iscte INSTITUTO UNIVERSITÁRIO DE LISBOA

ANNEX A - APPLICATION INSTALLATION MANUAL

HIGH-PERFORMANCE GPU-ACCELERATED FINITE ELEMENT ANALYSIS

António Cruz (140129), Bruno Santos (140586), Pedro Miranda (129268), Ricardo Kayseller (95813)

High-Performance GPU-Accelerated Finite Element Analysis

Annex B - FEMulator Pro Installation

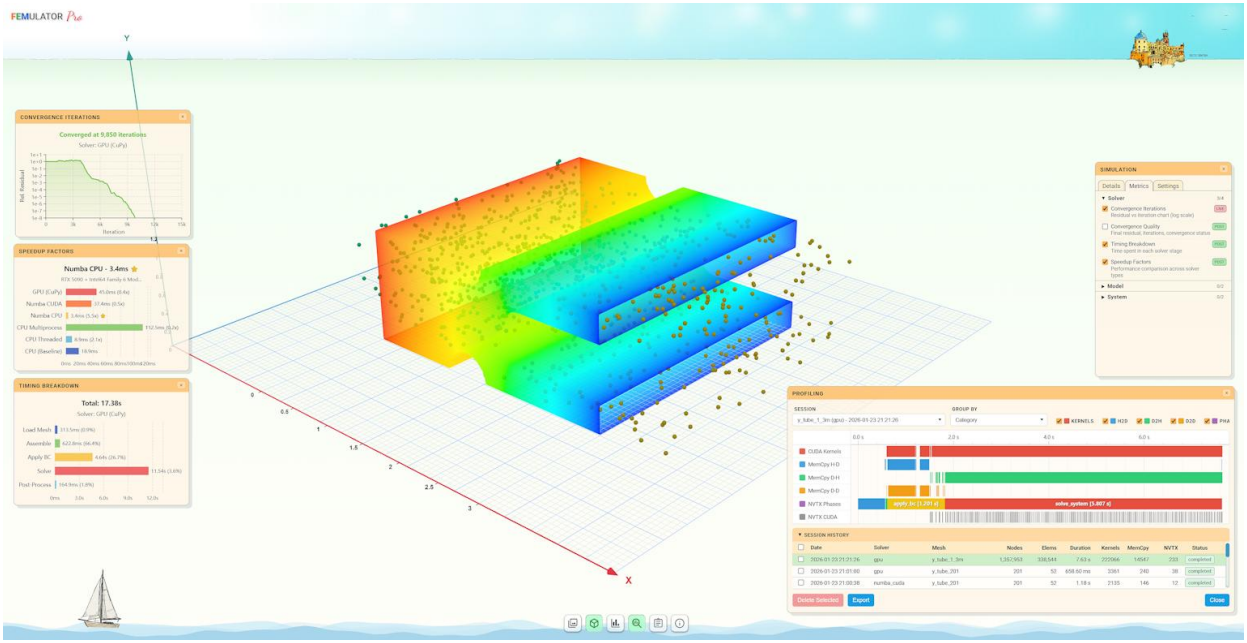

Figure 1: *FEMUlator Pro application screenshot upon running simulation and profiling sessions.*

1. Online Access (Recommended for Evaluation)

For project evaluation purposes, **FEMulator Pro** is available online at:

https://logus2k.com/fem

No installation required - simply access the URL above to use the application with full GPU acceleration.

If you prefer to run the application locally, follow the installation instructions below.

2. Run Locally using Docker (single command)

The simplest way to get **FEMulator Pro** running locally is with **one single docker run command**. This command has the **same effect** as the docker-compose.yml configuration shown later.

```
docker run -d --name femulator --hostname femulator --gpus all -p 5868:5868 logus2k/femulator
```

Notes: - --gpus all -> tries to use all available NVIDIA GPUs (ignored gracefully if no GPU or no NVIDIA support) - First run will automatically **pull** the latest image from Docker Hub - Access the application at: **http://localhost:5868**

3. Run Locally using Docker Compose

1. Prerequisites

- **Docker Desktop** (Windows / macOS) or **Docker Engine** + **Docker Compose plugin** (Linux)
- Install Docker here if not already installed

2. Create docker-compose.yml

Create a folder anywhere on your computer and save the following content as **docker-compose.yml**:

```yaml
services:
  femulator:
    image: logus2k/femulator:latest
    container_name: femulator
    hostname: femulator
    restart: unless-stopped
    deploy:
      resources:
        reservations:
          devices:
            - driver: nvidia
              count: "all"
              capabilities: [gpu]
    logging:
      options:
        max-size: "10m"
        max-file: "3"
    ports:
      - "5868:5868"
    networks:
      - femulator_network

networks:
  femulator_network:
    driver: bridge
```

3. Launch the Application

Open a terminal/command prompt in the folder containing docker-compose.yml and run:

```
docker compose up -d
```

(or older syntax: docker-compose up -d)

The image will be downloaded automatically, and the container will start in the background.

-> Access the application at **http://localhost:5868**

Management Commands (Docker Compose)

Action	Command
Start (or restart)	docker compose up -d
Stop and remove container	docker compose down
View logs (live)	docker logs -f femulator
Pull latest image & restart	docker compose pull && docker compose up -d
Stop, remove & clean up	docker compose down --rmi all

4. Platform-Specific Helper Scripts (Advanced/Development)

For users who prefer not to manage docker-compose.yml manually or who are working with local source code, the project provides **ready-to-use helper scripts**.

Important:

- **Windows users** -> use only files ending in **.bat**
- **Linux users** -> use only files ending in **.sh**

Purpose	Windows	Linux
Start application	start.bat	./start.sh
Stop application	stop.bat	./stop.sh
Update (soft rebuild)	update.bat	./update.sh
Full rebuild	rebuild_all.bat	./rebuild_all.sh
Complete removal	remove_all.bat	./remove_all.sh

First-Time Setup (Linux only)

chmod +x *.sh

Typical Usage Flow

1. **First start / daily use**
   Windows: start.bat
   Linux: ./start.sh

2. **Stop when finished**
   Windows: stop.bat
   Linux: ./stop.sh

3. **Update after code changes**
   Stop -> Update -> Start
   (Windows: stop.bat -> update.bat -> start.bat)

4. **Full reset (when something is broken)**
   Windows: rebuild_all.bat then start.bat
   Linux: ./rebuild_all.sh then ./start.sh

5. **Completely uninstall**
   Windows: remove_all.bat
   Linux: ./remove_all.sh

CPU vs GPU Behavior

- **CPU mode** -> always available, used by default
- **GPU mode** -> automatically enabled **only** if:
  - NVIDIA GPU is present
  - NVIDIA Container Toolkit / WSL2 GPU support is properly configured
  - The --gpus all flag or deploy.resources.reservations.devices is present

No manual choice is required - the container will use GPU **if possible**, otherwise, fallback to CPU.

Notes

- The official pre-built image is hosted on **Docker Hub**:
  -> **logus2k/femulator:latest**
- The single docker run command and the docker-compose.yml file are **functionally equivalent**
- All management can be done with the helper scripts **or** with plain Docker / Compose commands - choose whatever is more convenient for you

Enjoy using **FEMulator Pro**!