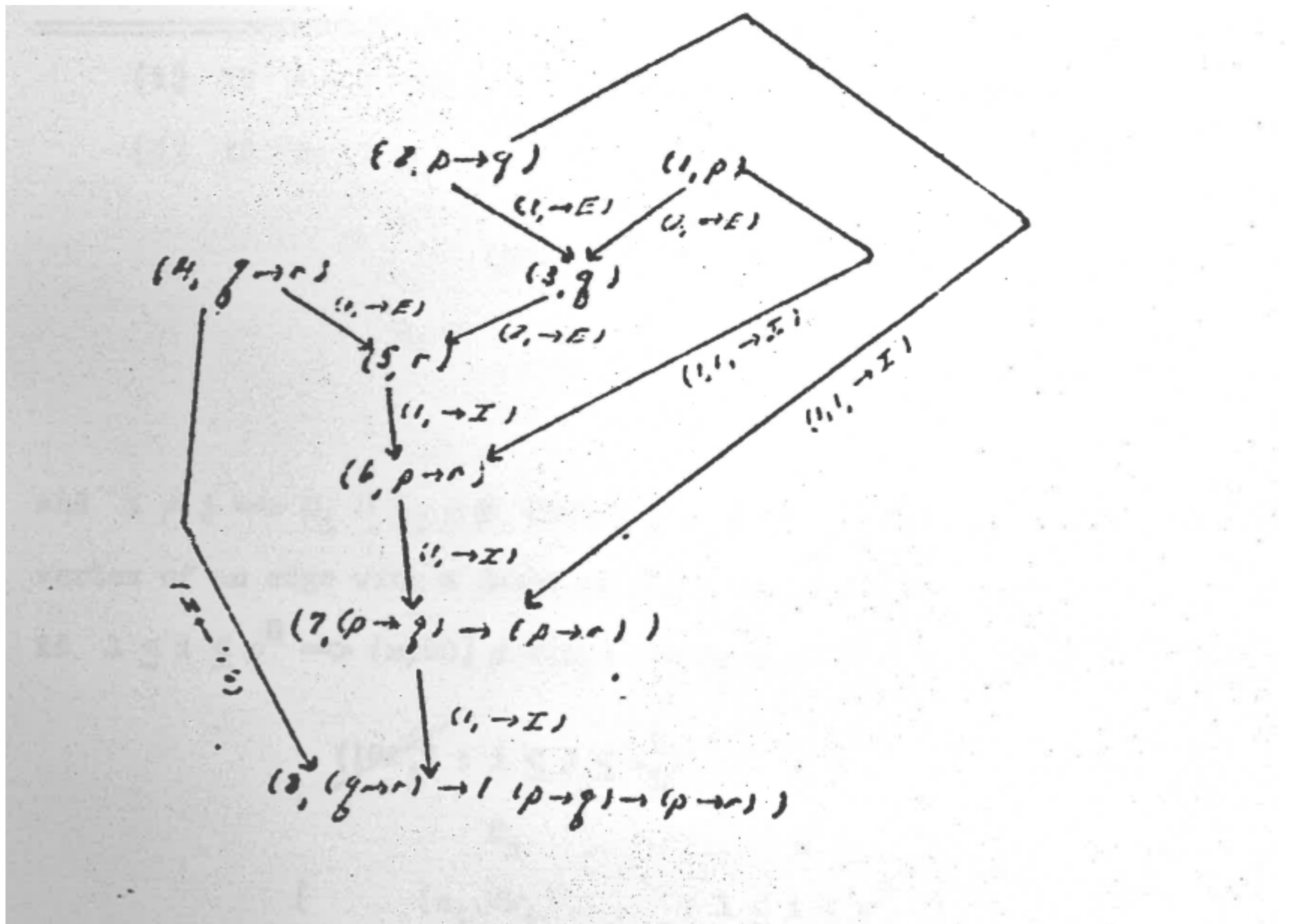
Representing terms as graphs
(an idea from the folklore)

Represent a term as a "tree with pointers", with lambda nodes pointing to the occurrences of the corresponding bound variable (or conversely).



This old idea is especially natural for **linear** terms.

Representing proofs as graphs
(a closely related idea)



A derivation in the System I (Section 3)

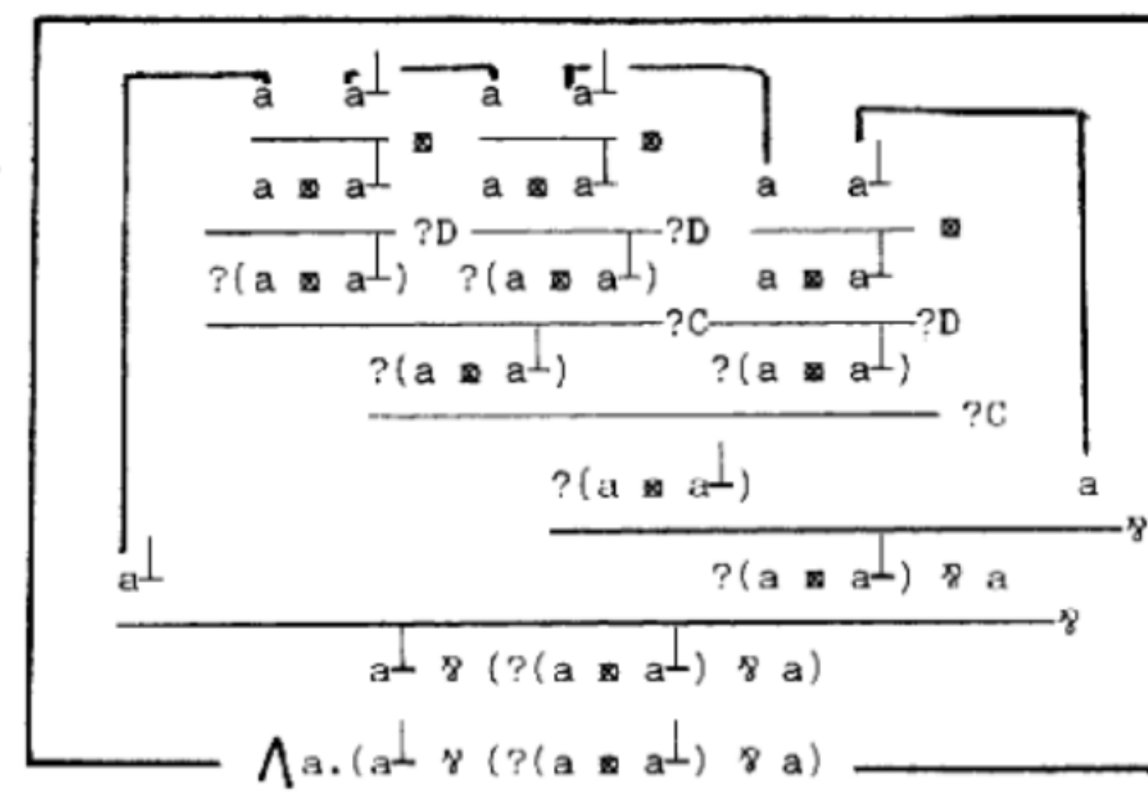


Fig. 57(b). The number "3" represented in PN2.

R. Statman (1974), Structural Complexity of Proofs, PhD Thesis, Stanford University

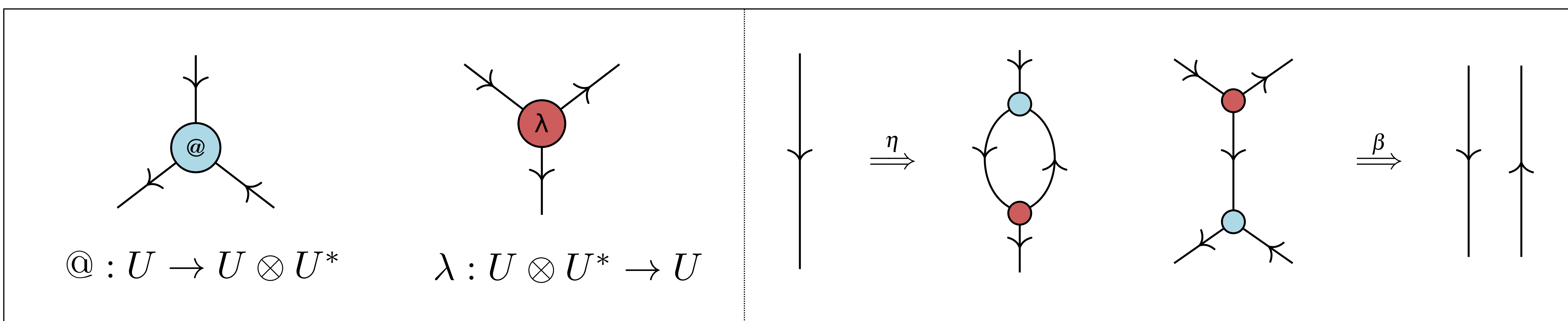
J.-Y. Girard (1987), Linear Logic, Theoretical Computer Science

λ-graphs as string diagrams

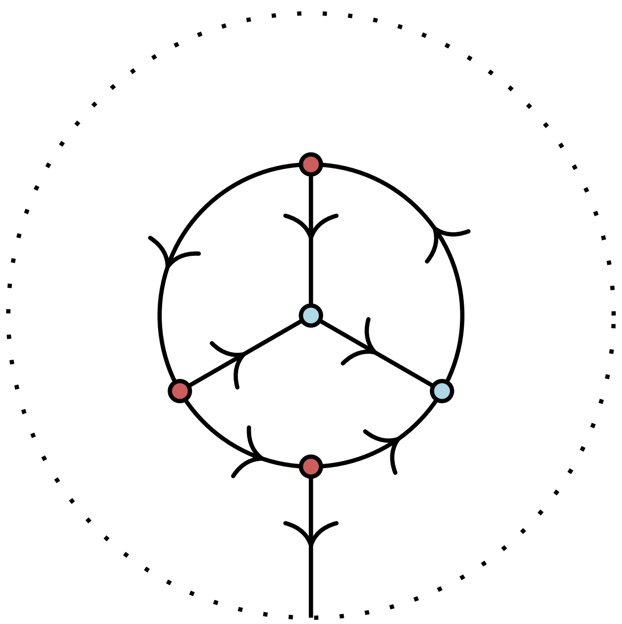
Idea (after D. Scott): a linear lambda term may be interpreted as an endomorphism of a reflexive object in a symmetric monoidal closed (bi)category.

$$U \underset{\lambda}{\overset{@}{\underset{\perp}{\rightleftarrows}}} U \multimap U$$

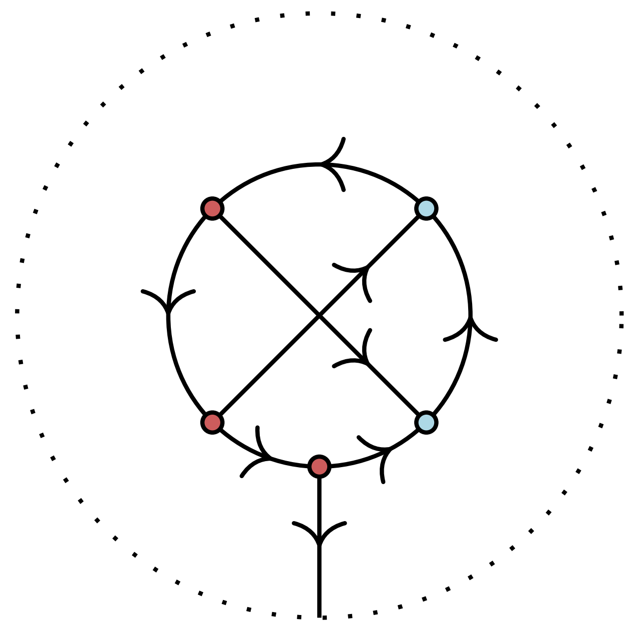
By interpreting this morphism in the graphical language of compact closed (bi)categories, we obtain the traditional diagram associated to the linear lambda term.



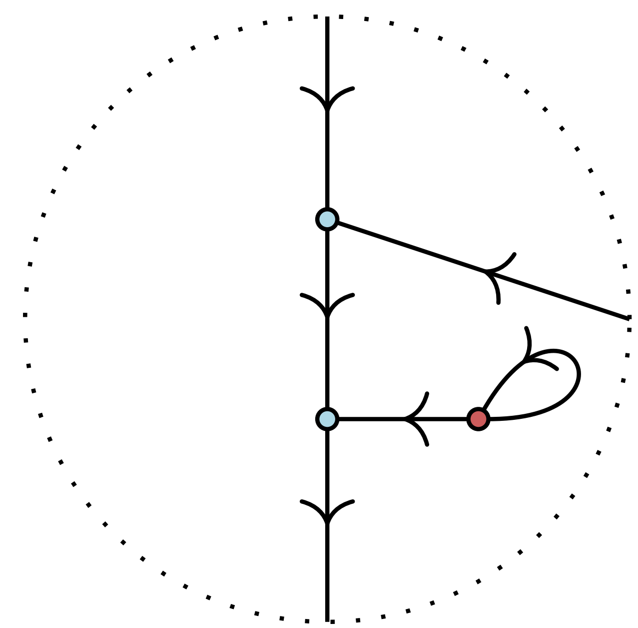
From linear terms to rooted 3-valent maps via string diagrams



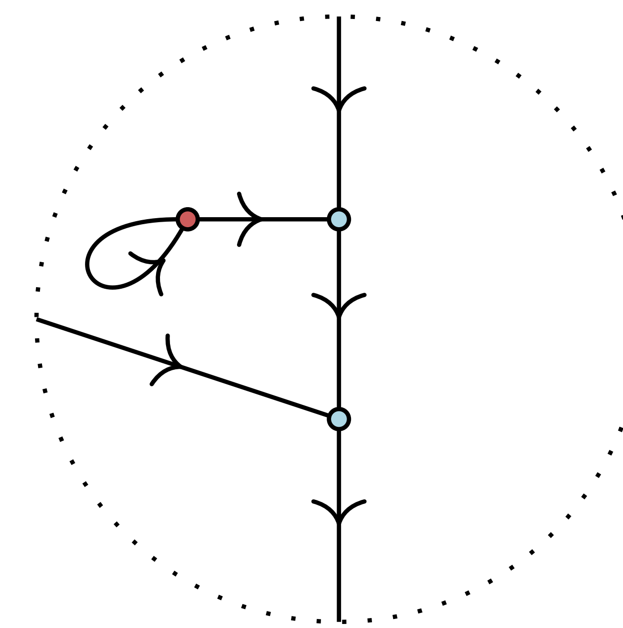
$\lambda x.\lambda y.\lambda z.x(yz)$



$\lambda x.\lambda y.\lambda z.(xz)y$

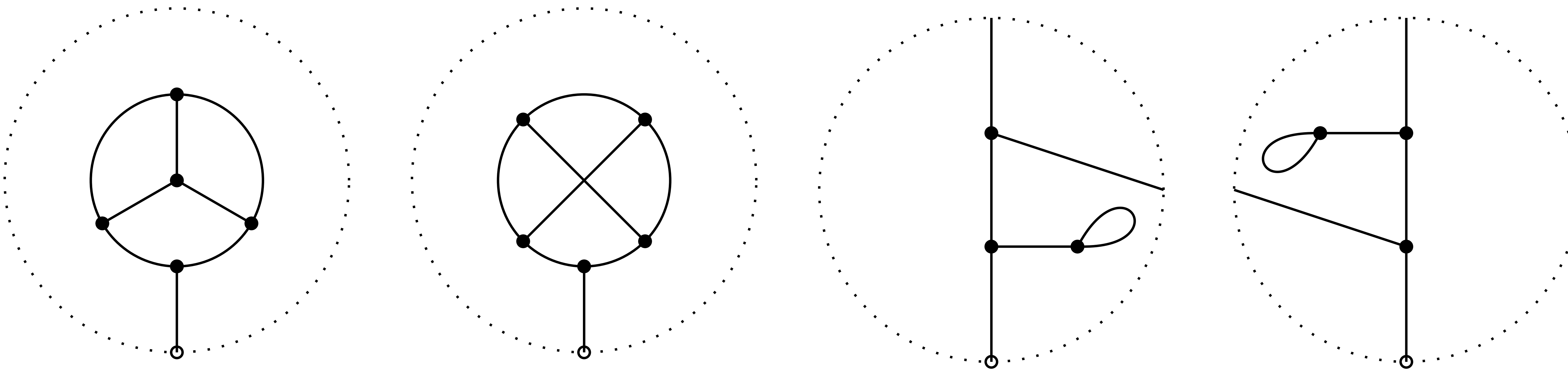


$x,y \vdash (xy)(\lambda z.z)$



$x,y \vdash x((\lambda z.z)y)$

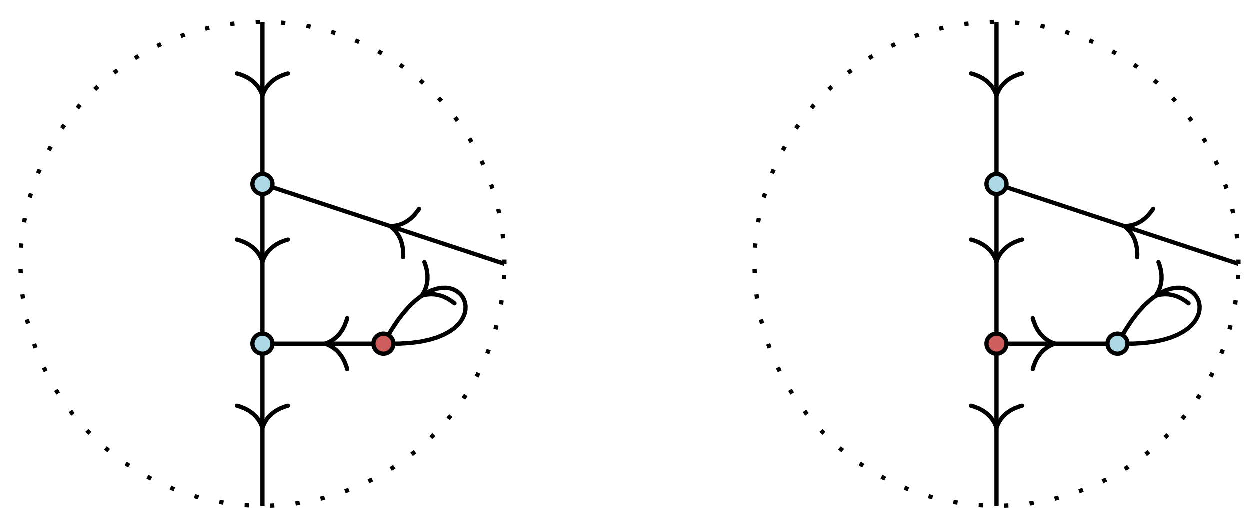
From linear terms to rooted 3-valent maps via string diagrams



$\lambda x.\lambda y.\lambda z.x(yz)$ $\lambda x.\lambda y.\lambda z.(xz)y$ $x,y \vdash (xy)(\lambda z.z)$ $x,y \vdash x((\lambda z.z)y)$

Diagrams versus Terms

Note: two different diagrams can correspond to the same underlying map.

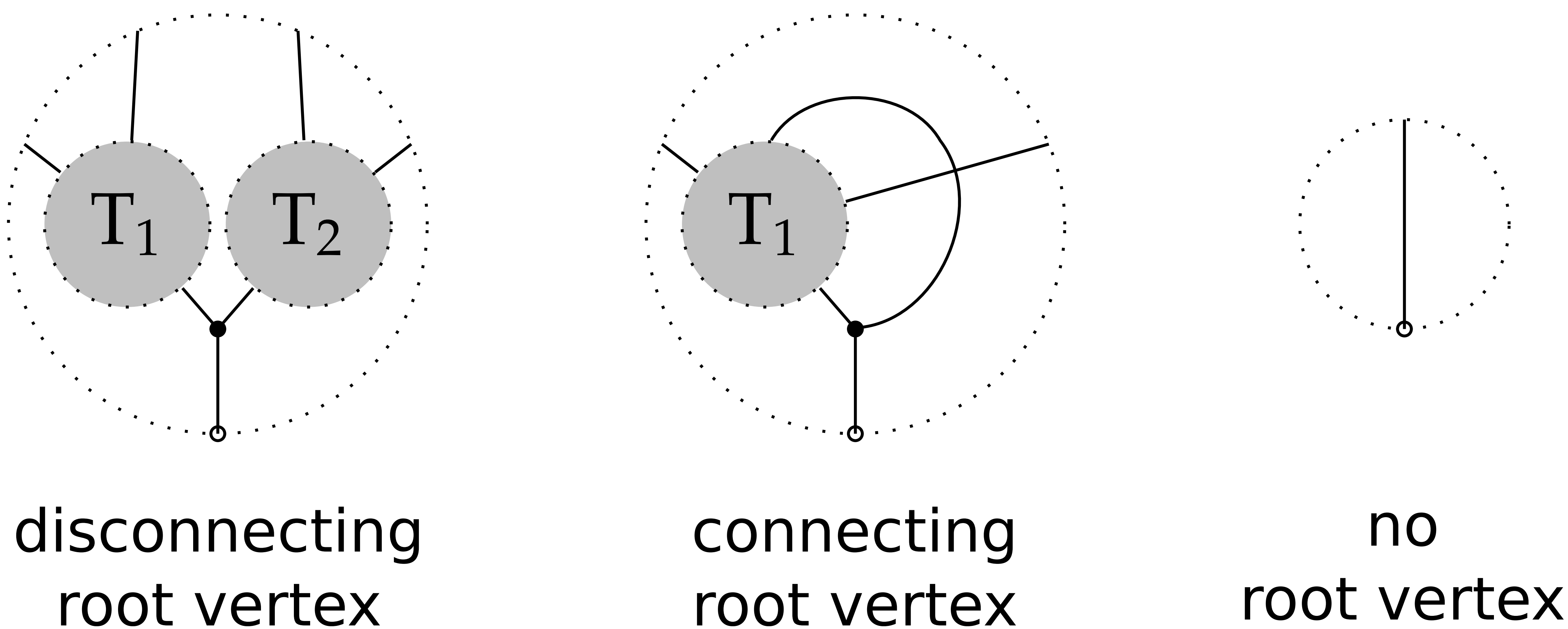


Indeed, a diagram is just a 3-valent map + a **proper orientation**.

Proposition: every rooted 3-valent map has a *unique* orientation corresponding to the diagram of a linear lambda term.

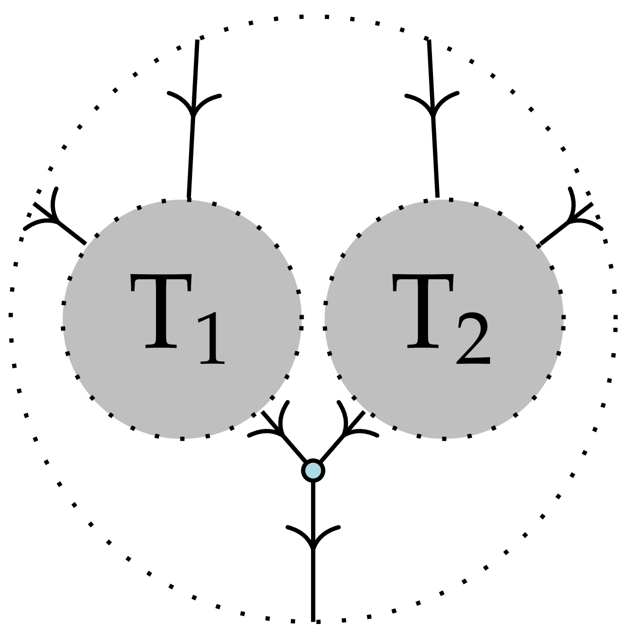
Rooted 3-valent maps, inductively

Observation: any rooted 3-valent map must have one of the following forms.

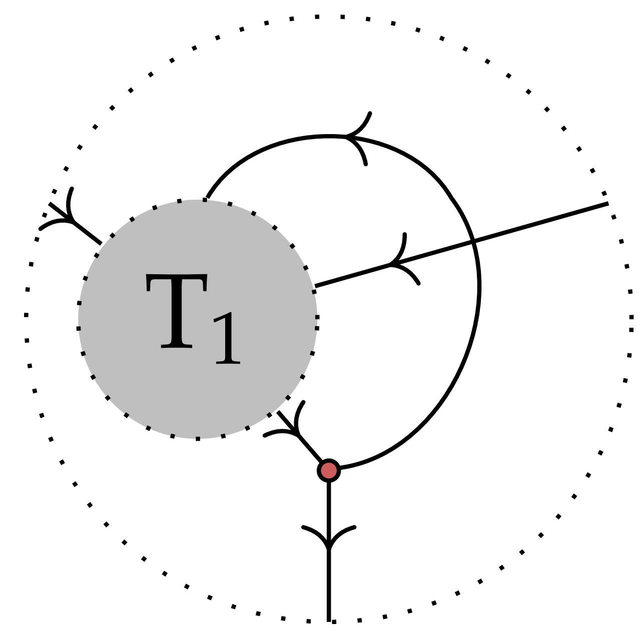


Linear lambda terms, inductively

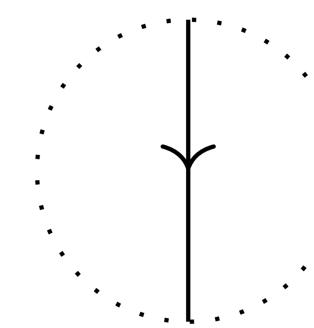
...but this exactly mirrors the inductive structure of linear lambda terms!



application

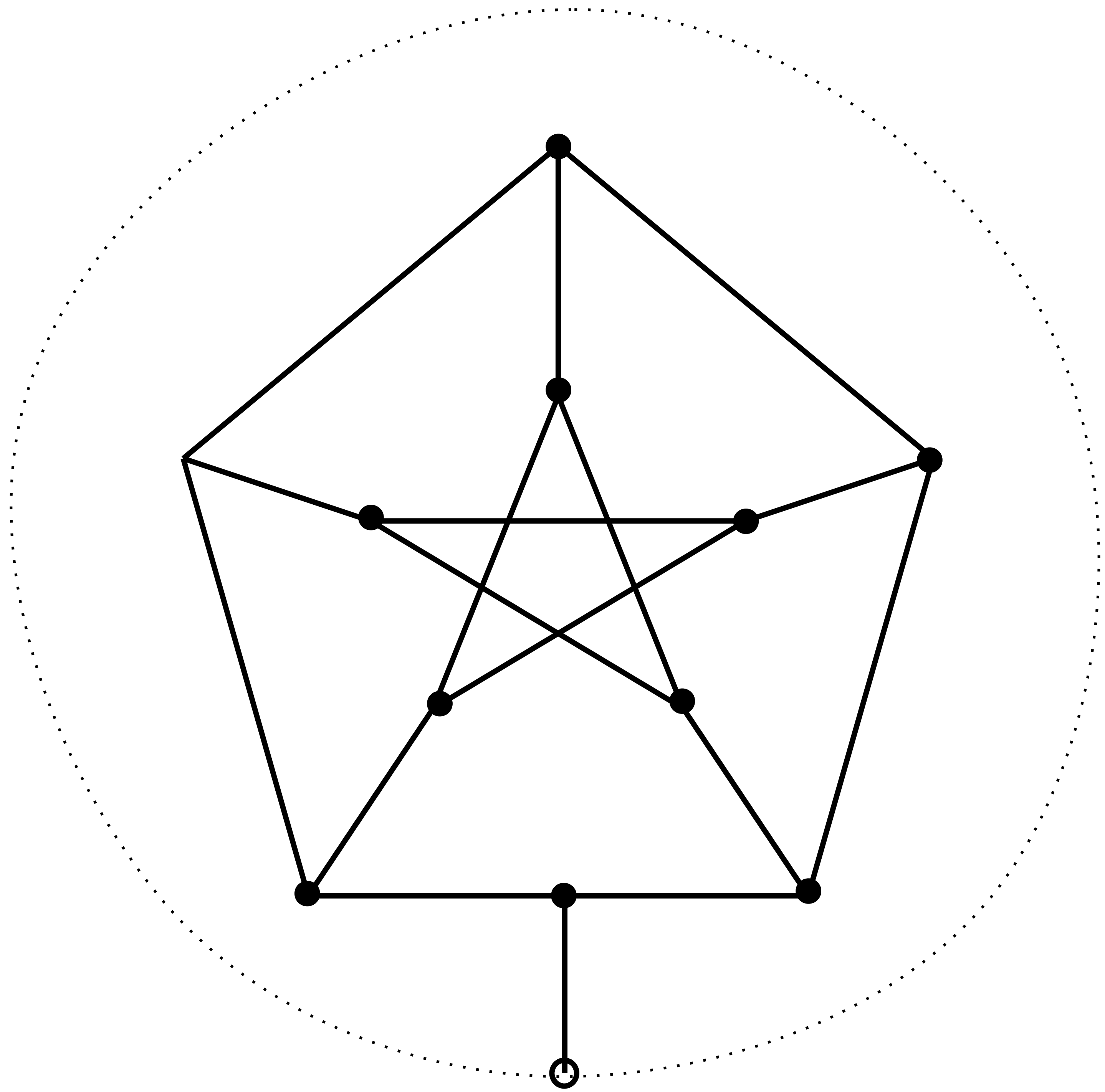


abstraction

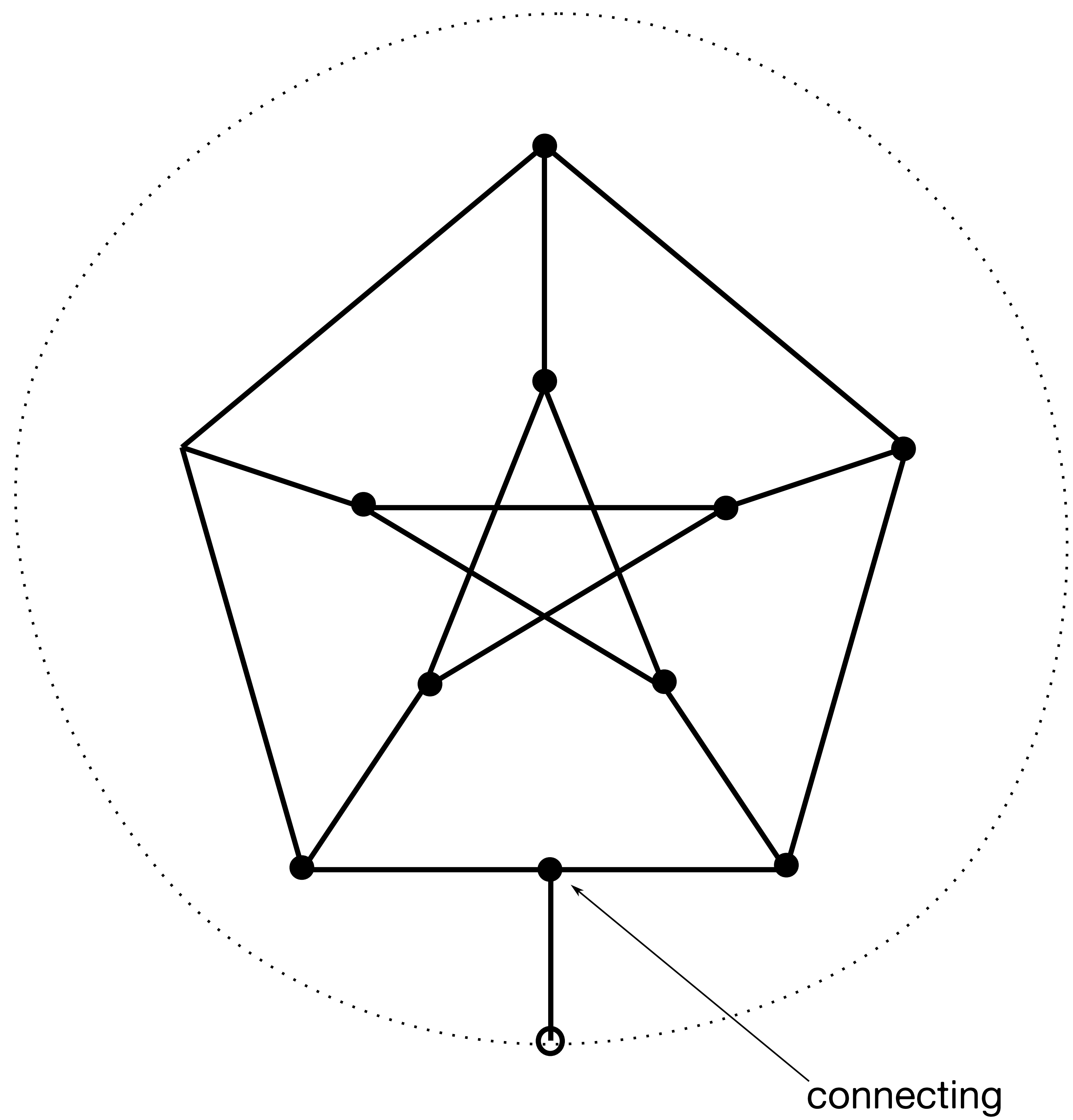


variable

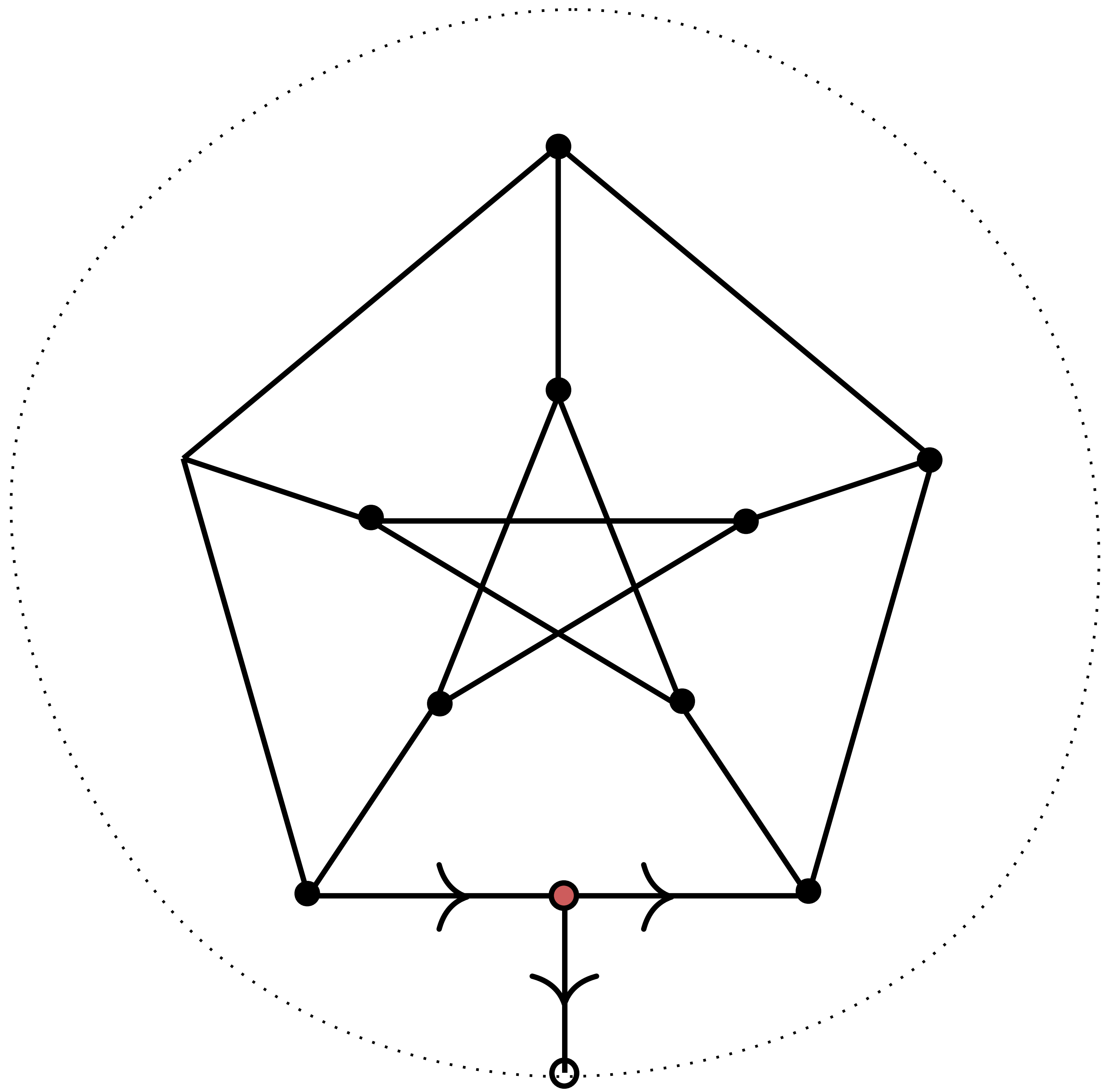
An example



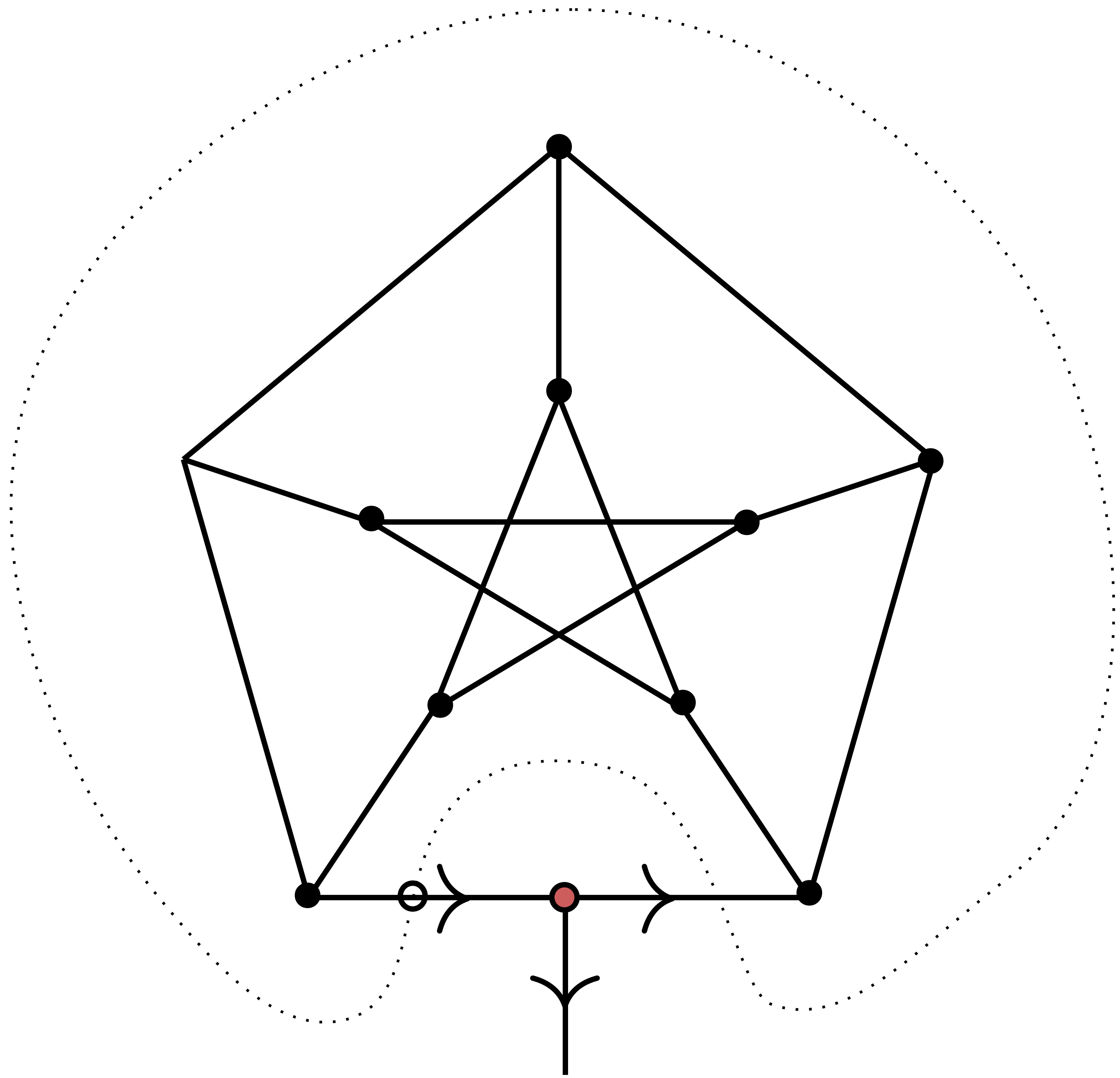
An example



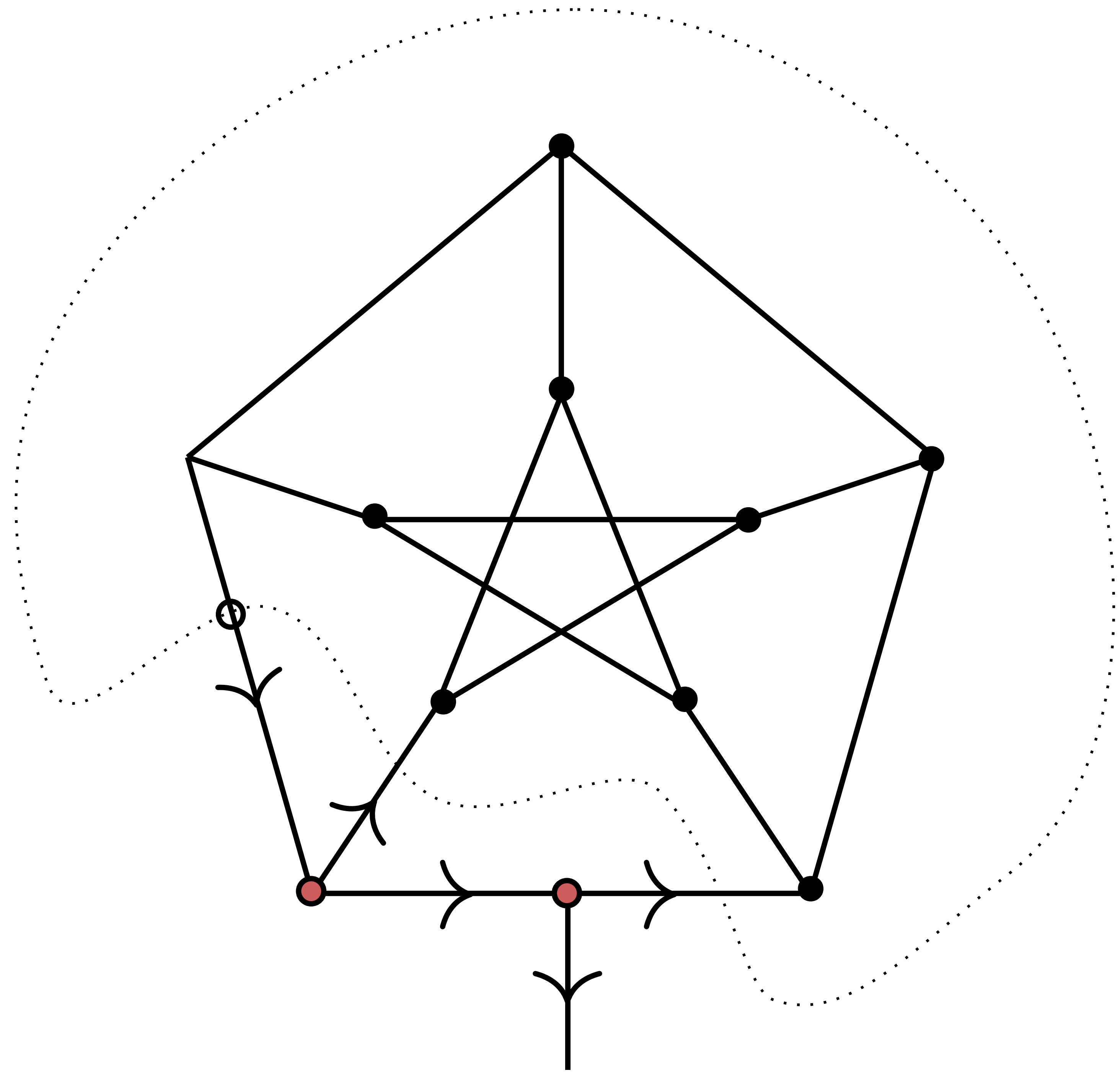
An example



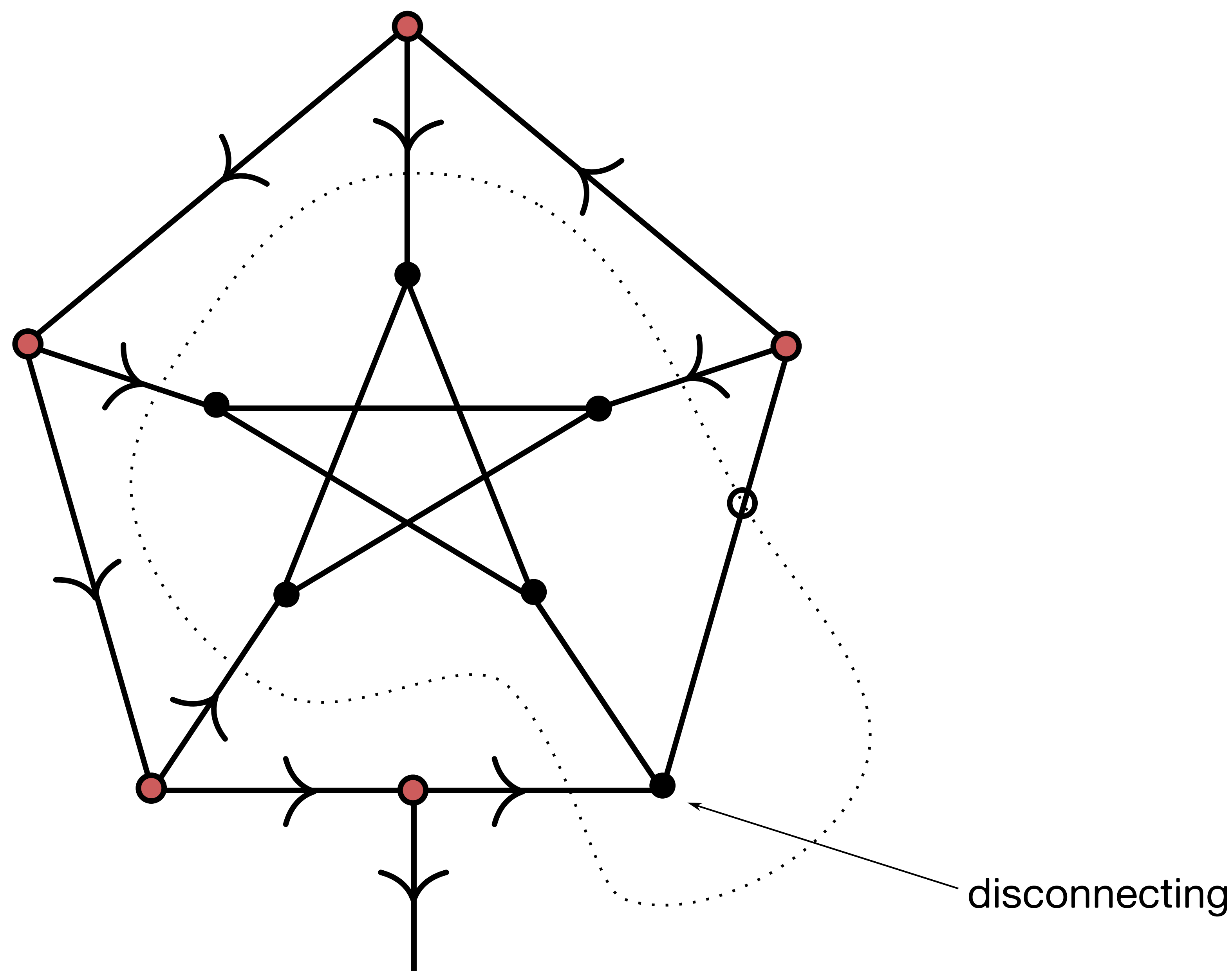
An example



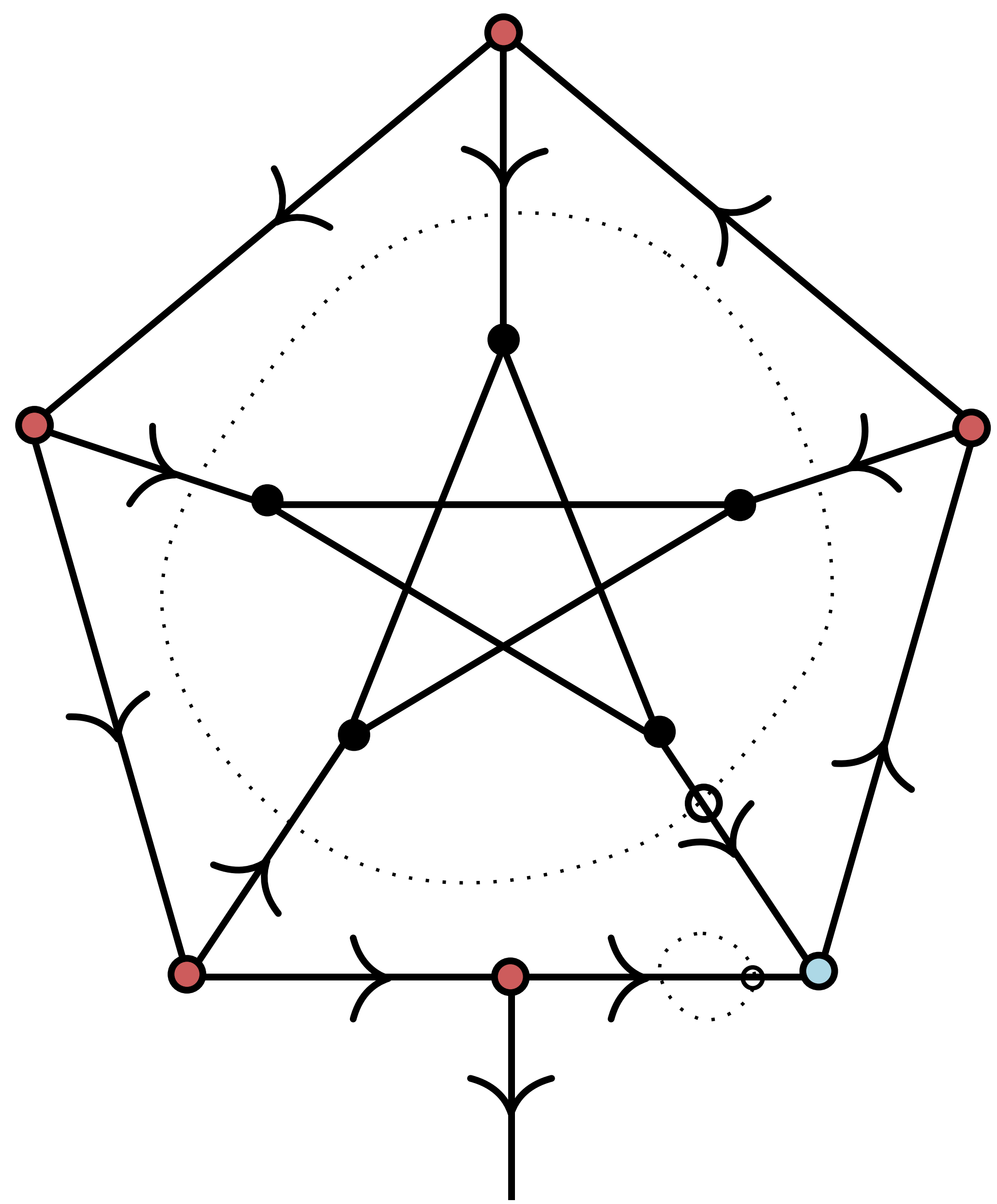
An example



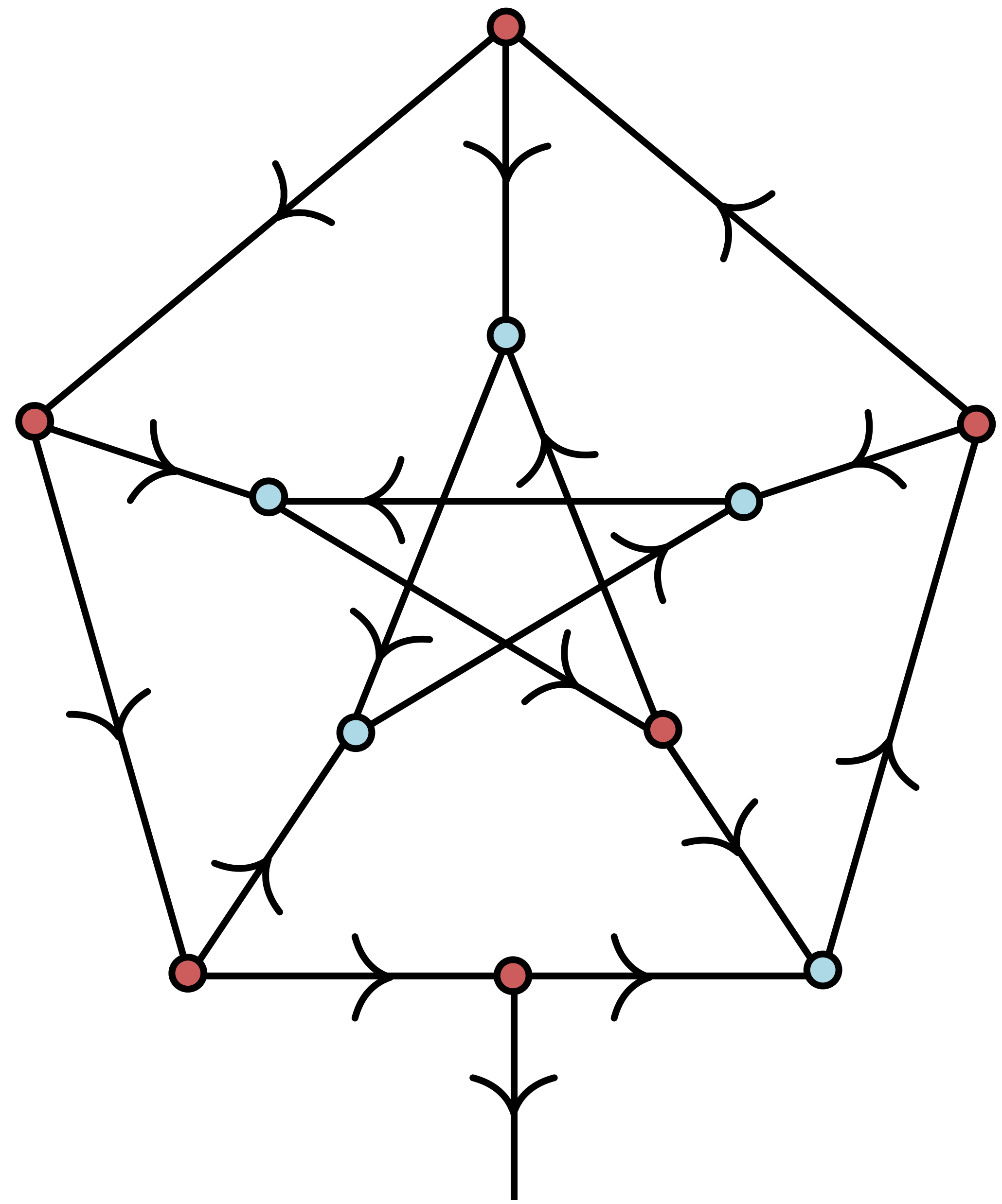
An example



An example



An example



λa.λb.λc.λd.λe.a(λf.c(e(b(df))))

An operadic perspective

Let $\Theta(n)$ = set of isomorphism classes of rooted 3-valent maps with n non-root boundary arcs.

Θ defines a **symmetric operad** equipped with operations

$$@ : \Theta(m) \times \Theta(n) \to \Theta(m+n)$$

$$\lambda_i : \Theta(m+1) \to \theta(m) \quad [1 \leq i \leq m+1]$$

naturally isomorphic to the operad of linear lambda terms.

An operadic perspective

Moreover, Θ has some natural suboperads:

Θ_0 = the *non-symmetric* operad of **planar** 3-valent maps
= **ordered** linear lambda terms (i.e., no exchange rule)

Θ^2 = the *constant-free* operad of **bridgeless** maps
= linear terms **with no closed subterms** ("unitless")

Θ_0^2 = rooted bridgeless planar 3-valent maps
= ordered linear terms with no closed subterms

Linear typing

$$\frac{\Gamma \vdash t : A \multimap B \qquad \Delta \vdash u : A}{\Gamma, \Delta \vdash t\,u : B} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \multimap B} \qquad \frac{}{x : A \vdash x : A}$$

$$\frac{\Gamma, y : B, x : A, \Delta \vdash t : C}{\Gamma, x : A, y : B, \Delta \vdash t : C}$$

Linear typing

$$\frac{\Gamma \vdash t : A \multimap B \qquad \Delta \vdash u : A}{\Gamma, \Delta \vdash t\,u : B} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \multimap B} \qquad \frac{}{x : A \vdash x : A}$$

$$\frac{\Gamma, y : B, x : A, \Delta \vdash t : C}{\Gamma, x : A, y : B, \Delta \vdash t : C}$$

Imagine (because why not?) that we draw types from the Klein Four Group $\mathbb{V} = \mathbb{Z}_2 \times \mathbb{Z}_2$, with $A \multimap B := B - A$.

Claim: Every unitless ordered linear term has a $\mathbb{V}$-typing such that no subterm is assigned the unit type (0,0).

Linear typing

$$\frac{\Gamma \vdash t : A \multimap B \qquad \Delta \vdash u : A}{\Gamma, \Delta \vdash t\,u : B} \qquad \frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x.t : A \multimap B} \qquad \frac{}{x : A \vdash x : A}$$

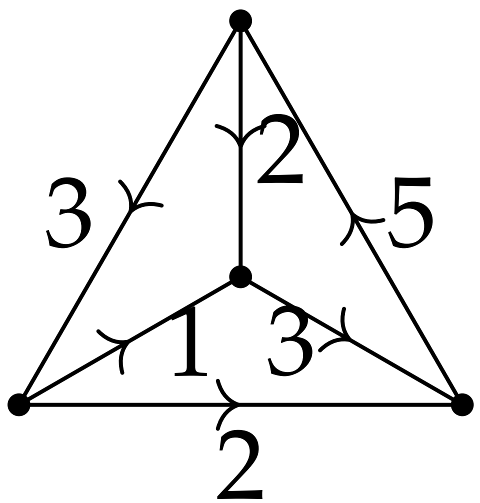
$$\frac{\Gamma, y : B, x : A, \Delta \vdash t : C}{\Gamma, x : A, y : B, \Delta \vdash t : C}$$

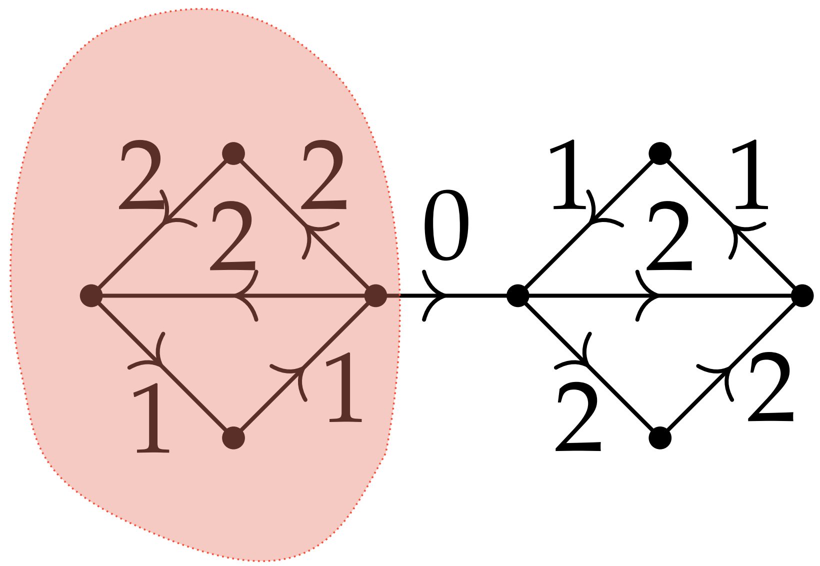
Imagine (because why not?) that we draw types from the Klein Four Group $\mathbb{V} = \mathbb{Z}_2 \times \mathbb{Z}_2$, with $A \multimap B := B - A$.

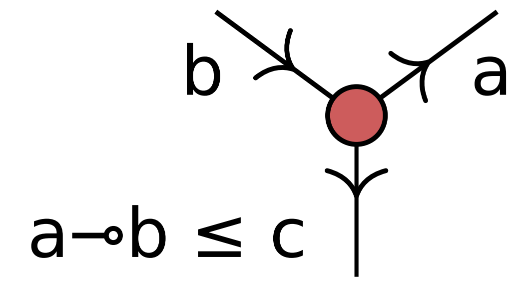
Claim: Every unitless ordered linear term has a $\mathbb{V}$-typing such that no subterm is assigned the unit type (0,0).

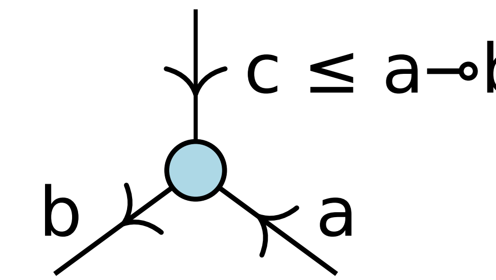
Proof: **This is equivalent to 4CT.**

punchline: linear typing is more subtle than you think!



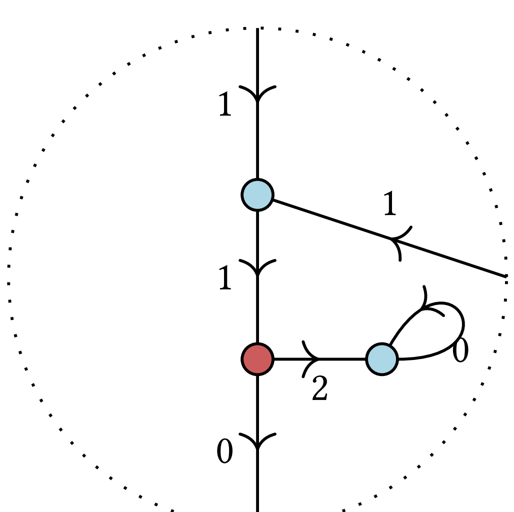


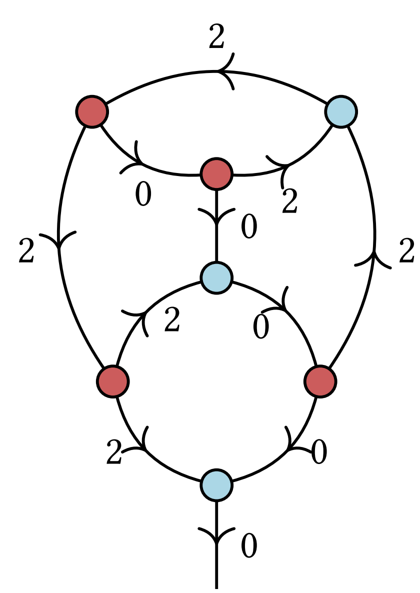




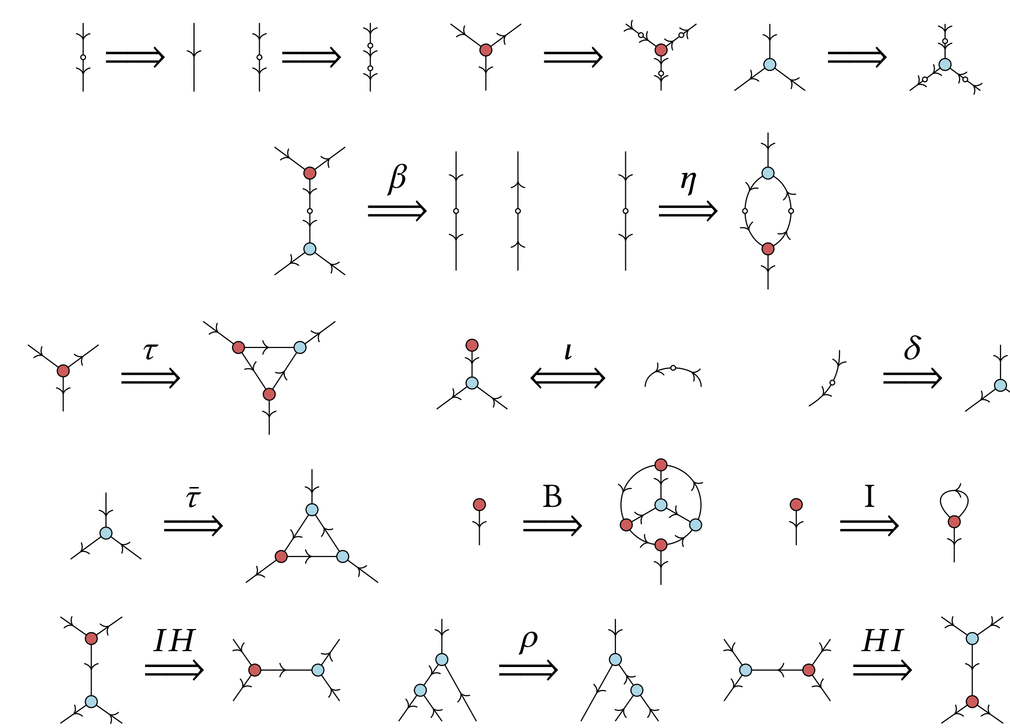
Part Two:
Idea of the Paper

Example 3.16. A pair of non-global $\hat{2}$-flows on non-lambda terms:





Proposition 4.2. *The following are imploid moves:*



Flows and nowhere-zero flows

Behind the scenes, what the lambda calculus formulation of 4CT really does is express the existence of a nowhere-zero $\mathbb{V}$-flow as a typing problem.

W. T. Tutte (1954). A contribution to the theory of chromatic polynomials.

A **flow** on an oriented graph, valued in an ab gp G, is an assignment φ : E → G such that

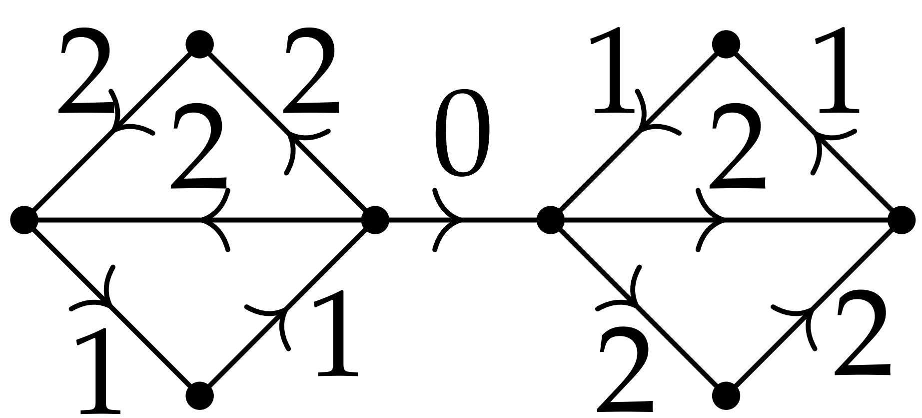
$$\sum_{x \in in(v)} \varphi(x) = \sum_{x \in out(v)} \varphi(x) \qquad \text{(Kirchhoff's law)}$$

holds at every vertex v ∈ V. A flow φ is **nowhere-zero** if φ(x) ≠ 0 for all x ∈ E.

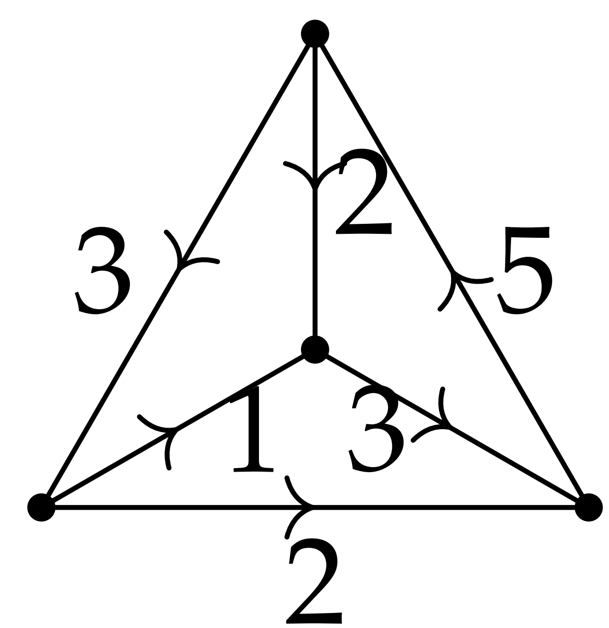
Flows and nowhere-zero flows

Behind the scenes, what the lambda calculus formulation of 4CT really does is express the existence of a nowhere-zero $\mathbb{V}$-flow as a typing problem.

W. T. Tutte (1954). A contribution to the theory of chromatic polynomials.



a $\mathbb{Z}_3$-flow



a nowhere-zero $\mathbb{Z}$-flow

Linear typings as flows

Goal: develop a more general theory of linear typings-as-flows on 3-valent maps.

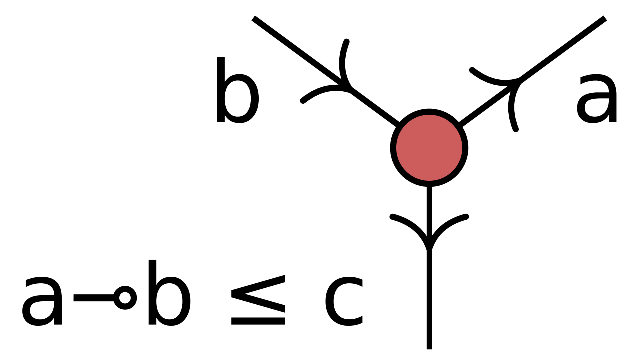
The LICS paper represents a preliminary exploration of such a theory, starting from the idea of replacing abelian groups by more general algebraic objects I call "imploids".

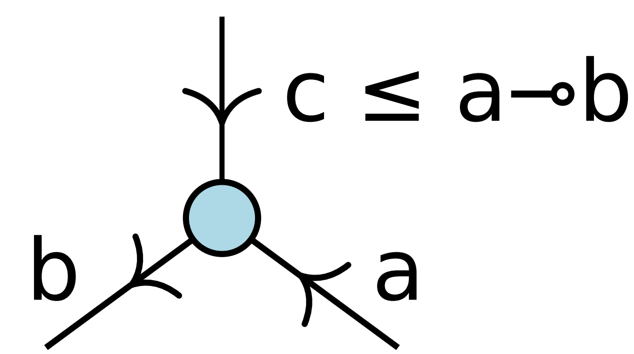
An **imploid** is just a preordered set equipped with an "implication" operation $\multimap$ and element I satisfying three natural laws of composition, identity, and unit.

(Another name for an imploid is a [skew-]**closed preorder**).

Linear typings as flows

Imploid-valued flows are defined by the following pair of local flow relations:





This notion makes sense for any well-oriented 3-valent map, but in the case of a linear lambda term it specializes to standard linear typing (with subtyping).

Also, we can speak of nowhere-unit flows (typings) as flows (typings) where no edge (subterm) is assigned a value above I.

Linear typings as flows

The paper mainly addresses two questions:

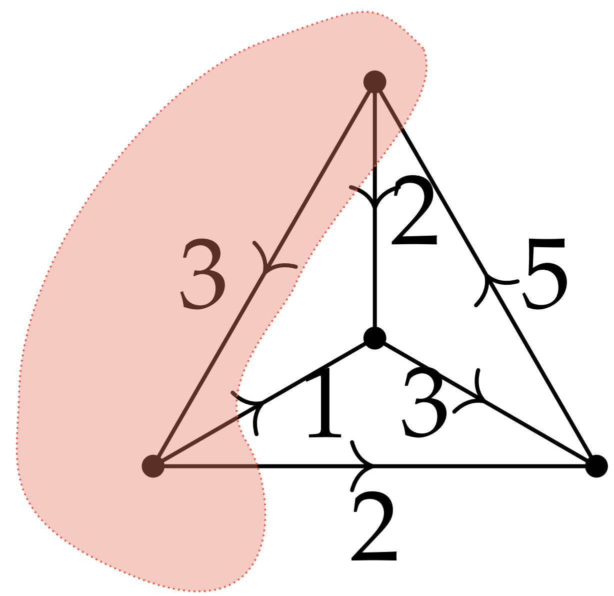
1. When does a well-oriented 3-valent map satisfy the **global extension** property?

2. How do moves such as **β-reduction** and **η-expansion** act on flows?

Additionally, the paper briefly discusses a polarized notion of flow, which draws connections to the theory of proof-nets in linear logic and to bidirectional typing.

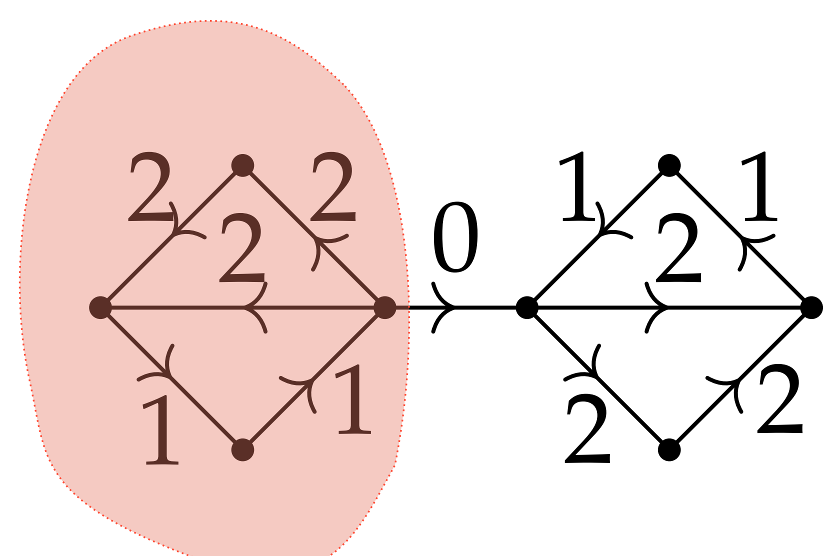
The global extension property

For classical (abelian group-valued) flows, it is easy to show that Kirchhoff's law extends to any induced subgraph.



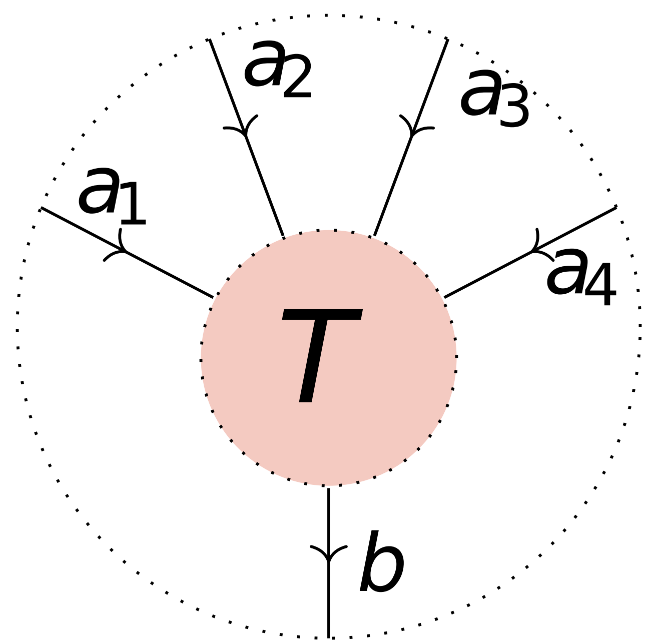
5 = 2 + 1 + 2

Corollary: any graph with a bridge cannot have a nowhere-zero flow.



The global extension property

For imploid-valued flows, we can similarly ask whether the local flow conditions may be lifted to a global flow relation across the boundary.

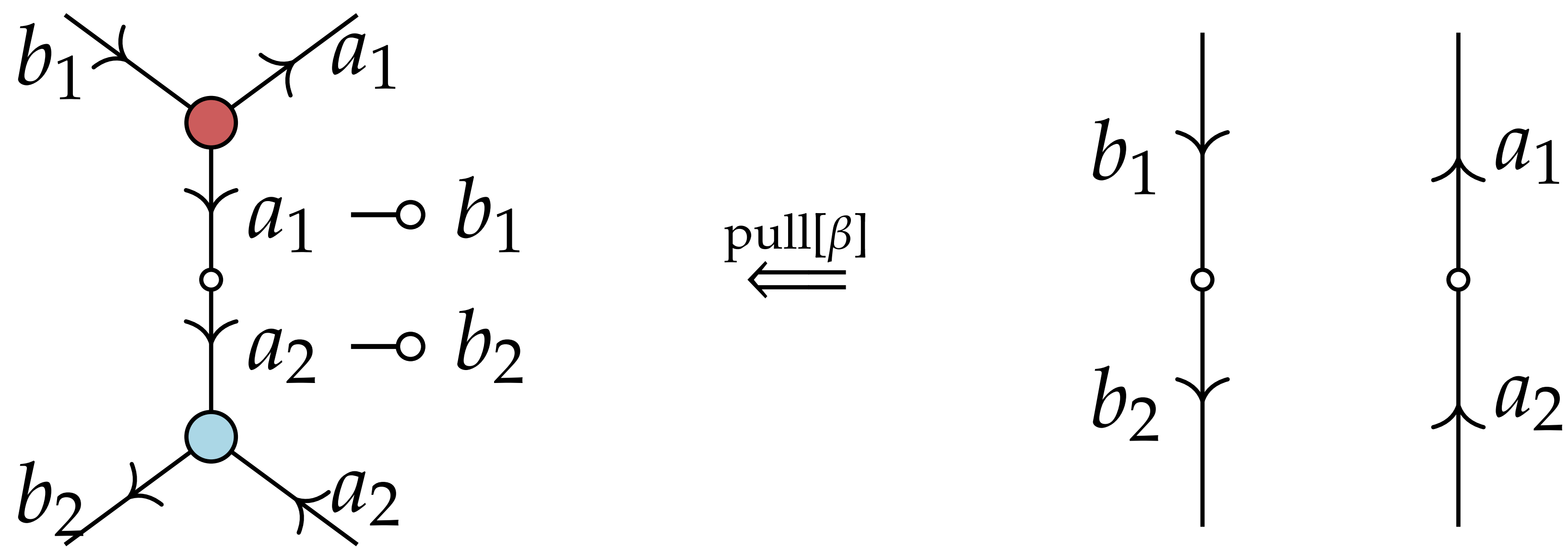


$I \leq a_1 \multimap a_2 \multimap a_3 \multimap a_4 \multimap b$

Theorem: T has the global extension property with respect to all symmetric imploids iff T has the unique orientation of a linear lambda term.
(In the planar case the symmetry condition may be dropped.)

Rewriting of flows

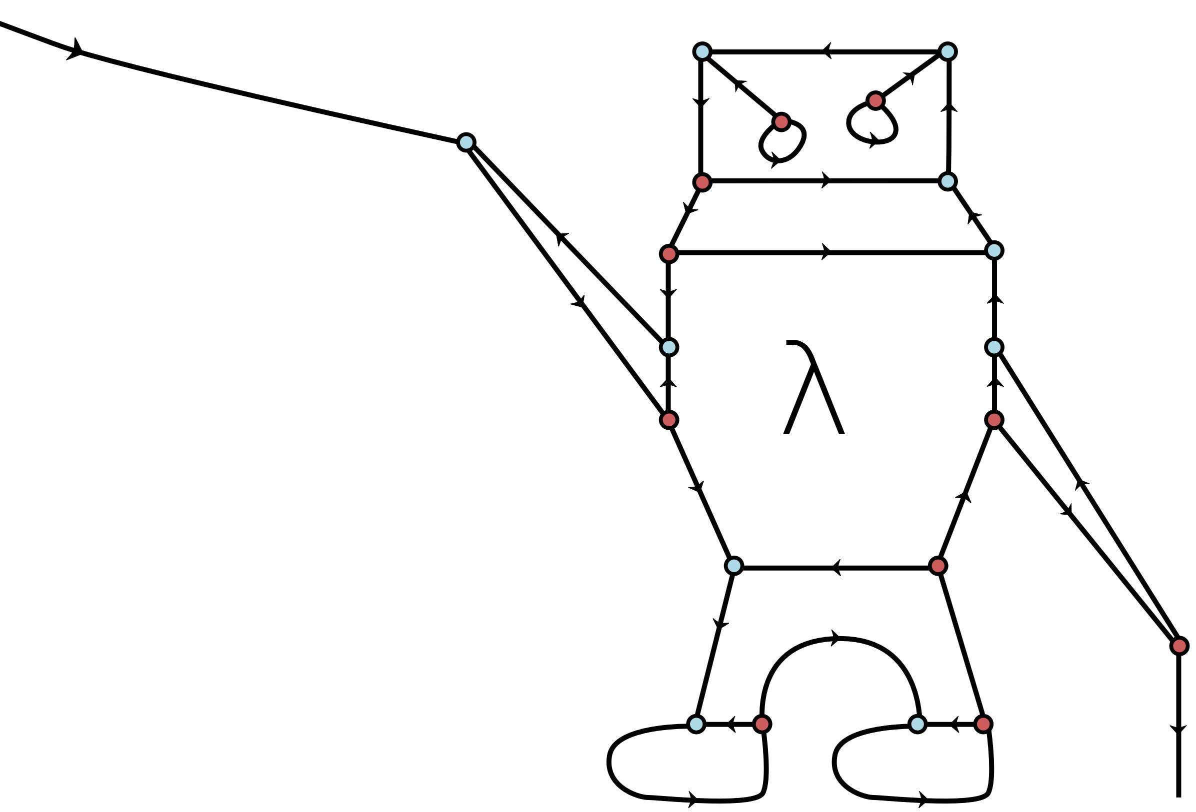
In general, flows can be pulled back across rewriting moves like β-reduction and η-expansion, but not necessarily pushed forward.



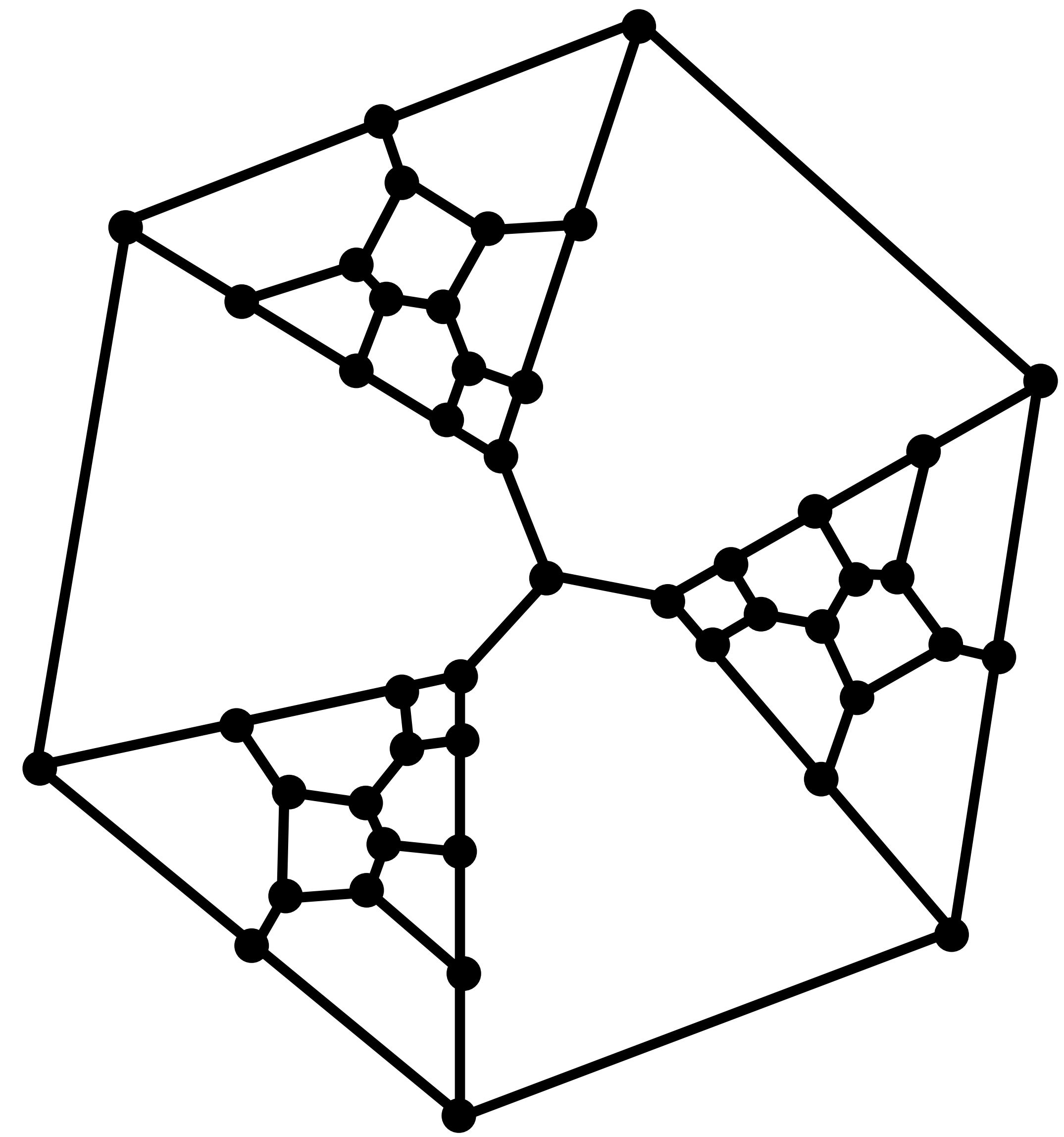
We refer to moves admitting such a pullback interpretation as "imploid moves".

Theorem (roughly): there are a finite set of imploid moves which generate all rooted 3-valent maps with their unique orientations as linear lambda terms. (This is closely related to the "BCI" completeness theorem in combinatory logic.)

Part Three:
One More Example

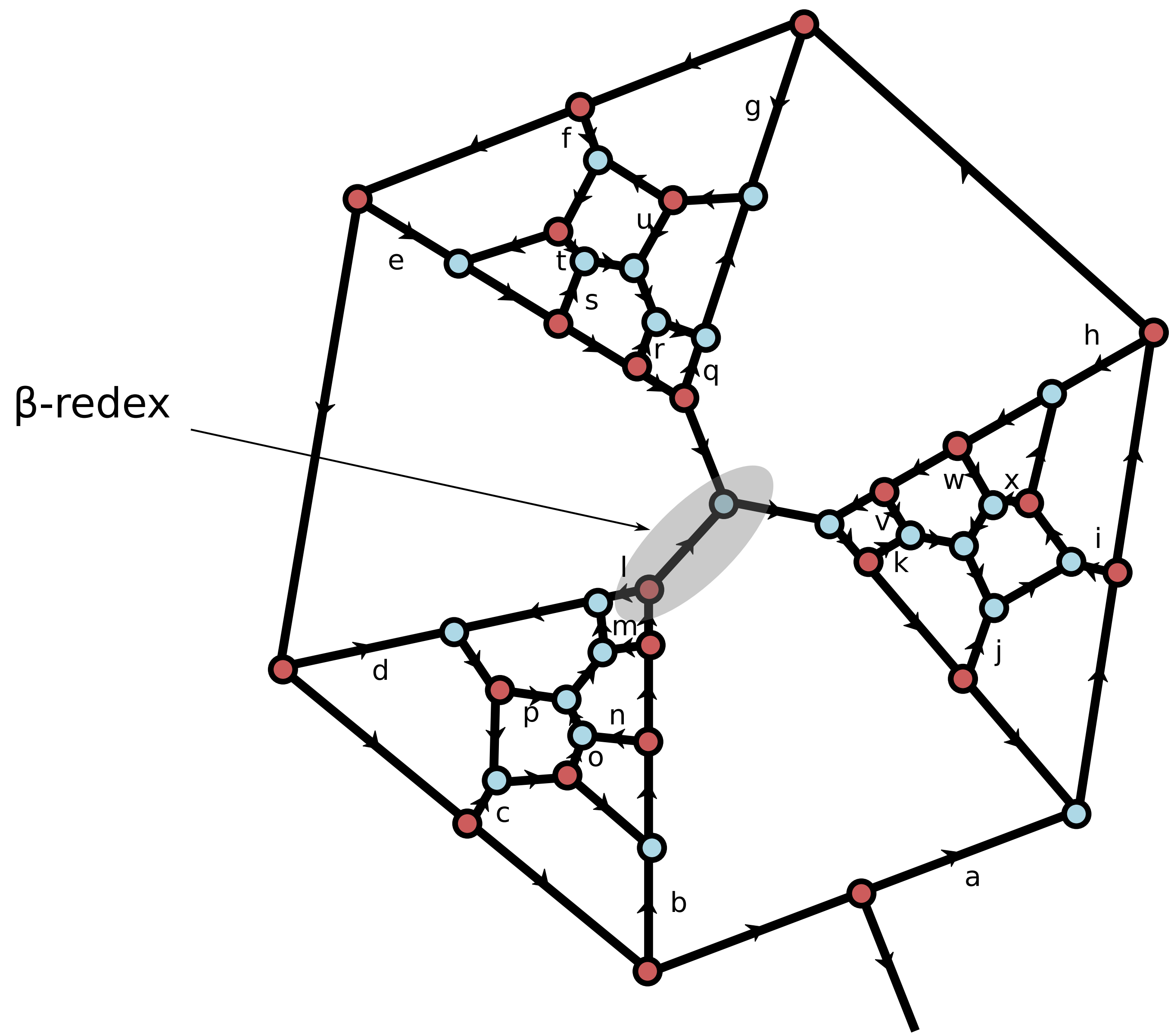


The Tutte Graph



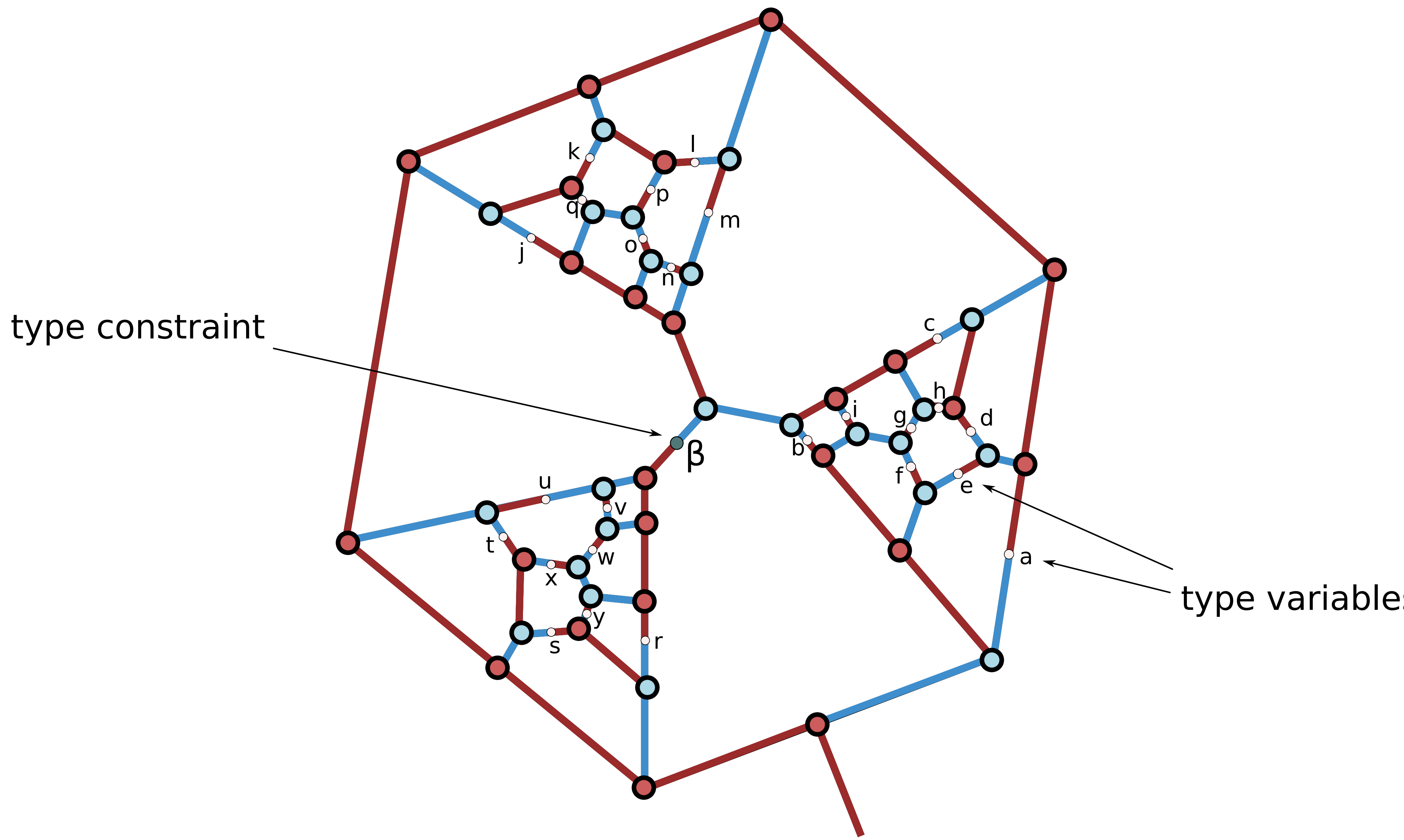
(From W. T. Tutte, “On Hamiltonian Circuits”, *Journal of the London Mathematical Society* 21 (1946), 98–101.)

The associated lambda term

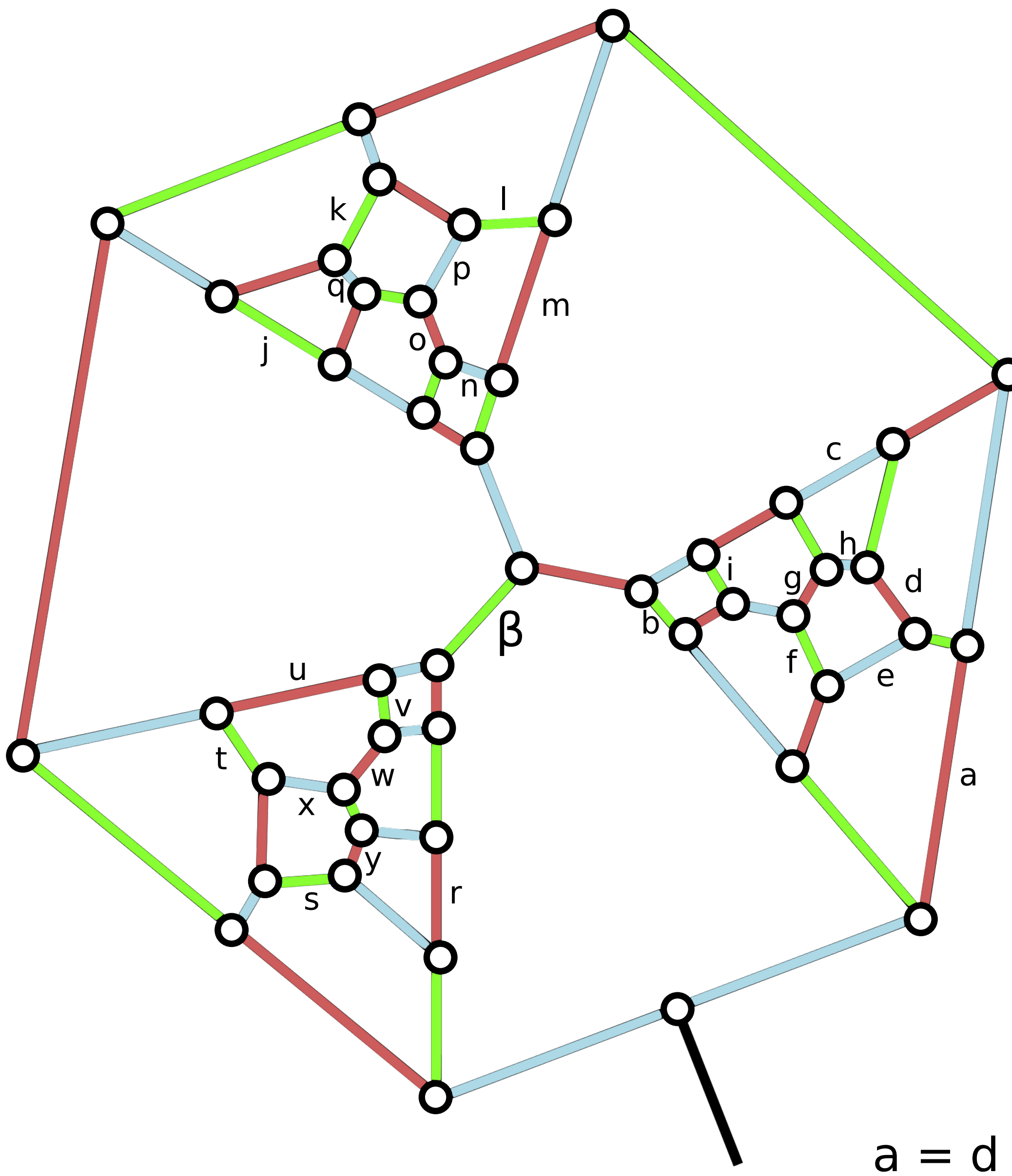


$$\lambda a\lambda b\lambda c\lambda d\lambda e\lambda f\lambda g\lambda h\lambda i.a(\lambda j\lambda k.\underline{((\lambda l\lambda m\lambda n.b(\lambda o.c(\lambda p.d(l(m((no)p)))))))(\lambda q\lambda r\lambda s.e(\lambda t.f(\lambda u.g(q(r((st)u)))))))}(\lambda v\lambda w.h(\lambda x.i(j((kv)(wx)))))))$$

The principal polarized flow



A $\mathbb{V}$-typing

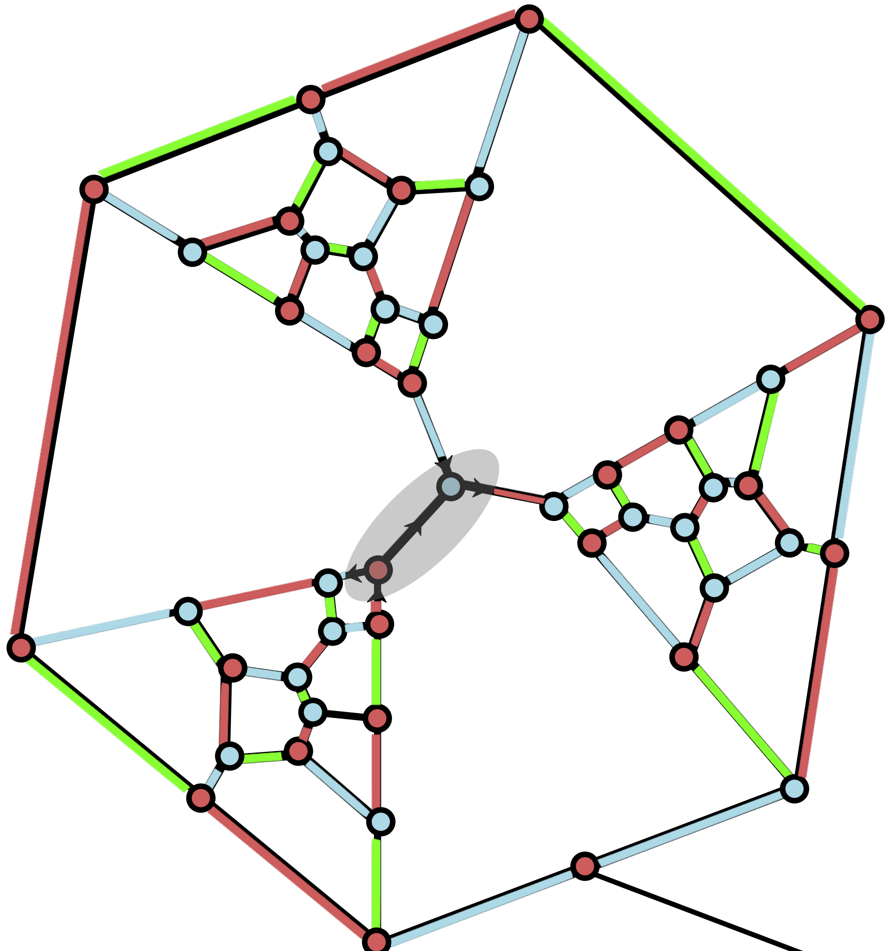


a = d = g = m = o = r = u = w = y = R

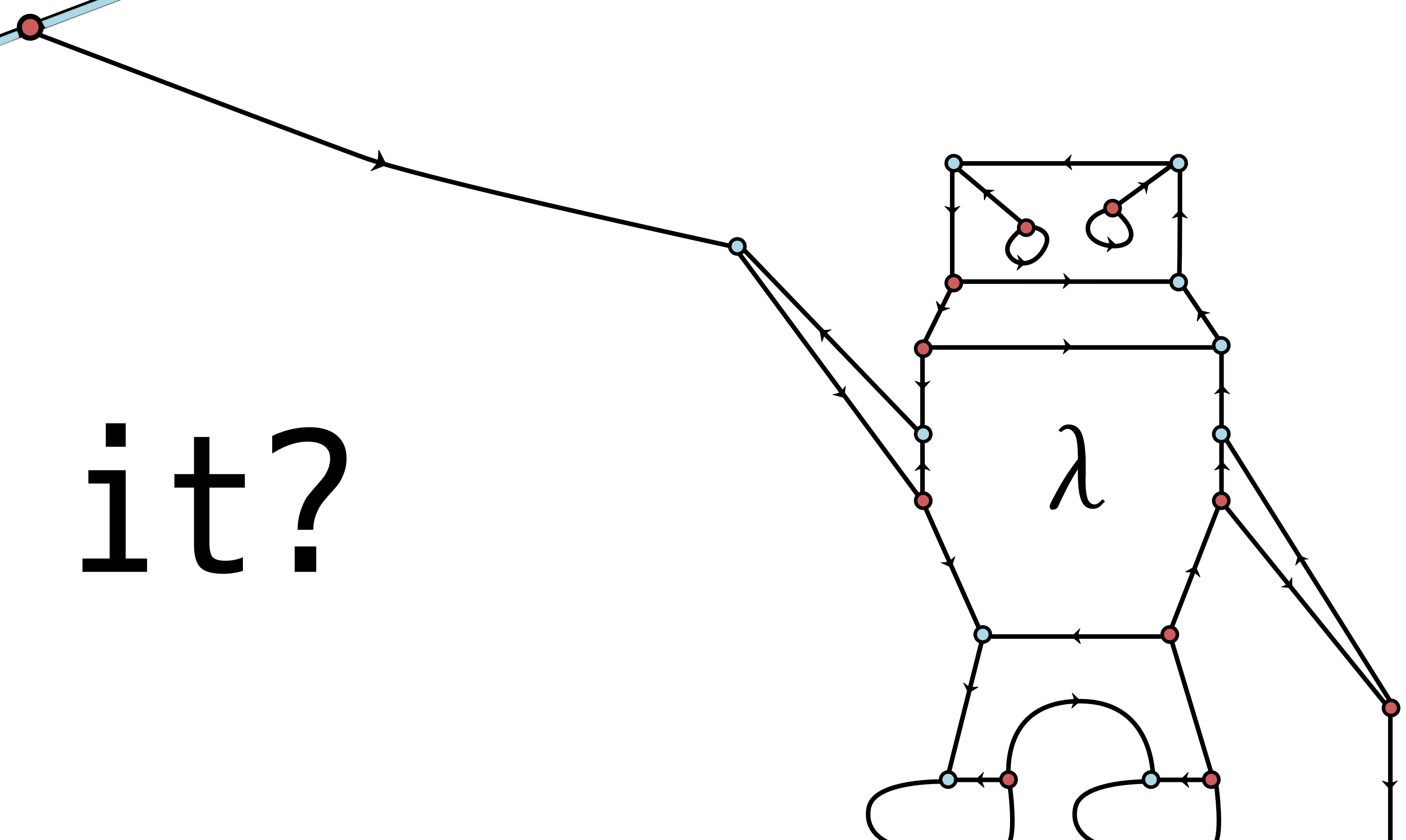
b = f = i = j = k = l = s = t = v = G

c = e = h = n = p = q = x =B

β : G = G



The End



...or is it?