

Compressible Flow Calculator

REST API for calculation of compressible flow relations. Includes Isentropic Flows, Normal Shocks, Oblique Shocks, Fanno Flows, and Rayleigh Flows.

More information: <http://www.sierrao.com>

Contact Info: sierrao@utexas.edu

Version: 1.0.0

BasePath: /

MIT

<https://github.com/SierraOG/CompressibleFlowCalculator/blob/master/LICENSE>

Access

Methods

[Jump to [Models](#)]

Table of Contents

[FannoController](#)

- [GET /fanno/{gamma}/{inputType}/{inputValue}](#).

[IsentropicController](#)

- [GET /isentropic/{gamma}/{inputType}/{inputValue}](#).

[NormalShockController](#)

- [GET /normalshock/{gamma}/{inputType}/{inputValue}](#).

[ObliqueShockController](#)

- [GET /oblique/{gamma}/{mach}/{secondInputType}/{secondInputValue}](#).

[RayleighController](#)

- [GET /rayleigh/{gamma}/{inputType}/{inputValue}](#).

FannoController

GET /fanno/{gamma}/{inputType}/{inputValue}

[Up](#)

Returns Fanno flow properties for given input. ([getFannoFlowUsingGET](#))

Path parameters

gamma (required)

Path Parameter — The value of gamma, the ratio of specific heats. For air, gamma = 1.4 at standard conditions. format: double

inputType (required)

Path Parameter — Describes the value passed to base the calculations on. Currently only an inputType of 'mach' is supported for Fanno Flow.

inputValue (required)

Path Parameter — The actual value passed to base the calculations on. Currently can only be the mach number for Fanno Flow. format: double

Return type[Fanno](#)**Example data**

Content-Type: application/json

```
{
  "popostar" : 5.962133916683182,
  "ttstar" : 7.061401241503109,
  "mach" : 1.4658129805029452,
  "flstar" : 0.8008281904610115,
  "ppstar" : 5.637376656633329,
  "gamma" : 6.027456183070403,
  "sstarsr" : 2.3021358869347655,
  "uustar" : 9.301444243932576
}
```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200

OK [Fanno](#)

401

Unauthorized

403

Forbidden

404

Not Found

IsentropicController

[Up](#)

GET /isentropic/{gamma}/{inputType}/{inputValue}

Returns Isentropic flow properties for given input. ([findIsentropicFlowUsingGET](#))**Path parameters****gamma (required)**

Path Parameter — The value of gamma, the ratio of specific heats. For air, gamma = 1.4 at standard conditions. format: double

inputType (required)

Path Parameter — Describes the inputValue passed to base the calculations on. Can be 'mach' for Mach number, 'temp' for T/T0 temperature ratio, 'pres' for P/P0 pressure ratio, 'rho' for rho/rho0 density ratio, 'areasub' for A/A* area ratio with the subsonic solution, 'areasuper' for A/A* area ratio with supersonic solution, 'pmangle' for Prandtl-Meyer angle in degrees, or 'machangle' for the Mach angle in degrees

inputValue (required)

Path Parameter — The corresponding value to do calculations on format: double

Return type[Isentropic](#)**Example data**

Content-Type: application/json

```
{
  "rhorhoo" : 9.301444243932576,
  "aastar" : 0.8008281904610115,
  "pmangle" : 5.637376656633329,
  "rhorhostar" : 3.616076749251911,
  "ttstar" : 4.145608029883936,
  "mach" : 1.4658129805029452,
  "machangle" : 5.962133916683182,
  "ppstar" : 7.061401241503109,
  "gamma" : 6.027456183070403,
  "ppo" : 2.3021358869347655,
  "tto" : 2.027123023002322
}
```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200

OK [Isentropic](#)

401

Unauthorized

403

Forbidden

404

Not Found

NormalShockController

GET /normalshock/{gamma}/{inputType}/{inputValue}

[Up](#)

Returns Normal Shock flow properties for given input. ([findNormalShockUsingGET](#))

Path parameters

gamma (required)

Path Parameter — The value of gamma, the ratio of specific heats. For air, gamma = 1.4 at standard conditions. format: double

inputType (required)

Path Parameter — Describes the inputValue passed to base the calculations on. Can be 'mach' for incoming Mach number before the shock wave, 'mach2' for Mach number after the shock wave, 'temp' for T2/T1 temperature ratio, 'pres' for P2/P1 pressure ratio, 'rho' for rho2/rho1 density ratio, 'stagpres' for P02/P01 stagnation pressure ratio, 'presstagpres' for P1/P02 pressure to stagnation pressure ratio

inputValue (required)

Path Parameter — The corresponding value to do calculations on format: double

Return type

[NormalShock](#)

Example data

Content-Type: application/json

```
{
  "p1po2" : 5.962133916683182,
  "p2p1" : 5.637376656633329,
```

```

"mach2" : 1.4658129805029452,
"mach" : 6.027456183070403,
"po2po1" : 2.3021358869347655,
"rho2rho1" : 7.061401241503109,
"gamma" : 0.8008281904610115,
"t2t1" : 9.301444243932576
}

```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200

OK [NormalShock](#)

401

Unauthorized

403

Forbidden

404

Not Found

ObliqueShockController

GET /oblique/{gamma}/{mach}/{secondInputType}/{secondInputValue} [Up](#)

Returns Oblique Shock flow properties for given inputs. ([findObliqueFlowUsingGET](#))

Path parameters

gamma (required)

Path Parameter — The value of gamma, the ratio of specific heats. For air, gamma = 1.4 at standard conditions. format: double

mach (required)

Path Parameter — The incoming Mach number before the shock. format: double

secondInputType (required)

Path Parameter — Describes the secondInputValue passed to base the calculations on. Can be 'mach1n' for the normal component for the incoming Mach number, 'beta' for the shock wave angle beta in degrees, 'thetaweak' for the turn angle theta in degrees assuming a weak shock solution, or 'thetastrong' for the turn angle theta in degrees assuming a strong shock solution

secondInputValue (required)

Path Parameter — The corresponding input value. format: double

Return type

[ObliqueShock](#)

Example data

Content-Type: application/json

```

{
  "p2p1" : 7.061401241503109,
  "mach2" : 2.3021358869347655,
  "m2n" : 5.962133916683182,
  "mach" : 5.637376656633329,
  "m1n" : 1.4658129805029452,
  "po2po1" : 9.301444243932576,
}

```

```

"rho2rho1" : 3.616076749251911,
"theta" : 4.145608029883936,
"beta" : 0.8008281904610115,
"gamma" : 6.027456183070403,
"t2t1" : 2.027123023002322
}

```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200

OK [ObliqueShock](#)

401

Unauthorized

403

Forbidden

404

Not Found

RayleighController

[Up](#)

GET /rayleigh/{gamma}/{inputType}/{inputValue}

returns Rayleigh flow properties for given input. ([findRayleighFlowUsingGET](#))

Path parameters

gamma (required)

Path Parameter — The value of gamma, the ratio of specific heats. For air, gamma = 1.4 at standard conditions. format: double

inputType (required)

Path Parameter — Describes the value passed to base the calculations on. Currently only an inputType of 'mach' is supported for Rayleigh Flow.

inputValue (required)

Path Parameter — The actual value passed to base the calculations on. Currently can only be the mach number for Rayleigh Flow. format: double

Return type

[Rayleigh](#)

Example data

Content-Type: application/json

```

{
  "popostar" : 1.4658129805029452,
  "totostar" : 2.3021358869347655,
  "ttstar" : 7.061401241503109,
  "mach" : 6.027456183070403,
  "ppstar" : 5.962133916683182,
  "gamma" : 0.8008281904610115,
  "sstarsr" : 5.637376656633329,
  "uustar" : 9.301444243932576
}

```

Produces

This API call produces the following media types according to the Accept request header; the media type will be conveyed by the Content-Type response header.

- application/json

Responses

200

OK [Rayleigh](#)

401

Unauthorized

403

Forbidden

404

Not Found

Models

[[Jump to Methods](#)]

Table of Contents

1. [Fanno - Fanno](#)
2. [Isentropic - Isentropic](#)
3. [NormalShock - NormalShock](#)
4. [ObliqueShock - ObliqueShock](#)
5. [Rayleigh - Rayleigh](#)

Fanno - Fanno

[Up](#)

flstard (optional)

[Double](#) format: double

gamma (optional)

[Double](#) format: double

mach (optional)

[Double](#) format: double

popostar (optional)

[Double](#) format: double

ppstar (optional)

[Double](#) format: double

sstarsr (optional)

[Double](#) format: double

ttstar (optional)

[Double](#) format: double

uustar (optional)

[Double](#) format: double

Isentropic - Isentropic

[Up](#)

aastar (optional)

[Double](#) format: double

gamma (optional)

[Double](#) format: double

mach (optional)

[Double](#) format: double

machangle (optional)

[Double](#) format: double

pmangle (optional)

[Double](#) format: double

ppo (optional)

[Double](#) format: double

ppstar (optional)

[Double](#) format: double

rhorrho (optional)

[Double](#) format: double

rhorrhostar (optional)

[Double](#) format: double

tto (optional)

[Double](#) format: double

ttstar (optional)

[Double](#) format: double

NormalShock - NormalShock

[Up](#)

gamma (optional)

[Double](#) format: double

mach (optional)

[Double](#) format: double

mach2 (optional)

[Double](#) format: double

p1po2 (optional)

[Double](#) format: double

p2p1 (optional)

[Double](#) format: double

po2po1 (optional)

[Double](#) format: double

rho2rho1 (optional)

[Double](#) format: double

t2t1 (optional)

[Double](#) format: double

ObliqueShock - ObliqueShock

[Up](#)

beta (optional)

[Double](#) format: double

gamma (optional)

[Double](#) format: double

m1n (optional)

[Double](#) format: double

m2n (optional)

[Double](#) format: double

mach (optional)

[Double](#) format: double

mach2 (optional)

[Double](#) format: double

p2p1 (optional)
[Double](#) format: double

po2po1 (optional)
[Double](#) format: double

rho2rho1 (optional)
[Double](#) format: double

t2t1 (optional)
[Double](#) format: double

theta (optional)
[Double](#) format: double

Rayleigh - Rayleigh

[Up](#)

gamma (optional)
[Double](#) format: double

mach (optional)
[Double](#) format: double

popstar (optional)
[Double](#) format: double

ppstar (optional)
[Double](#) format: double

sstarsr (optional)
[Double](#) format: double

totostar (optional)
[Double](#) format: double

ttstar (optional)
[Double](#) format: double

uustar (optional)
[Double](#) format: double