Renormalization and Computation Dyson–Schwinger equations in the theory of computation

Matilde Marcolli

based on:

- Colleen Delaney, Matilde Marcolli, *Dyson-Schwinger equations in the theory of computation*, arXiv:1302.5040
- Yuri Manin, *Renormalization and computation*, I and II, arXiv:0904.4921 and arXiv:0908.3430

Perturbative Quantum Field Theory

- Action functional in D dimensions

$$S(\phi) = \int \mathscr{L}(\phi) d^D x = S_0(\phi) + S_{int}(\phi)$$

- Lagrangian density

$$\mathscr{L}(\phi) = \frac{1}{2}(\partial\phi)^2 - \frac{m^2}{2}\phi^2 - \mathscr{L}_{int}(\phi)$$

- Perturbative expansion: Feynman rules and Feynman diagrams

$$S_{eff}(\phi) = S_0(\phi) + \sum_{\Gamma} \frac{\Gamma(\phi)}{\#\mathrm{Aut}(\Gamma)} \quad \text{(1PI graphs)}$$

- Generating functional $Z[J]$ of Green functions (source field J)

$$\frac{\delta^n Z}{\delta J(x_1)\cdots\delta J(x_n)}[0] = i^n Z[0]\langle\phi(x_1)\cdots\phi(x_n)\rangle$$

Algebraic renormalization in perturbative QFT

- A. Connes, D. Kreimer, *Renormalization in quantum field theory and the Riemann-Hilbert problem*, I and II, hep-th/9912092, hep-th/0003188
- A. Connes, M. Marcolli, *Renormalization, the Riemann-Hilbert correspondence, and motivic Galois theory*, hep-th/0411114
- K. Ebrahimi-Fard, L. Guo, D. Kreimer, *Integrable Renormalization II: the general case*, hep-th/0403118

Two step procedure:

• Regularization: replace divergent integral $U(\Gamma)$ by function with poles
• Renormalization: pole subtraction with consistency over subgraphs (Hopf algebra structure)

• Kreimer, Connes–Kreimer, Connes–M.: Hopf algebra of Feynman graphs and BPHZ renormalization method in terms of Birkhoff factorization and differential Galois theory

• Ebrahimi-Fard, Guo, Kreimer: algebraic renormalization in terms of Rota–Baxter algebras

Connes–Kreimer Hopf algebra $\mathscr{H} = \mathscr{H}(\mathscr{T})$ (depends on theory)

- Free commutative algebra in generators Γ 1PI Feynman graphs
- Grading: loop number (or internal lines)

$$\deg(\Gamma_1 \cdots \Gamma_n) = \sum_i \deg(\Gamma_i), \quad \deg(1) = 0$$

- Coproduct:

$$\Delta(\Gamma) = \Gamma \otimes 1 + 1 \otimes \Gamma + \sum_{\gamma \in \mathscr{V}(\Gamma)} \gamma \otimes \Gamma/\gamma$$

- Antipode: inductively

$$S(X) = -X - \sum S(X')X''$$

for $\Delta(X) = X \otimes 1 + 1 \otimes X + \sum X' \otimes X''$

Rota–Baxter algebra of weight $\lambda = -1$

$\mathscr{R}$ commutative unital algebra
$T : \mathscr{R} \to \mathscr{R}$ linear operator with

$$T(x)T(y) = T(xT(y)) + T(T(x)y) + \lambda T(xy)$$

• Example: $T =$ projection onto polar part of Laurent series

• T determines splitting $\mathscr{R}_+ = (1 - T)\mathscr{R}$, $\mathscr{R}_- =$ unitization of $T\mathscr{R}$; both $\mathscr{R}_\pm$ are algebras

Feynman rule

• $\phi : \mathscr{H} \to \mathscr{R}$ *commutative algebra homomorphism*

from CK Hopf algebra $\mathscr{H}$ to Rota–Baxter algebra $\mathscr{R}$ weight -1

$$\phi \in \mathrm{Hom}_{\mathrm{Alg}}(\mathscr{H}, \mathscr{R})$$

• Note: ϕ does *not know* that $\mathscr{H}$ Hopf and $\mathscr{R}$ Rota-Baxter, only commutative algebras

• Birkhoff factorization $\exists \phi_\pm \in \mathrm{Hom}_{\mathrm{Alg}}(\mathscr{H}, \mathscr{R}_\pm)$

$$\phi = (\phi_- \circ S) \star \phi_+$$

where $\phi_1 \star \phi_2(X) = \langle \phi_1 \otimes \phi_2, \Delta(X) \rangle$

• Connes-Kreimer inductive formula for Birkhoff factorization:

$$\phi_-(X) = -T(\phi(X) + \sum \phi_-(X')\phi(X''))$$

$$\phi_+(X) = (1 - T)(\phi(X) + \sum \phi_-(X')\phi(X''))$$

where $\Delta(X) = 1 \otimes X + X \otimes 1 + \sum X' \otimes X''$

• Recovers what known in physics as BPHZ renormalization procedure in physics

Hopf algebra of rooted trees

- Rooted tree τ: data $(F_\tau, V_\tau, v_\tau, \delta_\tau, j_\tau)$
  - F_τ set of half-edges (flags)
  - V_τ set of vertices
  - distinguished $v_\tau \in V_\tau$ (the root)
  - boundary map $\partial_\tau : F_\tau \to V_\tau$
  - involution $j_\tau : F_\tau \to F_\tau$, $j_\tau^2 = 1$ gluing half-edges to edges
  - E_τ internal edges, E_τ^{ext} external edges (fixed by involution)

Orientation: root vertex as output, all edges oriented along unique path to root

Decorations: $\phi_V : V_\tau \to \mathscr{D}_V$ labels of vertices, $\phi_F : F_\tau \to \mathscr{D}_F$ labels of flags (matched by involution) admissible cuts

• admissible cuts C of τ modify involution j_τ cutting a subset of internal edges into two flags f_i, f_i', so that every oriented path in τ from leaf to root contains at most one cut edge

• New graph is a forest

$$C(\tau) = \rho_C(\tau) \amalg \pi_C(\tau)$$

rooted tree $\rho_C(\tau)$; forest $\pi_C(\tau) = \amalg_i \pi_{C,i}(\tau)$, each tree $\pi_{C,i}(\tau)$ with single output (new roots)

Hopf algebras

• $\mathscr{H}^{nc}$ *noncommutative* Hopf algebra of *planar* rooted trees: free algebra generated by planar rooted trees, coproduct

$$\Delta(\tau) = \tau \otimes 1 + 1 \otimes \tau + \sum_C \pi_C(\tau) \otimes \rho_C(\tau)$$

grading by number of vertices, antipode

$$S(x) = -x - \sum S(x')x'', \quad \text{for } \Delta(x) = x \otimes 1 + 1 \otimes x + \sum x' \otimes x''$$

x', x'' lower order terms

• $\mathscr{H}$ *commutative* Hopf algebra of (planar) rooted trees: free commutative (polynomial) algebra generated by rooted trees, same form of coproduct, grading and antipode

• in Connes–Kreimer setting can equivalently work with Hopf algebra of rooted trees decorated by Feynman graphs or with Hopf algebra of Feynman graphs (coproduct: subgraphs and quotient graphs)

Dyson–Schwinger equations in QFT

• Equations of motion for Green functions (Euler–Lagrange equations)

• Infinite system of coupled differential equations

• obtained as formal Taylor series expansion at $J = 0$ of DS equation in the generating function $Z[J]$

$$\frac{\delta S}{\delta \phi(x)} \left[-i \frac{\delta}{\delta J} \right] Z[J] + J(x) Z[J] = 0$$

• in the Hopf algebraic approach to QFT, can lift the DS equations to the combinatorial level

Combinatorial Dyson–Schwinger equations

- C. Bergbauer and D. Kreimer, *Hopf algebras in renormalization theory: locality and Dyson-Schwinger equations from Hochschild cohomology*, hep-th/0506190
- K. Yeats, *Rearranging Dyson-Schwinger Equations*, AMS 2011.
- L. Foissy, *Systems of Dyson–Schwinger equations*, arXiv:0909.0358

Dyson–Schwinger equations and Hopf subalgebras

• If grafting operator satisfies *cocycle condition*, then solutions of Dyson–Schwinger equations form a *Hopf subalgebra*

Primitive recursive functions

- generated by *basic functions*
  - Successor $s : \mathbb{N} \to \mathbb{N}$, $s(x) = x + 1$;
  - Constant $c^n : \mathbb{N}^n \to \mathbb{N}$, $c^n(x) = 1$ (for $n \geq 0$);
  - Projection $\pi_i^n : \mathbb{N}^n \to \mathbb{N}$, $\pi_i^n(x) = x_i$ (for $n \geq 1$);
- with *elementary operations*
  - Composition
  - Bracketing
  - Recursion

Elementary operations:

- Composition $\mathfrak{c}_{(m,m,p)}$: for $f:\mathbb{N}^m\to\mathbb{N}^n$, $g:\mathbb{N}^n\to\mathbb{N}^p$,

$$g\circ f:\mathbb{N}^m\to\mathbb{N}^p,\quad \mathscr{D}(g\circ f)=f^{-1}(\mathscr{D}(g));$$

- Bracketing $\mathfrak{b}_{(k,m,n_i)}$: for $f_i:\mathbb{N}^m\to\mathbb{N}^{n_i}$, $i=1,\dots,k$,

$$f=(f_1,\dots,f_k):\mathbb{N}^m\to\mathbb{N}^{n_1+\cdots+n_k},\quad \mathscr{D}(f)=\mathscr{D}(f_1)\cap\cdots\cap\mathscr{D}(f_k);$$

- Recursion $\mathfrak{r}_n$: for $f:\mathbb{N}^n\to\mathbb{N}$ and $g:\mathbb{N}^{n+2}\to\mathbb{N}$,

$$h(x_1,\dots,x_n,1):=f(x_1,\dots,x_n),$$

$$h(x_1,\dots,x_n,k+1):=g(x_1,\dots,x_n,k,h(x_1,\dots,x_n,k)),\quad k\geq 1,$$

where recursively $(x_1,\dots,x_n,1)\in\mathscr{D}(h)$ iff $(x_1,\dots,x_n)\in\mathscr{D}(f)$ and $(x_1,\dots,x_n,k+1)\in\mathscr{D}(h)$ iff $(x_1,\dots,x_n,k,h(x_1,\dots,x_n,k))\in\mathscr{D}(g)$.

Manin's Hopf algebra of flow charts

• planar labelled rooted trees (bracketing and recursion are ordered: need planar)

• label set of vertices $\mathscr{D}_V = \{\mathfrak{c}_{(m,n,p)}, \mathfrak{b}_{(k,m,n_i)}, \mathfrak{r}_n\}$ (composition, bracketing, recursion)

• label set of flags $\mathscr{D}_F$ primitive recursive functions

• *admissible* labelings:

- $\phi_V(v) = \mathfrak{c}_{(m,n,p)}$: v valence 3; labels $h_1 = \phi_F(f_1)$, $h_2 = \phi_F(f_2)$ incoming flags with domains and ranges $h_1 : \mathbb{N}^m \to \mathbb{N}^n$ and $h_2 : \mathbb{N}^n \to \mathbb{N}^p$; outgoing flag composition $h_2 \circ h_1 = \mathfrak{c}_{(m,n,p)}(h_1, h_2)$.
- $\phi_V(v) = \mathfrak{r}_n$: v valence 3; labels $h_1 = \phi_F(f_1)$, $h_2 = \phi_F(f_2)$ incoming flags with domains and ranges $h_1 : \mathbb{N}^n \to \mathbb{N}$ and $h_2 : \mathbb{N}^{n+2} \to \mathbb{N}$, outgoing flag recursion $h = \mathfrak{r}_n(h_1, h_2)$.
- $\phi_V(v) = \mathfrak{b}_{(k,m,n_i)}$: v must have valence $k+1$; labels $h_i = \phi_F(f_i)$ incoming flags with domain $\mathbb{N}^m$; outgoing flag bracketing $f = (f_1, \ldots, f_k) = \mathfrak{b}_{(k,m,n_i)}(f_1, \ldots, f_k)$.

• Coproduct, grading, antipode from Hopf algebra of rooted trees

Variants on the Hopf algebra of flow charts

- noncommutative Hopf algebra $\mathscr{H}^{nc}_{\mathrm{flow},\mathscr{P}}$
- Hopf algebra with only vertex labels $\mathscr{H}^{nc}_{\mathrm{flow},\mathscr{V}}$
- Use only binary operations (valence 3 vertices): express bracketing as a composition of binary operations

$$\mathfrak{b}_{(k,m,n_i)} = \mathfrak{b}_{(2,m,n_1,n_2+\cdots+n_k)} \circ \cdots \circ \mathfrak{b}_{(2,m,n_{k-1},n_k)}$$

- Extend composition and recursion to k-ary operations
  - k-ary compositions $\mathfrak{c}_{(k,m,n_i)}(h_i) = h_k \circ \cdots \circ h_1$ of functions $h_i : \mathbb{N}^{n_{i-1}} \to \mathbb{N}^{n_i}$, for $i = 1, \ldots, k$, with $n_0 = m$
  - $(k+1)$-ary recursions with k initial conditions:

$$h(x_1, \ldots, x_n, 1) = h_1(x_1, \ldots, x_n), \ldots$$
$$h(x_1, \ldots, x_n, k) = h_k(x_1, \ldots, x_n),$$
$$h(x_1, \ldots, x_n, k+\ell) =$$
$$h_{k+1}(x_1, \ldots, x_n, h_1(x_1, \ldots, x_n), \ldots, h_k(x_1, \ldots, x_n), k+\ell-1),$$
for $\ell \geq 1$

Insertion and Hochschild 1-cocycles

• T =forest: *grafting operator* $B^+_\delta(T)$ = sum of planar trees with new root vertex added with incoming flags equal number of trees in T and a single output flag and decoration $\delta \in \{\mathfrak{b}, \mathfrak{c}, \mathfrak{r}\}$

• cocycle condition:

$$\Delta B^+_\delta = (id \otimes B^+_\delta)\Delta + B^+_\delta \otimes 1$$

equivalent to $\tilde{\Delta} B^+_\delta = (id \otimes B^+_\delta)\tilde{\Delta} + id \otimes B^+_\delta(1)$ with $\tilde{\Delta}(x) := \sum x' \otimes x''$ (non-primitive part) and $B^+_\delta(1) = v_\delta$ (single vertex, label δ): first term admissible cuts root vertex attached to $\rho_C(T)$, second term admissible cut separating root vertex.

• cocycle condition requires same type of label ($\mathfrak{b}$, $\mathfrak{c}$, or $\mathfrak{r}$) for all vertices of arbitrary valence: use version $\mathscr{H}^{nc}_{\mathrm{flow},\mathscr{V}'}$ with k-ary operations

Systems of Dyson–Schwinger equations (Foissy)

• non-constant formal power series in three variables $X = (X_\delta)$

$$F_\delta(X) = \sum_{k_1,k_2,k_3} a^{(\delta)}_{k_1,k_2,k_3} X_{\mathfrak{b}}^{k_1} X_{\mathfrak{c}}^{k_2} X_{\mathfrak{r}}^{k_3}$$

• associated system of Dyson–Schwinger equations

$$X_\delta = B^+_\delta(F_\delta(X))$$

• unique solution $X_\delta = \sum_\tau x_\tau \, \tau$ (sum over planar rooted trees root decoration δ)

$$x_\tau = (\prod_{k=1}^{3} \frac{(\sum_{l=1}^{m_k} p_{\delta,l})!}{\prod_{l=1}^{m_k} p_{\delta,l}!}) a^{(\delta)}_{\sum_{k=1}^{3} p_{1,k}, \sum_{k=1}^{3} p_{2,k}, \sum_{k=1}^{3} p_{3,k}} x_{\tau_{1,1}}^{p_{1,1}} \cdots x_{\tau_{3,m_3}}^{p_{3,m_3}}$$

when

$$\tau = B^+(\tau_{1,1}^{p_{1,1}} \cdots \tau_{1,m_1}^{p_{1,m_1}} \cdots \tau_{3,1}^{p_{3,1}} \cdots \tau_{3,m_3}^{p_{3,m_3}})$$

Dyson–Schwinger equations and Hopf subalgebras

(Bergbauer–Kreimer)

• Dyson–Schwinger equations in a Hopf algebra of the form

$$X = 1 + \sum_{n=1}^{\infty} c_n \, B_\delta^+(X^{n+1})$$

• associative algebra $\mathscr{A}$ (subalgebra of $\mathscr{H}$) generated by components x_n of unique solution of DS equation

• using cocycle condition for B_δ^+ get

$$\Delta(x_n) = \sum_{k=0}^{n} \Pi_k^n \otimes x_k, \quad \text{where} \quad \Pi_k^n = \sum_{j_1+\cdots+j_{k+1}=n-k} x_{j_1} \cdots x_{j_{k+1}}$$

$\Rightarrow$ Hopf subalgebra

• generalized by Foissy for broader class of DS equations in Hopf algebras, including systems

Variant: Hopf ideals

- DS equation $X = 1 + \sum_{n=1}^{\infty} c_n \, B_{\delta}^{+}(X^{n+1})$
- *ideal* $\mathscr{I}$ generated by the components x_n (with $n \geq 1$) of solution
- cocycle condition for $B_{\delta}^{+} \Rightarrow \mathscr{I}$ Hopf ideal

elements of $\mathscr{I}$ finite sums $\sum_{m=1}^{M} h_m x_{k_m}$ with $h_m \in \mathscr{H}$ and x_k components of unique solution of DS equation

Hopf ideal condition: $\Delta(\mathscr{I}) \subset \mathscr{I} \otimes \mathscr{H} \oplus \mathscr{H} \otimes \mathscr{I}$

coproduct $\Delta(x_k)$: primitive part $1 \otimes x_k + x_k \otimes 1$ in $\mathscr{H} \otimes \mathscr{I} \oplus \mathscr{I} \otimes \mathscr{H}$; other terms in $\mathscr{I} \otimes \mathscr{I}$, so coproducts $\Delta(h_m x_{k_m})$ in $\mathscr{H} \otimes \mathscr{I} \oplus \mathscr{I} \otimes \mathscr{H}$.

$\Rightarrow$ *quotient Hopf algebra* $\mathscr{H}_{\mathscr{I}} = \mathscr{H} / \mathscr{I}$

Note: commutative Hopf algebra; if noncommutative use two-sided ideals

Yanofsky's Galois theory of algorithms

- Yanofsky proposed equivalence relations on flowcharts = “implementing the same algorithm"
- algorithm as intermediate level between the flow chart (= labelled planar rooted tree) and the primitive recursive functions
- obtain “Galois correspondence"
- resulting automorphism groups are products of symmetric groups
- but there are *problems*:

Example: (Joachim Kock)

fix function f: infinitely many programs computing it; “Galois group" is symmetry group of that set; subgroup S_3 (or C_3) permuting (cyclically) three of the programs fixing others: same orbits but different groups

Proposal for a different form of Galois theory of algorithms

• *suggestion*: take the Hopf algebra structure into account in defining relations (= relations should be Hopf ideals)

• instead of the kind of groups described by Yanofsky, find a sub-group scheme $G_{\mathscr{I}} \subset G_{\rm flow}$ corresponding to the quotient $\mathscr{H}_{\mathscr{I}} = \mathscr{H}/\mathscr{I}$, with $G_{\rm flow}$ group scheme dual to Hopf algebra $\mathscr{H}$ of flow charts

• in particular get a $G_{\mathscr{I}}$ from a Dyson–Schwinger equation (system)

• the groups appearing in this way have a structure more similar to the "Galois groups" playing a role in QFT

From Hopf algebras to operads

- *operad of flow charts* $\mathcal{O}_{\text{flow},\mathcal{V}'}$
  - $\mathcal{O}(n) = \mathbb{K}$-vector space spanned by labelled planar rooted trees with n incoming flags
  - operad composition operations

$$\circ_{\mathcal{O}} : \mathcal{O}(n) \otimes \mathcal{O}(m_1) \otimes \cdots \otimes \mathcal{O}(m_n) \to \mathcal{O}(m_1 + \cdots + m_n)$$

on generators $\tau \otimes \tau_1 \otimes \cdots \otimes \tau_n$ by grafting output flag of τ_i to the i-th input flag of τ

Dyson–Schwinger equations in operads

- formal series $P(t) = 1 + \sum_{k=1}^{\infty} a_k t^k$
- collection $\beta = (\beta_n)$ with $\beta_n \in \mathscr{O}(n)$
- Dyson–Schwinger equation:

$$X = \beta(P(X))$$

with $X = \sum_k x_k$ a formal sum of $x_k \in \mathscr{O}(k)$

- *self-similarity* with respect to $X \mapsto \beta(P(X))$
- right-hand-side of equation: $\beta(P(X))_1 = 1 + \beta_1 \circ x_1$, with 1 identity in $\mathscr{O}(1)$, and for $n \geq 2$

$$\beta(P(X))_n = \sum_{k=1}^{n} \sum_{j_1+\cdots+j_k=n} a_k \ \beta_k \circ (x_{j_1} \otimes \cdots \otimes x_{j_k})$$

with $x_{j_1} \otimes \cdots \otimes x_{j_k} \in \mathscr{O}(j_1) \otimes \cdots \otimes \mathscr{O}(j_k)$, composition $\beta_k \circ_{\mathscr{O}} (x_{j_1} \otimes \cdots \otimes x_{j_k}) \in \mathscr{O}(n)$, with $j_1 + \cdots + j_k = n$

Inductive construction of solutions

- $\mathcal{O} = \mathcal{O}_{\mathrm{flow},\mathcal{V}'}$ operad of flow charts
- assume $a_1\beta_1 \neq 1 \in \mathcal{O}(1)$
- then operadic Dyson–Schwinger equation $X = \beta(P(X))$ has unique solution $X \in \prod_{n\geq 1} \mathcal{O}(n)$ given inductively by

$$(1 - a_1\beta_1) \circ x_{n+1} = \sum_{k=2}^{n+1} \sum_{j_1+\cdots j_k=n+1} a_k \ \beta_k \circ (x_{j_1} \otimes \cdots \otimes x_{j_k})$$

- $\mathcal{O}_{\beta,P}(n) = \mathbb{K}$-linear span of all compositions $x_k \circ (x_{j_1} \otimes \cdots \otimes x_{j_k})$ for $k = 1, \ldots, n$ and $j_1 + \cdots + j_k = n$, with x_k coordinates of solution $X \Rightarrow \mathcal{O}_{\beta,P}(n)$ is a sub-operad
- choosing $a_1 \neq 1$ and β_k single vertex k incoming flags, label δ gives operadic version of DS equation with B^+_δ, but more general DS equations in operadic setting (without cocycle condition)

Operads and Properads

• Manin: extend Hopf algebra of flow charts to graphs (not trees) with acyclic orientations

• replace operad with *properad*: compositions grafting outputs and inputs of acyclic graphs

• *properad* (Valette): operations with varying numbers of inputs and outputs labelled by connected acyclic graphs; (*operads*: trees varying number of inputs and single output; *props*: allow disconnected graphs)

• composition operations: m inputs, n outputs

$$\mathscr{P}(m,n)\otimes\mathscr{P}(j_1,k_1)\otimes\cdots\otimes\mathscr{P}(j_\ell,k_\ell)\to\mathscr{P}(j_1+\cdots+j_\ell,n)$$

for $k_1+\cdots+k_\ell=m$

- $\mathscr{P}_{\text{flow},\mathscr{V}'}$ properad of flow charts
- $\mathscr{P}(m,n) = \mathbb{K}$-vector space spanned by planar connected directed (acyclic) graphs with m incoming flags and n outgoing flags
- vertices decorated by operations including $\mathfrak{b}$, $\mathfrak{c}$, $\mathfrak{r}$ (m inputs, one output) and *macros* with m inputs and n outputs

Dyson–Schwinger equations in properads

- formal power series $P(t) = 1 + \sum_k a_k t^k$
- collection $\beta = (\beta_{m,n})$ with $\beta_{m,n} \in \mathscr{P}(m,n)$
- DS equation $X = \beta(P(X))$ (self-similarity)
- in components

$$\beta(P(X))_{m,n} = \sum_{k=1}^{m} a_k \sum_{\substack{j_1+\dots j_k=m \\ i_1+\cdots i_k=\ell}} \beta_{\ell,n} \circ (x_{j_1,i_1} \otimes \cdots \otimes x_{j_k,i_k})$$

Construction of solutions in properads

- transformations $\Lambda_n = \Lambda_n(a,\beta)$

$\Lambda_n(a,\beta) : \oplus_{k=1}^n \mathscr{P}(n,k) \to \oplus_{k=1}^n \mathscr{P}(n,k)$, with $\Lambda_n(a,\beta)_{ij} = a_j \beta_{j,i}$

- assume $I - \Lambda_n(a,\beta)$ invertible for all n (not always satisfied)
- then unique solution to DS equation $X = \beta(P(X))$
- inductive construction: $x_{1,1} = \Lambda_1^{-1}$ and for $m < n$

$$x_{m,n} = \sum_{k=1}^{m} a_k \beta_{k,n} \circ \left(\sum_{\ell=1}^{k} \sum_{\substack{j_1+\cdots+j_\ell=m \\ i_1+\cdots+i_\ell=k}} x_{j_1,i_1} \otimes \cdots \otimes x_{j_\ell,i_\ell} \right)$$

remaning components $m \geq n$ determined by

$$Y_n(x) = (I - \Lambda_n)^{-1} \Lambda_n V^{(n)}(x)$$

with $Y_n(x)^t = (x_{n,1}, \ldots, x_{n,n})$ and $V^{(n)}(x)^t = (V^{(n)}(x)_j)_{j=1,\ldots,n}$

$$V^{(n)}(x)_j = \sum_{k=2}^{n} \sum_{\substack{r_1+\cdots+r_k=n \\ s_1+\cdots+s_k=j}} x_{r_1,s_1} \otimes \cdots \otimes x_{r_k,s_k}$$

Manin's "renormalization of the halting problem"

• Idea: treat noncomputable functions like infinities in QFT

• Renormalization = extraction of finite part from divergent Feynman integrals; extraction of "computable part" from noncomputables

• First step: build a Hopf algebra (similar to flow charts case) and a Feynman rule that detects the presence of noncomputability (infinities)

• Second step: BPHZ type subtraction procedure

• Third step: what is the meaning of the "renormalized part" and of the "divergences part" of the Birkhoff factorization?

Partial recursive functions and the Hopf algebra

- enlarge from primitive recursive to partial recursive: same elementary operations $\mathfrak{c}, \mathfrak{b}, \mathfrak{r}$ of composition, bracketing and recursion but additional μ operation
- μ operation: input function $f : \mathbb{N}^{n+1} \to \mathbb{N}$, output

$$h : \mathbb{N}^n \to \mathbb{N}, \quad h(x_1, \ldots, x_n) = \min\{x_{n+1} \mid f(x_1, \ldots, x_{n+1}) = 1\},$$

with domain $\mathscr{D}(h)$ those $(x_1, \ldots, x_n)$ such that $\exists x_{n+1} \geq 1$

$$f(x_1, \ldots, x_{n+1}) = 1, \text{ with } (x_1, \ldots, x_n, k) \in \mathscr{D}(f), \forall k \leq x_{n+1}$$

- Church's thesis: get all semi-computable functions, for which $\exists$ program computing $f(x)$ for $x \in \mathscr{D}(f)$ and computed zero or never stops for $x \notin \mathscr{D}(f)$
- Hopf algebra: additional vertex decoration by μ operations, extended to arbitrary valence by combining with bracketing; edge decorations by partial recursive functions

Feynman rule for computation (Manin)

• $\mathscr{B}$ algebra of functions $\Phi : \mathbb{N}^k \to \mathscr{M}(D)$ from $\mathbb{N}^k$, for some k, to algebra $\mathscr{M}(D)$ of analytic functions in unit disk $D = \{z \in \mathbb{C} \,:\, |z| < 1\}$.

• Rota–Baxter operator T on $\mathscr{B}$ componentwise projection onto polar part at $z = 1$

• For any tree τ that computes f set

$$\Phi_\tau(\underline{k}, z) = \Phi(\underline{k}, f, z) := \sum_{n \geq 0} \frac{z^n}{(1 + n\bar{f}(\underline{k}))^2}$$

$\bar{f} : \mathbb{N}^m \to \mathbb{Z}_{\geq 0}$ computes $f(x)$ at $x \in \mathscr{D}(f)$ and 0 at $x \notin \mathscr{D}(f)$.

• $\Phi_\tau(\underline{k}, z)$ pole at $z = 1$ iff $\underline{k} \notin \mathscr{D}(f)$

• this Φ is algebraic Feynman rule: commutative algebra homomorphism from enlarged Hopf algebra of flow charts to Rota–Baxter algebra $\mathscr{B}$

apply BPHZ

• negative part of Birkhoff factorization becomes

$$\Phi_-(\underline{k}, f_\tau, z) = -T(\Phi(\underline{k}, f_\tau, z) + \sum_C \Phi_-(\underline{k}, f_{\pi_C(\tau)}, z)\Phi(\underline{k}, f_{\rho_C(\tau)}, z))$$

• Note: $f = f_\tau$ label of outgoing flag of τ: then $f_{\rho_C(\tau)} = f_\tau$

$$\Phi_-(\underline{k}, f_\tau, z) = -T\left(\Phi(\underline{k}, f_\tau, z)(1 + \Phi_-(\underline{k}, \sum_C f_{\pi_C(\tau)}, z))\right)$$

• What is happening here? Like in QFT, looking not only at "divergences" of program τ but also of *all subprograms* $\pi_C(\tau)$ and $\rho_C(\tau)$ determined by admissible cuts (the problem of subdivergences in renormalization)

Why subdivergences in computation?

- $\Phi_-(\underline{k}, f_\tau, z)$ detects not only if τ has infinities but if any subroutine does
- Note: $\Phi(\underline{k}, f_\tau, z)$ only depends on $f = f_\tau$ not on τ, but $\Phi_-(\underline{k}, f_\tau, z)$ really *depends on* τ
- Unlike QFT there are programs without divergences that do have subdivergences
- *Example:* (Joachim Kock)

identity function computed as composite of successor function followed by partial predecessor function $\mu(|y + 1 - x|)$ (undefined at 0, and $x - 1$ for $x > 0$), τ with a $\mathfrak{c}$ node and a μ node

Renormalized part What does it measure?

$$\Phi_+(\underline{k}, f_\tau, z) = (1-T)(\Phi(\underline{k}, f_\tau, z) + \sum_C \Phi_-(\underline{k}, f_{\pi_C(\tau)}, z)\Phi(\underline{k}, f_{\rho_C(\tau)}, z))$$

• Main question: is there a new $f_{\rm ren}$, now *primitive recursive*, such that $\Phi_+(\underline{k}, f_\tau, z) = \Phi(\underline{k}, f_{\rm ren}, z)$?

• in general not true simply as stated, but in QFT there is an *equivalence relation* on Feynman rules and renormalized values, a kind of gauge transformation by germs of holomorphic functions (Connes–Marcolli): correct statement of question is up to such an equivalence?

• *Useful viewpoint*: every partial recursive function can be computed by a Hopf-primitive program: Kleene normal form as μ of a total function