Parallel Simulation

Introduction

The activated modes of ePHASORSIM's parallel feature are determined using the following license keys:

EPHASOR_PARALLELIZE_GENUNITS	ePHASORSIM parallelizes the computation of generator units (GenUnits)
EPHASOR_PARTITION_POWERSYSTEM	ePHASORSIM also parallelizes the nodal equations

Note: If neither of the above license keys is enabled, no parallelization occurs.

The parallelization is controlled using:

- the EPHASOR_THREADS environment variable
- Two parameters in the **Simulation settings** tab of the Simulink mask: **Number of partitions** and **Optimize number of threads (Windows only)**

On Windows, the recommended way to use parallelism in ePHASORSIM is to enable the **Optimize number of threads** algorithm option in the **Solver** ma sk.

It is not recommended to use the EPHASOR_THREADS variable so that the algorithm can find the best setting. Defining EPHASOR_THREADS on Windows restricts the range of number of thread-pool sizes the algorithm will test.

Note: If MATLAB is started in standalone mode (i.e not through RT-LAB), the EPHASOR_THREADS variable potentially defined in the RT-LAB project will not be active, unless this variable is defined explicitly as an environment variable of the machine.

On Linux, by default, ePHASORSIM uses the 3rd core (core No. 2 with zero indexing) unless the EPHASOR_THREADS environment variable is defined. If the performance is not optimal (e.g. if there are overruns) parallelism can be controlled by defining EPHASOR_THREADS to change the number of threads, or to control core assignment to avoid conflicts with other RT-LAB features such as ARTEMiS or some drivers.

Refer to Parallel simulation rules for a more detailed overview of how to set up parallelism in ePHASORSIM.

Parallel Simulation Rules

The following abbreviations are used throughout this guide:

NP	**Number of partitions** parameter, set in the mask.
ONT	**Optimize number of threads (Windows only)** algorithm, set in the mask.
NC	Number of available physical cores of simulator.
	• on Windows: NC = number of exiting CPU cores. • on Linux: NC = min(number of existing CPU cores, value of **EPHASOR_NUM_CORES** in the license)
max NTH	maximum number of threads that can be used for parallelization.

	• if EPHASOR_THREADS is defined: maxNTH = number of cores specified in EPHASOR_THREADS, with an upper limit of NC - 2 • if EPHASOR_THREADS is not defined: maxNTH = 1
NTH	actual number of threads used by ePHASORSIM to parallelize GenUnits (and nodal equations if NP > 1)
	• if ONT is disabled: NTH = min(maxNTH, NC - 2) • if ONT is enabled (Windows only): During the first few steps of the simulation, the optimization algorithm finds the optimal value for NTH within the following range: 1<= NTH <= min(maxNTH, NC - 2)

The first and second table below summarizes calculations of NTH and also parallelization modes for Windows and Linux targets respectively.

Summary of Parallelization Modes for a Windows Target

ONT	Enabled		Disabled	
NP	=1	>1	=1	>1
NTH is:	Calculated optimally in this range: [1,min(maxNTH, NC-2)]		min(maxNTH, NC-2)	
Parallelization Mode is:	GenUnits only	Entire Network	GenUnits only	Entire Network

Summary of Parallelization Modes for a Linux Target

NP	=1	>1
NTH is:	min(maxNTH, NC-2)	
Parallelization Mode is:	GenUnits only	Entire Network

EPHASOR_THREADS Syntax

The syntax for EPHASOR_THREADS is EPHASOR_THREADS=<*core_list*>, where the <*core_list*> grammar is given in the following table.

Syntax Table for the EPHASOR_THREADS Variable

<*core_list*> :=	<*elem*> | <*elem*>,<*core_list*>
<*elem*> :=	<*core_id*> | <*range*>
<*range*> :=	<*core_id*>-<*core_id*>
<*core_id*> :=	<positive_integer>

Its meaning is the same on Windows and Linux, with the exception that:

- On **Linux**, it is used to define which cores specifically will be used in the simulation (sets the affinity).
- On **Windows**, it is used only to deduce the number of cores that the simulation will run on, but cannot assign specific cores.

For example, EPHASOR_THREADS = 3 and EPHASOR_THREADS = 5 mean that we want the simulation to run on only one core. On Linux, it also defines which core should be used (core 3 in the first case, core 5 in the second case).

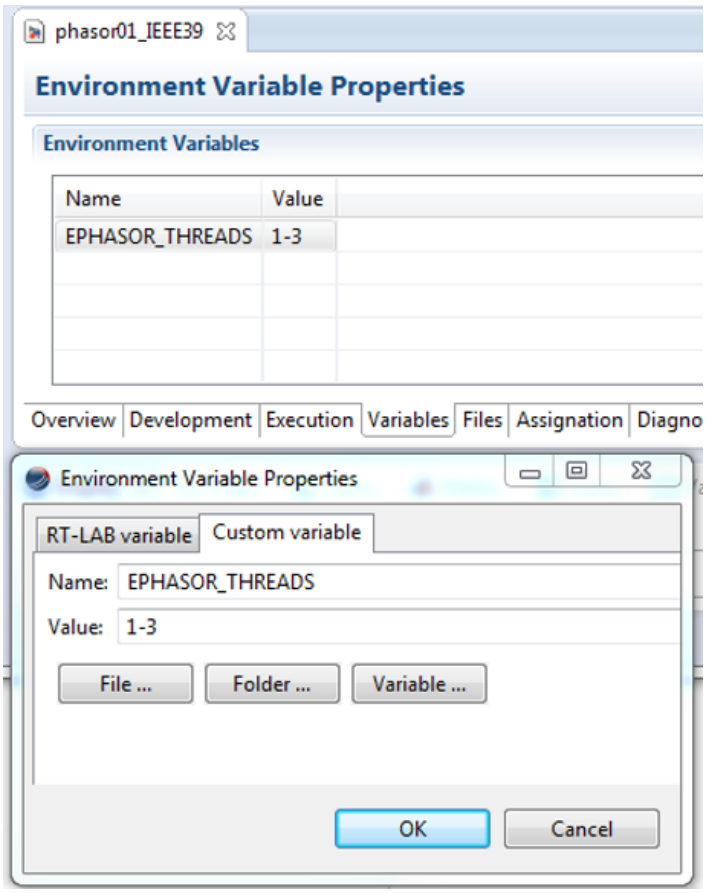
The following examples are all valid values:

EPHASOR_THREADS=2,3,5	in this case, we want to use 3 cores (on Linux, cores 2, 3, and 5)
EPHASOR_THREADS=3-6	this is equivalent to EPHASOR_THREADS=3,4,5,6 meaning we want to use 4 cores (on Linux, cores 3, 4, 5, and 6)
EPHASOR_THREADS=2,4-6	this is equivalent to EPHASOR_THREADS=2,4,5,6
If EPHASOR_THREADS is not defined, the default value is 2 - (NC - 2)	

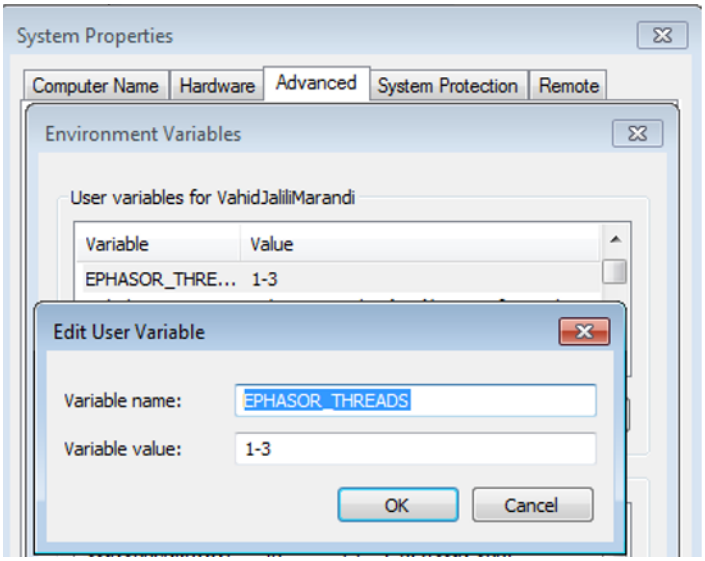
EPHASOR_THREADS Definition

EPHASOR_THREADS may be defined as a user environment variable.

Users can define the variable in an RT-LAB's project configuration, as in the following figure.



When using MATLAB directly, set EPHASOR_THREADS as a system environment variable prior to launching MATLAB. This can be done in System Properties, as in the figure below.



Notes on Linux Targets and EPHASOR_THREADS

- Care must be taken to avoid conflicts in core assignment. For example, if in the model there is a driver that uses core 4, this core should not be used in the value of EPHASOR_THREADS. Otherwise, unknown behavior could occur.
- The core IDs are 0 indexed.
- Cores 0 and 1 are reserved for RT-LAB. Only cores indexed 2 and greater are recommended for use with the EPHASOR_THREADS variable.
- Please refer to, Migration Notes, to see changes regarding to EPHASOR_THREADS in different versions.