Multilinear Lie bracket recursion formula

Michael Reed (2024), crucialflow.com, github.com/chakravala

Abstract

Presents multilinear Lie bracket optimized by Lie bracket recursion formula.

Let $\sigma \in S_n$ such that $\varepsilon(\sigma) = \varepsilon(\tau_1 \cdots \tau_i \cdots \tau_k) = (-1)^k$, where $\tau_i \in S_n$ are transpositions with $\varepsilon : S_n \to \{-1, 1\}$ being the sign of a permutation.

Definition 0.1. Let $[\cdot, \dots, \cdot] : \mathfrak{g}^n \to \mathfrak{g}$ define the n-linear Lie bracket with

$$[X_1, \dots, X_i, \dots, X_n] = \sum_{\sigma \in S_n} \varepsilon(\sigma) X_{\sigma(1)}(\dots(X_{\sigma(i)}(\dots(X_{\sigma(n)})\dots))\dots).$$

Theorem 0.2 (Anti-symmetric). *Multilinear Lie bracket is anti-symmetric,*

$$[X_{\rho(1)}, \dots, X_{\rho(i)}, \dots, X_{\rho(n)}] = \varepsilon(\rho)[X_1, \dots, X_i, \dots, X_n].$$

Proof. $[X_{\rho(1)}, \dots, X_{\rho(i)}, \dots, X_{\rho(n)}] =$

$$= \sum_{\sigma \in S_n} \varepsilon(\sigma) X_{\sigma\rho(1)}(\dots(X_{\sigma\rho(i)}(\dots(X_{\sigma\rho(n)})\dots))\dots)$$

$$= \varepsilon(\rho) \sum_{\sigma \in S_n} \varepsilon(\sigma\rho) X_{\sigma\rho(1)}(\dots(X_{\sigma\rho(i)}(\dots(X_{\sigma\rho(n)})\dots))\dots),$$

since for $\sigma \in S_n$ or $\sigma\rho \in S_n$ the sum is equivalent, the theorem is implied. □

Theorem 0.3 (Lie bracket recursion). *n-bracket is sum of $(n-1)$-brackets:*

$$[X_1, \dots, X_n] = \sum_{i=1}^{n} (-1)^{i-1} X_i([X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_n])$$

Proof. Define $\rho_i(\sigma) \in S_n$ for $i \in \{1, \dots, n\}$ and permuation $\sigma \in S_{n-1}$ with

$$\rho_i(\sigma) = \begin{pmatrix} i & 1 & \cdots & i-1 & i+1 & \cdots & n \\ \sigma(1) & \cdots & \sigma(i-1) & i & \sigma(i+1) & \cdots & \sigma(n) \end{pmatrix},$$

$$\sum_{i=1}^{n} (-1)^{i-1} X_i \left(\sum_{\sigma \in S_{n-1}} \varepsilon(\sigma) X_{\sigma(1)}(\dots(X_{\sigma(i-1)}(X_{\sigma(i+1)}(\dots(X_{\sigma(n)})\dots)))\dots) \right) =$$

$$\sum_{i=1}^{n} \sum_{\sigma \in S_{n-1}} (-1)^{i-1} \varepsilon(\sigma) X_i(X_{\sigma(1)}(\dots(X_{\sigma(i-1)}(X_{\sigma(i+1)}(\dots(X_{\sigma(n)})\dots)))\dots))$$

$$= \sum_{i=1}^{n} \sum_{\sigma \in S_{n-1}} \varepsilon(\rho_i(\sigma)) X_{\rho_i(\sigma)(1)}(\dots(X_{\rho_i(\sigma)(i-1)}(X_{\rho_i(\sigma)(i)}(X_{\rho_i(\sigma)(i+1)}(\dots(X_{\rho_i(\sigma)(n)})\dots))))\dots)$$

$$= \sum_{\rho_i(\sigma) \in S_n} \varepsilon(\rho_i(\sigma)) X_{\rho_i(\sigma)(1)}(\dots(X_{\rho_i(\sigma)(i-1)}(X_{\rho_i(\sigma)(i)}(X_{\rho_i(\sigma)(i+1)}(\dots(X_{\rho_i(\sigma)(n)})\dots))))\dots),$$

substitute RHS to apply anti-symmetry permutation, implying theorem. □

Example 0.4 (1-linear)**.** $[X]=X$ is trivial, computed with 0 evaluations.

Proof. Let $\lambda,\mu\in\mathbb{C}$ and $X_1,X_2\in\mathfrak{g}$, then $[\lambda X_1+\mu X_2]=\lambda[X_1]+\mu[X_2]$. $\square$

Example 0.5 (2-linear)**.** $[X,Y]=X([Y])-Y([X])=X(Y)-Y(X)$.

Proof. Let $\lambda,\mu\in\mathbb{C}$ and $X_1,X_2,Y\in\mathfrak{g}$ or $X,Y_1,Y_2\in\mathfrak{g}$, then

$$\begin{aligned}
[\lambda X_1+\mu X_2,Y] &= (\lambda X_1+\mu X_2)(Y)-Y(\lambda X_1+\lambda X_2)\\
&= \lambda(X_1(Y)-Y(X_1))+\mu(X_2(Y)-Y(X_2))\\
&= \lambda[X_1,Y]+\mu[X_2,Y],\\
[X,\lambda Y_1+\mu Y_2] &= X(\lambda Y_1+\lambda Y_2)-(\lambda Y_1+\mu Y_2)(X)\\
&= \lambda(X(Y_1)-Y_1(X))+\mu(X(Y_2)-Y_2(X))\\
&= \lambda[X,Y_1]+\mu[X,Y_2].
\end{aligned}$$

Hence $[\cdot,\cdot]:\mathfrak{g}\times\mathfrak{g}\to\mathfrak{g}$ is a bilinear Lie bracket with $2!=2\cdot 1$ evaluations. $\square$

Example 0.6 (3-linear)**.** Ternary Lie 3-bracket is defined by 12 evaluations:

$$[X,Y,Z]= X(Y(Z))-X(Z(Y))+Y(Z(X))-Y(X(Z))+Z(X(Y))-Z(Y(X))$$

Application of Lie bracket recursion results in $6=3!=3\cdot 2\cdot 1$ evaluations:

$$[X,Y,Z]=X([Y,Z])-Y([X,Z])+Z([X,Y]).$$

Proof. $\lambda,\mu\in\mathbb{C}$ and $X_1,X_2,Y,Z\in\mathfrak{g}$ or $X,Y_1,Y_2,Z\in\mathfrak{g}$ or $X,Y,Z_1,Z_2\in\mathfrak{g}$,

$$\begin{aligned}
&[\lambda X_1+\mu X_2,Y,Z]=\lambda[X_1,Y,Z]+\mu[X_2,Y,Z]\\
&=(\lambda X_1+\mu X_2)([Y,Z]))-Y([\lambda X_1+\mu X_2,Z])+Z([\lambda X_1+\mu X_2,Y])\\
&=\lambda X_1([Y,Z])+\mu X_2([Y,Z]))-Y(\lambda[X_1,Z]+\mu[X_2,Z])+Z(\lambda[X_1,Y]+\mu[X_2,Y])\\
&=\lambda(X_1([Y,Z])-Y([X_1,Z])+Z([X_1,Y]))+\mu(X_2([Y,Z])-Y([X_2,Z])+Z([X_2,Y])),
\end{aligned}$$

$$\begin{aligned}
&[X,\lambda Y_1+\mu Y_2,Z]=\lambda[X,Y_1,Z]+\mu[X,Y_2,Z]\\
&=X([\lambda Y_1+\mu Y_2,Z])-(\lambda Y_1+\mu Y_2)([X,Z])+Z([X,\lambda Y_1+\mu Y_2])\\
&=X(\lambda[Y_1,Z]+\mu[Y_2,Z])-\lambda Y_1([X,Z])-\mu Y_2([X,Z])+Z(\lambda[X,Y_1]+\mu[X,Y_2])\\
&=\lambda(X([Y_1,Z])-Y_1([X,Z])+Z([X,Y_1]))+\mu(X([Y_2,Z])-Y_2([X,Z])+Z([X,Y_2])),
\end{aligned}$$

$$\begin{aligned}
&[X,Y,\lambda Z_1+\mu Z_2]=\lambda[X,Y,Z_1]+\mu[X,Y,Z_2]\\
&=X([Y,\lambda Z_1+\mu Z_2])-Y([X,\lambda Z_1+\mu Z_2])+(\lambda Z_1+\mu Z_2)([X,Y])\\
&=X(\lambda[Y,Z_1]+\mu[Y,Z_2])-Y(\lambda[X,Z_1]+\mu[X,Z_2])+\lambda Z_1([X,Y])+\mu Z_2([X,Y])\\
&=\lambda(X([Y,Z_1])-Y([X,Z_1])+Z_1([X,Y]))+\mu(X([Y,Z_2])-Y([X,Z_2])+Z_2([X,Y])).
\end{aligned}$$

Hence $[\cdot,\cdot,\cdot]:\mathfrak{g}\times\mathfrak{g}\times\mathfrak{g}\to\mathfrak{g}$ is a trilinear Lie 3-bracket. $\square$

Example 0.7 (4-linear)**.** Lie 4-bracket is defined by 72 evaluations, while application of Lie bracket recursion results in $24 = 4! = 4\cdot3\cdot2\cdot1$ evaluations:

$$[W,X,Y,Z] = W([X,Y,Z]) - X([W,Y,Z]) + Y([W,X,Z]) - Z([W,X,Y]).$$

Proof of the 4-linearity follows similar principles as previous examples and is left as an exercise for the reader.

Proof. Let $W,X,Y,Z \in \mathfrak{g}$, then apply the multilinear Lie bracket recursion,

$$\begin{aligned}
[W,X,Y,Z] &= W(X(Y(Z))) + W(Y(Z(X))) + W(Z(X(Y))) \\
&\quad + X(W(Z(Y))) + X(Y(W(Z))) + X(Z(Y(W))) \\
&\quad + Y(W(X(Z))) + Y(X(Z(W))) + Y(Z(W(X))) \\
&\quad + Z(W(Y(X))) + Z(X(W(Y))) + Z(Y(X(W))) \\
&\quad - W(X(Z(Y))) - W(Y(X(Z))) - W(Z(Y(X))) \\
&\quad - X(W(Y(Z))) - X(Y(Z(W))) - X(Z(W(Y))) \\
&\quad - Y(W(Z(X))) - Y(X(W(Z))) - Y(Z(X(W))) \\
&\quad - Z(W(X(Y))) - Z(X(Y(W))) - Z(Y(W(X))), \\
[W,X,Y,Z] &= W(X([Y,Z])) - W(Y([X,Z])) + W(Z([X,Y])) \\
&\quad - X(W([Y,Z])) + X(Y([W,Z])) - X(Z([W,Y])) \\
&\quad + Y(W([X,Z])) - Y(X([W,Z])) + Y(Z([W,X])) \\
&\quad - Z(W([X,Y])) + Z(X([W,Y])) - Z(Y([W,X])), \\
[W,X,Y,Z] &= W([X,Y,Z]) - X([W,Y,Z]) + Y([W,X,Z]) - Z([W,X,Y]).
\end{aligned}$$

Hence the recursion formula reduces the evaluation count by $\frac{1}{3}$. □

Example 0.8 (5-linear)**.** Lie 5-bracket is defined by 480 evaluations, while application of Lie bracket recursion has $120 = 5! = 5\cdot4\cdot3\cdot2\cdot1$ evaluations:

$$[V,W,X,Y,Z] = V([W,X,Y,Z]) - W([V,X,Y,Z]) + X([V,W,Y,Z]) - Y([V,W,X,Z]) + Z([V,W,X,Y]).$$

Proof of the 5-linearity follows similar principles as previous examples and is left as an exercise for the reader.

Proof. Let $V,W,X,Y,Z \in \mathfrak{g}$, then apply multilinear Lie bracket recursion,

$$\begin{aligned}
[V,W,X,Y,Z] &= V(W([X,Y,Z])) - V(X([W,Y,Z])) + V(Y([W,X,Z])) - V(Z([W,X,Y])) \\
&\quad - W(V([X,Y,Z])) + W(X([V,Y,Z])) - W(Y([V,X,Z])) + W(Z([V,X,Y])) \\
&\quad + X(V([W,Y,Z])) - X(W([V,Y,Z])) + X(Y([V,W,Z])) - X(Z([V,W,Y])) \\
&\quad - Y(V([W,X,Z])) + Y(W([V,X,Z])) - Y(X([V,W,Z])) + Y(Z([V,W,X])) \\
&\quad + Z(V([W,X,Y])) - Z(W([V,X,Y])) + Z(X([V,W,Y])) - Z(Y([V,W,X])),
\end{aligned}$$

$$\begin{aligned}
[V,W,X,Y,Z] = {} & V(W(X([Y,Z]))) - V(W(Y([X,Z]))) + V(W(Z([X,Y]))) \\
& - V(X(W([Y,Z]))) + V(X(Y([W,Z]))) - V(X(Z([W,Y]))) \\
& + V(Y(W([X,Z]))) - V(Y(X([W,Z]))) + V(Y(Z([W,X]))) \\
& - V(Z(W([X,Y]))) + V(Z(X([W,Y]))) - V(Z(Y([W,X]))) \\
& - W(V(X([Y,Z]))) + W(V(Y([X,Z]))) - W(V(Z([X,Y]))) \\
& + W(X(V([Y,Z]))) - W(X(Y([V,Z]))) + W(X(Z([V,Y]))) \\
& - W(Y(V([X,Z]))) + W(Y(X([V,Z]))) - W(Y(Z([V,X]))) \\
& + W(Z(V([X,Y]))) - W(Z(X([V,Y]))) + W(Z(Y([V,X]))) \\
& + X(V(W([Y,Z]))) - X(V(Y([W,Z]))) + X(V(Z([W,Y]))) \\
& - X(W(V([Y,Z]))) + X(W(Y([V,Z]))) - X(W(Z([V,Y]))) \\
& + X(Y(V([W,Z]))) - X(Y(W([V,Z]))) + X(Y(Z([V,W]))) \\
& - X(Z(V([W,Y]))) + X(Z(W([V,Y]))) - X(Z(Y([V,W]))) \\
& - Y(V(W([X,Z]))) + Y(V(X([W,Z]))) - Y(V(Z([W,X]))) \\
& + Y(W(V([X,Z]))) - Y(W(X([V,Z]))) + Y(W(Z([V,X]))) \\
& - Y(X(V([W,Z]))) + Y(X(W([V,Z]))) - Y(X(Z([V,W]))) \\
& + Y(Z(V([W,X]))) - Y(Z(W([V,X]))) + Y(Z(X([V,W]))) \\
& + Z(V(W([X,Y]))) - Z(V(X([W,Y]))) + Z(V(Y([W,X]))) \\
& - Z(W(V([X,Y]))) + Z(W(X([V,Y]))) - Z(W(Y([V,X]))) \\
& + Z(X(V([W,Y]))) - Z(X(W([V,Y]))) + Z(X(Y([V,W]))) \\
& - Z(Y(V([W,X]))) + Z(Y(W([V,X]))) - Z(Y(X([V,W]))),
\end{aligned}$$

final step of the multilinear Lie bracket recursion is left as an exercise for the reader. Hence the recursion formula reduces the evaluation count by $\frac{1}{4}$. ☐

Corollary 0.9. Multilinear Lie bracket definition has $n!(n-1)$ evaluations.

Corollary 0.10. Multilinear Lie bracket recursion has $n!$ evaluations.

Corollary 0.11. Multilinear Lie bracket recursion is more efficient by $\frac{1}{n-1}$.

In conclusion, the presented multilinear Lie bracket definition properly generalizes the bilinear Lie bracket to the n-linear case, while the proof of the multilinear Lie bracket recursion enables optimizing evaluations by $\frac{1}{n-1}$. The multilinear Lie bracket recursion is analogous to the Koszul complex of the Grassmann algebra; although it is fundamentally different due to the multilinear Lie bracket being non-associative, unlike the analogous exterior product. The order of n-linearity is not restricted by the dimension of the underlying vector space. Instead, the multilinear Lie bracket is limited by the order of differentiation or unlimited by smoothness of the coefficients.