YAML Configuration Files Additional Syntax

All YAML 1.2.2 constructs are supported. Below are the specific structures used in configuration files for defining operation constraints, profiles, and approximation parameters.

Key-Value Pairs

Key-value pairs are given as standard YAML mappings:
`key: <value>`.

Lists

Lists are given as standard YAML lists:
`[ <item1>, <item2>, ... ]`.

Tuples

A tuple is given as a string bounded by parenthesis wrapped colons:

- `"(: <first>, <second> :)"`.
  - In a list of tuples, all tuples must be of the same length.

Intervals

An interval is given as a string bounded by parenthesis wrapped bracket:

- `"([ <lowerbound>, <upperbound> ])"`.
  - The brackets distinguish intervals from tuples.

Expressions

An expression is given as a string bounded by parenthesis wrapped parenthesis:

- `"(( <expression> ))"`
- a list is given: `"(( [ <item1>, <item2>, ... ] ))"`.
- a tuple is given: `"(( (: <first>, <second>, ... :) ))"`.
- an interval is given: `"(( ([ <lowerbound>, <upperbound> ]) ))"`.

Iterators

In an expression, string interpolating iterators may be used to generate lists.

String interpolation is done with `"prefix${i}suffix"`, where 'i' is replaced:

```
"(( [ Binary4p{i}sf for i in [1..3] ] ))"
```

↦

```
[ Binary4p1sf, Binary4p2sf, Binary4p3sf ]
```

Multiple substitutions may be done within a single item

```
"(( [ (Binary4p${i}s${j} for i in [1..2] for j in ["f", "e"]) ] ))"
```

↦

```
[ Binary4p1sf, Binary4p2sf, Binary4p1se, Binary4p2se ]
```

Two generated lists are concatenated when they are wrapped in parenthesis

```
"(( ([ Binary6p${i}sf for i in [1, 3, 4] ], [ Binary8p${k}se for k in [2..4] ]) ))"
```

↦

```
[ Binary6p1sf, Binary6p3sf, Binary6p4sf, Binary8p2se, Binary8p3se, Binary8p4se ]
```

.