

いし い ひろみ  
**石井 大海**

**基本情報**

DeepFlow 株式会社

E-MAIL: [h-konn.jinro@gmail.com](mailto:h-konn.jinro@gmail.com)

WEB SITE: <https://konn-san.com>

GITHUB: <https://github.com/konn>

**教育・学位**

- |                |  |
|----------------|--|
| <b>博士 (理学)</b> | 2019 年 3 月, 筑波大学数理物質科学研究科数学専攻 博士後期課程・修了<br><b>博士論文:</b> Bidirectional Interplay between Mathematics and Computer Science: Safety and Extensibility in Computer Algebra and Haskell |
| <b>修士 (理学)</b> | 2016 年 3 月, 筑波大学数理物質科学研究科 博士前期課程・修了<br><b>修士論文:</b> On Regularity Properties of Sets of Reals and Inaccessible Cardinals   |
| <b>学士 (理学)</b> | 2014 年 3 月, 早稲田大学基幹理工学部数学科・卒   |

**職歴・採用歴**

- 2019 年 4 月-現在 DeepFlow 株式会社 研究開発部
- 2019 年 4 月-2020 年 3 月 統計数理研究所 外来研究員
- 2017 年 4 月-2019 年 3 月 日本学術振興会特別研究員 DC2  
研究課題:『実数の集合の性質の集合論的解明と工学的応用』
- 2014 年-2017 年 筑波大学数学類『計算機演習』ティーチング・アシスタント
- 2014 年 4 月 Google Summer of Code 2014 採択  
題目:『Haskell による効率的な Gröbner 基底計算とそのための疎行列対角化アルゴリズムの実装』
- 2014 年 4 月-2017 年 3 月 筑波大学数学専攻計算機管理アルバイト (www-admin)
- 2010 年 10 月-2014 年 3 月 株式会社 Preferred Infrastructure アルバイト
- 2010 年 8 月-9 月 株式会社 Preferred Infrastructure インターン生

**賞罰・特記事項**

- 日本学術振興会特別研究員 DC2, (2017 年~2019 年)
- 第十四回茗溪会賞 (2016 年)
- 2013 年度早稲田大学基幹理工学部賞最優秀賞 (第一回), 基幹理工学部卒業生総代
- Web 数式描画ライブラリ KaTeX および Haskell 製 Web フレームワーク Yesod コミット

## 研究業績

### 査読付き会議論文

August 2021, **Automatic Differentiation with Higher Infinitesimals, or Computational Smooth Infinitesimal Analysis in Weil Algebra**. Computer Algebra in Scientific Computing 2021, Sochi, Russia.

September 2018, **A Purely Functional Computer Algebra System Embedded in Haskell**. Computer Algebra in Scientific Computing 2018, Lille, France.

September 2015. Oleg Kiselyov and Hiromi ISHII, **Freer Monads, More Extensible effects**. Haskell Symposium 2015, Vancouver, Canada.

### 学会発表

August 2021, **Automatic Differentiation with Higher Infinitesimals, or Computational Smooth Infinitesimal Analysis in Weil Algebra**. Computer Algebra in Scientific Computing 2021, Sochi, Russia, 査読あり.

September 2018, **A Purely Functional Computer Algebra System Embedded in Haskell**. Computer Algebra in Scientific Computing 2018, Lille, France, 査読あり.

March 2016, **Freer Monads, More Extensible Effects**. Programming and Programming Language Workshop (PPL) 2016, Okayama-prefecture, Japan, 査読あり.

November 2017, **Reflection Principle and construction of saturated ideals on  $\mathcal{P}_{\omega_1} \lambda$** . Workshop on Iterated Forcing Theory and Cardinal Invariants, Kyoto-prefecture, Japan, 査読なし.

### 論文

- [1] Hiromi Ishii, **On regularity properties of set of reals and inaccessible cardinals**, MA thesis, Tsukuba University, 2016.
- [2] Oleg Kiselyov and Hiromi Ishii, **Freer monads, more extensible effects**, Proceedings of the 2015 ACM SIGPLAN Symposium on Haskell, Haskell '15, Vancouver, BC, Canada: ACM, 2015, pp. 94–105, ISBN: 978-1-4503-3808-0, DOI: 10.1145/2804302.2804319.

## 開発実績

主要言語・技術要素: Haskell, MPI, ParaView, Python, 数値シミュレーション, 計算機代数, FFI, GitHub, CI/CD, GitOps, REST や GraphQL などの Web API 利用・設計

経験済環境: JavaScript, TypeScript, Rust, Ruby, C.

### レポート自動採点システム $\lambda$ eport

生徒の提出したレポート (Haskell プログラム) を自動採点するためのツールである。演習のティーティング・アシスタントにはそこまで Haskell に習熟していない担当者もあり、また単なる目視による採点では見落としがあり得る。そこで、関数型プログラミングにおける性質ベーステスト (Property-based Testing)

の方法論を応用し、QuickCheck ライブラリを用いてレポートの答案の形式仕様を与え、**生徒の答案が仕様を満たすかを自動でチェック**し、結果を報告する。

実装に当たっては、生徒の答案というプライバシーに関わる情報を取り扱うため、**Let's Encrypt** で証明書を取得し HTTPS 上で通信を行うようにした。また、複数人で同時に採点を行うため、分散コンピューティングライブラリ **Cloud Haskell** を用いて **Work-Stealing モデルを採用した分散システム**を構築した。また、各プロセスの内部でも様々な並行処理が必要であるため、Haskell の強力な **Software Transactional Memory** の機構を活用し、簡潔で安全な実装を実現した。QuickCheck によるテストでは任意のコードが実行される可能性があるため、**Linux Containers** を用いて隔離されたコンテナ上でこれらの検証を行う。Linux Containers は CPU やメモリの占有率を制御出来、この点でも安全性が担保される。小間ごとに採点された結果は、Linux Containers の **WebSocket API** を介してワーカにリアルタイムで伝達され、**EventSource** 経由で Web フロントエンドに反映される。

## 計算機代数システム computational-algebra

WEB SITE: <https://konn.github.io/computational-algebra>

関数型言語 Haskell の処理系 Glasgow Haskell Compiler (以下, GHC) が提供する先進的型システムを活かし、高度な代数計算を安全に行える計算代数システムを EDSL としてゼロから設計・実装した。具体的には、型システムや形式手法を応用することで以下の長所を実現した：

**型安全** 係数体や変数の数、順番などを型で表すことで、誤操作を防止する。

**拡張性** 多項式や代数系を表す型クラスを提供し、内部実装に依らず様々なアルゴリズムを適用可能にした。

**直感的**  $\mathbb{Q}[x, y, z]$  と  $\mathbb{Q}[x, z, y, w]$  は型上区別されるが、これらの間の埋め込み写像を型情報だけから自動的に計算出来る。また、多項式も本来の数式に近い形で  $\#x^2 + \#z * \#x - 2$  のように書ける。

**静的保証** QuickCheck を用いて**アルゴリズムの形式仕様を静的に検証**し、ライブラリの品質を保っている。

上述の **Google Summer of Code 2014** では、高速な Gröbner 基底計算アルゴリズムとして知られる  $F_4$  および  $F_5$  アルゴリズムの実装を試み、複数の行列ライブラリを統一的に扱うための枠組みや、ライブラリの整備を進めた。その他に、有理係数多項式の因数分解や、代数的数の計算機能なども実装されている。

特に、型安全性を最大限実現するために、GHC の型レベル自然数機能を強化する **type-natural** パッケージや、GHC の型検査器に Presburger 算術ソルバを組み込むコンパイラプラグイン **ghc-typelits-presburger** を開発した。それらを用い、リストや配列、ベクトルなど任意のシーケンシャルなデータ型から固定長のコンテナ型を創り出せる **sized** パッケージを実装した。

本ライブラリを用いた筑波大学数学類の卒業研究をティーチング・アシスタントとして指導した。また、エクアドル大学のヤチャイ技術大学の研究プロジェクト等でも用いられているようである。

## 大規模数値計算ソルバー開発

DeepFlow 株式会社にて、最新鋭の数学的アルゴリズムに基づいた、Haskell 製大規模並列数値ソルバープログラムの開発に従事。当該ソルバーはスレッド並列や MPI 通信などを活かした Haskell による高性能プログラミングの技術を用い、数億セルの並列計算を実現。高い抽象性と拡張性・効率性を両立するための Haskell 内への埋め込みドメイン特化型言語の設計や、型シス

テムの設計等を主に担当する。その際に取り組んだ技術的課題と解決策については、数年を経て古くなっているものの、Haskell Day 2019 での発表資料 (<https://speakerdeck.com/konn/da-gui-mo-shu-zhi-ji-suan-wozhi-eru-haskell-nil-nil-pragmatic-haskell-in-large-scale-numerical-comp>) を参考にされたい。

## モノレポ依存関係管理ツール Guardian の実装

上述の大規模数値計算ソルバーは非常に多くのパッケージから成っており、それらの間の依存関係が複雑化することでビルド時間に大きな悪影響が出ていた。こうした状況を改善するため、パッケージを複数のグループに分類し、グループ間の依存関係に関する制約を設定させ CI で絶えずチェックさせることで、依存関係において疎結合・関心の分離を実現するためのツール Guardian の開発を社内で主導し、OSS として公開した。

## 研究・シミュレーション用ワークフローツール開発

Python, PyTorch, Rye, DVC

### 機械学習案件

Python, PyTorch, Rye, DVC

### その他の開発活動

- Haskell の公式 Language Server である Haskell Language Server のメンテナを務める。主な貢献事項は以下の通り：
  1. Splice Plugin: Template Haskell のスプライスを評価し、その結果でマクロ呼び出しを置換するためのプラグインを実装した。
  2. Disambiguate Imports: 複数モジュールまたは現在のモジュールで定義され衝突が発生している曖昧な識別子を一意化するための機構を提案し、実装した。
  3. Eval Plugin の改良: パーザの効率改善、`:type` や `:kind` コマンドの実装などを行った。
- Web 上で高速かつ高品質な数式描画を可能とする **KaTeX** に幾つかのコマンド・環境を実装し、コミット権限を得た。これまで本格的な JavaScript 開発は経験していなかったが、`npm` や `browserify`, `flow` などを用いたワークフローには半日程度でキャッチアップ出来た。
- Haskell の **Web フレームワーク Yesod** の OAuth 認証ライブラリの開発 (現在は引退)。
- 『型システム入門 —プログラミング言語と型の理論—』『Haskell による並列・並行プログラミング』(ともにオーム社) など定評のある技術書・学術書の邦訳に、出版前レビューとして参画・貢献した。
- その他の開発プロジェクトは、GitHub 上 (<https://github.com/konn>) にて見る事が出来る。

FiXme■  
Fa-  
tal:  
ちゃ  
んと  
書く

FiXme■  
Fa-  
tal:  
ちゃ  
んと  
書く