The function `cubequad` takes a quadrature rule on the unit cube and returns either a quadrature value for one of the predefined integrals, a quadrature rule suitable for the integration of functions with a singularity on the unit cube or on a physical parallelotope [1].

1 Input

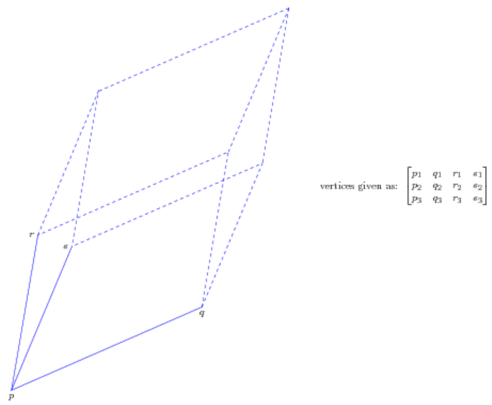

Figure 1: Example for giving the defining vertices of a parallelotope.

- `-d`, `--dimension=DIM`
  This argument gives the spatial dimension in which the integral should be evaluated. If `--vertexlist` is given it is not necessary to give this argument.

- `-k`, `--intersection=DIM`
  This argument gives the dimension of the intersection of the two parallelotopes. As in the case of `--dimension` this argument can be calculated if `--vertexlist` is given and doesn't have to be entered if `--vertexlist` is.

- `-f`, `--function=NUMBER`
  Integrate one of the predefined functions. Possible choices are:

1	$\|z\|^{\alpha}$,	where $\alpha = -2d + k + \frac{1}{\pi}$	
2	$\|y - x\|^{\alpha}$,	where $\alpha = -2d + k + \frac{1}{\pi}$	Note that this is the same as `1` except for the method of calculating z. This method leads to more subtractive cancellation [2, Section 6.2].
3	$\|z\|^{\alpha}$,	where $\alpha = -\frac{1}{2}$	
4	$\|y - x\|^{\alpha}$,	where $\alpha = -\frac{1}{2}$	Note that this is the same as `3` except for the method of calculating z.
5	$\|z\|^{\alpha}$,	where $\alpha = -d\frac{1}{\pi}$	
7	$\left(\sum_{i=1}^{d} v_i^2 + (1+u_i)^2\right)^{\alpha/2}$,	where $\alpha = -2d + k + \frac{1}{\pi}$	
11	$\sin(\hat{u})\cos(\hat{z})\sin(\check{u})\sin(\check{v})\|z\|^{\alpha}$,	where $\alpha = -2d + k + \frac{1}{\pi}$	

- `-l`, `--vertexlist=FILE`
  This argument gives the filename of the file containing the list of vertices of the physical parallelotopes. If this argument is not given `cubequad` returns a quadrature rule on the unit cube which must still be transformed to the physical coordinates. Some predefined vertexlists can be found in the folder `vertexlists/`.
  The file must contain the arguments in the following order:
  First an integer giving the size of the vertexlist in the first coordinate, ie. spatial dimension and then an integer giving the size in the second coordinate, ie. $2d - k$. Then the k common vertices of the two parallelotopes are given, then the remaining $d - k$ defining vertices of the first parallelotope and then the remaining $d - k$ defining vertices of the second parallelotope. (See Figure 1 for an example)

- `-m`, `--combination-method=METHOD`
  Argument giving the combination method for producing high-dimensional quadrature in regular coordi-

nates. METHOD must be sparse, full-tensor, or sobol. If this argument is not given full-tensor product quadrature is used.

- `-n`, `--n-regular=NUMBER`
  Number of quadrature points in each regular coordinate or in the highest level of the sparse-grid quadrature. This argument must be entered unless a file containing the quadrature points for the regular direction is given.

- The following flags can be given if only certain parts of the quadrature rule are needed. If a predefined function is used (i.e., `--function` is set) these are calculated appropriety do not need to be given.
  `--no-x` Don't calculate x-part of the quadrature
  `--no-y` Don't calculate y-part of the quadrature
  `--no-z` Don't calculate z-part of the quadrature

- `-o`, `--output=FILE`
  This argument gives the filename into which the output quadrature rule will be written. If this argument is not given, a filename is generated. If `--function` is set this is not necessary since the quadrature value will be written directly to the console. Note that the quadrature rules will be on the unit cube unless `--vertexlist` is set.

- `-p`, `--regular-sparse-p=NUMBER`
  Parameter used for sparse grid quadrature, it is only needed if `--regular=sparse`. If this argument is not given it is set to 1, giving standard sparse grid quadrature.

- `-r`, `--regular=NAME`
  One-dimensional quadrature rule for regular coordinates. If `--combination-method=sobol` this argument is not needed.
  NAME must be gauss-legendre, kronrod-patterson, or clenshaw-curtis or the path to a file containing a one-dimensional quadrature rule.

- `--quadrature-rule-regular=FILE`
  Read the entire $2d - 1$ dimensional regular quadrature rule from FILE (Don't specify with `--regular` or `--combination-method`).

- `-s`, `--singular=NAME`
  One-dimensional quadrature rule for singular coordinate. This argument should only be given if a predefined quadrature rule should be used for the singular coordinate.
  NAME must be composite-gauss-legendre or gauss-jacobi.

- `--quadrature-rule-singular=FILE`
  Read singular quadrature rule from FILE (Don't specify with `--singular`).

- `--piecewise`
  This flag calculates portions of the sparsegrid quadrature and evaluates them piece by piece. This leads to more quadrature points since several points may coincide, but get evaluated several times. However, one can calculate with more points in the maximal level. It can only be used with `--combination-method=sparse` and `--whichF`.

- `-?`, `--help` Give help list.

- `--usage` Give a short usage message.

2 Examples

2.1 Integrate one of the predefined functions

```
./cubequad --function=1 --vertexlist=vertexlists/33.dat --n-regular=3 --regular=gauss-legendre \
          --singular=gauss-jacobi --combination-method=full-tensor
```

This example returns the integral value, number of degrees of freedom needed to calculate it and the time taken to calculate it. The vertexlist `33.dat` ist used, this file contains the vertices of the unit cube in 3 dimensions. The function integrated is 1, so the integral solved is

$$\int_{x\in[0,1]^3}\int_{y\in[0,1]^3}\|y-x\|^\alpha dydx.$$

The rest of the arguments give the types of the quadrature rules to use. In this example Gauss-Legendre quadrature with 3 points is tensorised for the regular direction and Gauss- Jacobi is used in the singular direction.

2.2 Get a quadrature rule on the unit cube

```
./cubequad --dimension=2 --intersection=0 --n-regular=4 --singular=composite-gauss-legendre \
           --combination-method=sobol --output=example1.dat
```

This example returns a quadrature rule on the unit cube suitable for integrating functions with a singularity. The quadrature rule still needs to be transferred to a physical parallelotope.

`--output=example1.dat` specifies the filename the resultant quadrature rule is written into.

The rest of the arguments give the types of the quadrature rules to use. In this example Sobol quadrature is used in the regular direction and composite-Gauss-Legendre is used in the singular direction.

2.3 Get a quadrature rule on the physical parallelotopes

```
./cubequad --vertexlist=vertexlists/29.dat --n-regular=3 --regular=kronrod-patterson \
           --singular=gauss-jacobi --combination-method=sparse --regular-sparse-p=-1 \
           --output=example2.dat
```

This example returns a quadrature rule suitable for integrating functions with a singularity. The vertexlist `29.dat` ist used, this file contains the vertices of two disjoint cubes in 3 dimensions.

`--output=example2.dat` specifies the filename the resultant quadrature rule is written into.

The rest of the arguments give the types of the quadrature rules to use. In this example sparsegrid Kronrod-Patterson is used in the regular direction and Gauss-Jacobi is used in the singular direction.

References

[1] A. Chernov and A. Reinarz. Numerical quadrature for high-dimensional singular integrals over parallelotopes. *Computers & Mathematics with Applications*, 66(7):1213 – 1231, 2013.

[2] Alexey Chernov, Tobias von Petersdorff, and Christoph Schwab. Exponential convergence of hp quadrature for integral operators with Gevrey kernels. *M2AN*, 45(3):387–422, 2011.