

1. Do uruchomienia będzie potrzebny system Linux.

Możliwe jest uruchomienie na maszynie wirtualnej (VMPlayer + Linux Ubuntu image)

2. Installation of IGA-ADS solver on Linux

Instalacja (starsza wersja) opisana jest w rozdziale 8 artykułu

https://www.researchgate.net/publication/313532745_IGA-ADS_Isogeometric_analysis_FEM_using_ADS_solver

a) Install cmake (co najmniej wersja 3.13)

```
sudo apt install cmake
```

b) Install gfortran

```
sudo apt install gfortran
```

c) Install lapack

```
sudo apt install liblapack-doc
```

```
sudo apt install liblapack-dev
```

d) Install blas

```
sudo apt install libblas-doc
```

```
sudo apt install libblas-dev
```

e) Install boost (co najmniej wersja 1/58)

```
sudo apt install libboost-all-dev
```

f) Install llvm

```
sudo apt install llvm-12 llvm-12-dev
```

g) Download IGA-ADS solver

```
git clone https://github.com/marcinlos/iga-ads
```

```
git checkout develop
```

h) Proszę uruchomić skrypt który instaluje bibliotekę GALOIS z zależnościami

```
DEPS=$(realpath deps)
```

```
iga-ads/scripts/install-dependencies.sh $(realpath deps-build) "${DEPS}"
```

UWAGA: Galois nie kompiluje się poprawnie z nowszymi wersjami GCC (≥ 11) (<https://github.com/IntelligentSoftwareSystems/Galois/issues/401>). Aby naprawić ten problem, przed uruchomieniem skryptu install-dependencies.sh należy dodać do niego następującą linijkę:

```
sed -i '36i #include <optional>' Galois/tools/graph-convert/graph-convert.cpp
```

pod liniijką 82. Po tej zmianie ten kawałek skryptu powinien wyglądać następująco:

```
# Install Galois
GALOIS_VER=6.0
git clone --branch release-${GALOIS_VER} --depth=1 --quiet https://github.com/IntelligentSoftwareSystems/Galois
sed -i '36i #include <optional>' Galois/tools/graph-convert/graph-convert.cpp
```

i) Kompilujemy IGA-ADS

```
cd iga-ads
```

```
mkdir build
```

```
cmake -S . -B build -D CMAKE_BUILD_TYPE=Release -D ADS_USE_GALOIS=ON -D USE_MUMPS=OFF -
D CMAKE_PREFIX_PATH="${DEPS}"
```

```
cmake --build build -j $(nproc)
```

j) Przykładowe uruchomienie

```
cd build/examples (tutaj są różne exeki)
```

```
./heat_2d
```

Liczy 10,000 kroków czasowych transportu ciepła na siatce 40x40

Wypluwa out_0.data out_*.data

k) Generacja rysunków

```
gnuplot
```

```
gnuplot> plot "out_100.data" with image
```

Proszę przygotować następujący raport

**W każdym zespole 3 osobowym proszę zainstalować IGA-ADS na każdym z 3 komputerów
[W tym ćwiczeniu przyznajemy 3 osobne oceny dla 3 osób]**

1. Dane procesora i pamięci

2. Log z instalacji (lub z podłączenia do ścieżki) cmake, gfortran, lapack, blas i boost

3. Log z wywołania

```
cmake -S . -B build -D CMAKE_BUILD_TYPE=Release -D ADS_USE_GALOIS=ON -D USE_MUMPS=OFF -
D CMAKE_PREFIX_PATH="${DEPS}"
```

wskazujący na możliwość kompilacji IGA-ADS

4. Log z kompilacji IGA-ADS

5. Listing katalogu po uruchomieniu ./heat_2d

6. Jaki największy krok czasowy nie powoduje eksplozji symulacji?

Animacja z wynikami symulacji zrobiona za pomocą gnupłota lub innego ulubionego narzędzia oraz sklejacza png (lub innego wybranego narzędzia do generacji animacji)

7. Jaki najmniejszy krok czasowy powoduje eksplozję symulacji? Proszę wygenerować animacje

Animacja z wynikami symulacji zrobiona za pomocą gnupłota lub innego ulubionego narzędzia oraz sklejacza png (lub innego wybranego narzędzia do generacji animacji)