# Semi-quantitative semantics

**Vincent Moreau**
Tallinn University of Technology, Estonia

**Lê Thành Dũng (Tito) Nguyễn**
LIS, CNRS & Aix-Marseille University, France

**Abstract.** *We introduce a new exponential modality for coherence spaces that can "count up to 2". Its Kleisli category provides an interesting example of finitary yet intensional semantics of the simply typed $\lambda$-calculus. The eventual goal is to present this semantics using an intersection type system, and derive a corresponding notion of graded resource approximants, with a view towards applications to higher-order transducers.*

The past two decades have seen the development of *quantitative semantics* of linear logic, such as the relational model, with an associated toolkit: non-idempotent intersection types (a.k.a. multi types) and resource approximation (variants of the Taylor expansion of $\lambda$-terms). Many references are given in the introductions to [21, 11, 1]. For now, these tools have had an *infinitary* character:

- The quantitative semantics introduced until now are not finitary (i.e. locally finite categories), even in the simply typed case. Contrapositively: the known finitary semantics of simple types — variants of Scott domains, sequential algorithms, ... — do not record how many times a function uses its argument.
- Relatedly, the Taylor expansion is an infinite sum: a $\lambda$-term may have infinitely many resource approximants. This is also the case for the polyadic linear approximants of Mazza, Pellissier and Vial [13].

We would like to "finitarize" the quantitative toolkit, in order to make it computable in principle. Inspired by the successes in higher-order model checking of intersection types [12, 20] and finitary semantics [26, 10, 16], we are aiming towards applications to automata theory, discussed at the end of this extended abstract. Our desiderata are:

- A *semi-quantitative semantics* which counts whether a function uses its argument zero, one or multiple times. This amounts to the following change of semiring of interest:

$$\text{quantitative } \mathbb{N} = \{0, 1, 2, 3, \dots\} \qquad \rightsquigarrow \qquad \text{semi-quantitative } \mathbb{S} = \{0, 1, \infty\}$$

  where $\infty$ is a common value for all natural numbers $2, 3, \dots$ strictly greater than 1. This allows us to track multiplicities, thus getting rid of the main source of infinitary behavior while still keeping track of some intensional information about resource usage.
- More precisely, the semantics should be a "web model"[1] of linear logic, with an exponential based on $\mathbb{S}$-weighted multisets (functions with codomain $\mathbb{S}$). Web models have the advantage of being presentable as intersection type assignment systems.
- Following the spirit of Mazza et al. [13], our intersection type derivations should correspond to some kind of polyadic approximants. The calculus of approximants should be (the polyadic version of) some simply typed $\mathbb{S}$-*graded* $\lambda$-calculus, for instance[2] [4, Section 4].

[1] See Tasson & Walch [24] for a recent generic treatment of web models, and Tsukada & Asada [25] for a more general notion of vector-space-like models of linear logic.

[2] There are many such graded calculi; a reason for choosing the one of Choudhury, Eades, Eisenberg and Weirich [4] is that it allows excluding the weakening rule. The use of $\mathbb{S} = \{0, 1, \infty\}$ in a type system goes back at least to the work of Mogensen in the late 1990s [17], while $\mathbb{S}$-grading in its modern form was first used by McBride to combine linearity and dependent types [14]. See also Scherer's term counting approximation [23, Chapter 9] for another application of $\mathbb{S}$ to programming language theory.

**Choosing the right web model.** An exponential modality parameterized by a "multiplicity semiring" has already been studied for the relational model of linear logic [3, 2]. However, multiplicity semirings are always infinite. In fact, it is well known that the relational model does not support a "qualitative" exponential, and thus we cannot expect it either to support a semi-quantitative one. The problem with the qualitative exponential non-comonad lies in the lack of naturality of the counit (dereliction), cf. [15, Section 8.7]. In the semi-quantitative setting we have another problem: the action on morphisms

$$
\begin{aligned}
&!\colon \mathbf{Rel}(X, Y) \to \mathbf{Rel}(\mathbb{S}^X, \mathbb{S}^Y) \\
&\qquad\qquad f \mapsto \{([x_1, \ldots, x_n], [y_1, \ldots, y_n]) \mid n \in \mathbb{N},\ \{(x_1, y_1), \ldots\} \subseteq f\}
\end{aligned}
$$

(where square brackets are used to denote $\mathbb{S}$-weighted multisets, with $[a, a] = [\infty \cdot a] = [a, a, a]$) is *not even functorial*! Here is a counterexample: for $f = \{1, 2\} \times \{*\}$ and $g = \{*\} \times \{1, 2, 3\}$, we have $([1, 2], [1, 2, 3]) \in (!g \circ !f) \setminus !(g \circ f)$.

Surprisingly, this lack of functoriality of the semi-quantitative exponential persists in the category of ordered relations (a.k.a. "Scott model of linear logic", or Bool-enriched profunctors, cf. [8]) even though it admits both a qualitative and a quantitative exponential modality. Therefore, we move on to the next simplest web model: *coherence spaces.*

**Semi-quantitative multicliques.** Our new exponential $!_{\mathbb{S}}$ is halfway between the usual qualitative and quantitative exponentials of the category $\mathbf{Coh}$ of coherence spaces. A semi-quantitative multiclique $u$ on a coherence space $X = (|X|, \mathrel{\substack{\frown \\ \smile}}_X)$ is a function

$$
u \quad : \quad |X| \ \longrightarrow\ \mathbb{S} \qquad \text{such that } \mathrm{supp}(u) := u^{-1}(\mathbb{S} \setminus \{0\}) \text{ is a finite clique of } X
$$

Two semi-quantitative multicliques $u$ and $v$ are coherent when their supports $\mathrm{supp}(u)$ and $\mathrm{supp}(v)$ are coherent as cliques of $X$. In this way, we get a new coherence space $!_{\mathbb{S}}X$ of semi-quantitative multicliques of $X$.

An action on morphisms can be defined as for $\mathbf{Rel}$ above, but here it is functorial. What is more, this construction yields a comonad $!_{\mathbb{S}}$ on $\mathbf{Coh}$, whose counit and comultiplication come from the semiring structure of $\mathbb{S}$. More concretely, for any coherence space $X$ we have

$$
\begin{aligned}
&\Big\{([x], x) \ : \ x \in |X|\Big\} \quad \in \quad \mathbf{Coh}(!_{\mathbb{S}}X, X) \\
&\Big\{(u_1 + \cdots + u_n, [u_1, \ldots, u_n]) \ : \ [u_1, \ldots, u_n] \in |!_{\mathbb{S}}!_{\mathbb{S}}X|\Big\} \quad \in \quad \mathbf{Coh}(!_{\mathbb{S}}X, !_{\mathbb{S}}!_{\mathbb{S}}X)
\end{aligned}
$$

In this way, we build a model of propositional classical linear logic.

**Intensional finitary semantics.** The finiteness of the semiring $\mathbb{S}$ implies that our comonad $!_{\mathbb{S}}$ preserves the finiteness of webs. Still, the comonad $!_{\mathbb{S}}$ captures quantitative aspects, and in particular has a coKleisli category $\mathbf{Coh}_{!_{\mathbb{S}}}$ that is not well-pointed, which amounts to a form of intensional computation. The combination of these two properties is remarkable, and contrasts with the usual, well-known qualitative and quantitative exponentials used to interpret linear logic in the category of coherence spaces, respectively based on cliques and multicliques. The situation is summarized in Table 1.

Cartesian closed categories that are locally finite are at the heart of the growing connection between automata theory and the $\lambda$-calculus. The notion of higher-order regular language, due to Salvati [22], yields a conservative extension of regular languages of finite words and ranked trees to the full type hierarchy of the simply typed $\lambda$-calculus. Rooted in denotational

**Table 1** Properties of the qualitative, quantitative, and semi-quantitative exponentials on **Coh**.

| | cliques | multicliques | semi-quantitative multicliques |
|---|---|---|---|
| Intensional? | $\times$ | ✓ | ✓ |
| Finitary? | ✓ | $\times$ | ✓ |

semantics, this approach extends the notion of language of finite words recognized by a monoid to one of languages of $\lambda$-terms recognized by a cartesian closed category. In particular:

$$\text{every cartesian closed category } \mathbf{C} \quad \xrightarrow{\text{induces}} \quad \text{a class of languages of } \lambda\text{-terms}$$

A natural hypothesis to establish links between denotational semantics and automata theory consists in considering cartesian closed categories **C** that are locally finite. A recent result of the authors establishes that all locally finite cartesian closed categories that are well-pointed (and non-posetal) recognize exactly higher-order regular languages [19], see the second author's PhD thesis for more details [18, Theorem 7.4.3]. Knowing whether this statement extends to all locally finite cartesian closed categories, potentially non-well pointed, is an important open problem on higher-order regular languages [18, Section 10.2].

Our new semi-quantitative exponential $!_{\mathbb{S}}$ restricts to the full subcategory **FinCoh** of finite coherence spaces, and hence induces a cartesian closed coKleisli category $\mathbf{FinCoh}_{!_{\mathbb{S}}}$ which is locally finite and not well-pointed. We are conducting ongoing investigations about the class of languages recognized by this category:

$$\text{coKleisli category } \mathbf{FinCoh}_{!_{\mathbb{S}}} \quad \xrightarrow{\text{induces}} \quad \text{higher-order regular languages?}$$

**Towards applications to higher-order tree transducers.** We aim to generalize the work of Engelfriet and Maneth [5, 6] on macro tree transducers (MTTs) to higher-order transducers, devices that use simply typed $\lambda$-terms to compute tree-to-tree functions. (MTTs are the order-1 special case, while compositions of MTTs correspond to *safe* higher-order transducers.)

One of their results is that every macro tree transducer can be turned into a *nondeleting* MTT (with regular lookahead) [5, Lemma 6.6]. This has been applied to decide asymptotic growth properties of MTTs [7, 9]. To extend this "nondeleting normal form" to higher-order transducers, we plan to use $\mathbb{S}$-graded approximants. Actually, for this only purpose, relevant (i.e. Bool-graded) approximation would suffice.

But we also have in mind a "bridge theorem" on output languages of compositions of MTTs [6, Section 5] whose proof involves a duplication analysis. Morally:[3] in a nondeleting computation that produces an output of a certain form, "complicated" subtrees cannot be duplicated; so either an argument is used linearly, or it can only take "simple" values. Here, $\{0, 1, \infty\}$-grading looks like the right tool for the job. Extending this "bridge theorem" to higher-order transducers is the subject of an ongoing collaboration of the second author with Paweł Parys, which is actually the initial motivation for the work presented here.

[3] In the original text [6, p. 370]: "Consider the right-hand side of a rule of $M$ in which some parameter $y_j$ occurs more than once. If, during the derivation of a tree which has as yield a $\delta$-string, this rule was applied, then the tree which is substituted for $y_j$ in this derivation contains at most one occurrence of a symbol in $A$. Because otherwise, due to copying, the resulting string would not be a $\delta$-string. Hence, when deriving a $\delta$-string, a rule which contains multiple occurrences of a parameter $y_j$ is only applicable if the yield of the tree being substituted for $y_j$ contains at most one occurrence of a symbol in $A$."

## References


1. Beniamino Accattoli. Semantic bounds and multi types, revisited. In Aniello Murano and Alexandra Silva, editors, *32nd EACSL Annual Conference on Computer Science Logic, CSL 2024, Naples, Italy, February 19-23, 2024*, LIPIcs, pages 7:1–7:24. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.CSL.2024.7.
2. Flavien Breuvart and Michele Pagani. Modelling coeffects in the relational semantics of linear logic. In Stephan Kreutzer, editor, *24th EACSL Annual Conference on Computer Science Logic, CSL 2015, Berlin, Germany, September 7-10, 2015*, LIPIcs, pages 567–581. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015. doi:10.4230/LIPICS.CSL.2015.567.
3. Alberto Carraro, Thomas Ehrhard, and Antonino Salibra. Exponentials with infinite multiplicities. In Anuj Dawar and Helmut Veith, editors, *Computer Science Logic, 24th International Workshop, CSL 2010, 19th Annual Conference of the EACSL, Brno, Czech Republic, August 23-27, 2010. Proceedings*, Lecture Notes in Computer Science, pages 170–184. Springer, 2010. doi:10.1007/978-3-642-15205-4_16.
4. Pritam Choudhury, Harley Eades III, Richard A. Eisenberg, and Stephanie Weirich. A graded dependent type system with a usage-aware semantics. *Proc. ACM Program. Lang.*, 5(POPL):1–32, 2021. doi:10.1145/3434331.
5. Joost Engelfriet and Sebastian Maneth. Macro Tree Transducers, Attribute Grammars, and MSO Definable Tree Translations. *Information and Computation*, 154(1):34–91, October 1999. doi:10.1006/inco.1999.2807.
6. Joost Engelfriet and Sebastian Maneth. Output string languages of compositions of deterministic macro tree transducers. *Journal of Computer and System Sciences*, 64(2):350–395, 2002. doi:10.1006/jcss.2001.1816.
7. Joost Engelfriet and Sebastian Maneth. Macro tree translations of linear size increase are MSO definable. *SIAM Journal on Computing*, 32(4):950–1006, 2003. doi:10.1137/S0097539701394511.
8. Zeinab Galal. A profunctorial Scott semantics. In Zena M. Ariola, editor, *5th International Conference on Formal Structures for Computation and Deduction, FSCD 2020, Paris, France (Virtual Conference), June 29 - July 6, 2020*, LIPIcs, pages 16:1–16:18. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPICS.FSCD.2020.16.
9. Paul Gallot, Sebastian Maneth, Keisuke Nakano, and Charles Peyrat. Deciding linear height and linear size-to-height increase of macro tree transducers. In Karl Bringmann, Martin Grohe, Gabriele Puppis, and Ola Svensson, editors, *51st International Colloquium on Automata, Languages, and Programming, ICALP 2024, July 8-12, 2024, Tallinn, Estonia*, volume 297 of *LIPIcs*, pages 138:1–138:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.ICALP.2024.138.
10. Charles Grellois. *Semantics of linear logic and higher-order model-checking.* PhD thesis, Université Paris 7, April 2016. URL: https://hal.science/tel-01311150.
11. Axel Kerinec, Giulio Manzonetto, and Federico Olimpieri. Why are proofs relevant in proof-relevant models? *Proc. ACM Program. Lang.*, 7(POPL):218–248, 2023. doi:10.1145/3571201.
12. Naoki Kobayashi. Model checking higher-order programs. *Journal of the ACM*, 60(3):20:1–20:62, 2013. doi:10.1145/2487241.2487246.
13. Damiano Mazza, Luc Pellissier, and Pierre Vial. Polyadic approximations, fibrations and intersection types. *Proc. ACM Program. Lang.*, 2(POPL):6:1–6:28, 2018. doi:10.1145/3158094.
14. Conor McBride. I got plenty o' nuttin'. In Sam Lindley, Conor McBride, Philip W. Trinder, and Donald Sannella, editors, *A List of Successes That Can Change the World - Essays Dedicated to Philip Wadler on the Occasion of His 60th Birthday*, Lecture Notes in Computer Science, pages 207–233. Springer, 2016. doi:10.1007/978-3-319-30936-1_12.
15. Paul-André Melliès. Categorical semantics of linear logic. In *Interactive models of computation and program behaviour*, volume 27 of *Panoramas et Synthèses*, pages 1–196. Société Mathématique de France, 2009. URL: https://www.irif.fr/~mellies/papers/panorama.pdf.

**16** Paul-André Melliès. Higher-order parity automata. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–12, Reykjavik, Iceland, June 2017. IEEE. doi:10.1109/LICS.2017.8005077.

**17** Torben Æ. Mogensen. Types for 0, 1 or many uses. In Chris Clack, Kevin Hammond, and Antony J. T. Davie, editors, *Implementation of Functional Languages, 9th International Workshop, IFL'97, St. Andrews, Scotland, UK, September 10-12, 1997, Selected Papers*, Lecture Notes in Computer Science, pages 112–122. Springer, 1997. doi:10.1007/BFB0055427.

**18** Vincent Moreau. *A topological and fibrational approach to higher-order automata*. PhD thesis, Université Paris Cité, July 2025. URL: https://theses.hal.science/tel-05428993.

**19** Vincent Moreau and Lê Thành Dũng Nguyễn. Syntactically and semantically regular languages of $\lambda$-terms coincide through logical relations. In Aniello Murano and Alexandra Silva, editors, *32nd EACSL Annual Conference on Computer Science Logic, CSL 2024, Naples, Italy, February 19-23, 2024*, LIPIcs, pages 40:1–40:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPICS.CSL.2024.40.

**20** C.-H. Luke Ong. Higher-order model checking: An overview. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015*, pages 1–15. IEEE Computer Society, 2015. doi:10.1109/LICS.2015.9.

**21** C.-H. Luke Ong. Quantitative semantics of the lambda calculus: Some generalisations of the relational model. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017*, pages 1–12. IEEE Computer Society, 2017. doi:10.1109/LICS.2017.8005064.

**22** Sylvain Salvati. Recognizability in the simply typed lambda-calculus. In Hiroakira Ono, Makoto Kanazawa, and Ruy J. G. B. de Queiroz, editors, *Logic, Language, Information and Computation, 16th International Workshop, WoLLIC 2009, Tokyo, Japan, June 21-24, 2009. Proceedings*, Lecture Notes in Computer Science, pages 48–60. Springer, 2009. doi:10.1007/978-3-642-02261-6_5.

**23** Gabriel Scherer. *Which types have a unique inhabitant? : Focusing on pure program equivalence. (Quels types ont un habitant unique ? : Focalisons-nous sur l'équivalence de programmes)*. PhD thesis, Paris Diderot University, France, 2016. URL: https://theses.hal.science/tel-01309712.

**24** Christine Tasson and Aymeric Walch. Absolute convergence and Taylor expansion in web based models of linear logic, 2026. arXiv:2603.25215.

**25** Takeshi Tsukada and Kazuyuki Asada. Linear-algebraic models of linear logic as categories of modules over Σ-semirings. In Christel Baier and Dana Fisman, editors, *LICS '22: 37th Annual ACM/IEEE Symposium on Logic in Computer Science, Haifa, Israel, August 2 - 5, 2022*, pages 60:1–60:13. ACM, 2022. doi:10.1145/3531130.3533373.

**26** Igor Walukiewicz. Automata theory and higher-order model-checking. *ACM SIGLOG News*, 3(4):13–31, 2016. doi:10.1145/3026744.3026745.