OpTion Manual

Haobo Yan (燕浩波) HAOBO@STU.PKU.EDU.CN

Institute of Theoretical Physics, School of Physics, Peking University

Contents

1. Installation

OpTion (Operator generaTion) is a Mathematica package for operator construction in lattice QCD. In the current version, **OpTion** can construct arbitrary-body operators for mesons, baryons, or arbitrary spin particles. **OpTion** can also construct one-hadron operators from gamma matrices and covariant derivatives.

There is a `Test.nb` file under the repository Tests and can run directly. To install the package,

1. Clone the zip file from GitHub.

2. Extract the zip file and move the repository to the Applications directory inside `FileNameJoin[{$UserBaseDirectory, "Applications"}]`.

To load the package, run

```
<< OpTion`
```

2. Basics

In **OpTion**, most functions read in arguments include the following quantities:

- `Par`: the particle to build the operators. It is either a **Mathematica** string of `"P"`, `"Psudoscalar"` or a length-2 list `{0,"-"}` that holds the spin and parity of the particle,
- `Ptot`: the center-of-mass momentum of the system,
- `Rep`: the lattice representation to generate in,
- `RepRow`: the row in the representation,
- `MaxMom`: maximum momentum of individual particles. For typical lattice QCD calculations, taking `RepRow` as large as 3 would be enough
- `MaxND`: the maximum number of covariant derivatives, used for one-hadron operator construction,
- `ParTuple`: tuple of `Par`, used for N-hadron operators

3. Tutorials

3.1 One-hadron operator

The function `OneHadronOperatorAll[Ptot, Rep, RepRow, MaxND]` searches all possible one-hadron operators for the number of covariant derivatives less than `MaxND` and returns the operator lists:

```
(* Particle settings *)
Ptot = {0, 0, 0};
Rep = "T1-";
RepRow = 3;
MaxND = 1;
(* Generate the one-hadron operators *)
Print /@ OneHadronOperatorAll[Ptot, Rep, RepRow, MaxND];
```

Output:

V_z

D_z

$A_y D_x - A_x D_y$

It is also possible to construct particular operators by specifying the helicity, various angular momentum, type of gamma matrices, and number of covariant derivatives by `OneHadronOperator[ptot, rep,r, J, lambda, JD, J13, TG, ND]`.

```
(* For particular operators (not all) *)
OneHadronOperator[Ptot, Rep, RepRow, 1, 0, 0, 0, Par2, 2]
```

Output:

$V_z(D_x^2+D_y^2+D_z^2)$

3.2 Two-hadron operator

There are projection methods and partial-wave methods for two-hadron operator constructions. The function `TwoHadronOperatorAll[Ptot, Rep, RepRow, MaxMom, Par1, Par2]` searches all possible two-hadron operators for the maximum momentum of individual particles less than $MaxMom$ and returns the operator lists:

```
(* Particle settings *)
Par1 = "P";
Par2 = "V";
Ptot = {0, 0, 0};
Rep = "T1−";
RepRow = 3;
MaxMom = 1;
(* Generate the two−hadron operators *)
Print /@ TwoHadronOperatorAll[Ptot, Rep, RepRow, MaxMom, Par1, Par2];
```

Output:

$-P_1[e_y]V_{2x}[-e_y]+P_1[-e_y]V_{2x}[e_y]+P_1[e_x]V_{2y}[-e_x]-P_1[-e_x]V_{2y}[e_x]$

To produce operators that are mostly L-wave, use

`TwoHadronOperatorPartialWaveAll[Ptot, Rep, RepRow, MaxMom, Par1, Par2,L]`,

```
(* Partial Wave *)
Print /@ TwoHadronOperatorPartialWaveAll[Ptot, Rep, RepRow, MaxMom, Par1, Par2, 1];
```

Output:

$-P_1[e_y]V_{2x}[-e_y]+P_1[-e_y]V_{2x}[e_y]+P_1[e_x]V_{2y}[-e_x]-P_1[-e_x]V_{2y}[e_x]$

The output is all P-wave operators with MaxMom = 1.

To construct particular operators by specifying relative momentum, the z-component of the angular momentum or the total spin angular momentum, use the following functions:

- `TwoHadronOperator[Ptot,Rep,RepRow,Mom,Par1,ms1,Par2,ms2]`
- `TwoHadronOperatorPartialWave[Ptot,Rep,RepRow,Mom,J,L,S,Par1,Par2]`

```
TwoHadronOperator[Ptot, Rep, RepRow, {1, 0, 0}, Par1, Par2, 0, 1]
TwoHadronOperatorPartialWave[Ptot, "T1", RepRow, {1, 0, 0}, 1, 1, 1, Par1, Par2]
```

The output is both:

$-P_1[e_y]V_{2x}[-e_y]+P_1[-e_y]V_{2x}[e_y]+P_1[e_x]V_{2y}[-e_x]-P_1[-e_x]V_{2y}[e_x]$

3.3 N-hadron operators

OpTion is capable of constructing operators from arbitrary numbers of particles. The argument `ParTuple` sets the number of particles. As an example, here shows $\pi\pi\pi$ operators subduced into T_1^+ in the rest frame:

```
Rep = "T1+";
ParTuple = {"P", "P", "P"};
Print /@ NHadronOperatorAll[Ptot, Rep, RepRow, MaxMom, ParTuple];
```

Output:

$P_1[e_z]P_2[0]P_3[-e_z]-P_1[-e_z]P_2[0]P_3[e_z]$

$P_1[e_z]P_2[-e_z]P_3[0]-P_1[-e_z]P_2[e_z]P_3[0]$

$P_1[0]P_2[e_z]P_3[-e_z]-P_1[0]P_2[-e_z]P_3[e_z]$

4. Acknowledging OpTion

Please cite

[1] Haobo Yan, Liuming Liu, Chuan Liu, and other authors, arXiv:2312.xxxx

Contact: haobo@stu.pku.edu.cn