Introducing DyMonDS-as-a-Service (DyMaaS) for Internet of Things

©Marija Ilic, Rupamathi Jaddivada

ilic@mit.edu, rjaddiva@mit.edu

IEEE High Performance Extreme Computing (HPEC) Conference

September 24-26, 2019

EESG @MIT

Acknowledgements

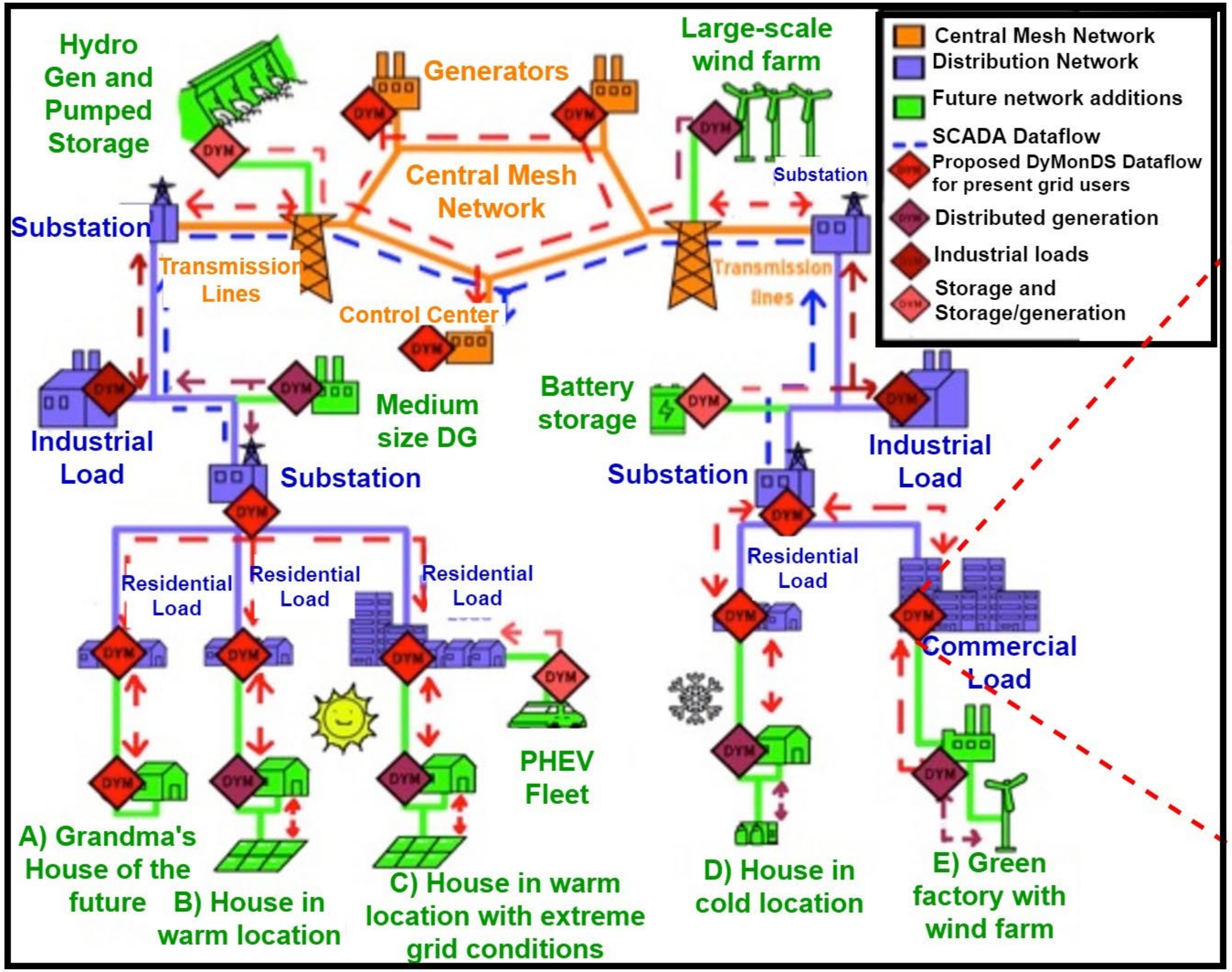
DISTRIBUTION STATEMENT A. Approved for public release. Distribution is unlimited.
This material is based upon work supported by the Department of Energy under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of Energy .

EESG @MIT

Outline

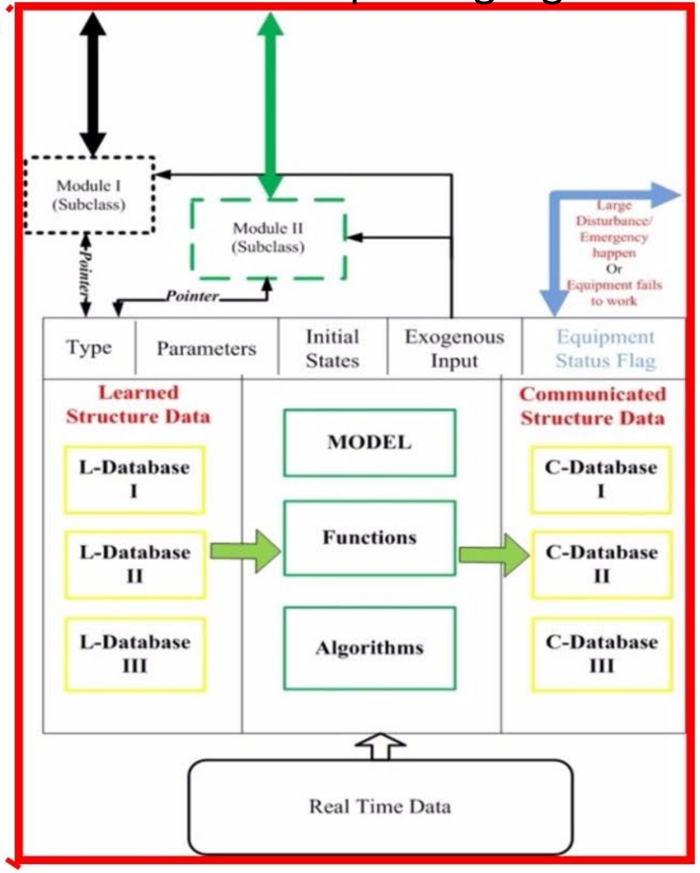
- Emerging physical systems with the advent of IoT
- Toward DyMaaS
- Modeling framework as a basis for iterative co-design
- Physics-based adaptive computer architecture design
- Physics-based multi-rate numerical methods for physical response emulation
  - Intra-processor computations
  - Inter-processor emulation computations
- Case of a microgrid system
- Future work

Emerging systems with the advent of IoT



Extensive data collection and storage exists at the distributed IoT devices already.

Propose to utilize local computations with minimal communication for exploiting algorithm-level parallelism



EESG @MIT

Computational challenges and the need for DyMaaS

- Physical systems involve multi-rate dynamical evolution ranging from microseconds to hours
- Spread across large geographical areas
- Solutions advocated presently **HPECaaS**
  - Increase computational power and communication rate
  - Include hardware implementations such as GPGPUs, etc.

Not scalable to arbitrarily large systems

Quickly gets expensive

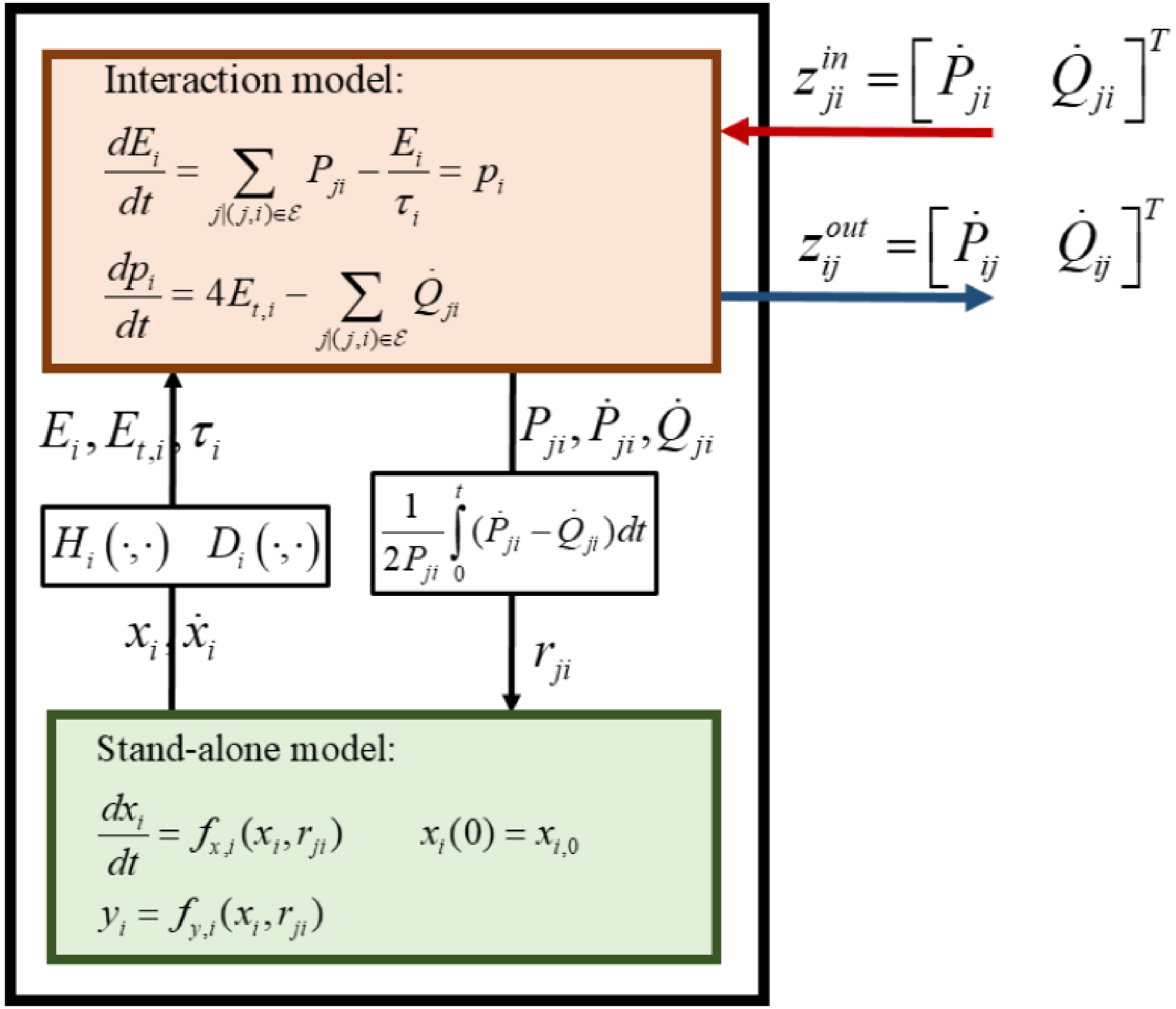
- Above largely make use of data-level and/or task-level parallelism
- We propose to exploit algorithm-level parallelism by understanding the underlying structure --- ***Towards DyMaaS***

EESG @MIT

Modeling framework as the basis for providing DyMaaS

New aggregate variables

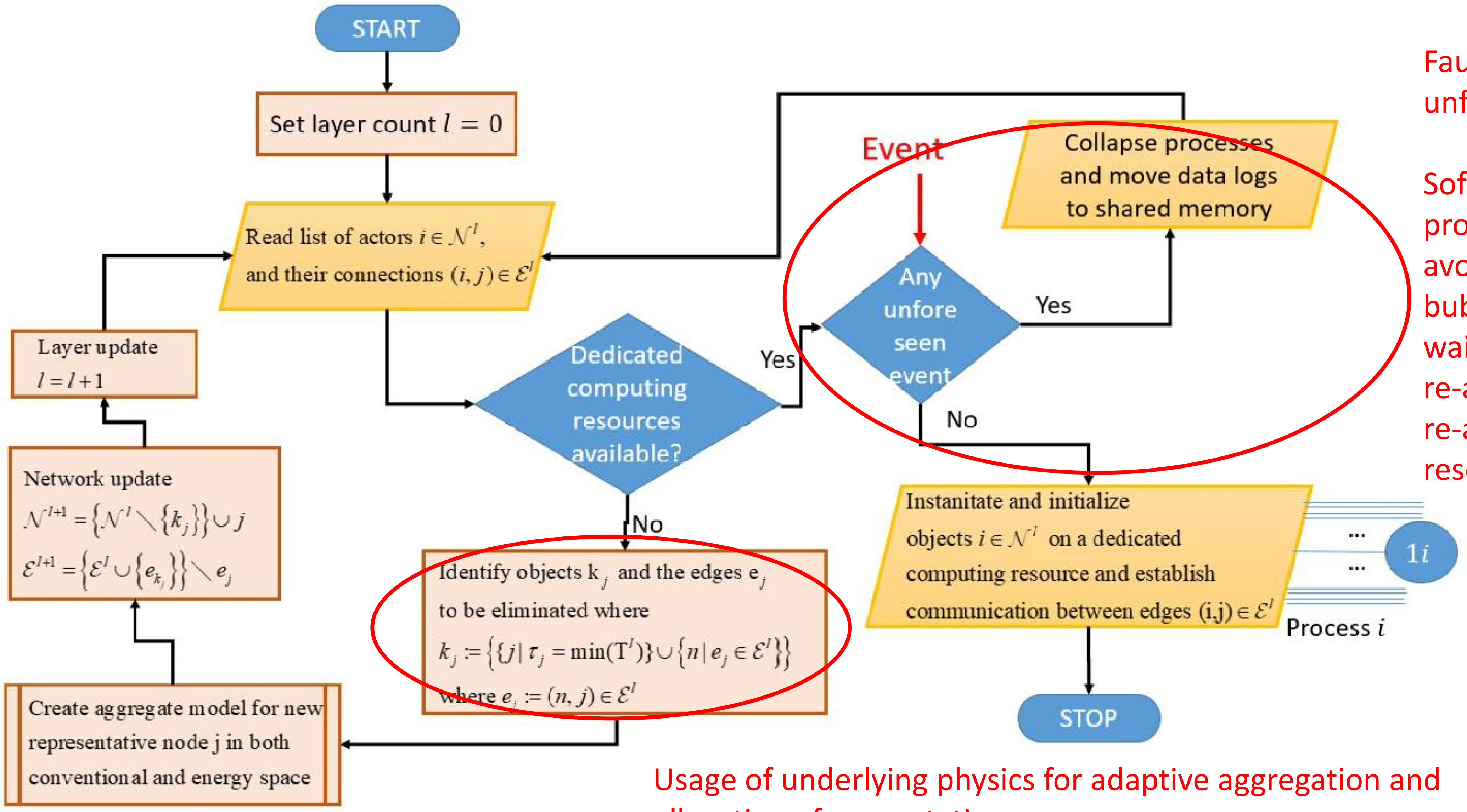
$z_i = [E, P]^T$



- Physics-based unified aggregation variables- *common information model*
- Dictates adaptive communication rates needed – *need not be trial-and-error based scheduling*
- Inherently interactive underlying model facilitates predictive computation– *achieves numerically stable computations*
- Internal information abstracted sufficiently - *decreased security vulnerabilities*

EESG @MIT

Physics-based adaptive design of computing architecture



Fault-tolerant for unforeseen events

Software interrupts programmable to avoid pipeline bubles, excessive wait times through re-aggregation and re-allocation of resources

Usage of underlying physics for adaptive aggregation and allocating of computation resources

EESG @MIT

State-machine implementation of actors

$1i$

State 1:

- **ReceivePortPowerInfo** – Receive the information of Port power interactions
- **ReceiveNextTimeStep –** Receive desirable present time step of communication from neighbors

Input Port power Information

Adaptive time step

Output Port power Information

Input Port power Information

Commonly understood physical variables as the minimum communication needs

State 2:

Sub-state 1:

- Simulate interaction model for present values of input port power
- Compute average output port power values

Stored energy, its rate of change

Stored energy in tangent space

Port inputs in conventional space

Sub-state 2:

- Simulate internal dynamics for present values of port inputs
- Compute fine granularity output port power values

- **SendPortPowerInfo** – After convergence of average of output port power values as computed by sub-sates 1 and 2, send the Port power output variables
- **SendNextTimeStep –** Find the desirable next time step of communication utilizing value of the time constant computed at present operating conditions

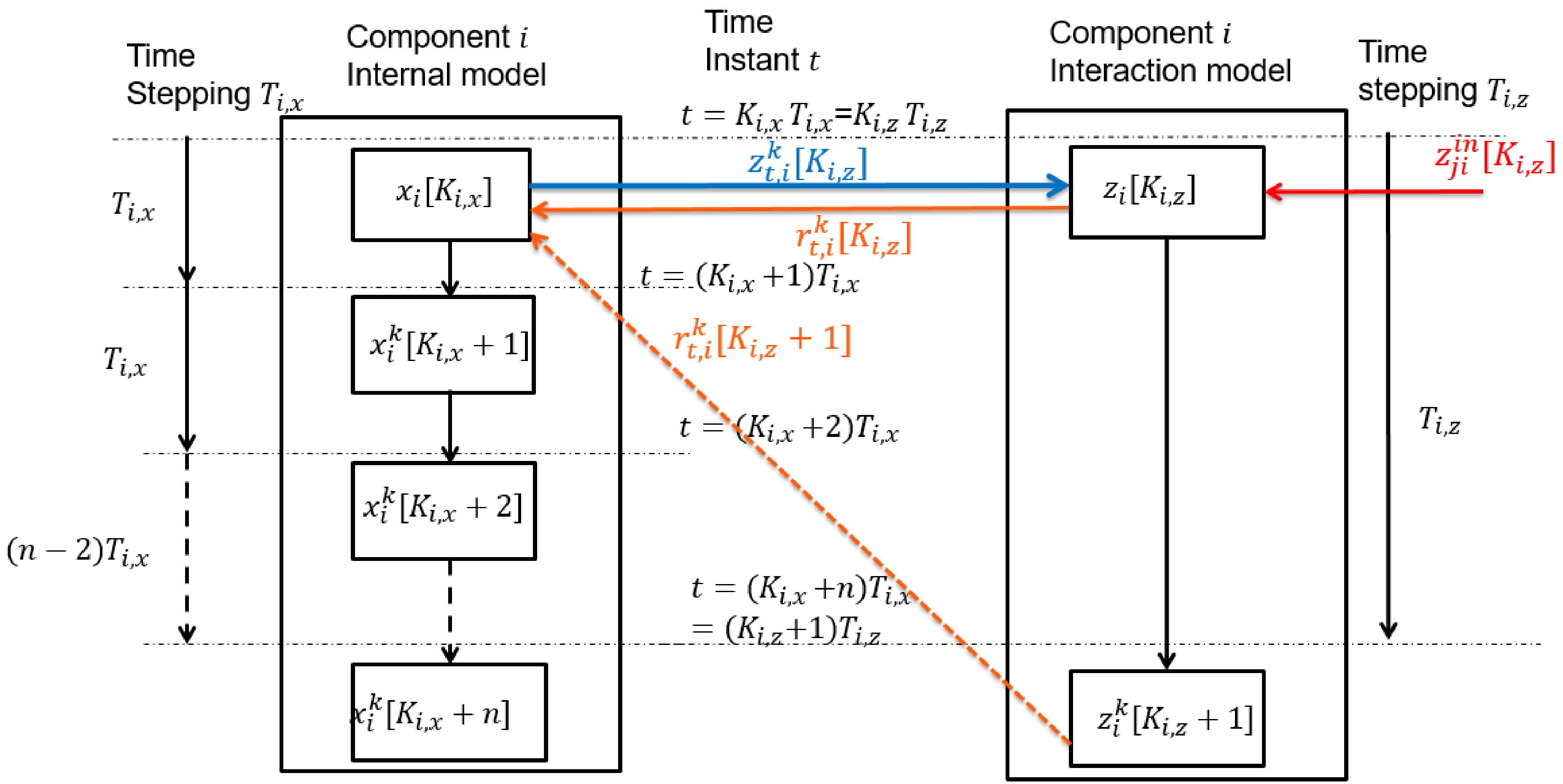
More granular simulations – order is very low,
- Can use existing HPEC methods to exploit data-level and task-level parallelism if needed

Output Port power Information

Adaptive time step

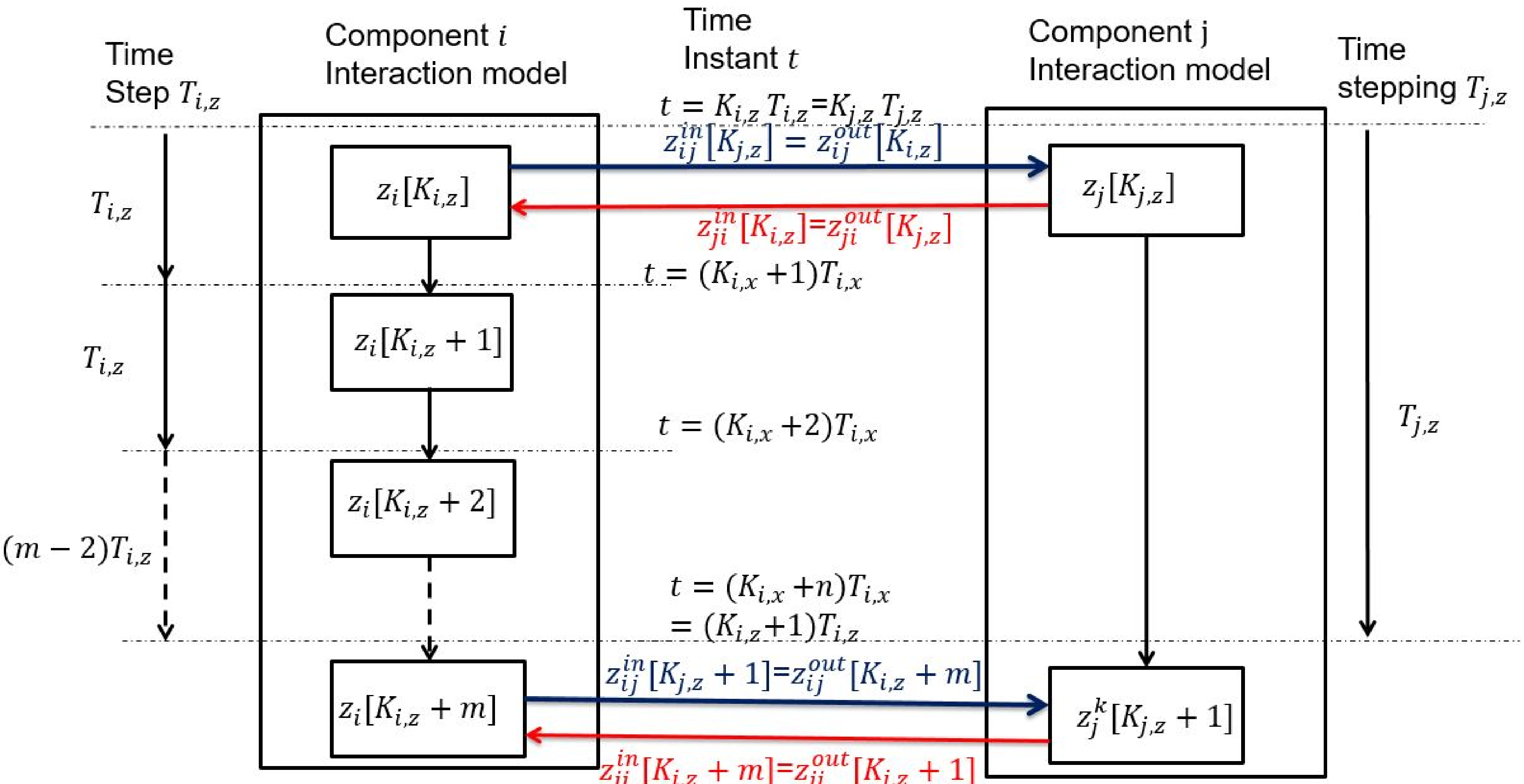
EESG @MIT

Physics-based interactive numerical methods for intra-processor simulations



EESG @MIT

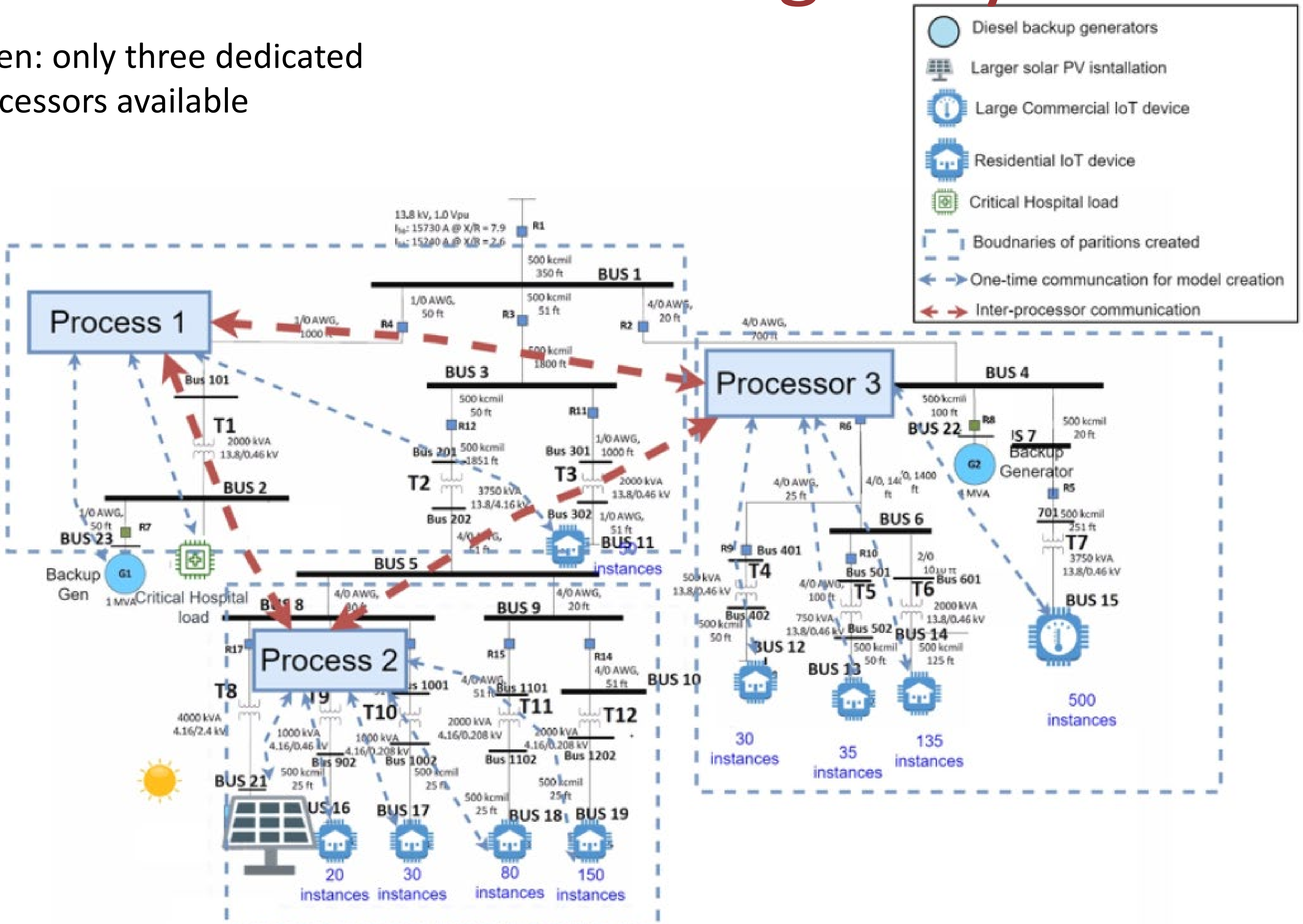
Physics-based interactive numerical methods for inter-processor simulations



EESG @MIT

Physics-based adaptive resource allocation of a microgrid system

Given: only three dedicated processors available



System simulation computation complexity

- 160 state variables without even modeling dynamics of homes.
- Timescales range from microseconds at PV to seconds in the backup diesel generator

Partitions created dynamically using the operating conditions dependent time constants of the aggregation variables

EESG @MIT

Conclusions

- Modeling framework utilized for co-design of computing architecture and interactive numerical schemes to be embedded
- Modeling is cognizant of possible non-determinism and asynchronism of communicated information
- Inherently modular facilitating plug-and-play of devices
- Promising results obtained in the field of control – dual of the numerical problems studied typically
- Extension of the framework to provide grid services is a straight-forward extension being pursued as well
- Effectiveness of the numerical methods and the scalability analysis of the iterative co-design is work in progress