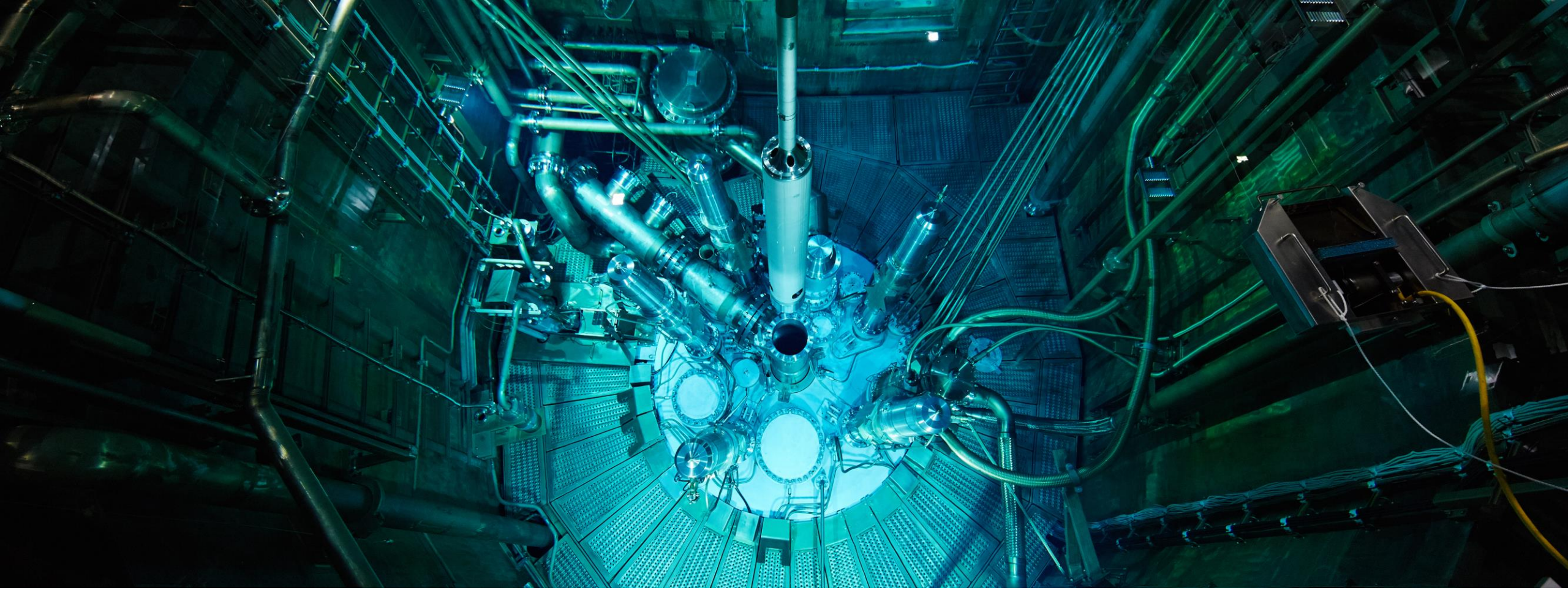


IMPLEMENTATION OF INVOLUTES IN SERPENT

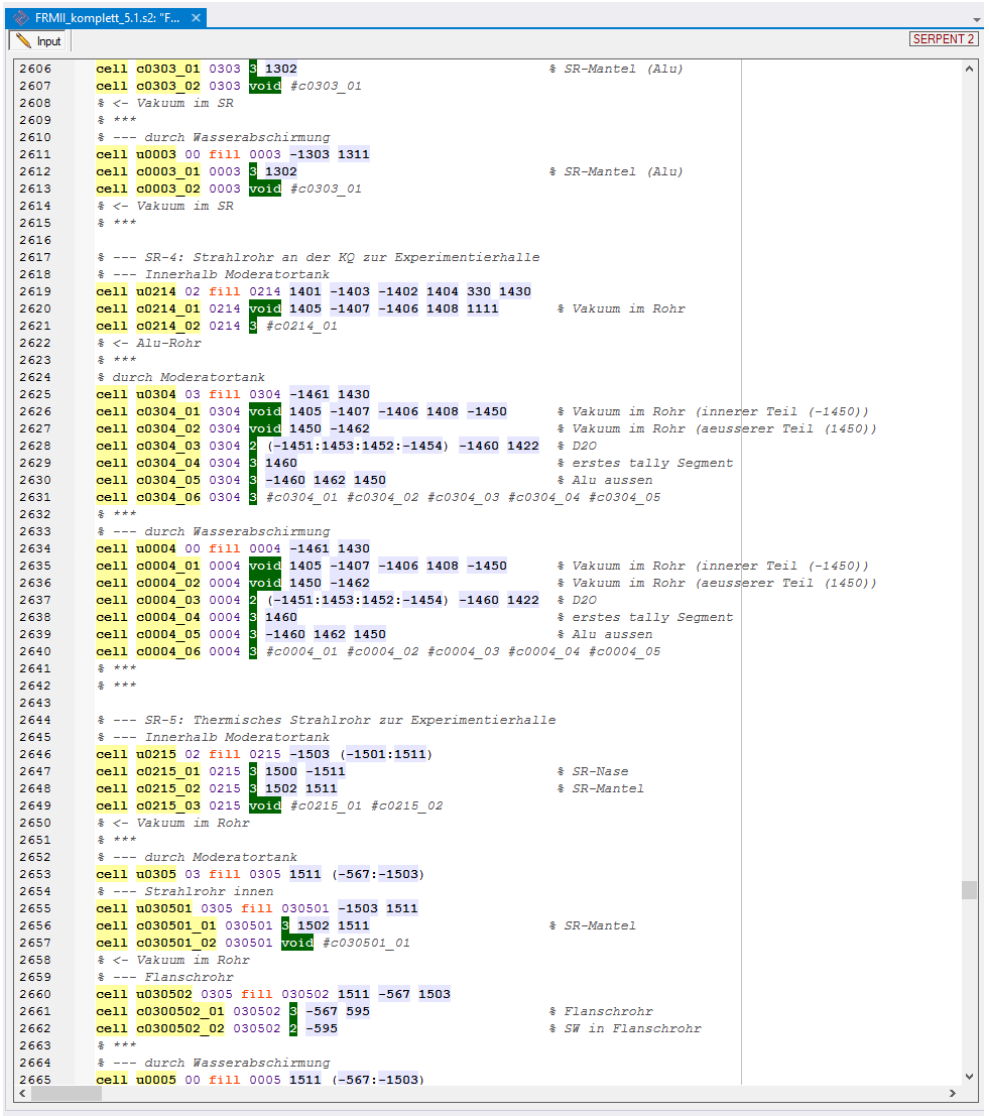
DR. CHRISTIAN REITER

Serpent User Group Meeting
October 28th 2020
TUM

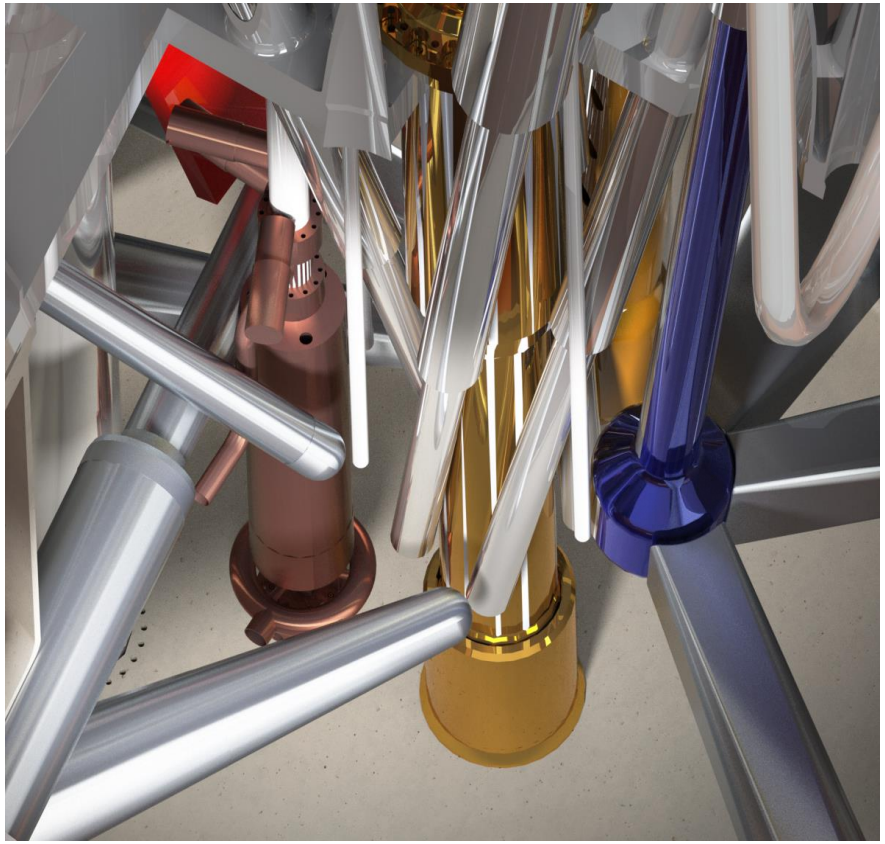
CORE DESIGN

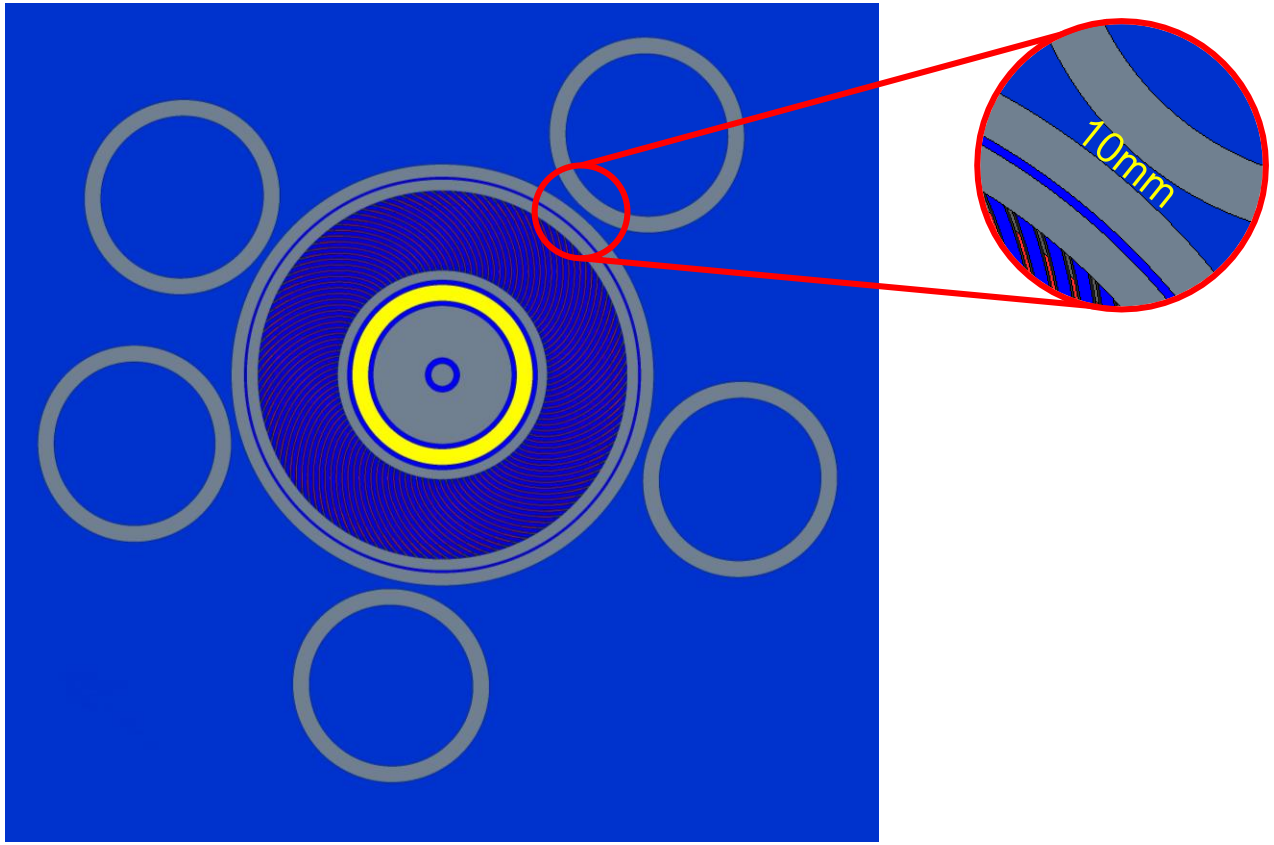
- Coupled calculations
- Codes have to deal with constraint conditions
  - Very compact core
  - Complex geometry
  - High flux gradients
  - Tight cooling channels
  - High aspect ratio
- Steady state & transient study
- “Ready for licensing”

TUM is looking for a compatible core with the lowest enrichment, as technically and economically feasible, with the condition of only marginal losses of scientific performance



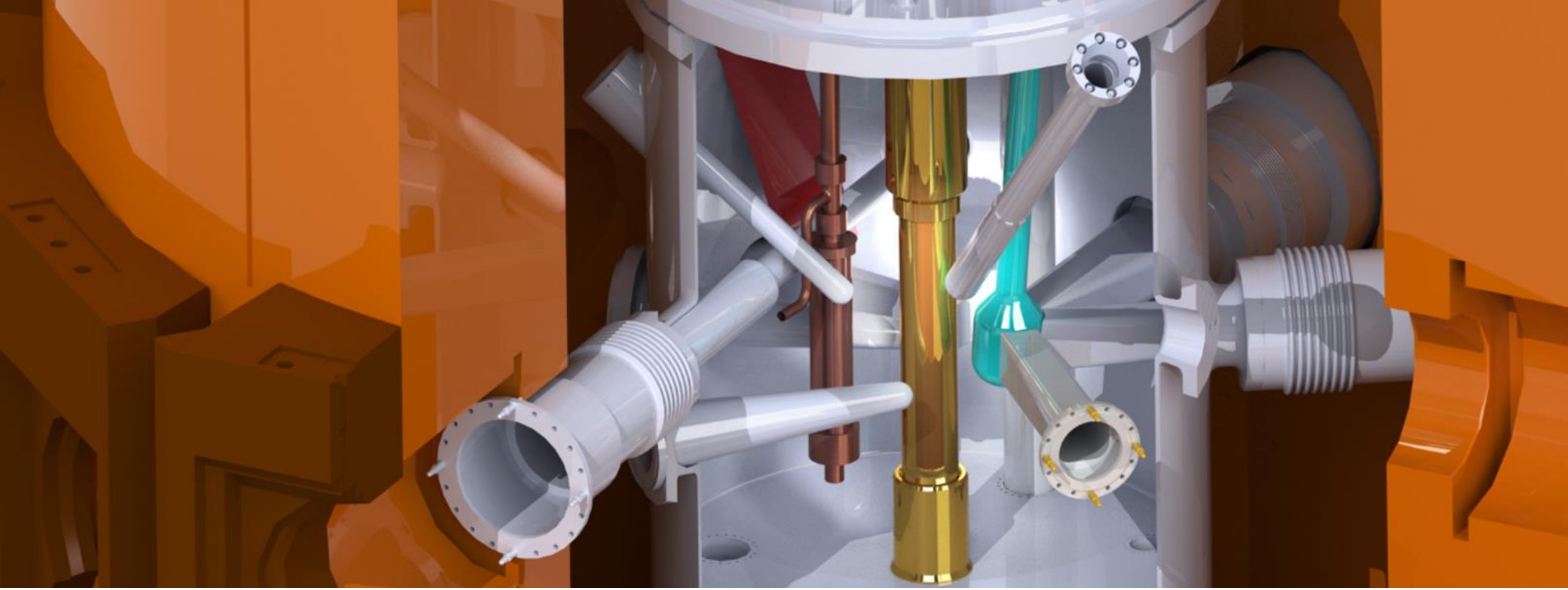
NO SPACE LEFT





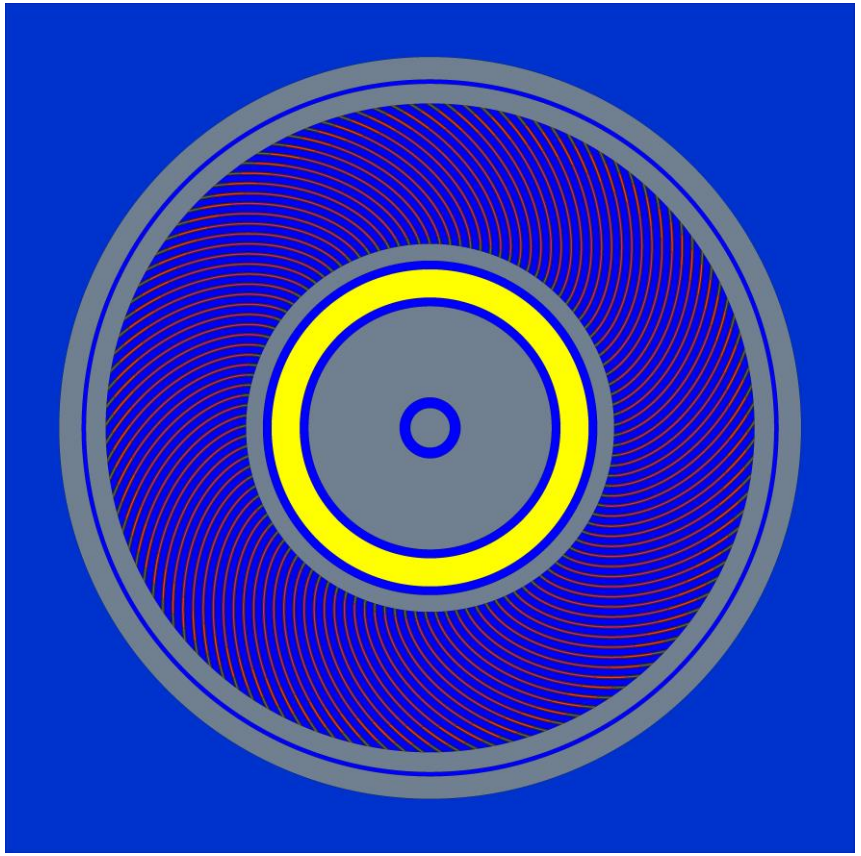
POSSIBLE PARAMETERS

▪ Three fuel candidates: U_3Si_2, disperse and monolithic U-Mo ▪ Meat & cladding thickness ▪ Molybdenum content ▪ Number of plates ▪ Density jump and/or burnable poison	Fuel element
▪ Central channel tube: diameter and material ▪ Control rod tube: diameter and material	Reactor



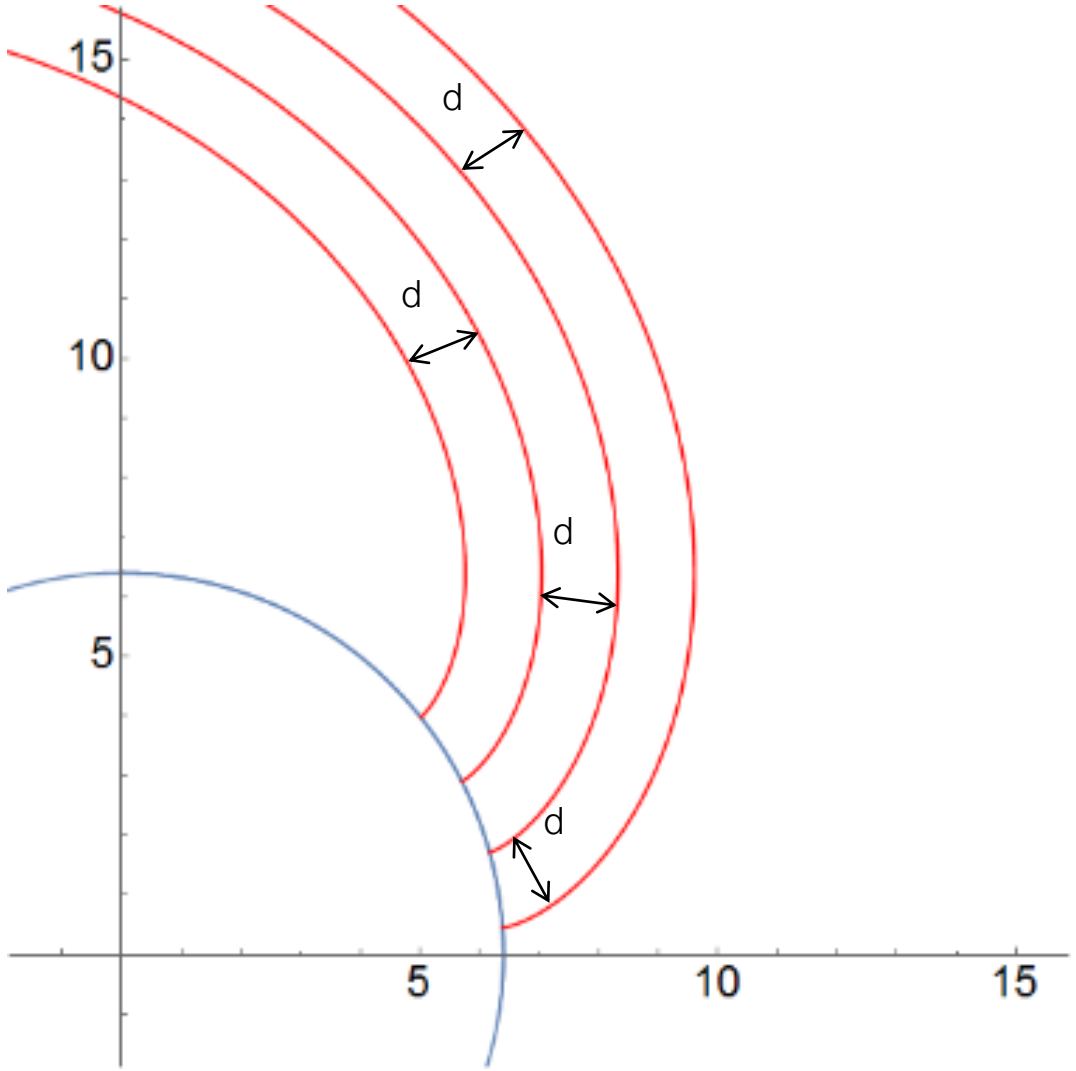
INVOLUTES

CURRENT FUEL ELEMENT



WHY INVOLUTES?

Constant cooling channel thickness solves the challenging cooling of high-flux-neutron-reactor FRM II



MATHEMATICAL BACKGROUND

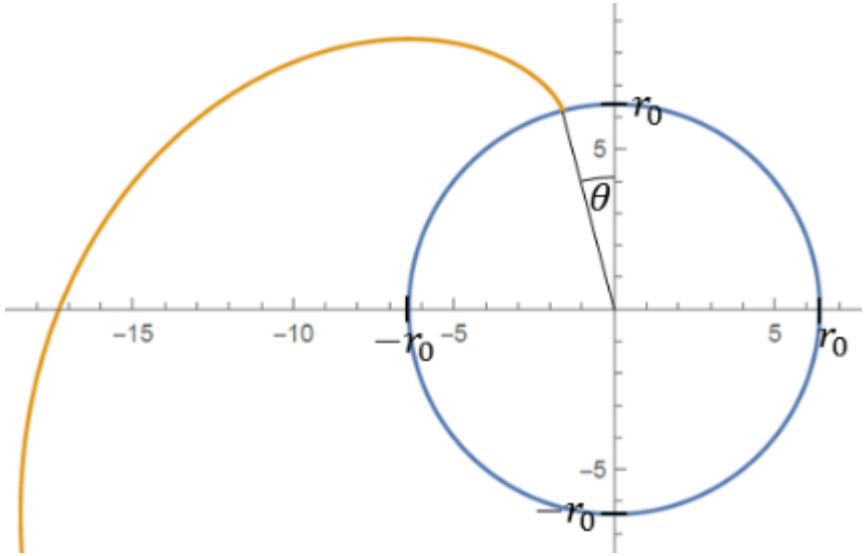
$$\begin{pmatrix} x(r_0,\theta,\varphi) \\ y(r_0,\theta,\varphi) \end{pmatrix} = r_0 \begin{pmatrix} \varphi \cdot \cos(\varphi+\theta) - \sin(\varphi+\theta) \\ \cos(\varphi+\theta) + \varphi \cdot \sin(\varphi+\theta) \end{pmatrix}$$

Defining Parameter:

$$r_0 \in [0,\infty[$$

$$\theta \in [0,2\pi]$$

Parameter: $\varphi \in [0,\infty[$



DEFINITION OF SURFACE-TYPES

Surface-type (Parameters)

1. Testing if point (x, y, z) is inside or outside (left or right) the surface

2. Calculating the shortest distance to the surface from point (x, y, z) in direction (u, v, w)

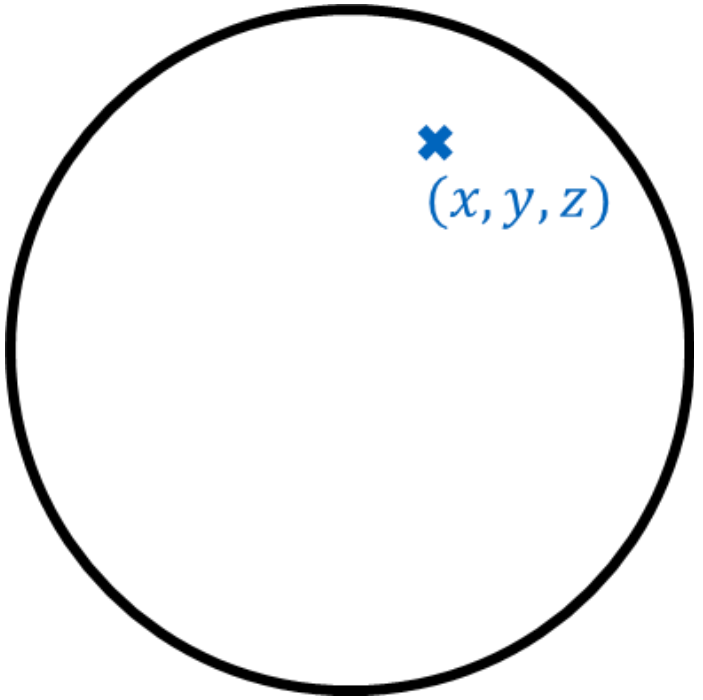
DEFINITION OF SURFACE-TYPES

Surface-type (Parameters)

1. Testing if point (x, y, z) is inside or outside (left or right) the surface

2. Calculating the shortest distance to the surface from point (x, y, z) in direction (u, v, w)

Example: Sphere



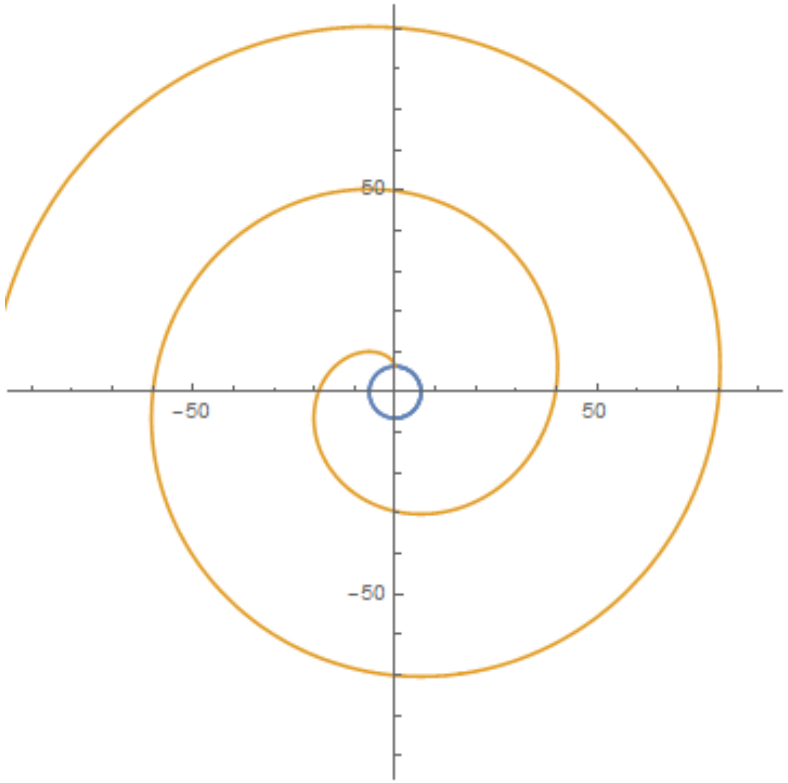
DEFINITION OF SURFACE-TYPES

Surface-type (Parameters)

1. Testing if point (x, y, z) is inside or outside (left or right) the surface

2. Calculating the shortest distance to the surface from point (x, y, z) in direction (u, v, w)

Involute



DEFINITION AS MACROBODY

In x-y-plane defined by parameters:

r_0: involute radius

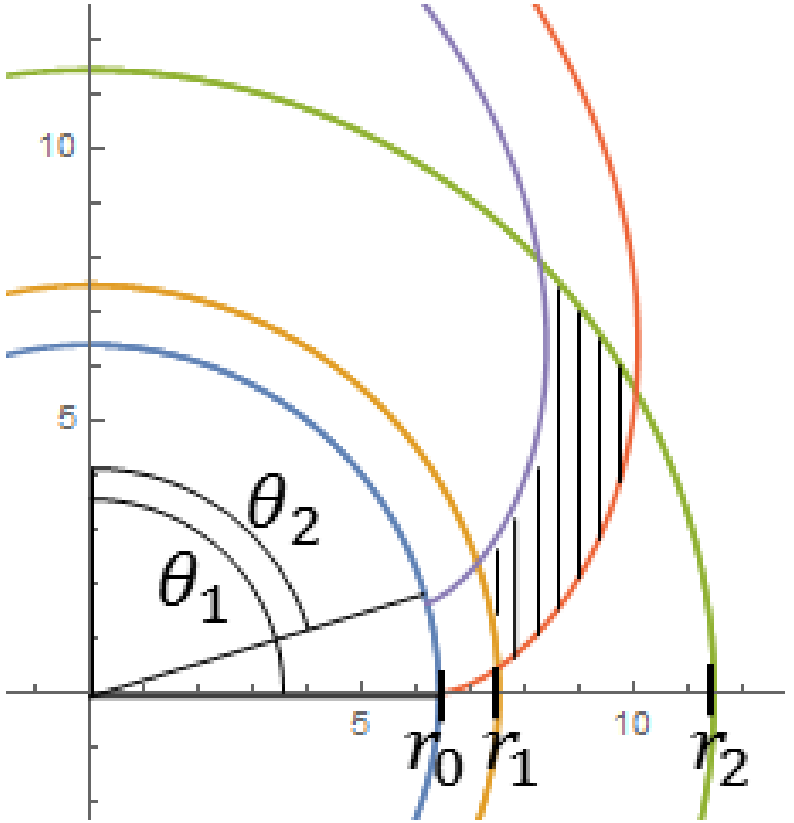
θ_1: starting angle of first involute

θ_2: starting angle of second involute

r_1: radius of inner cylinder

r_2: radius of outer cylinder

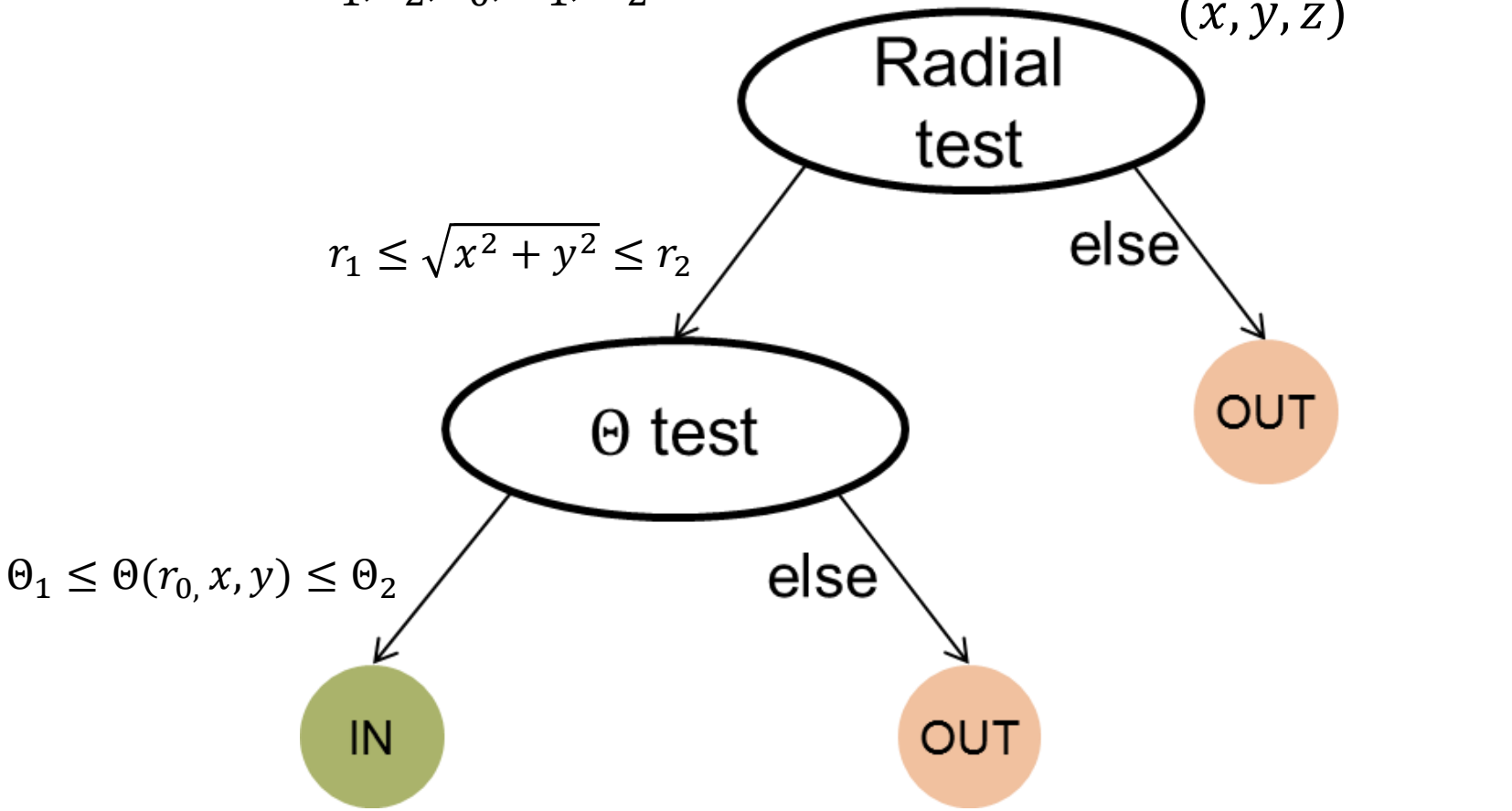
Body infinite extended in z-direction



TEST FOR “INSIDE” AND “OUTSIDE”

Involute with Parameters
$r_1, r_2, r_0, \Theta_1, \Theta_2$

Arbitrary Point
(x, y, z)



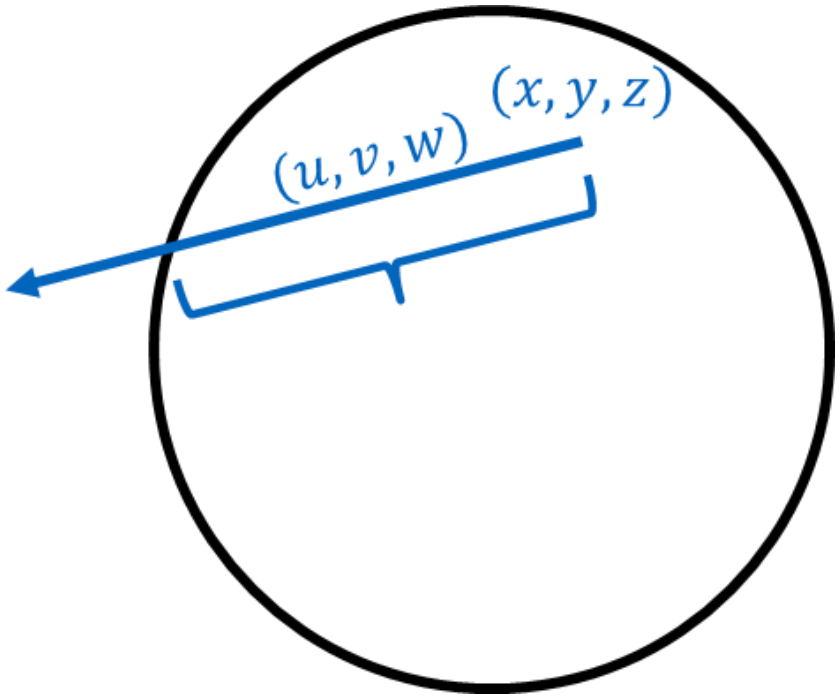
DEFINITION OF SURFACE-TYPES

Surface-type (Parameters)

1. Testing if point (x, y, z) is inside or outside (left or right) the surface

2. Calculating the shortest distance to the surface from point (x, y, z) in direction (u, v, w)

Example: Sphere



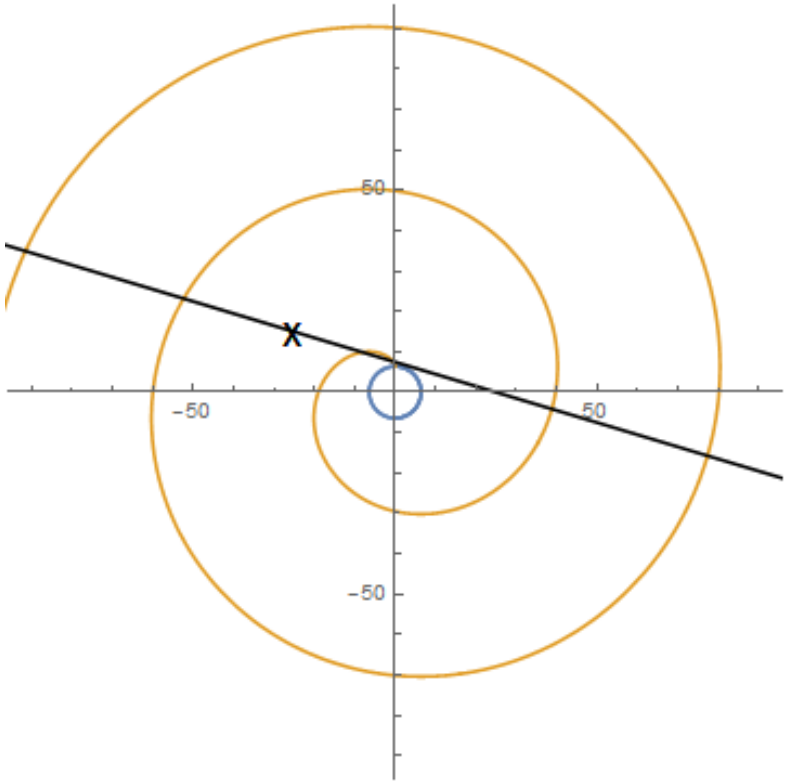
DEFINITION OF SURFACE-TYPES

Surface-type (Parameters)

1. Testing if point (x, y, z) is inside or outside (left or right) the surface

2. Calculating the shortest distance to the surface from point (x, y, z) in direction (u, v, w)

Example: Involute



NEED FOR NUMERICS

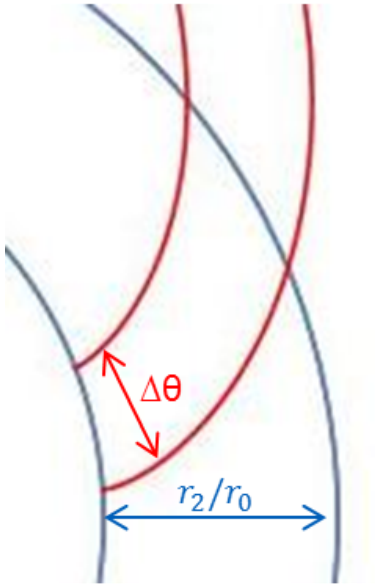
Intersection point of an involute with a line is not analytically solvable:

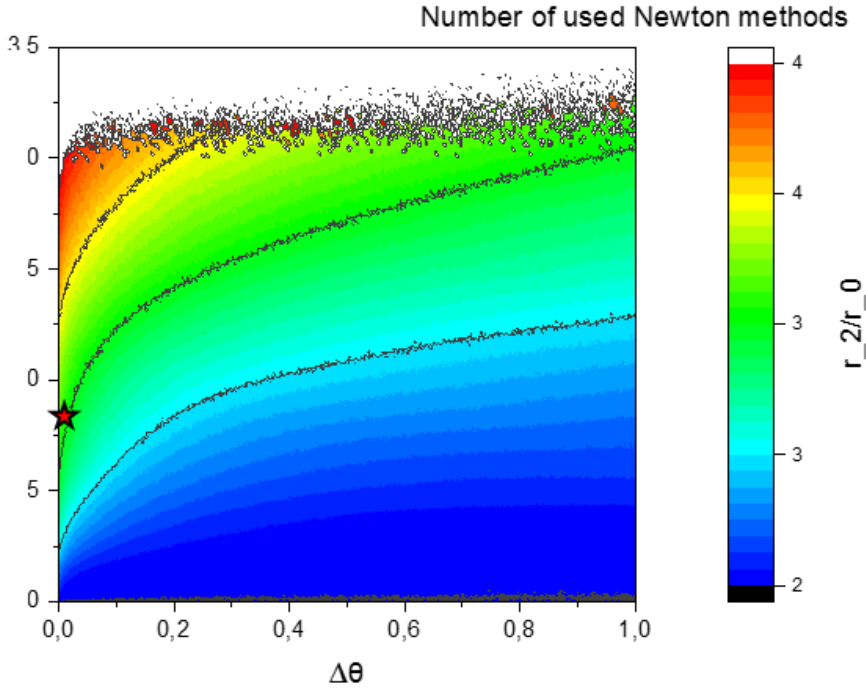
$$f(\varphi) = \{r \cdot [\varphi \cdot \cos(\varphi + \theta) - \sin(\varphi + \theta)] - x_0\} \cdot dy$$
$$+\{(-r) \cdot [\cos(\varphi + \theta) + \varphi \cdot \sin(\varphi + \theta)] + y_0\} \cdot dx = 0$$

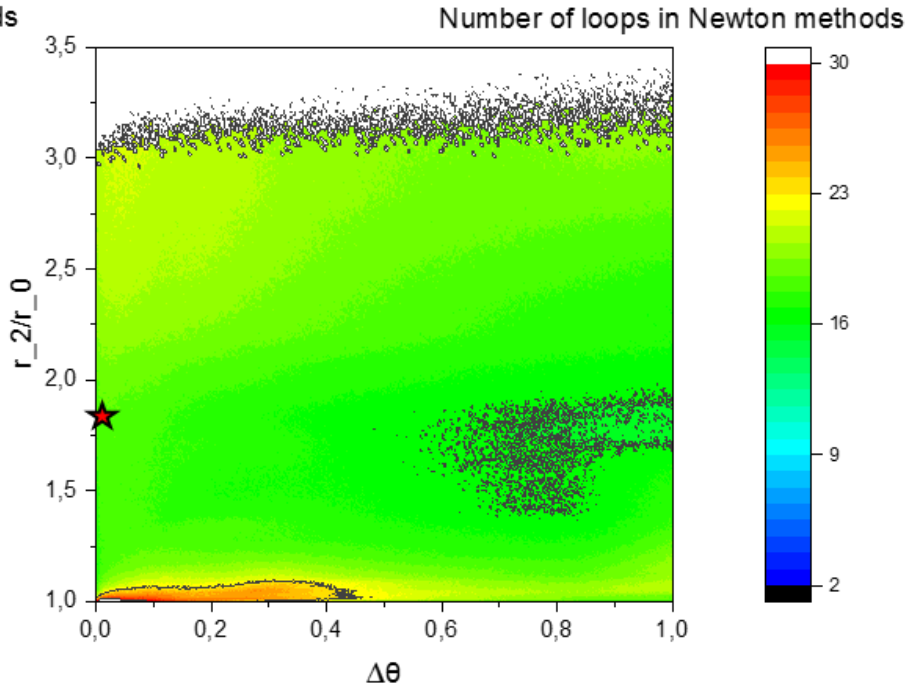
First version: Newton's method

Second version: Regular Falsi

CONVERGENCE OF NEWTONS METHOD

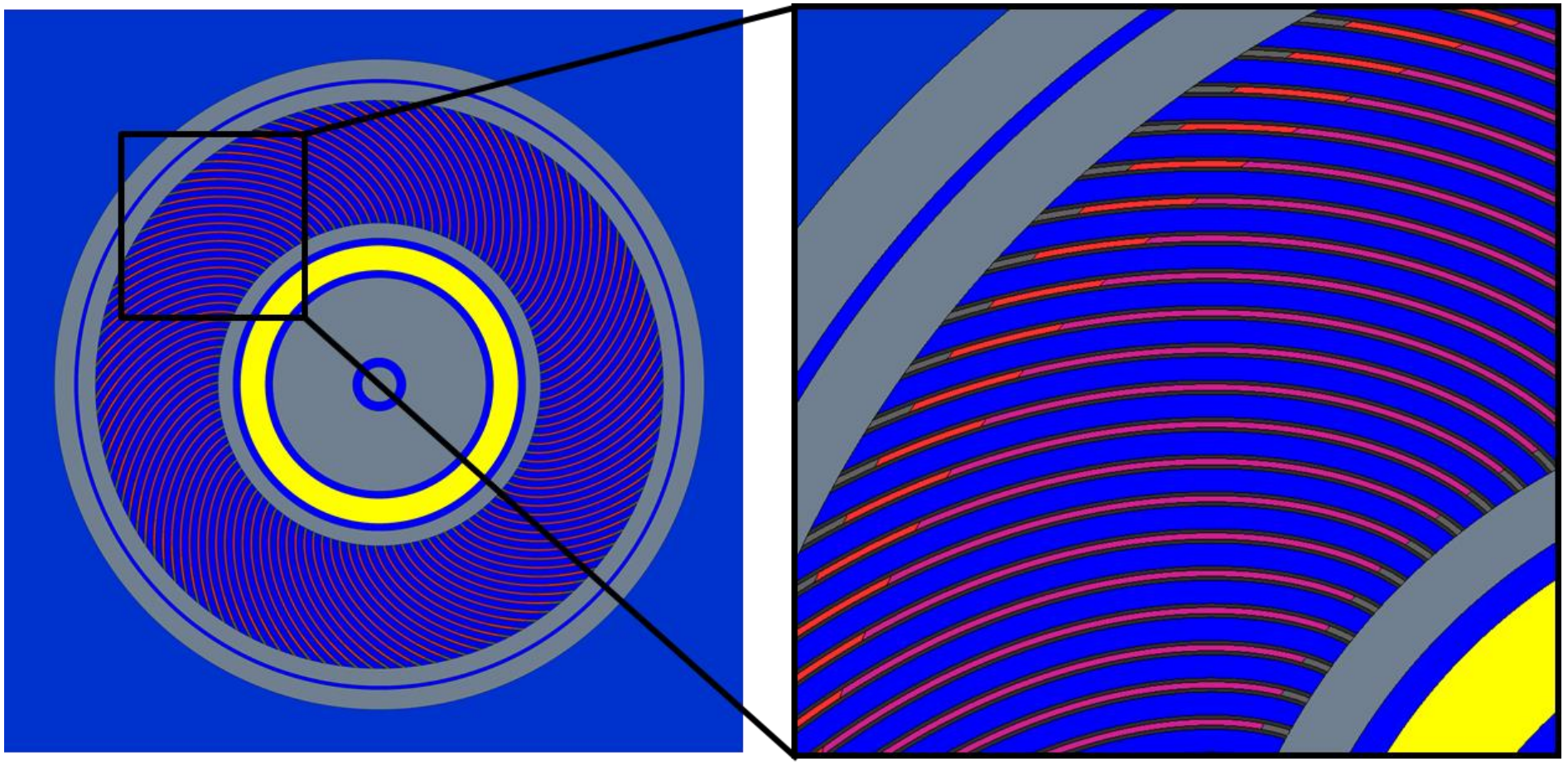




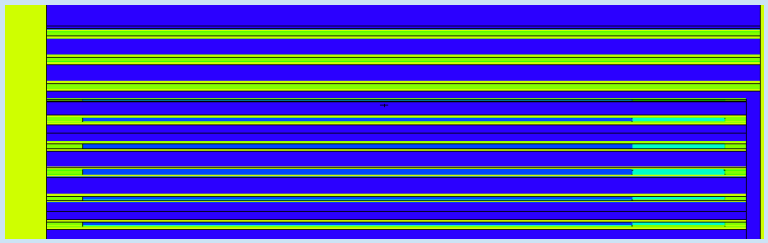
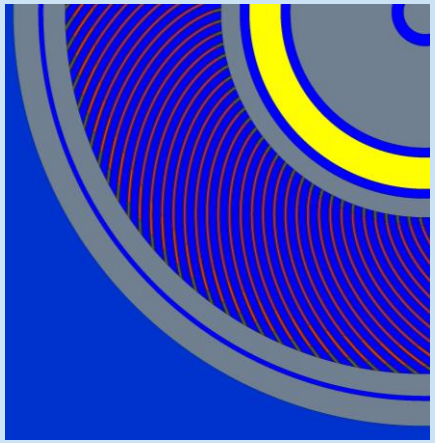
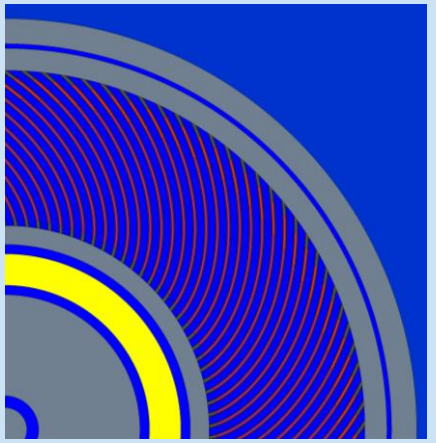


★ = FRM II

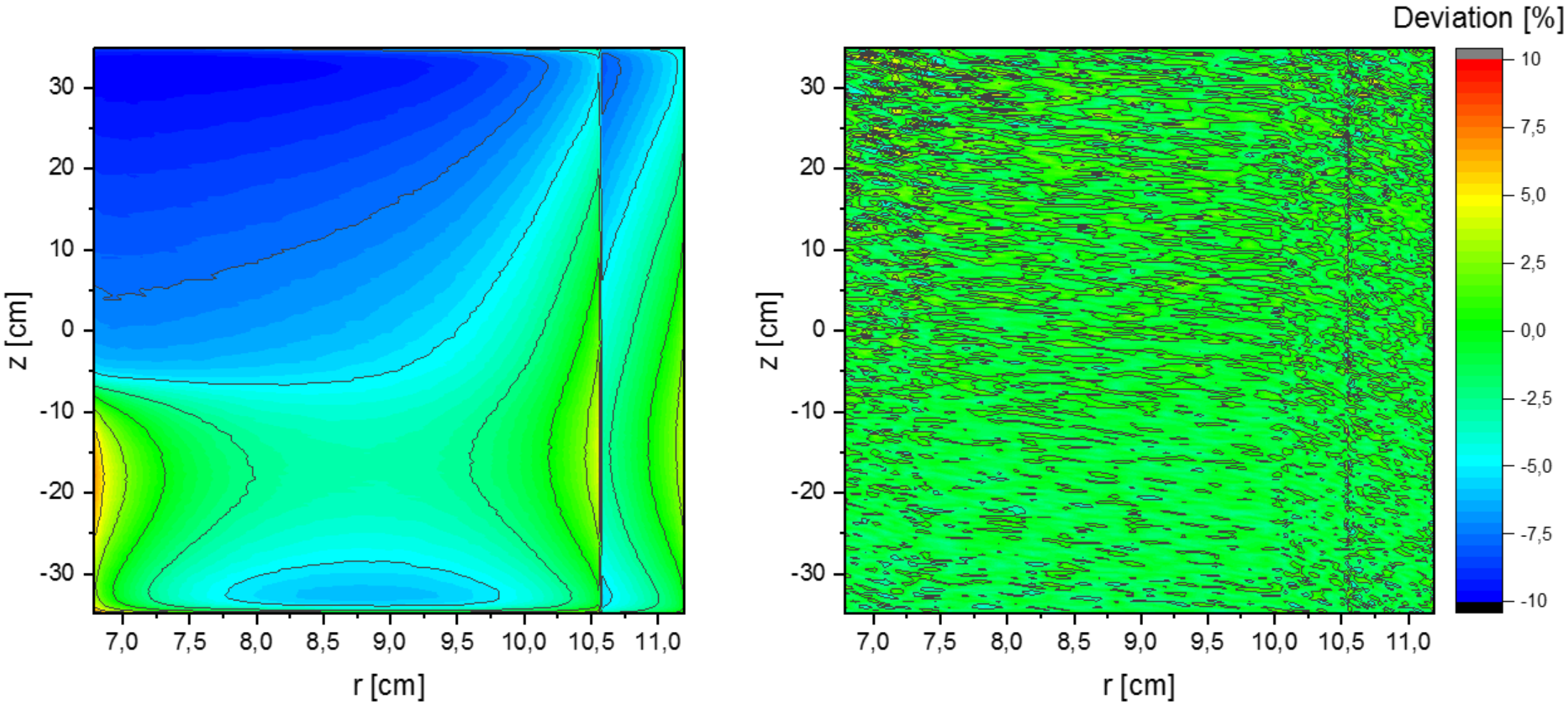
EXACT INVOLUTE GEOMETRY IN SERPENT 2



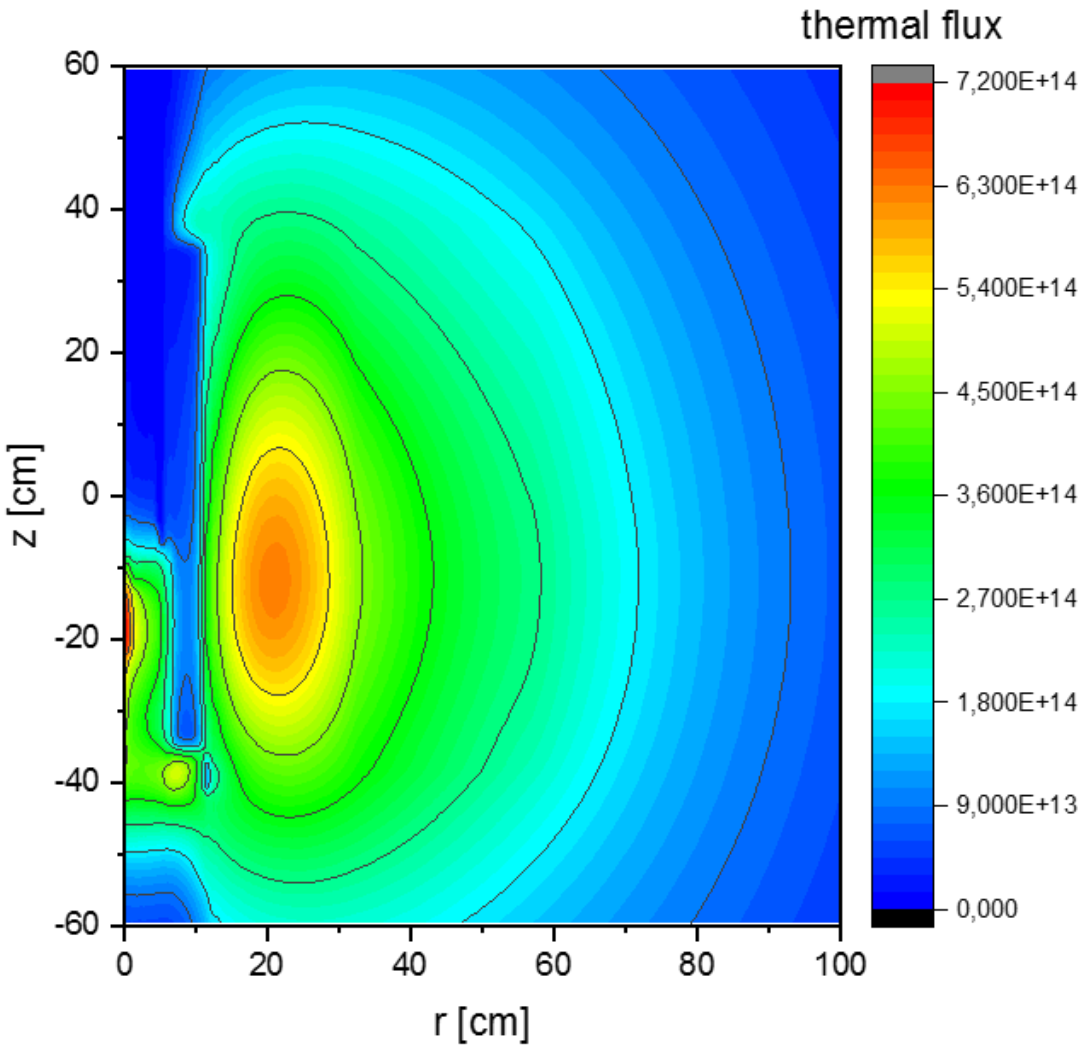
MULTIPLICATION FACTOR

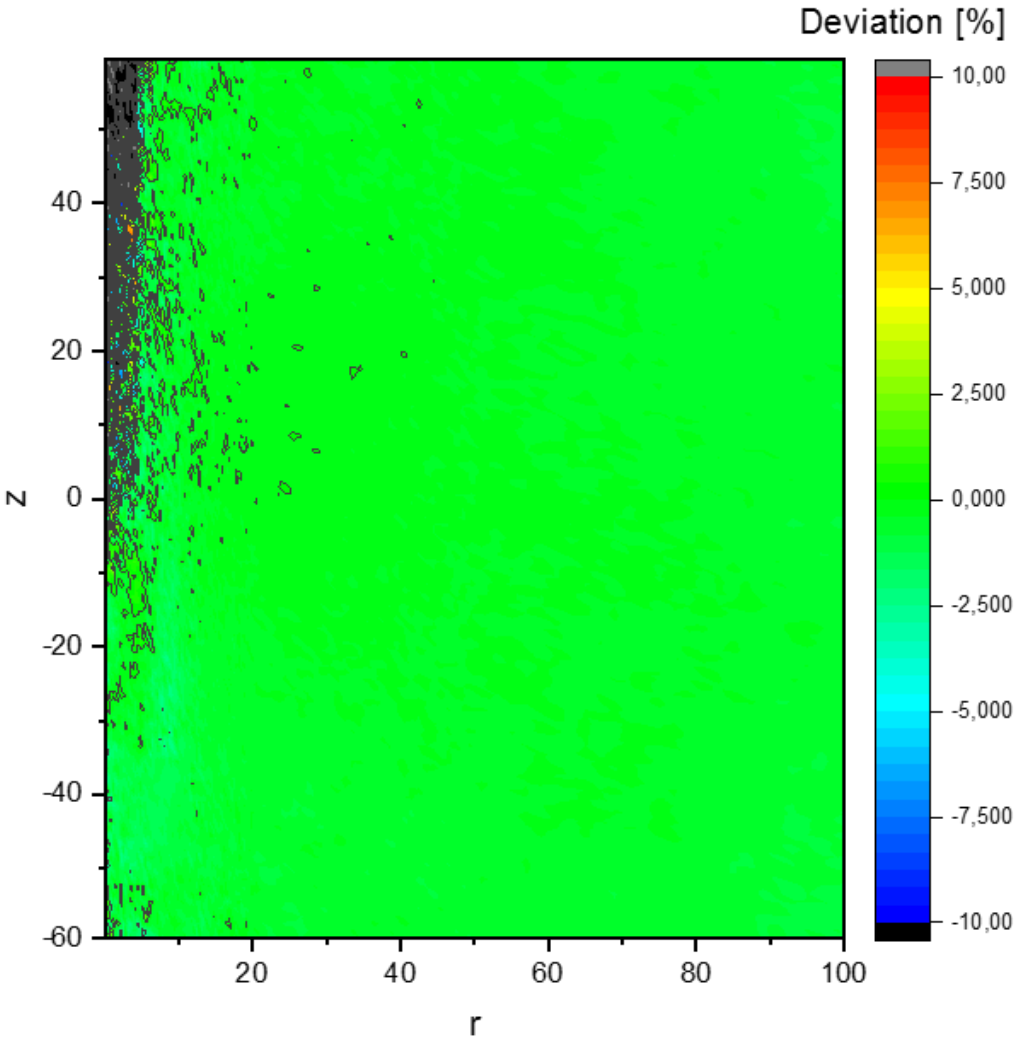
Substitutional Geometry	Parabola – Ellipse Approximation	Exact involutes
		
0,99850 ± 0.0001		

FISSION RATE

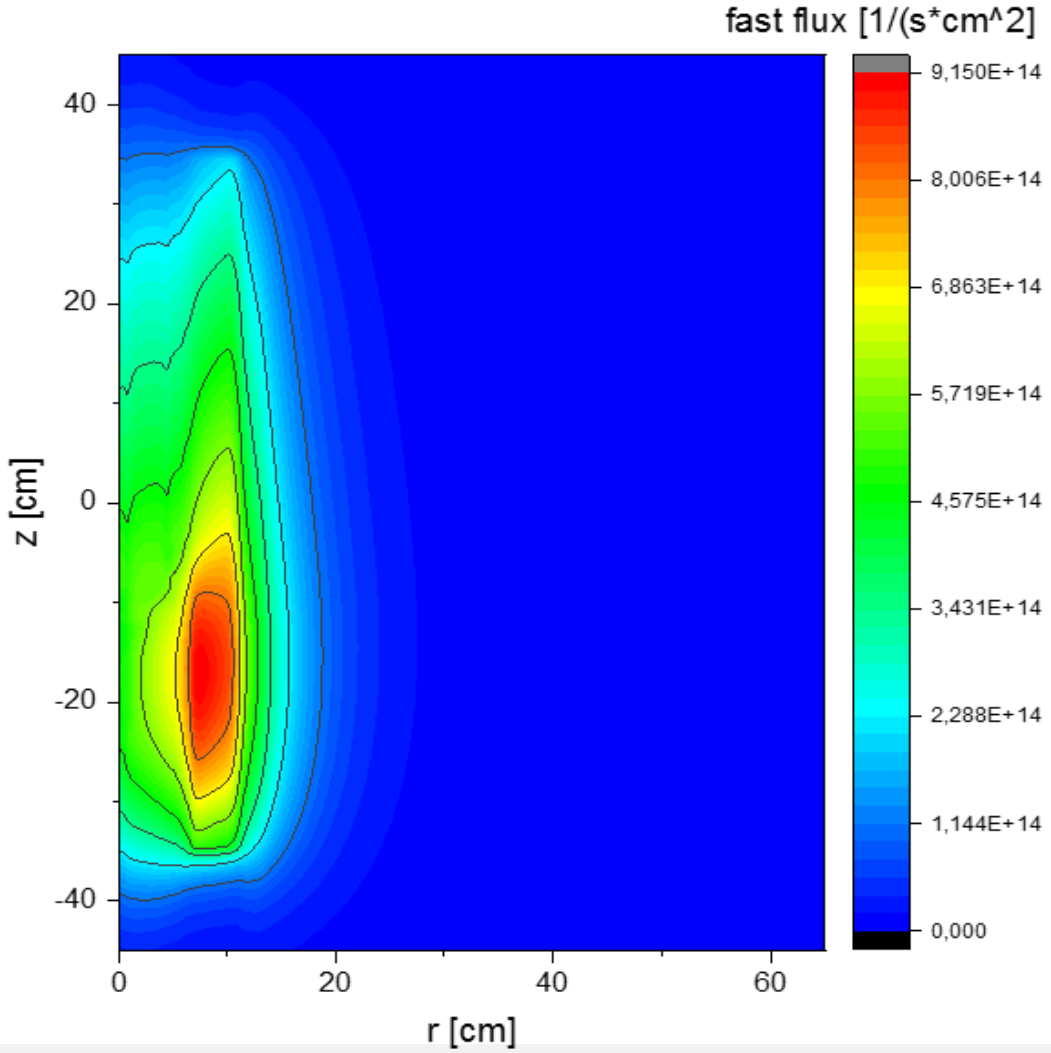


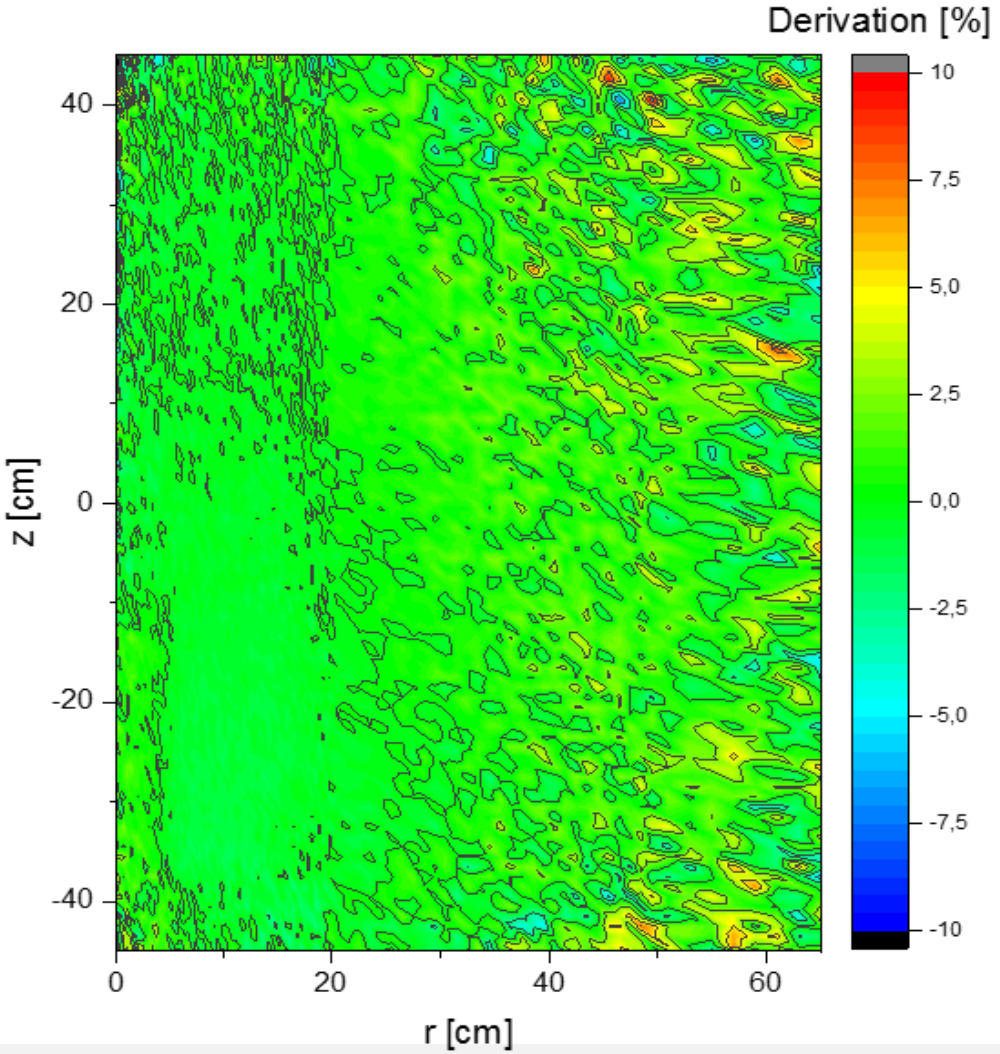
THERMAL FLUX





THERMAL FLUX





ISSUES TO BE SOLVED

- Newtons method works, but:
  - Particles go lost
  - Numerical issues with checkvolumes routine

- Regular Falsi is more stable and faster

THANK YOU