Normalized Completion for Stratified Linear Rewriting Systems

Zuan Liu – Philippe Malbos

Institut Camille Jordan

Université Claude Bernard Lyon 1

International Workshop on Confluence 2026

July 24, 2026

Outline

1. Introduction
2. Stratified rewriting systems
3. Normalized completion by stratification
4. Stratified diagrammatic rewriting systems
5. Conclusion

Contents

Background

- **Algebraic rewriting** is a rewriting theory for the presentation of algebraic structures, aimed at solving

Background

- **Algebraic rewriting** is a rewriting theory for the presentation of algebraic structures, aimed at solving
  - Decision problems, e.g. the word problem;
  - Basis theorems via unique normal forms;
  - Symbolic computation in homological algebra.

Background

- **Algebraic rewriting** is a rewriting theory for the presentation of algebraic structures, aimed at solving
  - Decision problems, e.g. the word problem;
  - Basis theorems via unique normal forms;
  - Symbolic computation in homological algebra.
- **Linear rewriting in various algebraic structures**

Background

- **Algebraic rewriting** is a rewriting theory for the presentation of algebraic structures, aimed at solving
  - Decision problems, e.g. the word problem;
  - Basis theorems via unique normal forms;
  - Symbolic computation in homological algebra.
- **Linear rewriting in various algebraic structures**
  - Commutative algebras: Buchberger '65;
  - Associative algebras: Bokut '76, Bergman '78, Mora '94;
  - Operads: Dotsenko-Khoroshkin '10, Malbos-Ren '21;
  - Diagrammatic algebras: Alleaume '18, Dupont '21, Schelstraete '24.

Background

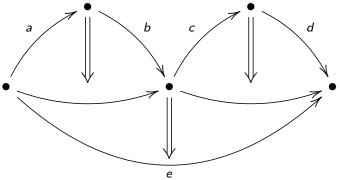
- **Algebraic rewriting** is a rewriting theory for the presentation of algebraic structures, aimed at solving
  - Decision problems, e.g. the word problem;
  - Basis theorems via unique normal forms;
  - Symbolic computation in homological algebra.
- **Linear rewriting in various algebraic structures**
  - Commutative algebras: Buchberger '65;
  - Associative algebras: Bokut '76, Bergman '78, Mora '94;
  - Operads: Dotsenko-Khoroshkin '10, Malbos-Ren '21;
  - Diagrammatic algebras: Alleaume '18, Dupont '21, Schelstraete '24.
- **Computing linear bases by confluence**

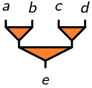
Background

- **Algebraic rewriting** is a rewriting theory for the presentation of algebraic structures, aimed at solving
  - Decision problems, e.g. the word problem;
  - Basis theorems via unique normal forms;
  - Symbolic computation in homological algebra.
- **Linear rewriting in various algebraic structures**
  - Commutative algebras: Buchberger '65;
  - Associative algebras: Bokut '76, Bergman '78, Mora '94;
  - Operads: Dotsenko-Khoroshkin '10, Malbos-Ren '21;
  - Diagrammatic algebras: Alleaume '18, Dupont '21, Schelstraete '24.
- **Computing linear bases by confluence**
  - Every element of a terminating and confluent rewriting system admits a unique normal form.
  - These normal forms provide canonical representatives.
  - The set of normal forms yields a **linear basis** of the presented algebraic structure.

Diagrammatic rewriting

- **Diagrammatic rewriting** is a rewriting model for **linear 2-categories**, in which 2-cells are represented by 2-dimensional diagrams, such as





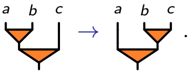
.

Diagrammatic rewriting

- **Example (Associative category).** The associative theory is presented by the relation

$$(a \star_0 b) \star_0 c = a \star_0 (b \star_0 c).$$

Diagrammatically, this relation can be oriented as the rewriting rule on 2-cells

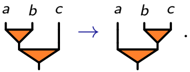


Diagrammatic rewriting

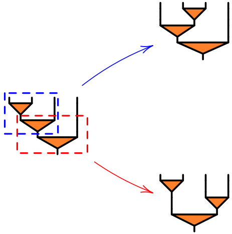
- **Example (Associative category).** The associative theory is presented by the relation

$$(a \star_0 b) \star_0 c = a \star_0 (b \star_0 c).$$

Diagrammatically, this relation can be oriented as the rewriting rule on 2-cells



- Overlapping applications of this rule naturally give rise to a critical branching:

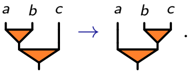


Diagrammatic rewriting

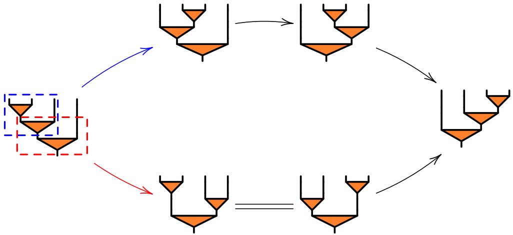
- **Example (Associative category).** The associative theory is presented by the relation

$$(a \star_0 b) \star_0 c = a \star_0 (b \star_0 c).$$

Diagrammatically, this relation can be oriented as the rewriting rule on 2-cells



- Overlapping applications of this rule naturally give rise to a critical branching, which is confluent:



Problem

- In general, proving convergence for diagrammatic rewriting can be challenging.

Problem

- In general, proving convergence for diagrammatic rewriting can be challenging.
  - Large sets of rewriting rules may require

Problem

- In general, proving convergence for diagrammatic rewriting can be challenging.
  - Large sets of rewriting rules may require
    - **Complex termination functions** (e.g. KLR algebras);
    - The analysis of **many types of critical branchings** (e.g. the q-Schur category).

Problem

- In general, proving convergence for diagrammatic rewriting can be challenging.
  - Large sets of rewriting rules may require
    - **Complex termination functions** (e.g. KLR algebras);
    - The analysis of **many types of critical branchings** (e.g. the q-Schur category).
  - Two-dimensional rewriting involves **complex interactions between rules**.

Problem

- In general, proving convergence for diagrammatic rewriting can be challenging.
  - Large sets of rewriting rules may require
    - **Complex termination functions** (e.g. KLR algebras);
    - The analysis of **many types of critical branchings** (e.g. the q-Schur category).
  - Two-dimensional rewriting involves **complex interactions between rules**.
- **Our objective**
  Develop completion procedures and termination methods adapted to diagrammatic rewriting by stratification.

Ideas

- **Stratified normalisation completion modulo**
  - Rewriting modulo (Huet '80, Jouannaud-Kirchner '84);
  - Normalisation and completion modulo (Marche '96);
  - Stratification ideas (Jouannaud-Li '12).

Ideas

- **Stratified normalisation completion modulo**
  - Rewriting modulo (Huet '80, Jouannaud-Kirchner '84);
  - Normalisation and completion modulo (Marche '96);
  - Stratification ideas (Jouannaud-Li '12).

- **Our approach**

  We stratify the set R of rewriting rules according to their termination and confluence properties.

Ideas

- **Stratified normalisation completion modulo**
    - Rewriting modulo (Huet '80, Jouannaud-Kirchner '84);
    - Normalisation and completion modulo (Marche '96);
    - Stratification ideas (Jouannaud-Li '12).

- **Our approach**

  We stratify the set R of rewriting rules according to their termination and confluence properties.

    - Let (X, R) be a rewriting system with a stratification

$$R = R_1 \sqcup R_2 \sqcup \cdots \sqcup R_k.$$

Ideas

- **Stratified normalisation completion modulo**
  - Rewriting modulo (Huet '80, Jouannaud-Kirchner '84);
  - Normalisation and completion modulo (Marche '96);
  - Stratification ideas (Jouannaud-Li '12).

- **Our approach**

  We stratify the set R of rewriting rules according to their termination and confluence properties.

  - Let (X, R) be a rewriting system with a stratification

$$R = R_1 \sqcup R_2 \sqcup \cdots \sqcup R_k.$$

  - Even if each stratum R_i is confluent or terminating, their union R is, in general, not. Thus, confluence and termination property are not **modular**.

Ideas

- **Stratified normalisation completion modulo**
  - Rewriting modulo (Huet '80, Jouannaud-Kirchner '84);
  - Normalisation and completion modulo (Marche '96);
  - Stratification ideas (Jouannaud-Li '12).

- **Our approach**

  We stratify the set R of rewriting rules according to their termination and confluence properties.
  - Let (X, R) be a rewriting system with a stratification

  $$R = R_1 \sqcup R_2 \sqcup \cdots \sqcup R_k.$$

  - Even if each stratum R_i is confluent or terminating, their union R is, in general, not. Thus, confluence and termination property are not **modular**.

- **Question**

  Can one construct a **Tietze equivalent** rewriting system (X, R') endowed with a stratification for which these properties become modular?

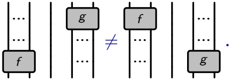
Contents

Stratified rewriting systems

- A **linear 2-precategory** $\mathcal{C}$ is a linear 2-category consisting of 0-cells, 1-cells, and 2-cells, where 2-cells can be composed both horizontally and vertically, such that
  - The **interchange laws** are not imposed, that is, the vertical composition of 2-cells is not required to be functorial:

$$(f \circ_0 id \circ_0 s_1(g)) \circ_1 (t_1(f) \circ_0 id \circ_0 g) \neq (s_1(f) \circ_0 id \circ_0 g) \circ_1 (f \circ_0 id \circ_0 t_1(g)).$$

  Diagrammatically,

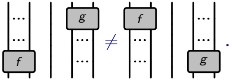


Stratified rewriting systems

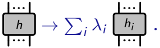
- A **linear 2-precategory** $\mathcal{C}$ is a linear 2-category consisting of 0-cells, 1-cells, and 2-cells, where 2-cells can be composed both horizontally and vertically, such that
  - The **interchange laws** are not imposed, that is, the vertical composition of 2-cells is not required to be functorial:

$$(f \circ_0 id \circ_0 s_1(g)) \circ_1 (t_1(f) \circ_0 id \circ_0 g) \neq (s_1(f) \circ_0 id \circ_0 g) \circ_1 (f \circ_0 id \circ_0 t_1(g)).$$

Diagrammatically,



- A **rewriting rule** on $\mathcal{C}$ is of the form



A **stratification** of length k of the set R of rewriting rules is a partition

$$R = F_s^1 R \sqcup F_s^2 R \sqcup \cdots \sqcup F_s^k R,$$

denoted by $F_s^* R$. For every $1 \leq i \leq k$, set

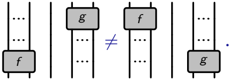
$$F^i R = F_s^1 R \sqcup \cdots \sqcup F_s^i R.$$

Stratified rewriting systems

- A **linear 2-precategory** $\mathcal{C}$ is a linear 2-category consisting of 0-cells, 1-cells, and 2-cells, where 2-cells can be composed both horizontally and vertically, such that
  - The **interchange laws** are not imposed, that is, the vertical composition of 2-cells is not required to be functorial:

$$(f \circ_0 id \circ_0 s_1(g)) \circ_1 (t_1(f) \circ_0 id \circ_0 g) \neq (s_1(f) \circ_0 id \circ_0 g) \circ_1 (f \circ_0 id \circ_0 t_1(g)).$$

  Diagrammatically,



- A **rewriting rule** on $\mathcal{C}$ is of the form

  A **stratification** of length k of the set R of rewriting rules is a partition

$$R = F_s^1 R \sqcup F_s^2 R \sqcup \cdots \sqcup F_s^k R,$$

  denoted by $F_s^* R$. For every $1 \leq i \leq k$, set

$$F^i R = F_s^1 R \sqcup \cdots \sqcup F_s^i R.$$

- A set R of rewriting rules equipped with such a stratification is called a ***k*-stratified rewriting system**.

Stratified termination functions

- A **weighted function** on a precategory $\mathcal{C}$ is a hom-set–wise function

$$\delta : \mathcal{C} \longrightarrow (Z, \prec),$$

such that

$$\delta(f) \prec \delta(g) \quad \text{implies} \quad \delta(C[f]) \prec \delta(C[g]),$$

for every context C.

Stratified termination functions

- A **weighted function** on a precategory $\mathcal{C}$ is a hom-set–wise function

$$\delta : \mathcal{C} \longrightarrow (Z, \prec),$$

such that

$$\delta(f) \prec \delta(g) \quad \text{implies} \quad \delta(C[f]) \prec \delta(C[g]),$$

for every context C.

- A **termination function** of a k-stratified rewriting system $F_s^* R$ is a family of weighted functions $(\delta_i)_{1 \leq i \leq k}$, such that

$$\begin{cases} \delta_i(t\alpha) < \delta_i(s\alpha), & \alpha \in F_s^i R, \\ \delta_i(t\beta) \leq \delta_i(s\beta), & \beta \in F^{i-1} R. \end{cases}$$

Stratified termination functions

- A **weighted function** on a precategory $\mathcal{C}$ is a hom-set–wise function

$$\delta : \mathcal{C} \longrightarrow (Z, \prec),$$

such that

$$\delta(f) \prec \delta(g) \quad \text{implies} \quad \delta(C[f]) \prec \delta(C[g]),$$

for every context C.

- A **termination function** of a k-stratified rewriting system $F_s^* R$ is a family of weighted functions $(\delta_i)_{1 \leq i \leq k}$, such that

$$\begin{cases} \delta_i(t\alpha) < \delta_i(s\alpha), & \alpha \in F_s^i R, \\ \delta_i(t\beta) \leq \delta_i(s\beta), & \beta \in F^{i-1} R. \end{cases}$$

- **Theorem.** If a k-stratified rewriting system $F_s^* R$ admits a termination function, then it is terminating.

Stratified termination functions

- A **weighted function** on a precategory $\mathcal{C}$ is a hom-set–wise function

$$\delta : \mathcal{C} \longrightarrow (Z, \prec),$$

such that

$$\delta(f) \prec \delta(g) \quad \text{implies} \quad \delta(C[f]) \prec \delta(C[g]),$$

for every context C.

- A **termination function** of a k-stratified rewriting system $F_s^* R$ is a family of weighted functions $(\delta_i)_{1 \leq i \leq k}$, such that

$$\begin{cases} \delta_i(t\alpha) < \delta_i(s\alpha), & \alpha \in F_s^i R, \\ \delta_i(t\beta) \leq \delta_i(s\beta), & \beta \in F^{i-1} R. \end{cases}$$

- **Theorem.** If a k-stratified rewriting system $F_s^* R$ admits a termination function, then it is terminating.

- **Remark.** Stratified termination functions generalize the derivation approach introduced by Guiraud–Malbos '09, which is widely used in diagrammatic rewriting.

Contents

Normalisation

Input. Let $F_s^* R$ be a k-stratified rewriting system with a compatible **monomial order** $\preccurlyeq$.

Output. A **normalisation** $F_s^* \widehat{R}$ of $F_s^* R$.

Normalisation

Input. Let $F_s^* R$ be a k-stratified rewriting system with a compatible **monomial order** $\preccurlyeq$.

Output. A **normalisation** $F_s^* \widehat{R}$ of $F_s^* R$.

Step 1. Set $F_s \widehat{R} := F_s R$ and choose a **normalisation strategy** $(-)\!\downarrow$ on $F^1 \widehat{R}$.

Normalisation

Input. Let $F_s^* R$ be a k-stratified rewriting system with a compatible **monomial order** $\preccurlyeq$.

Output. A **normalisation** $F_s^* \widehat{R}$ of $F_s^* R$.

Step 1. Set $F_s \widehat{R} := F_s R$ and choose a **normalisation strategy** $(-)\!\downarrow$ on $F^1 \widehat{R}$.

Step 2. Normalise every $\alpha \in F_s^2 R$ in each admissible context C to define a new rule $C[\alpha]\!\downarrow$:

$$\begin{tikzcd} C[s\alpha] \arrow[r, "C[\alpha]"] \arrow[d, twoheadrightarrow] & C[t\alpha] \arrow[d, twoheadrightarrow] \\ C[s\alpha]\!\downarrow \arrow[r, leftrightarrow, dashed, "C[\alpha]\downarrow"] & C[t\alpha]\!\downarrow \end{tikzcd}$$

Normalisation

Input. Let $F_s^* R$ be a k-stratified rewriting system with a compatible **monomial order** $\preccurlyeq$.

Output. A **normalisation** $F_s^* \widehat{R}$ of $F_s^* R$.

Step 1. Set $F_s \widehat{R} := F_s R$ and choose a **normalisation strategy** $(-)\!\downarrow$ on $F^1 \widehat{R}$.

Step 2. Normalise every $\alpha \in F_s^2 R$ in each admissible context C to define a new rule $C[\alpha]\!\downarrow$:

$$\begin{tikzcd}
C[s\alpha] \arrow[r, "C[\alpha]"] \arrow[d, two heads] & C[t\alpha] \arrow[d, two heads] \\
C[s\alpha]\!\downarrow & C[t\alpha]\!\downarrow \arrow[l, dashed, leftrightarrow, "C[\alpha]\downarrow"']
\end{tikzcd}$$

Step 3. Define the set of rewriting rules

$$[\![\alpha]\!]\!\downarrow := \{C[\alpha]\!\downarrow\}. \qquad \text{and} \qquad F_s^2 \widehat{R} := \bigsqcup_{\alpha \in F_s^2 R} \min_{\sqsubseteq} [\![\alpha]\!]\!\downarrow,$$

where $\beta_1 \sqsubseteq \beta_2$ if $\beta_2 = D[\beta_1]$ for some context D.

Normalisation

Input. Let $F_s^* R$ be a k-stratified rewriting system with a compatible **monomial order** $\preccurlyeq$.

Output. A **normalisation** $F_s^* \widehat{R}$ of $F_s^* R$.

Step 1. Set $F_s \widehat{R} := F_s R$ and choose a **normalisation strategy** $(-)\!\downarrow$ on $F^1 \widehat{R}$.

Step 2. Normalise every $\alpha \in F_s^2 R$ in each admissible context C to define a new rule $C[\alpha]\!\downarrow$:

$$
\begin{tikzcd}
C[s\alpha] \arrow[r, "C[\alpha]"] \arrow[d, twoheadrightarrow] & C[t\alpha] \arrow[d, twoheadrightarrow] \\
C[s\alpha]\!\downarrow & C[t\alpha]\!\downarrow \arrow[l, dashed, leftrightarrow, "C[\alpha]\downarrow"']
\end{tikzcd}
$$

Step 3. Define the set of rewriting rules

$$
[\![\alpha]\!]\!\downarrow := \{C[\alpha]\!\downarrow\}. \qquad \text{and} \qquad F_s^2 \widehat{R} := \bigsqcup_{\alpha \in F_s^2 R} \min_{\sqsubseteq} [\![\alpha]\!]\!\downarrow,
$$

where $\beta_1 \sqsubseteq \beta_2$ if $\beta_2 = D[\beta_1]$ for some context D.

Iteration. Continue recursively to define $F_s^i \widehat{R}$ for all $3 \leq i \leq k$.

Main results

- The **confluence** of the normalisation $\widehat{R}$ is **modular**:

  Theorem (Modularity of stratified rewriting).
  If every stratum $F_s^i\widehat{R}$ is confluent for $1 \leq i \leq k$, then their union $\widehat{R} = F^k\widehat{R}$ is confluent.

Main results

- The **confluence** of the normalisation $\widehat{R}$ is **modular**:

  Theorem (Modularity of stratified rewriting).
  If every stratum $F_s^i\widehat{R}$ is confluent for $1 \leq i \leq k$, then their union $\widehat{R} = F^k\widehat{R}$ is confluent.

- **Corollary.** Let R be a stratified rewriting system presenting a linear monoidal category $\mathcal{C}$. Then the set $\mathrm{Nf}_m(\widehat{R})$ forms a linear hom-basis of $\mathcal{C}$.

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} \boldsymbol{xy} & \to & \boldsymbol{yx}, \\ \boldsymbol{yz} & \to & \boldsymbol{zy}, \\ \boldsymbol{xz} & \to & \boldsymbol{zx} \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : \boldsymbol{xyz} \to \boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2 \right\}.$$

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} \boldsymbol{xy} & \rightarrow & \boldsymbol{yx}, \\ \boldsymbol{yz} & \rightarrow & \boldsymbol{zy}, \\ \boldsymbol{xz} & \rightarrow & \boldsymbol{zx} \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : \boldsymbol{xyz} \rightarrow \boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{ z^i y^j x^k \mid i, j, k \geq 0 \}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \, \square \, z^p y^q x^r .$$

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} \boldsymbol{xy} & \to & \boldsymbol{yx}, \\ \boldsymbol{yz} & \to & \boldsymbol{zy}, \\ \boldsymbol{xz} & \to & \boldsymbol{zx} \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : \boldsymbol{xyz} \to \boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{ z^i y^j x^k \mid i, j, k \geq 0 \}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \, \square \, z^p y^q x^r.$$

Step 2. Compute the normal reduction:

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} \boldsymbol{xy} & \to & \boldsymbol{yx}, \\ \boldsymbol{yz} & \to & \boldsymbol{zy}, \\ \boldsymbol{xz} & \to & \boldsymbol{zx} \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : \boldsymbol{xyz} \to \boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{z^i y^j x^k \mid i, j, k \geq 0\}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \,\Box\, z^p y^q x^r.$$

Step 2. Compute the normal reduction:

$$C[\alpha] : \boldsymbol{z}^{\boldsymbol{i}} \boldsymbol{y}^{\boldsymbol{j}} \boldsymbol{x}^{\boldsymbol{k}} \, \boldsymbol{xyz} \, \boldsymbol{z}^{\boldsymbol{p}} \boldsymbol{y}^{\boldsymbol{q}} \boldsymbol{x}^{\boldsymbol{r}} \quad \to \quad \boldsymbol{z}^{\boldsymbol{i}} \boldsymbol{y}^{\boldsymbol{j}} \boldsymbol{x}^{\boldsymbol{k}} \, (\boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2) \, \boldsymbol{z}^{\boldsymbol{p}} \boldsymbol{y}^{\boldsymbol{q}} \boldsymbol{x}^{\boldsymbol{r}}.$$

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} \boldsymbol{xy} & \to & \boldsymbol{yx}, \\ \boldsymbol{yz} & \to & \boldsymbol{zy}, \\ \boldsymbol{xz} & \to & \boldsymbol{zx} \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : \boldsymbol{xyz} \to \boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{ z^i y^j x^k \mid i, j, k \geq 0 \}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \,\square\, z^p y^q x^r.$$

Step 2. Compute the normal reduction:

$$C[\alpha]\!\downarrow\colon \boldsymbol{z^{i+p+1} y^{j+q+1} x^{k+r+1}} \to \boldsymbol{z^{i+p} y^{j+q} x^{k+r+2}} + \boldsymbol{z^{i+p} y^{j+q+2} x^{k+r}} + \boldsymbol{z^{i+p+2} y^{j+q} x^{k+r}}.$$

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} \boldsymbol{xy} & \to & \boldsymbol{yx}, \\ \boldsymbol{yz} & \to & \boldsymbol{zy}, \\ \boldsymbol{xz} & \to & \boldsymbol{zx} \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : \boldsymbol{xyz} \to \boldsymbol{x}^2 + \boldsymbol{y}^2 + \boldsymbol{z}^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{ z^i y^j x^k \mid i, j, k \geq 0 \}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \,\square\, z^p y^q x^r.$$

Step 2. Compute the normal reduction:

$$C[\alpha]{\downarrow}\colon \boldsymbol{z}^{\cancel{i+p}+1} \boldsymbol{y}^{j+q+1} \boldsymbol{x}^{\cancel{k+r}+1} \to \boldsymbol{z}^{\cancel{i+p}} \boldsymbol{y}^{j+q} \boldsymbol{x}^{\cancel{k+r}+2} + \boldsymbol{z}^{\cancel{i+p}} \boldsymbol{y}^{j+q+2} \boldsymbol{x}^{\cancel{k+r}} + \boldsymbol{z}^{\cancel{i+p}+2} \boldsymbol{y}^{j+q} \boldsymbol{x}^{\cancel{k+r}}.$$

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} xy & \rightarrow & yx, \\ yz & \rightarrow & zy, \\ xz & \rightarrow & zx \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : xyz \rightarrow x^2 + y^2 + z^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{ z^i y^j x^k \mid i, j, k \geq 0 \}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \, \square \, z^p y^q x^r.$$

Step 2. Compute the normal reduction:

$$F_s^2 \widehat{R} := \left\{ zy^{n+1}x \overset{C[\alpha]\downarrow}{\rightarrow} y^n x^2 + y^{n+2} + z^2 y^n \;\middle|\; n \geq 0 \right\}.$$

Example

Goal. Compute a linear basis of the presented algebra by stratification.

Input. A 2-stratified rewriting system $F_s^* R$ with generators x, y, z and

$$F_s^1 R = \left\{ \begin{array}{ccc} xy & \to & yx, \\ yz & \to & zy, \\ xz & \to & zx \end{array} \right\}, \qquad F_s^2 R = \left\{ \alpha : xyz \to x^2 + y^2 + z^2 \right\}.$$

Step 1. Let $\preccurlyeq$ be the lexicographic order with $z \prec y \prec x$. The first stratum is confluent:

$$F_s^1 \widehat{R} := F_s^1 R, \qquad \mathrm{Nf}(F^1 \widehat{R}) = \{ z^i y^j x^k \mid i, j, k \geq 0 \}.$$

Hence every admissible context is of the form

$$C = z^i y^j x^k \, \square \, z^p y^q x^r.$$

Step 2. Compute the normal reduction:

$$F_s^2 \widehat{R} := \left\{ zy^{n+1}x \overset{C[\alpha]\downarrow}{\to} y^n x^2 + y^{n+2} + z^2 y^n \;\middle|\; n \geq 0 \right\}.$$

Conclusion.
The stratum $F_s^2 \widehat{R}$ has no critical branching. Hence $\widehat{R}$ is confluent. Therefore, the presented algebra admits the linear basis

$$\mathrm{Nf}_m(\widehat{R}) = \{ z^i x^j, \; y^i x^j, \; z^i y^j \mid i, j \geq 0 \}.$$

Stratified normalisation and Knuth–Bendix completion

Combining the stratified normalisation procedure with the Knuth-Bendix completion procedure yields the following **completion procedure for stratified rewriting systems**.

$$\underbrace{F_s^1 \quad \sqcup \quad F_s^2 \quad \sqcup \quad \cdots \quad \sqcup \quad F_s^k}_{\text{SN}}$$

$$\begin{tikzcd}
\widehat{F_s^1} \arrow[d, "\text{KB}"'] & \sqcup & \widehat{F_s^2} \arrow[d, "\text{KB}"] & \sqcup & \cdots & \sqcup & \widehat{F_s^k} \arrow[d, "\text{KB}"] \\
\widetilde{\widehat{F_s^1}} & \sqcup & \widetilde{\widehat{F_s^2}} & \sqcup & \cdots & \sqcup & \widetilde{\widehat{F_s^k}}
\end{tikzcd}$$

$$\underbrace{\widetilde{\widehat{F_s^1}} \quad \sqcup \quad \widetilde{\widehat{F_s^2}} \quad \sqcup \quad \cdots \quad \sqcup \quad \widetilde{\widehat{F_s^k}}}_{\text{SN}}$$

$$\widehat{\widetilde{\widehat{F_s^1}}} \quad \sqcup \quad \widehat{\widetilde{\widehat{F_s^2}}} \quad \sqcup \quad \cdots \quad \sqcup \quad \widehat{\widetilde{\widehat{F_s^k}}}$$

$$\vdots$$

Contents

Filtration of admissible contexts

- For **string rewriting systems**, admissible contexts can often be described recursively.

$$\rightarrow \ \square \ \leftarrow \qquad (e.g.\ z^i y^j x^k \ xyz \ z^p y^q x^r)$$

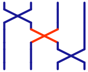
Filtration of admissible contexts

- For **string rewriting systems**, admissible contexts can often be described recursively.

$$\rightarrow \square \leftarrow \qquad (e.g.\ z^i y^j x^k\ xyz\ z^p y^q x^r)$$

- For **diagrammatic rewriting systems**, however, admissible contexts are typically much harder to compute explicitly.

$$\begin{array}{ccc} & \downarrow & \\ \rightarrow & \square & \leftarrow \\ & \uparrow & \end{array} \qquad (e.g.$$



$)$

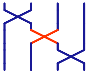
Filtration of admissible contexts

- For **string rewriting systems**, admissible contexts can often be described recursively.

$$\rightarrow \ \square \ \leftarrow \qquad (e.g.\ z^i y^j x^k \ xyz \ z^p y^q x^r)$$

- For **diagrammatic rewriting systems**, however, admissible contexts are typically much harder to compute explicitly.

$$\begin{array}{ccc} & \downarrow & \\ \rightarrow & \square & \leftarrow \\ & \uparrow & \end{array} \qquad (e.g.$$

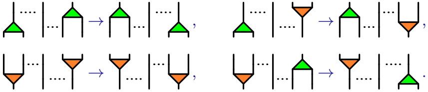


$)$

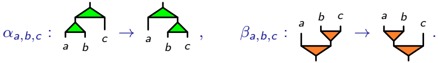
- Hence, we introduce a technical method to deal with these complicated contexts according to their **length**.

Example

Input. A 2-category $\mathcal{C}$ presented by a 2-stratified rewriting system $F_s^* R$.
The first stratum $F_s^1 R$ consists of

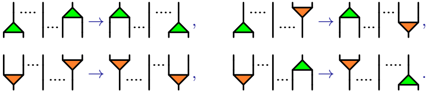


and $F_s^2 R$ consists of

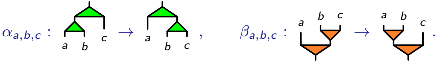


Example

Input. A 2-category $\mathcal{C}$ presented by a 2-stratified rewriting system $F_s^* R$.
The first stratum $F_s^1 R$ consists of



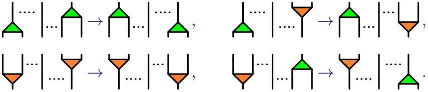
and $F_s^2 R$ consists of



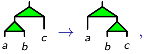
Contexts of length 0. Set $F_s^1 \widehat{R} = F_s^1 R$. Since $C[\alpha]{\downarrow} = \alpha$ for contexts C of length 0, we obtain $\min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^0 = \{\alpha\}$.

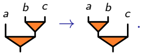
Example

Input. A 2-category $\mathcal{C}$ presented by a 2-stratified rewriting system $F_s^* R$.
The first stratum $F_s^1 R$ consists of



and $F_s^2 R$ consists of

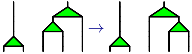
$\alpha_{a,b,c}$: 

, $\beta_{a,b,c}$: 

.

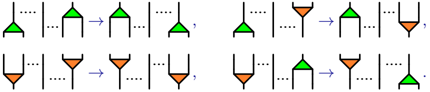
Contexts of length 0. Set $F_s^1 \widehat{R} = F_s^1 R$. Since $C[\alpha]{\downarrow} = \alpha$ for contexts C of length 0, we obtain $\min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^0 = \{\alpha\}$.

Contexts of length 1. Only four cases give non-trivial normalisations. For example,

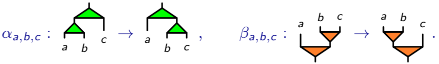
$C[\alpha]$: 

Example

Input. A 2-category $\mathcal{C}$ presented by a 2-stratified rewriting system $F_s^* R$.
The first stratum $F_s^1 R$ consists of



and $F_s^2 R$ consists of

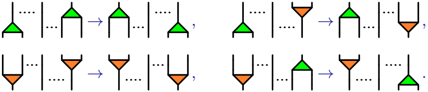


Contexts of length 0. Set $F_s^1 \widehat{R} = F_s^1 R$. Since $C[\alpha]{\downarrow} = \alpha$ for contexts C of length 0, we obtain $\min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^0 = \{\alpha\}$.

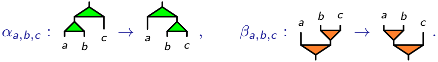
Contexts of length 1. Only four cases give non-trivial normalisations. For example,

Example

Input. A 2-category $\mathcal{C}$ presented by a 2-stratified rewriting system $F_s^* R$.
The first stratum $F_s^1 R$ consists of

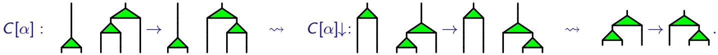


and $F_s^2 R$ consists of



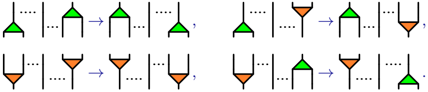
Contexts of length 0. Set $F_s^1 \widehat{R} = F_s^1 R$. Since $C[\alpha]{\downarrow} = \alpha$ for contexts C of length 0, we obtain $\min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^0 = \{\alpha\}$.

Contexts of length 1. Only four cases give non-trivial normalisations. For example,

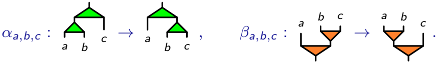


Example

Input. A 2-category $\mathcal{C}$ presented by a 2-stratified rewriting system $F_s^* R$.
The first stratum $F_s^1 R$ consists of

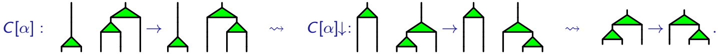


and $F_s^2 R$ consists of



Contexts of length 0. Set $F_s^1 \widehat{R} = F_s^1 R$. Since $C[\alpha]{\downarrow} = \alpha$ for contexts C of length 0, we obtain $\min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^0 = \{\alpha\}$.

Contexts of length 1. Only four cases give non-trivial normalisations. For example,



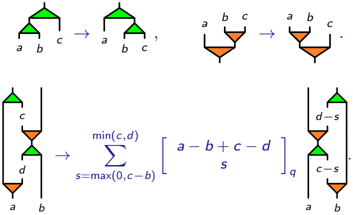
Conclusion. Hence

$$\min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^1 = \{\alpha\} = \min_{\sqsubseteq} [\![\alpha]\!]{\downarrow}_1^0, \qquad F_s^2 \widehat{R} = F_s^2 R.$$

Since $F_s^1 \widehat{R}$ and $F_s^2 \widehat{R}$ are confluent, $\mathrm{Nf}_m(\widehat{R})$ forms a hom-basis of $\mathcal{C}$.

Normal form basis of q-Schur category by stratification

- In Brundan '2025, a presentation of the q-Schur category is given:

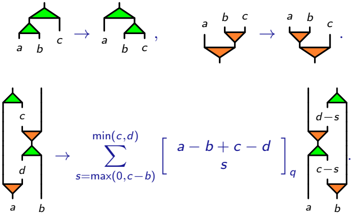


for all $a, b, c, d \geq 0$ with $d \leq a$ and $c \leq b + d$.

- From this presentation, we construct a 5-stratified rewriting system $F_s^*\mathrm{qS}$, compute its normalisation $\widehat{\mathrm{qS}}$ of the q-Schur category qS.

Normal form basis of q-Schur category by stratification

- In Brundan '2025, a presentation of the q-Schur category is given:



  for all $a, b, c, d \geq 0$ with $d \leq a$ and $c \leq b + d$.

- From this presentation, we construct a 5-stratified rewriting system $F_s^* \mathrm{qS}$, compute its normalisation $\widehat{\mathrm{qS}}$ of the q-Schur category qS.

- **Theorem.** The set $\mathrm{Nf}(\widehat{\mathrm{qS}})$ forms a hom-basis of the q-Schur category.

- This hom-basis, called the **normal form basis**, differs from the one obtained in Brundan '25.

Contents

Future work

- Develop a stratified normalisation procedure for **rewriting modulo**, with applications to:
  - Affine oriented Brauer categories Brundan-Comes-Nash-Reynolds '17;
  - Affine web categories Song-Wang '25;
  - Categories arising from Soergel bimodules Libedinsky '08.

Future work

- Develop a stratified normalisation procedure for **rewriting modulo**, with applications to:
  - Affine oriented Brauer categories Brundan-Comes-Nash-Reynolds '17;
  - Affine web categories Song-Wang '25;
  - Categories arising from Soergel bimodules Libedinsky '08.
- Extend stratified normalisation beyond compatible **monomial orders** by using **termination functions**.
  - This would allow the treatment of rewriting systems without suitable monomial orders.

Future work

- Develop a stratified normalisation procedure for **rewriting modulo**, with applications to:
  - Affine oriented Brauer categories Brundan-Comes-Nash-Reynolds '17;
  - Affine web categories Song-Wang '25;
  - Categories arising from Soergel bimodules Libedinsky '08.
- Extend stratified normalisation beyond compatible **monomial orders** by using **termination functions**.
  - This would allow the treatment of rewriting systems without suitable monomial orders.
- Improve stratified normalisation procedures through:
  - **Filtration** of admissible contexts according to their length;
  - **Automata descriptions** of admissible contexts.

Thank you for your attention.

Institut Camille Jordan

Université Claude Bernard Lyon 1