Affine Standard Lyndon words

Yehor Avdieiev
Mentor: Alexander Tsymbaliuk

14/05/2023

Main Results

- Generalization of Leclerc's algorithm describing Lalonde-Ram's bijection
$$\ell : \Delta^+ \xrightarrow{\sim} SL = \{\textit{standard Lyndon words}\}$$
from finite to affine Lie algebras
- Finding all *SL-words* for the "standard order" of simple roots in type $A_n^{(1)}$
- Finding the structure and some order properties for all *SL-words* for general order in type $A_n^{(1)}$
- Writing a Python code that founds all *SL-words* up to degree $k\delta$ (for any type, any order, and any k)

Part 1

- **Introduction**

Simple Lie algebras and root systems

- $\mathfrak{g}$ - Lie algebra: vector space with a skew-symmetric $[\cdot,\cdot] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$ satisfying

$$[[a,b],c] + [[b,c],a] + [[c,a],b] = 0 \quad \forall\, a,\ b,\ c \in \mathfrak{g}$$

- $\mathfrak{h} \subset \mathfrak{g}$ - Lie subalgebra: vector subspace s.t. $[\mathfrak{h},\mathfrak{h}] \subseteq \mathfrak{h}$
- $\mathfrak{h} \subset \mathfrak{g}$ - ideal if $[\mathfrak{g},\mathfrak{h}] \subseteq \mathfrak{h}$
- $\mathfrak{g}$ - **simple** if it is not abelian and has no nonzero proper ideals
- **Root system** is a pair $(V,\ \Delta)$, where V is a finite dimensional vector space over $\mathbb{R}$ with a positive definite bilinear form $(\cdot,\cdot)$ and $\Delta \subset V$ is a finite subset, such that:
  1. $0 \notin \Delta$; $\mathbb{R}\Delta = V$
  2. If $\alpha \in \Delta$, then $n\alpha \in \Delta$ if and only if $n = \pm 1$
  3. (String property). For any $\alpha, \beta \in \Delta$ we have:

$$\{\beta + j\alpha \,|\, j \in \mathbb{Z}\} \cap (\Delta \cup 0) = \{\beta + p\alpha, \ldots, \beta, \ldots, \beta - q\alpha\},$$

where p–$q = 2\frac{(\alpha,\beta)}{(\alpha,\alpha)}$

Simple roots and Cartan subalgebra

- Let (V, Δ) - root system, $f : V \to \mathbb{R}$ - linear map s.t. $f(\alpha) \neq 0$ $\forall \alpha \in \Delta$. Then:
  (i) $\alpha \in \Delta$ is *positive* if $f(\alpha) > 0$ and negative if $f(\alpha) < 0$
  (ii) Such root is *simple* if it is not a sum of two positive roots
  (iii) A *highest root* $\theta \in \Delta$ is such that $f(\theta) \geq f(\alpha)$ for all $\alpha \in \Delta$
- $\Delta^+ = \{\text{all positive roots}\}$, and $\Delta^- = -\Delta^+ = \{\text{all negative roots}\}$
- $\Pi \subset \Delta^+$ is the set of simple roots
- A Lie subalgebra $\mathfrak{h} \subset \mathfrak{g}$ is *Cartan* if it satisfies the following two conditions:
  1) $\mathfrak{h}$ *is nilpotent*, i.e $[\mathfrak{h}, [\mathfrak{h}, \cdots, [\mathfrak{h}, [\mathfrak{h}, \mathfrak{h}]] \cdots]] = 0$ for a finite number of brackets.
  2) $n_{\mathfrak{g}}(\mathfrak{h}) = \mathfrak{h}$, where $n_{\mathfrak{g}}(\mathfrak{h}) = \{x \in \mathfrak{g} | [x, \mathfrak{h}] \subset \mathfrak{h}\}$.

Root space decomposition

- Let $\mathfrak{g}$ - simple Lie algebra, $\mathfrak{h} \subseteq \mathfrak{g}$ - Cartan subalgebra. Then:

$$\mathfrak{g} = \bigoplus_{\alpha \in \mathfrak{h}^*} \mathfrak{g}_\alpha,$$

where

$$\mathfrak{g}_\alpha := \{x \in \mathfrak{g} | [h, x] = \alpha(h)x \ \forall h \in \mathfrak{h}\}$$

- Define a finite set of nonzero weights, called **roots of $\mathfrak{g}$ relative to $\mathfrak{h}$**:

$$\Delta = \{\alpha \in \mathfrak{h}^* | \mathfrak{g}_\alpha \neq 0\} \setminus \{0\}$$

- Above provides the *root space decomposition* of $\mathfrak{g}$:

$$\mathfrak{g} = \mathfrak{h} \oplus \bigoplus_{\alpha \in \Delta} \mathfrak{g}_\alpha, \ \textit{with } \mathit{dim}(\mathfrak{g}_\alpha) = 1 \ \forall \alpha \in \Delta$$

SL-words

- I - ordered finite alphabet, I^* - all finite length words in I
- For $u = i_1 i_2 \dots i_k \in I^*$, its length is $|u| = k$
- Get lexicographic order on I^*: $u = i_1 i_2 \dots i_k < v = j_1 j_2 \dots j_n$ iff $i_1 = j_1,\ i_2 = j_2, \dots,\ i_r > j_r$, or $i_1 = j_1,\ i_2 = j_2, \dots,\ i_k = j_k$ and $n > k$
- **Definition 1**: $\ell \in I^*$ is a **Lyndon word** if it is lexicographically smaller than all of its cyclic rearrangement
- $\mathfrak{a}$ - Lie algebra generated by a finite set $\{e_i\}_{i \in I}$ labelled by the alphabet I
- The standard bracketing of a Lyndon word ℓ is given inductively: $b[i] := e_i$ for $i \in I$, $b[\ell] := [b[m], b[n]]$, where $\ell = mn$ and n is the longest Lyndon word appearing as a proper right suffix of ℓ
- **Definition 2**: Lyndon word ℓ is **Lie-standard** w.r.t. $\mathfrak{a}$ if $b[\ell]$ cannot be written as a sum of bracketings of strictly larger Lyndon words

Leclerc's algorithm

- $\Pi = \{\alpha_i\}_{i \in I}$ - set of simple roots, I - our alphabet
- The weight of a word $w = i_1 i_2 \dots i_k \in I^*$ is defined by:

$$wt(w) = \alpha_{i_1} + \alpha_{i_2} + \dots + \alpha_{i_k}$$

- **Proposition** (Lyndon):
  $\mathfrak{g}_\alpha$ *is spanned by* $\{b[\ell] | \ \ell \text{ - } Lyndon, \ wt(\ell) = \alpha\}$
- **Theorem** (Lalonde-Ram, 1995):
  There is a bijection

$$\ell : \ \Delta^+ = \{positive\ roots\} \xrightarrow{\sim} SL = \{standard\ Lyndon\ words\}$$

  such that $deg \ell(\alpha) = \alpha$
- **Explicit algorithm** (Leclerc, 2004):

$$\ell(\alpha) = \max \left\{ \ell(\gamma_1)\ell(\gamma_2) \middle| \substack{\alpha = \gamma_1 + \gamma_2 \\ \gamma_1, \gamma_2 \in \Delta^+ \\ \ell(\gamma_1) < \ell(\gamma_2)} \right\}$$

Affine Lie algebras

- $\mathfrak{g}$ - simple finite dimensional Lie algebra
- $\{\alpha_i\}_{i\in I}$ - simple roots, $\theta \in \Delta^+$ - the highest root
- $\widehat{I} := I \sqcup \{0\}$
- The affine root lattice $\widehat{Q} = Q \times \mathbb{Z}$ with the generators $\{(\alpha_i, 0)\}_{i\in I}$ and $\alpha_0 := (-\theta, 1)$
- The affine root system $\widehat{\Delta} = \widehat{\Delta}^+ \sqcup \widehat{\Delta}^-$:

$$\widehat{\Delta}^+ = \left\{\Delta^+ \times \mathbb{Z}_{\geq 0}\right\} \sqcup \left\{0 \times \mathbb{Z}_{>0}\right\} \sqcup \left\{\Delta^- \times \mathbb{Z}_{>0}\right\}$$

$$\widehat{\Delta}^- = \left\{\Delta^- \times \mathbb{Z}_{\leq 0}\right\} \sqcup \left\{0 \times \mathbb{Z}_{<0}\right\} \sqcup \left\{\Delta^+ \times \mathbb{Z}_{<0}\right\}$$

- The corresponding Lie algebra $\widehat{\mathfrak{g}}$ is a central extension of loops into $\mathfrak{g}$, i.e.

$$\widehat{\mathfrak{g}} \simeq \mathfrak{g}[t, t^{-1}] \oplus \mathbb{C} \cdot c \text{ as a vector space}$$

Generalized Leclerc's algorithm

- $\widehat{\Delta}^{+,\mathrm{re}} := \left\{\Delta^+ \times \mathbb{Z}_{\geq 0}\right\} \sqcup \left\{\Delta^- \times \mathbb{Z}_{>0}\right\}$ - set of real roots
  $\widehat{\Delta}^{+,\mathrm{im}} := \left\{0 \times \mathbb{Z}_{>0}\right\}$ - set of imaginary roots
- **Proposition:** For simple roots, $\ell(\alpha_i) = [i]$. For other real $\alpha \in \widehat{\Delta}^{+,\mathrm{re}}$:

$$\ell(\alpha) \ = \max \left\{ \ell_*(\gamma_1)\ell_*(\gamma_2) \Big|_{\substack{\alpha=\gamma_1+\gamma_2,\, \gamma_k \in \widehat{\Delta}^+ \\ \ell_*(\gamma_1)<\ell_*(\gamma_2) \\ [b[\ell_*(\gamma_1)], b[\ell_*(\gamma_2)]]\neq 0}} \right\}, \qquad (1)$$

  where $\ell_*(\gamma)$ denotes $\ell(\gamma)$ for real γ, and any one of $\ell_k(\gamma)$ for imaginary γ
- **Proposition:** For imaginary $\alpha \in \widehat{\Delta}^{+,\mathrm{im}}$, the corresponding $\{\ell_k(\alpha)\}_{k=1}^{|I|}$ are the $|I| = rk(\mathfrak{g})$ lexicographically largest words from the list as in the right-hand side of (1) whose standard bracketings are linearly independent

Part 2

- **SL-words for the standard order** $1 < 2 < 3 < \cdots < n < 0$ **in type** $A_n^{(1)}$

SL-words for the standard order on $A_n^{(1)}$ with $n \geq 3$

- Define $\alpha_{i\to j} := \alpha_i + \alpha_{i+1} + \cdots + \alpha_j$, where letters are viewed as *mod* $(n+1)$ residues placed on a circle.
- **Theorem:** The *SL-words* for $k \geq 1$:

$$k\delta \leftrightarrow \begin{cases} 10n\dots(r+2)23\dots r\underbrace{10n\dots(r+1)23\dots r}_{k\text{ times}}(r+1), \ \textit{for } 1 \leq r < n \\ 123\dots n\underbrace{1023\dots n}_{k\text{ times}}0 \end{cases}$$

$$k\delta + \alpha_{i\to j} \leftrightarrow \underbrace{10n\cdots 23\dots(i-1)}_{k\text{ times}}i(i+1)\dots j, \ \textit{for } 2 < i \leq j$$

$$k\delta + \alpha_{1\to i} \leftrightarrow 123\dots n\underbrace{1023\dots n}_{(k-1)\text{ times}}1023\dots i, \ \textit{for } i \neq 0$$

$$k\delta + \alpha_2 \leftrightarrow \underbrace{10n\dots 32}_{k\text{ times}}2$$

SL-words for the standard order on $A_n^{(1)}$ with $n \geq 3$

$$k\delta + \alpha_{2\to j} \leftrightarrow \begin{cases} \underbrace{10n\dots32}_{\frac{k}{2}\text{ times}}\,2\,\underbrace{10n\dots32}_{\frac{k}{2}\text{ times}}\,34\dots j,\ k-\ \mathit{even} \\ \underbrace{10n\dots32}_{\frac{k+1}{2}\text{ times}}\,34\dots j\,\underbrace{10n\dots32}_{\frac{k-1}{2}\text{ times}}\,2,\ k-\ \mathit{odd} \end{cases},\ \mathit{for}\ j > 2$$

$$k\delta+\alpha_{j\to i} \leftrightarrow 10n\dots j23\dots(j-2)\,\underbrace{10n\dots(j-1)23\dots(j-2)}_{(k-1)\text{ times}}\,10n\dots(j-1)23\dots i$$

for $i+1<j$

The rest of the SL-words: $\alpha_{i\to j} \leftrightarrow i(i+1)\dots j$ (for $i \leq j$),
$\alpha_{j\to i} \leftrightarrow 10n\dots j23\dots i$ (for $i+1<j$),

$$\delta \leftrightarrow \begin{cases} 10n\dots(r+2)23\dots(r+1),\ \mathit{for}\ 1 \leq r < n \\ 123\dots n0 \end{cases}$$

SL-words for the standard order on $A_2^{(1)}$

The structure for $A_2^{(1)}$ is slightly different from the $n \geq 3$ case.

- **Theorem:** For $k \geq 1$:

$$k\delta + \alpha_1 \leftrightarrow 12 \underbrace{102}_{k-1\,\text{times}} 10$$

$$k\delta + \alpha_2 \leftrightarrow \underbrace{102}_{k\,\text{times}} 2$$

$$k\delta + \alpha_0 \leftrightarrow \underbrace{102}_{k\,\text{times}} 0$$

$$k\delta + \alpha_1 + \alpha_2 \leftrightarrow 12 \underbrace{102}_{k\,\text{times}}$$

$$k\delta + \alpha_1 + \alpha_0 \leftrightarrow 10 \underbrace{102}_{k\,\text{times}}$$

SL-words for the standard order for $A_2^{(1)}$

$$k\delta + \alpha_2 + \alpha_0 \leftrightarrow \begin{cases} \underbrace{102}_{\frac{k}{2}\text{ times}}\ 2\ \underbrace{102}_{\frac{k}{2}\text{ times}}\ 0 & , k\text{−even} \\ \underbrace{102}_{\frac{k+1}{2}\text{ times}}\ 0\ \underbrace{102}_{\frac{k-1}{2}\text{ times}}\ 2 & , k\text{−odd} \end{cases}$$

$$(k+1)\delta \leftrightarrow \begin{cases} 10\ \underbrace{102}_{k\text{ times}}\ 2 \\ 12\ \underbrace{102}_{k\text{ times}}\ 0 \end{cases}$$

For the remaining roots: $\alpha_1 \leftrightarrow 1$, $\alpha_1 + \alpha_2 \leftrightarrow 12$, $\alpha_2 \leftrightarrow 2$, $\alpha_1 + \alpha_0 \leftrightarrow 10$, $\alpha_0 \leftrightarrow 0$, $\alpha_2 + \alpha_0 \leftrightarrow 20$, $\alpha_1 + \alpha_2 + \alpha_0 \leftrightarrow \begin{cases} 102 \\ 120 \end{cases}$

Both results are proved by induction on k using generalized Leclerc's algorithm

Part 3

- **Structure and some order properties of** *SL-words* **for general order in type** $A_n^{(1)}$

Key features of the general order in type $A_n^{(1)}$

- Alike the standard order, the core is to compute *SL-words* for the root δ
- The structure of the SL-words is defined by the words between δ and 2δ:

$$\ell(\alpha + k\delta) = \ell_1 \underbrace{\ell(\delta)}_{k-1\,\text{times}} \ell_2, \text{ where } \ell(\alpha+\delta) = \ell_1\ell_2$$

- Using this fact we can prove some properties of the lex. order on SL-words
- **Lemma:** $\forall \alpha \in \widehat{\Delta}^{+,\mathrm{re}}$, the sequence $\ell(\alpha), \ell(\alpha+\delta), \ell(\alpha+2\delta), \ldots$ is monotonous
- **Conjecture:** For any $\alpha, \beta \in \widehat{\Delta}^{+,\mathrm{re}}$ such that $\alpha + \beta \in \widehat{\Delta}^{+,\mathrm{re}}$, have:

$$\alpha < \alpha + \beta < \beta \quad \text{or} \quad \beta < \alpha + \beta < \alpha$$

(analogue of Rosso's convexity property in finite types)

Part 4

- **Python code that founds all** *SL-words* **up to degree** $k\delta$

Python code

- Define function that computes standard bracketing of SL-words
- The code will work inductively. Suppose that we have a list of SL-words up to $k\delta$
- For each root between $k\delta$ and $(k+1)\delta$ find all possible variants of rewriting it into sum of two words from the list (knowing up to 2δ is enough)
- For real roots: count bracketing, find the largest word with the bracketing $\neq 0$
- For imaginary root: count bracketing, find the largest $|I| = rk(\mathfrak{g})$ words with linear independent bracketing

The End

Thank you!

SLAVA UKRAINE!

GEROYAM SLAVA!