

TVORBA GLOBÁLNÍHO TERMÁLNÍHO MODELU TOKAMAKU COMPASS-U ANEB JAK V COMSOLU PROGRAMOVĚ VYBRAT 2000 PLOCH?

P. BARTOŇ^{1,2}, J. ZELDA¹, J. PŘEVRÁTIL¹, J. HAVLÍČEK¹

1) Ústav fyziky plazmatu, Akademie věd České Republiky, Praha

2) Matematicko-fyzikální Fakulta, Karlova Univerzita, Praha

- Co je tokamak COMPASS-U?
- Globální termální model
- Proč potřebujeme vybírat plochy?
- COMSOL API a „application builder metody“
- Představení vyvinuté aplikace

- TOKAMAKy jsou vědecká zařízení pro výzkum termonukleární fúze
 - slučování jader izotopů vodíku (D+D, D+T)
 - v budoucnu by na jejich principu měla být postavena elektrárna

- Na ÚFP stavíme nový tokamak pojmenovaný **COMPASS-U**

- Hlavní parametry

- Toroidální mag. pole 5 T
- Proud plazmatem 2 MA
- Velký poloměr $R = 0.894 \text{ m}$
- Malý poloměr $a = 0.27 \text{ m}$

Původní COMPASS

- 2.1 T
- 0.35 MA
- 0.56 m
- 0.23 m

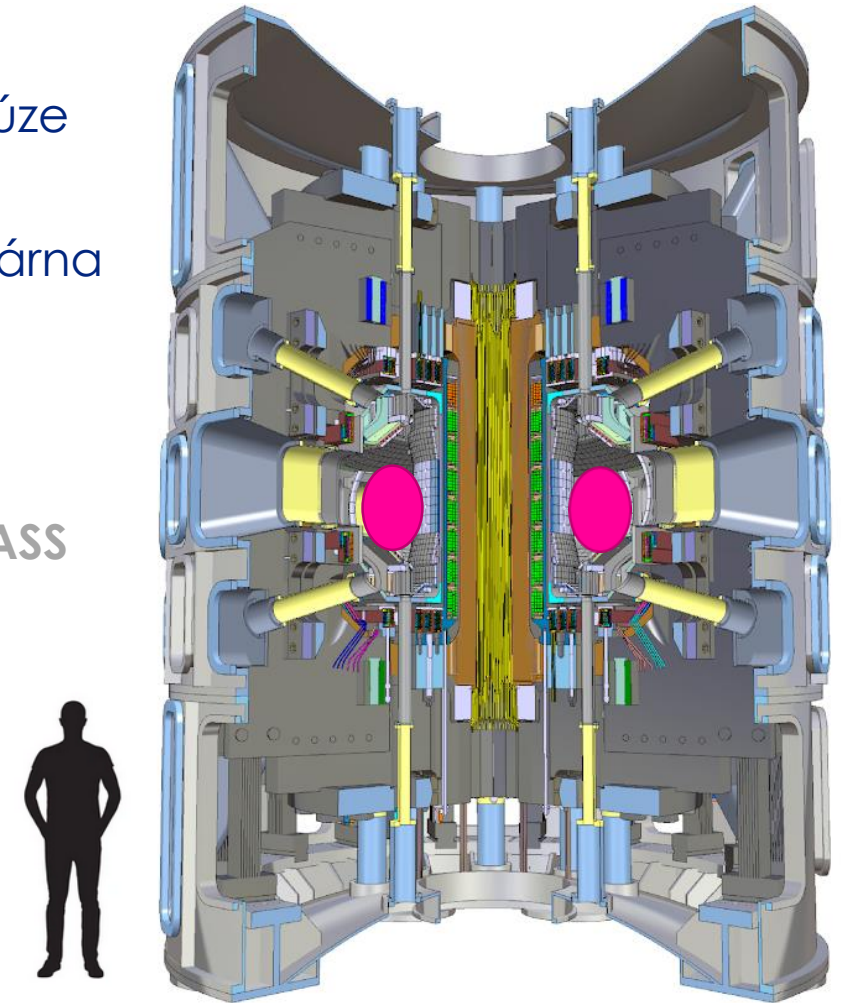


Fig. 1.: Řez tokamakem COMPASS Upgrade

- COMPASS-U bude obsahovat
 - měděné magnetické cívky chlazené na -190 °C a
 - reakční komoru vyhřívanou na 500 °C .
- To znamená rozdíl 700 stupňů na minimální délce!
 - Vzdálenost mezi komponentami $\sim 20\text{ mm}$
- Nutnost podrobného teplotního modelu

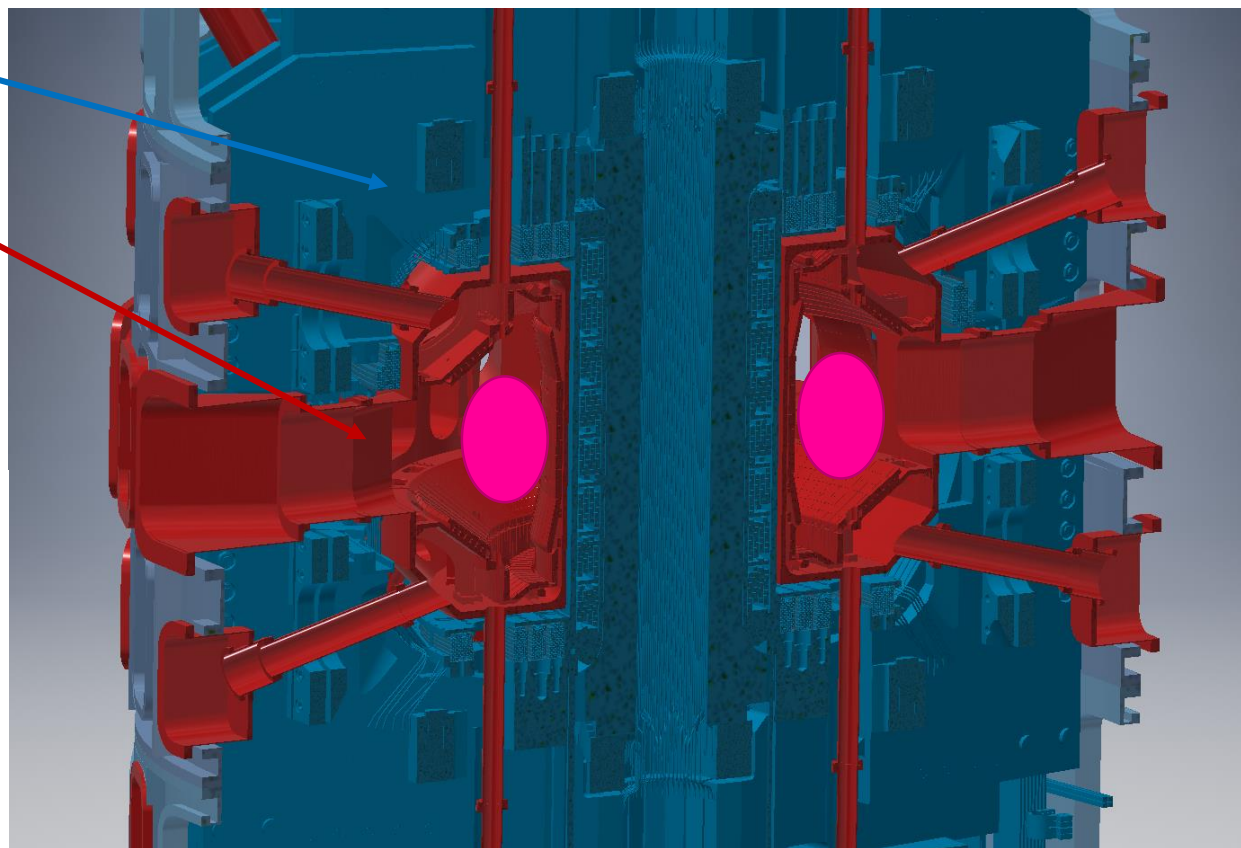


Fig. 2.: Řez tokamaku COMPASS-U s vyznačením teplot jednotlivých komponent

- Trojrozměrný **globální termální model**
 - Zjednodušená geometrie
 - 1/8 celku (využití symetrií)
 - Vnitřek komory překreslen na homogenní
 - 7 použitých materiálů
 - Pevné látky z Material Library
 - Helium z CoolPropu
 - Použité fyziky
 - **Heat Transfer in Solids**
 - **Non-isothermal Pipeflow**
 - Dva heliové okruhy
 - **Surface-to-surface radiation**
 - Dvě skupiny ploch pro tepelnou výměnu

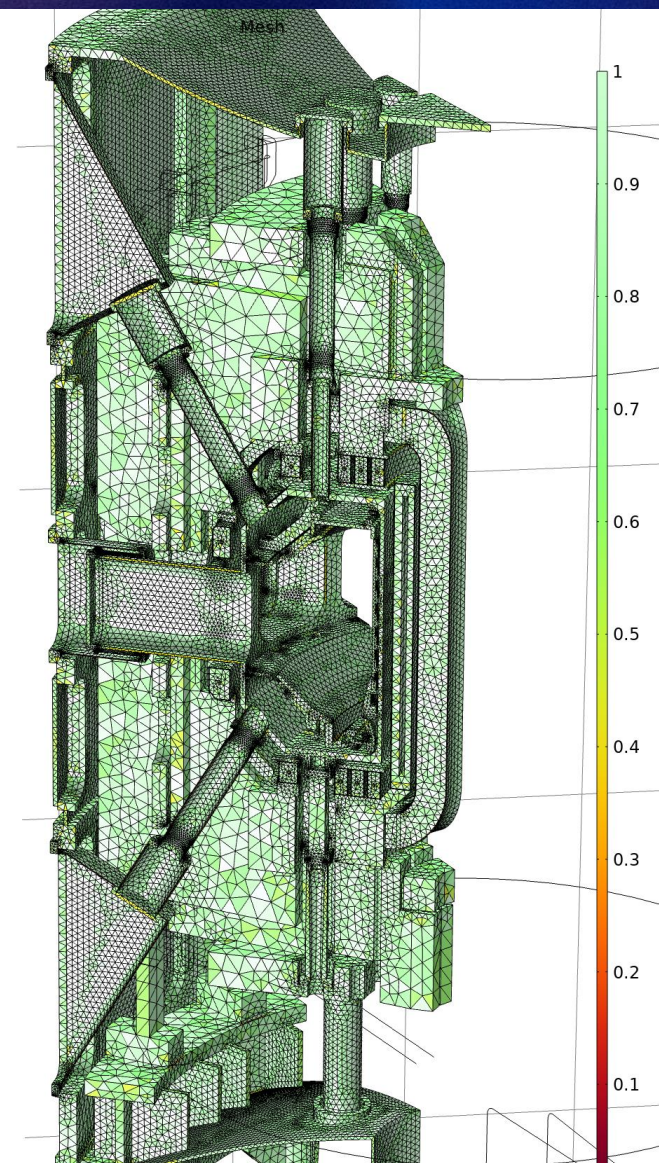
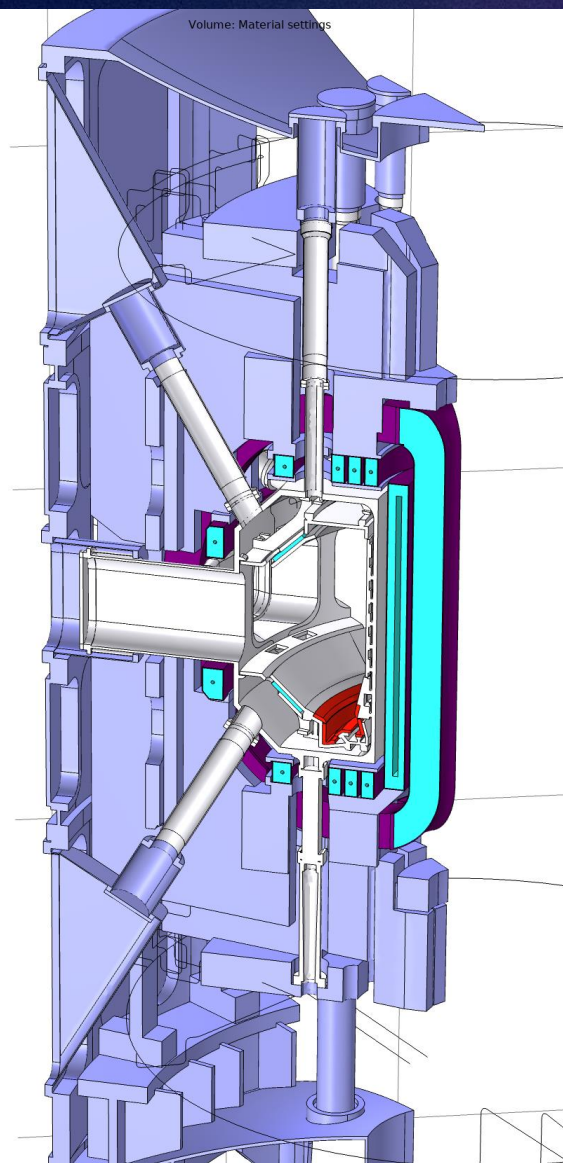


Fig. 3.: Geometrie s vyznačenými materiály

Fig. 4.: Výsledná síť

- Pro surface-to-surface radiation je potřeba zvolit tři skupiny ploch
 - Uvnitř komory
 - Zvenku komory && uvnitř kryostatu
 - Zvenku kryostatu (bez záření)
- Jenže celý model má cca 4000 ploch!
- Volba „Group by continuous tangent“ pomůže
 - Při nastavení na 89 stupňů „nepřeलेze“ na plochu řezu
 - Ale stejně to je zdlouhavý proces,
 - který je navíc po každé změně geometrie nutné opakovat!

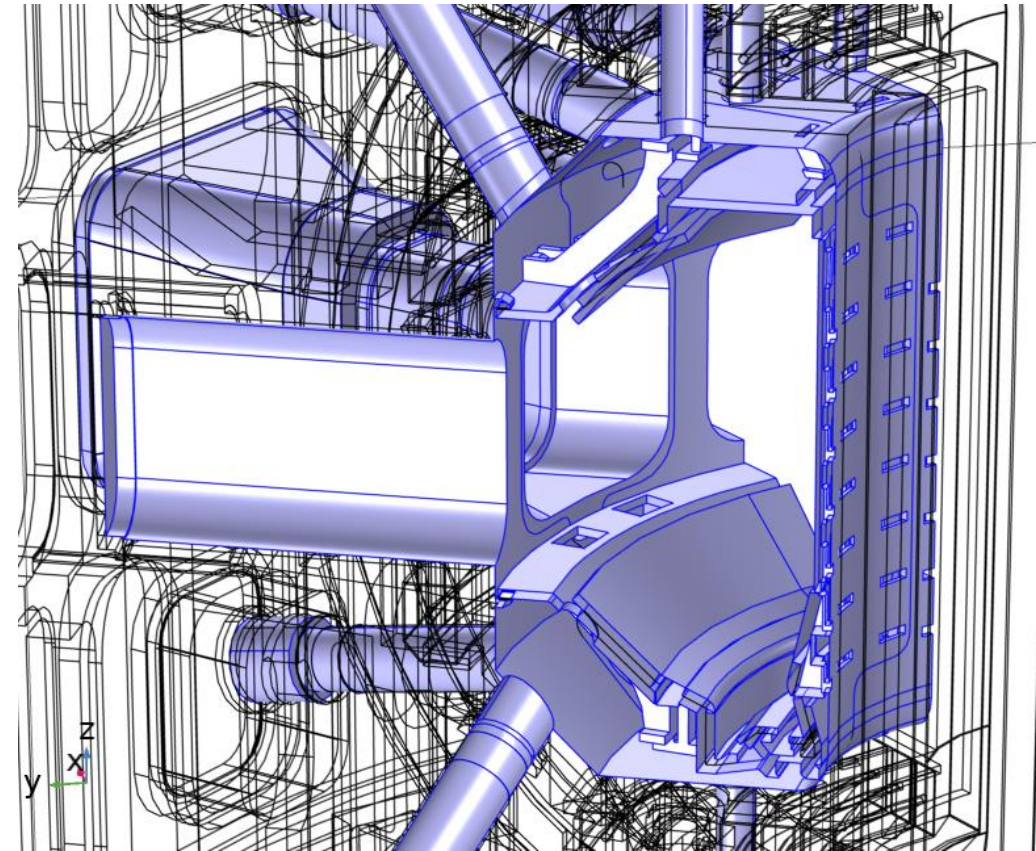


Fig. 5.: Plochy tvořící vnitřní radiační smyčku komory

- COMSOL je postaven nad Javovým API, přes které jde konat všechny GUI akce (a něco navíc)
- API lze volat
 - externě z Javových programů,
 - externě z Matlabu (LiveLink for MATLAB)
 - Interně z Application Builderu
- Application builder ale umí **nejen** „standalone aplikace“!
 - Jeho součástí je možnost tvorby formulářů/dialogů pro použití uvnitř COMSOLu
 - A nebo jenom tvorba samostatných metod

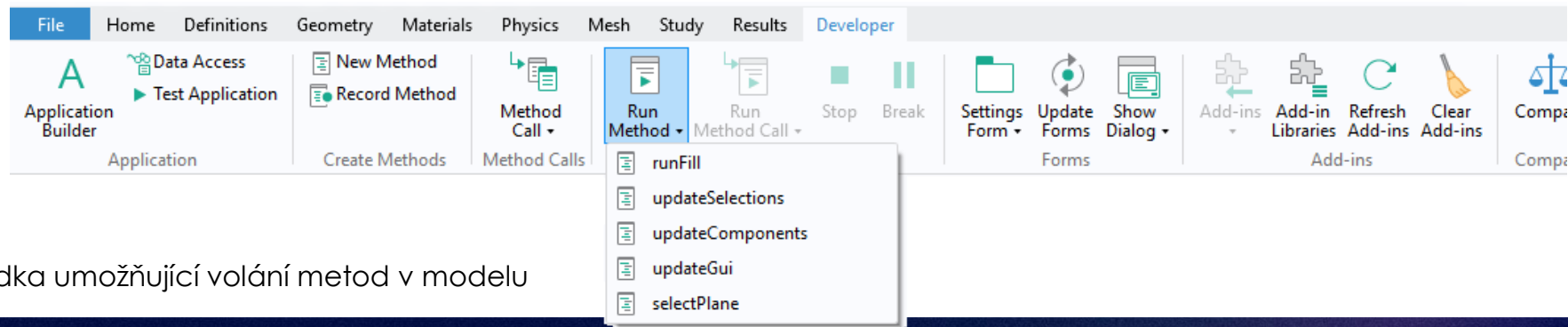
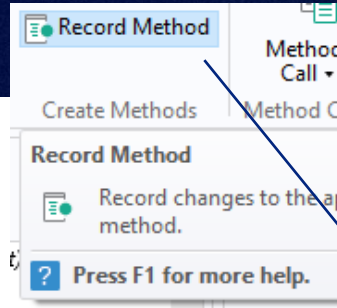
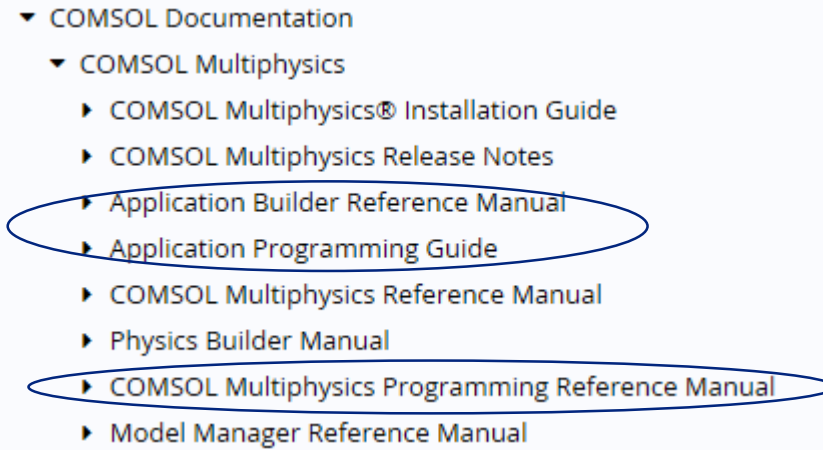


Fig. 6.: Nabídka umožňující volání metod v modelu

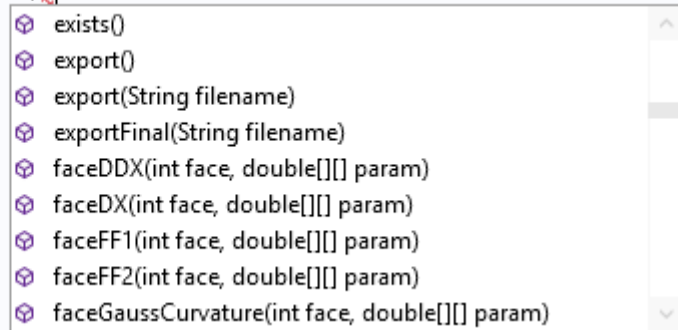
- Jak na to?
- COMSOL umí „nahrávat“ akce!
- Má bohatou dokumentaci
- Má překvapivě schopné IDE



```
model.component("comp1").selection().create("sel8", "Explicit");
model.component("comp1").selection("sel8").geom(2);
model.component("comp1").selection("sel8").set(1701);
model.component("comp1").selection("sel8").set(1301, 1701);
model.component("comp1").selection().create("com1", "Complement");
model.component("comp1").selection("com1").set("entitydim", 2);
model.component("comp1").selection("com1").set("input", new String[]{"sel8"});
```



```
model.component("comp1").geom("geom1").|
```



- COMSOL API je primárně určené pro komunikaci „GUI – backend“
 - není tam přehršel „užitečných funkcí“ které by člověk čekal
 - např. zjistit polohy vrcholů tvořících stěnu je překvapivě komplikované
- V AppBuilder metodách nelze importovat třídy z java stdlib
 - To znamená že jsou dostupné jen inbuilt datové typy
 - Boolean, int, double, string, array, array2d
 - např. nelze použít java.util.ArrayList!

```
3 | import java.util.ArrayList; // import the ArrayList class
```

- Pro náš úkol chceme zvolit všechny plochy, které spolu **sousedí**, ale přitom neleží **na řezech**
- Jednoduchý „floodfill“ algoritmus

```
} while (stack.length > 0);
```

- Dokud máme něco v zásobníku

```
int workingBoundary = stack[0];
stack = remove(stack, 0);
```

- Vytáhneme plochu ze zásobníku

- Najdeme všechny její sousedy

```
int[] adjacentBoundaries = comp.geom("geom1").getAdj(2, 2, workingBoundary);
```

- Všechny sousedy prověříme, že

- se nenachází na řezu a

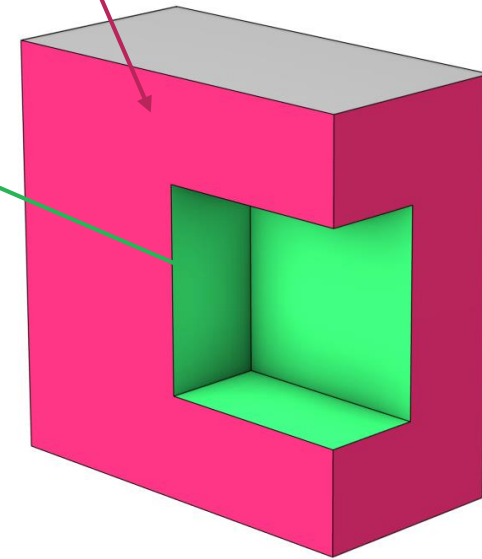
```
boolean isIgnored = (ignoreList[adjacentBoundary] == 1);
```

- už jsme tam nebyli a

```
boolean isAlreadySelected = (selected[adjacentBoundary] == 1);
```

- v tom případě ho přidáme do výběru a do zásobníku

```
selected[adjacentBoundary] = 1;
stack = append(stack, adjacentBoundary);
comp.selection(resultSelection).add(adjacentBoundary);
```



- Metoda v pozadí
- GUI formulář v popředí

```

40  do {
41      int workingBoundary = stack[0];
42      stack = remove(stack, 0);
43
44      int[] adjacentBoundaries = comp.geom("geom1").getAdj(2, 2, workingBoundary);
45      for (int adjacentBoundary : adjacentBoundaries) {
46          boolean isIgnored = (ignoreList[adjacentBoundary] == 1);
47          boolean isAlreadySelected = (selected[adjacentBoundary] == 1);
48
49          if (!isIgnored && !isAlreadySelected) {
50              int adjacentDomains = comp.geom("geom1").getAdj(2, 3, adjacentBoundary).length;
51              if ((adjacentDomains < 2) || !ignore_inner_boundaries) {
52                  selected[adjacentBoundary] = 1;
53                  stack = append(stack, adjacentBoundary);
54                  comp.selection(resultSelection).add(adjacentBoundary);
55
56                  boundaries_selected++;
57                  stack_size = stack.length;
58                  if (slowdown) {
59                      sleep(slowdown_time);
60                  }
61              }
62          }
63      }
64  } while (stack.length > 0);
    
```

Fig. 7.: Ukázka metody pro výběr sousedících ploch v dutině

FloodFiller

Working component: -- Please select --

Selection to floodfill: -- Please select --

Selection to ignore: -- Please select --

Ignore inner boundaries

Slow down the process 20 ms/step

Start FloodFill

Stack size: 4

Boundaries selected: 1186

Fig. 8.: Vytvořený formulář pro výběr ploch v dutině

The screenshot displays the COMPASS software interface. On the left is a tree view of the model's structure. The main window shows a 3D wireframe model of a complex vessel interior. A 'FloodFill' dialog box is open in the center, with the following settings:

- Working component:** Component 1 (com1)
- Selection to floodfill:** vessel_interior
- Selection to ignore:** to_ignore
- Ignore inner boundaries
- Slow down the process: 10 ms/step
- Stack size:** 4
- Boundaries selected:** 1186
- Log:**
 - 201 boundaries to ignore (in uni1)
 - 1 boundaries to start with (in sel2)
 - Got 4095 boundaries
 - 201 boundaries to ignore (in uni1)
 - 1 boundaries to start with (in sel2)
 - Got 4095 boundaries

The 'Start FloodFill' button is highlighted with a mouse cursor. A small portion of the vessel interior is highlighted in purple in the 3D view. At the bottom, a 'Messages' panel shows the execution log.

- COMSOL umí vyexportovat metody a formuláře ve formě Add-In
- Ty lze katalogizovat v knihovně, používat v jiných modelech a distribuovat dál

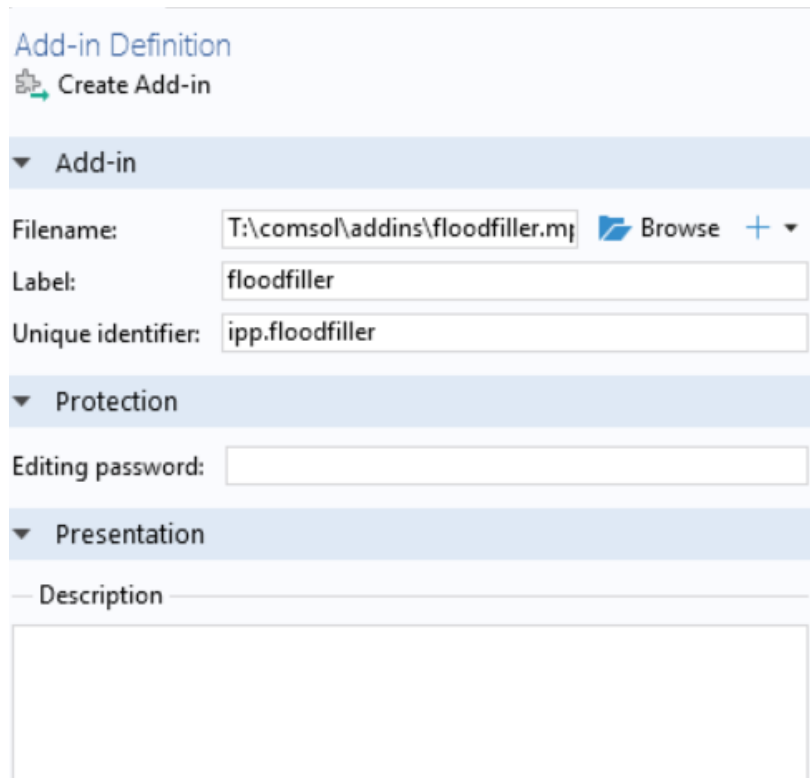


Fig. 9.: Rozhraní pro tvorbu add-inů


 floodfiller.mph	27.05.2022 8:59	COMSOL Applicati...	709 kB
---	-----------------	---------------------	--------

Fig. 10.: Vyexportovaný add-in

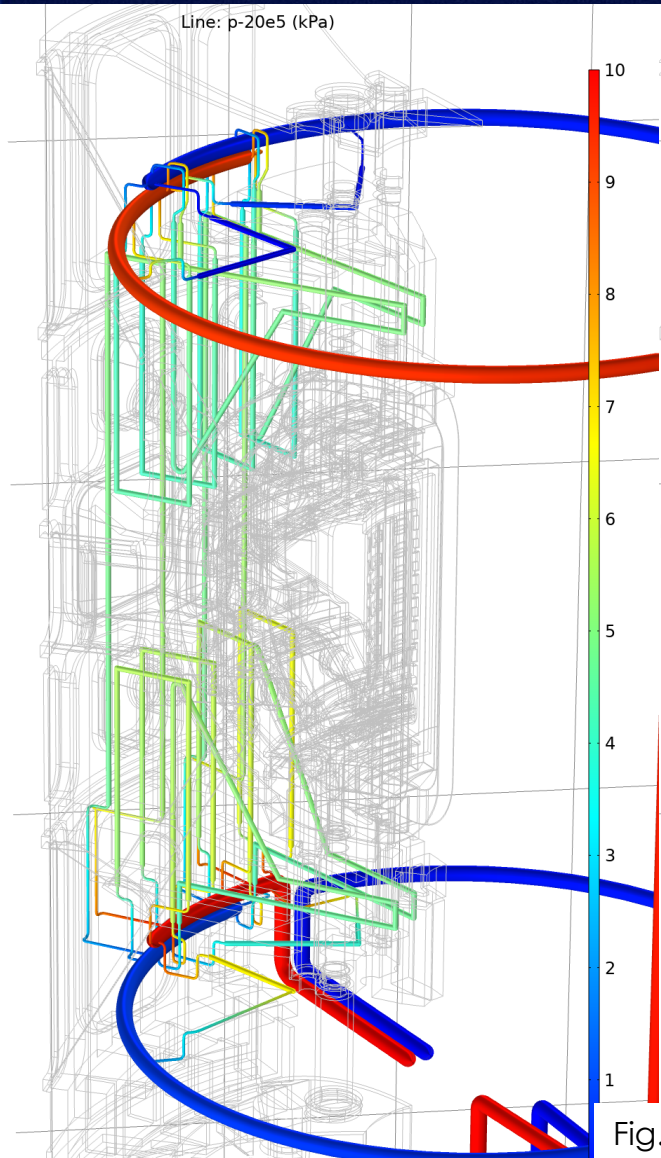


Fig. 11.: Rozložení tlaku na chladícím okruhu

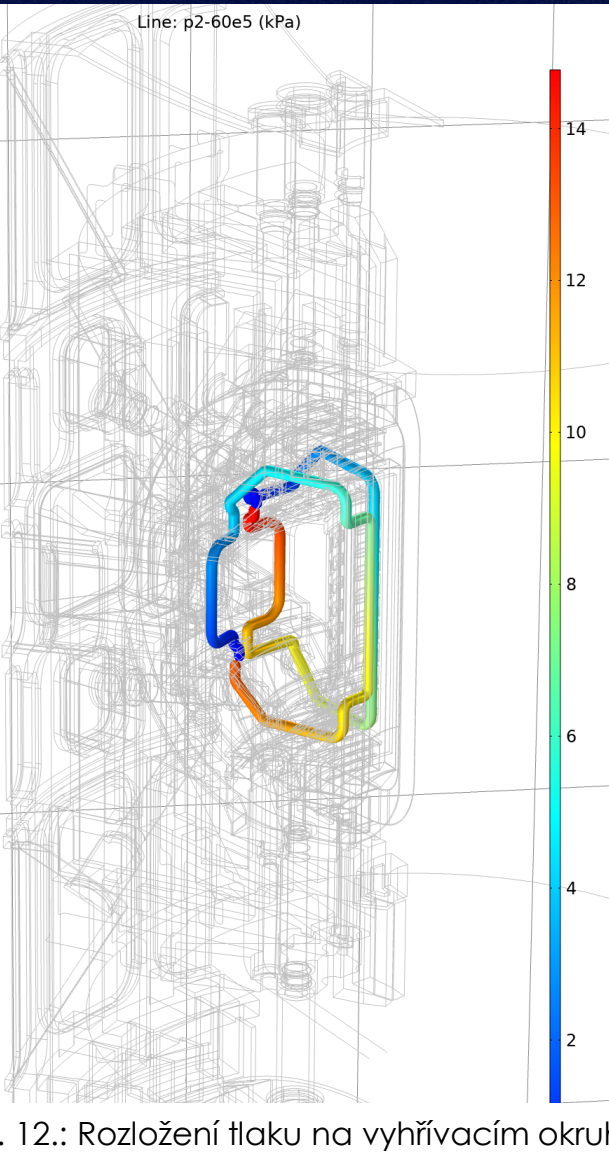


Fig. 12.: Rozložení tlaku na vyhřívacím okruhu

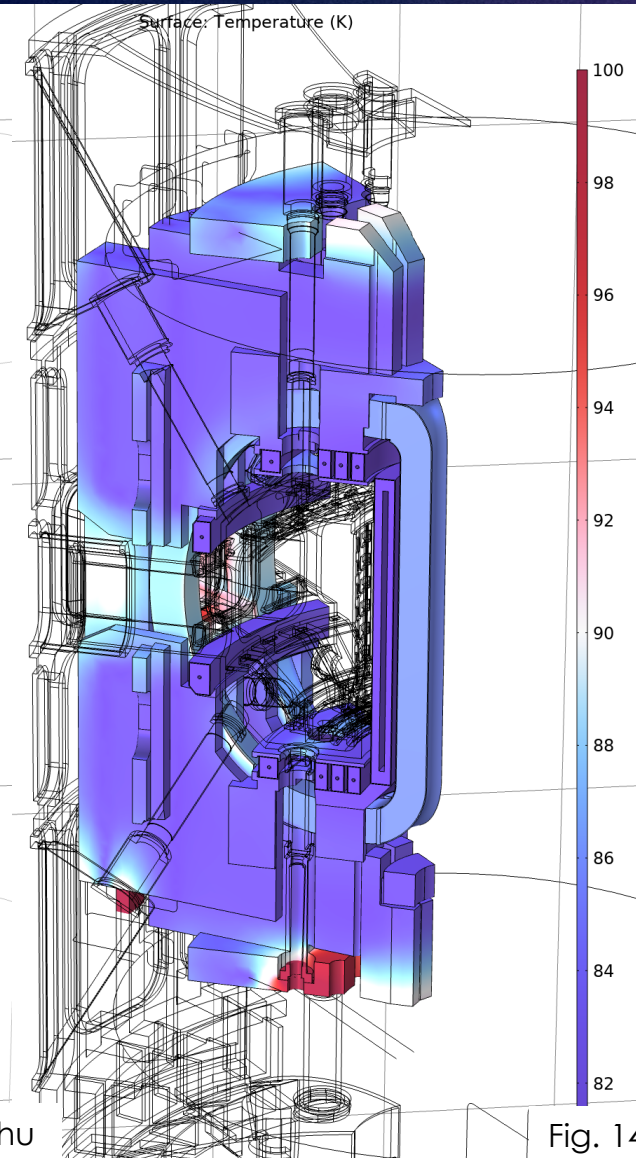


Fig. 13.: Výsledné rozložení teplot chladných komponent

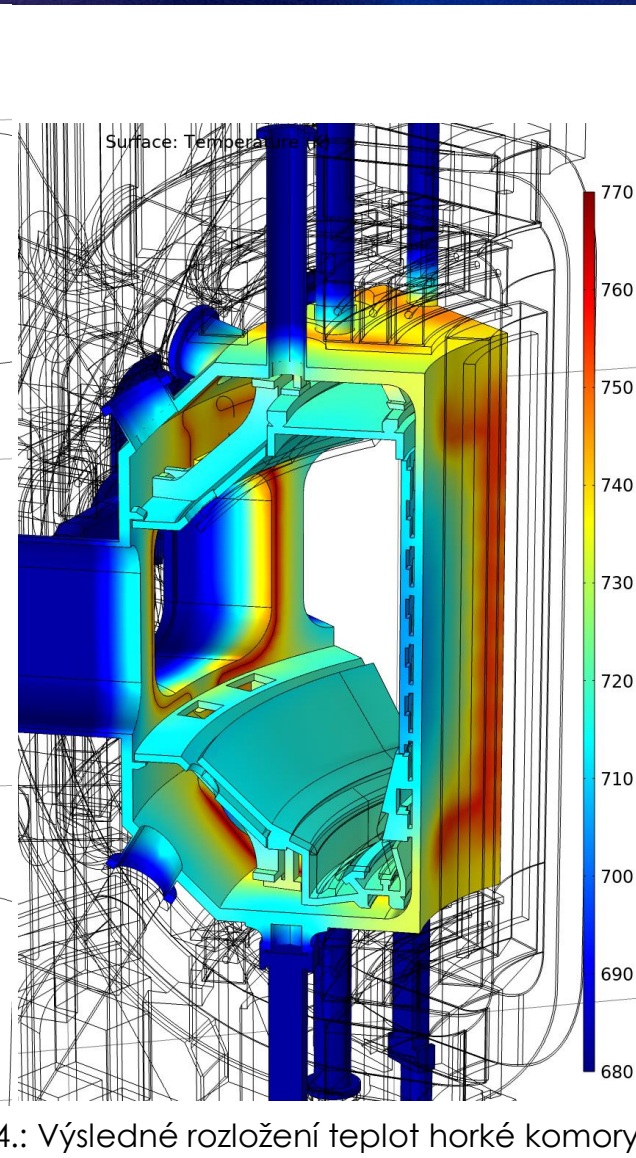


Fig. 14.: Výsledné rozložení teplot horké komory