Solving the Kuramoto Oscillator Model of Power Grids on GPU

Lilla Barancsuk[1,2]
Bálint Hartmann[1,2], Jeffrey Kelling[3], Géza Ódor[2]

[1]Budapest University of Technology and Economics
[2]Centre for Energy Research
[3]Helmholtz-Zenter Dresden-Rossendorf

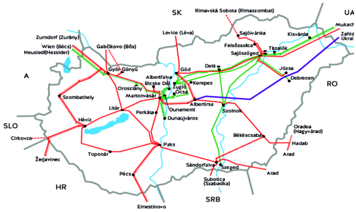
barancsuk.lilla@ek-cer.hu

MŰEGYETEM 1782

Centre for Energy Research

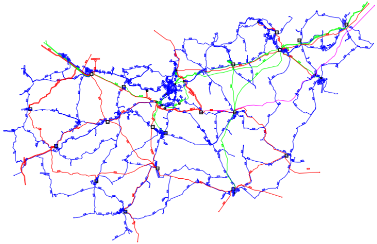
GPU Day 2021, Budapest

Introduction: power grid in a nutshell



- Critical infrastructure
- Primarily powered by electromechanical generators ⇝ stability in frequency
- Recent increase in distributed production (renewables) ⇝ instability
- More frequent outages, extreme events

Modelling power grids for stability investigation



The topography of Hungarian transmission and sub-transmission networks

- Graph representation
- Complex, hierarchical networks
- Coupled oscillator model – mathematical formulation using the Kuramoto model[1]
- Outages:
  – line, node failures
  – failure propagation ⇝ cascade phenomena

[1] Ódor & Hartmann: Power-Law Distributions of Dynamic Cascade Failures in Power-Grid Models. Entropy, Vol. 22, No. 6 (2020)

Problem formulation: the Kuramoto model

Modelling each node as an oscillator

Second-order Kuramoto equation

$$\ddot{\Phi}_i(t) = \omega_{i,0} - \alpha\dot{\Phi}_i(t) + \frac{K}{N}\sum_{j=1}^{N} A_{ij}\sin(\Phi_j(t) - \Phi_i(t))$$

- Φ_i angle of the ith oscillator
- N number of nodes
- K global coupling
- A_{ij} admittance matrix
- α dissipation
- $\omega_{i,0}$ intrinsic frequencies (unit variance Gaussian distribution centered at $50\,$Hz)

Coupled non-linear ordinary differential equations $\rightsquigarrow$ numerical solution by integration

Implementation on GPU's

$$\ddot{\Phi}_i(t) = \omega_{i,0} - \alpha\dot{\Phi}_i(t) + \frac{K}{N}\sum_{j=1}^{N} A_{ij}\sin(\Phi_j(t) - \Phi_i(t))$$

Implementation builds on the code of Jeffrey Kelling[2]

- First- and second-order Kuramoto model for arbitrary graphs
- Integration scheme: Runge–Kutta 4 (provided by `boost::numeric::odeint`[3])
- Accelerator for `odeint`: VexCL library
- Special graph data structure for efficient memory access and load balancing on GPU

[2] Jeffrey Kelling: Solving the Kuramoto Oscillator Model on Random Graphs, GPU Day (2019)
[3] boost.org

Recent developments

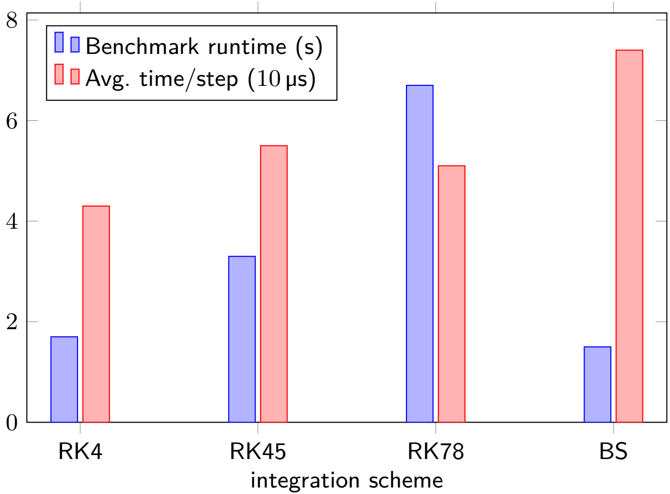
- Higher order integration schemes
- Line failure modelling by threshold cuts
- Stopping criteria for steady state
- In progress: region-specific analysis of the network

Integration schemes

We integrated additional adaptive steppers by `boost::numeric::odeint`

- Runge–Kutta Fehlberg 78
  - High order method with error estimation
  - Order: 8
  - Worst overall performance
- Runge–Kutta Cash–Karp 54
  - General scheme with error estimation
  - Order: 5
- Bulirsch–Stoer
  - Stepper with step size and order control
  - Very good if high precision is required
  - Best in terms of overall performance

Stepper performance evaluation



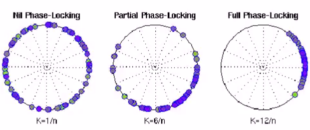
Benchmark: 2D lattice with 10000 nodes, for 100 time steps, precision: 10^{-4}.
Using GeForce Titan GTX Black

Modelling outages: threshold line cuts

- Intentional line cuts for perturbing the network:
  – thermalization
  – setting A_{ij} from 1 to 0 for a predefined set of lines
  – analyse failure cascade

- Line overcurrent protection
  If power flow between nodes exceeds a threshold:
  $|\sin(\Phi_j - \Phi_i)| > T \rightarrow A_{ij} = 0$

- Measuring synchronization with the Kuramoto order parameter

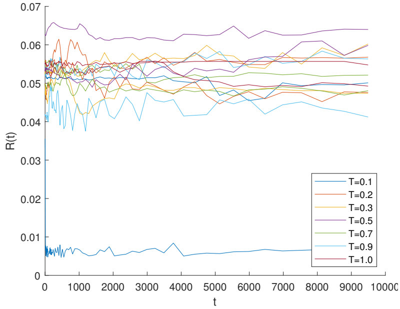


Phase locking oscillators[4]

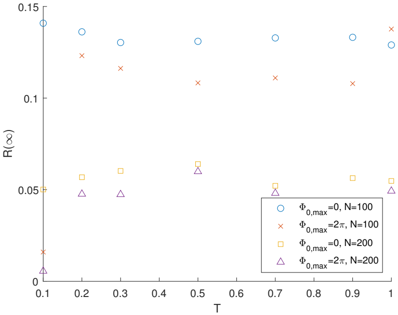
$$R(t) = \left\langle \frac{1}{N} \left| \sum_{j=1}^{N} \exp\left(\mathrm{i}\Phi_j(t)\right) \right| \right\rangle$$

[4] Source: https://en.wikipedia.org/wiki/Kuramoto_model

Experimental results on a 2D lattice



2D lattice with 40000 nodes, starting from a steady state.



Summary

- Evaluating power grid vulnerability
- Second-order Kuramoto model
- Numerical solution using integration schemes
- Invoke line failures to analyse cascade phenomena
- Measure synchronization properties
- Perspectives:
  – Region-specific parameters (e.g. local order parameter)
  – Stability and vulnerability measures
  – Test with real and synthetic power networks
  – Validate the results by comparison with real-world outage statistics