

Speeding up Simulink

Murali Yeddanapudi

Agenda

- Typical use cases
- Accelerator mode
- Performance Advisor
- Fast Restart and parsim
- Incremental workflows
- Solver Profiler

Agenda

- Typical use cases
- Accelerator mode
- Performance Advisor
- Fast Restart and parsim
- Incremental workflows
- Solver Profiler

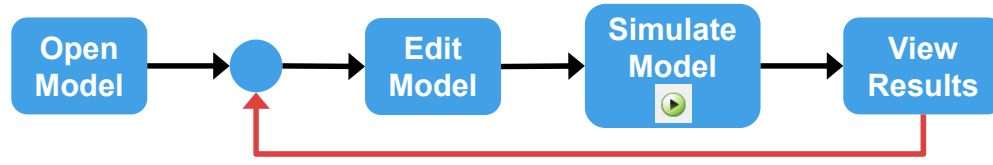
Typical simulation use cases

Edit-Sim-Repeat

Tune-Sim-Repeat

Multi-Sim

Edit-Sim-Repeat



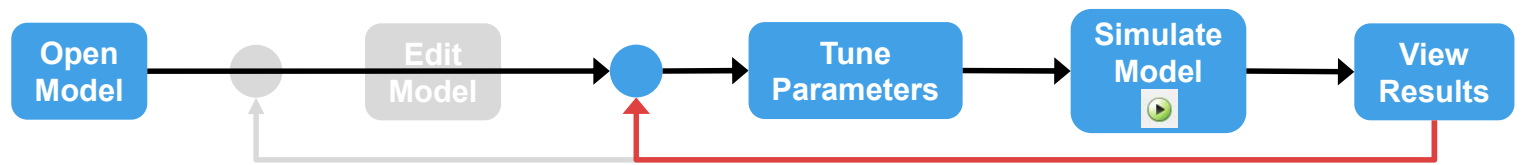
Tune-Sim-Repeat

Multi-Sim

Edit-Sim-Repeat

Tune-Sim-Repeat

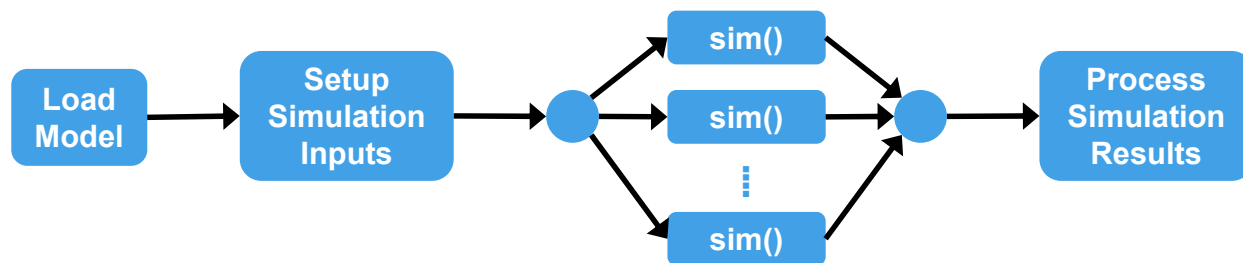
Multi-Sim



Edit-Sim-Repeat

Tune-Sim-Repeat

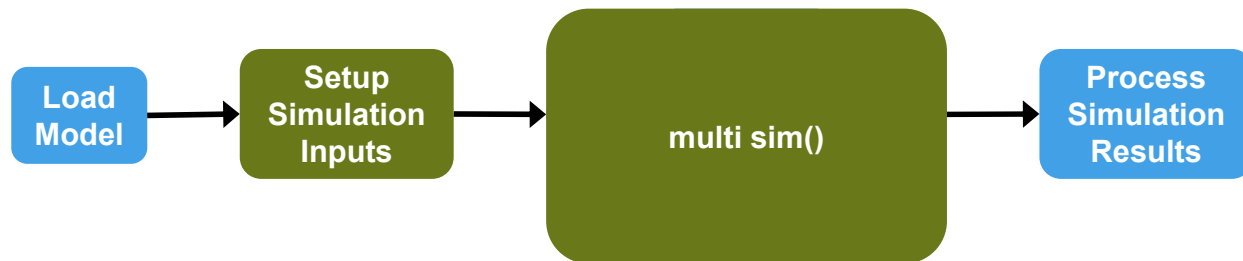
Multi-Sim



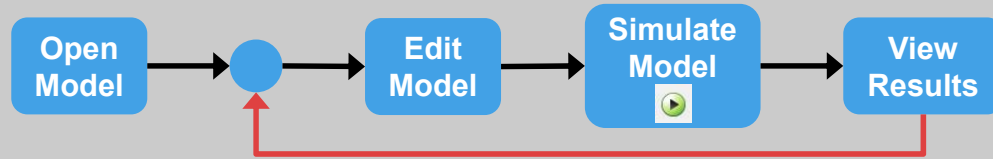
Edit-Sim-Repeat

Tune-Sim-Repeat

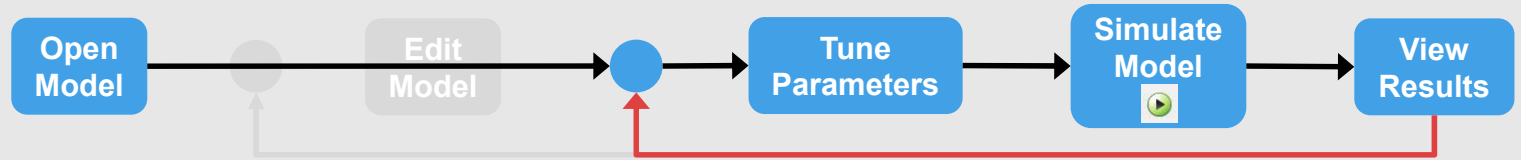
Multi-Sim



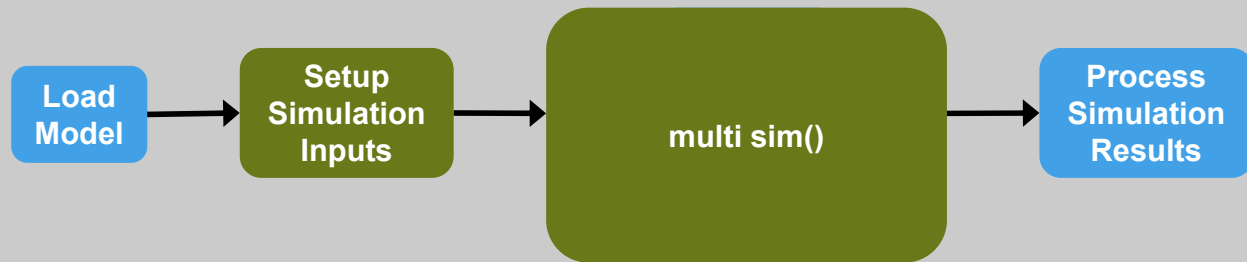
Edit-Sim-Repeat



Tune-Sim-Repeat



Multi-Sim



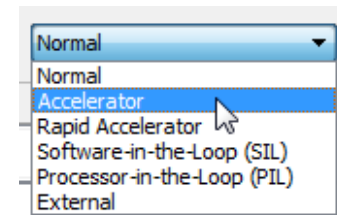
Agenda

- Typical use cases
- Accelerator mode
- Performance Advisor
- Fast Restart and parsim
- Incremental workflows
- Solver Profiler

Accelerator Mode

Why would Simulink speed up?

- JIT compiles (or generates C-code for) portions of the model
- Running compiled code has less overhead



What's the tradeoff?

- There is overhead to generate code
- Some run time diagnostics are disabled, e.g., inf/nan checking
- May not speedup all models

Introduced before **R2006a**

Help Search: how acceleration modes work

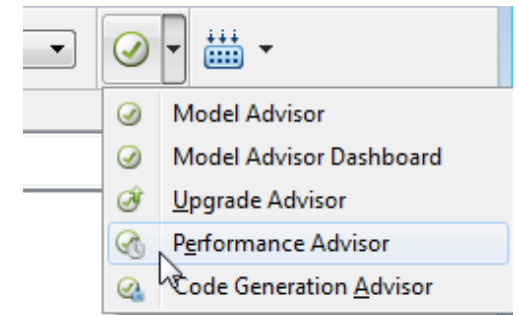
Performance Advisor

Why would Simulink speed up?

- Checks your model for speedup options
- Validates its own advice, only applies changes that:
 - give the same answer
 - and improve speed

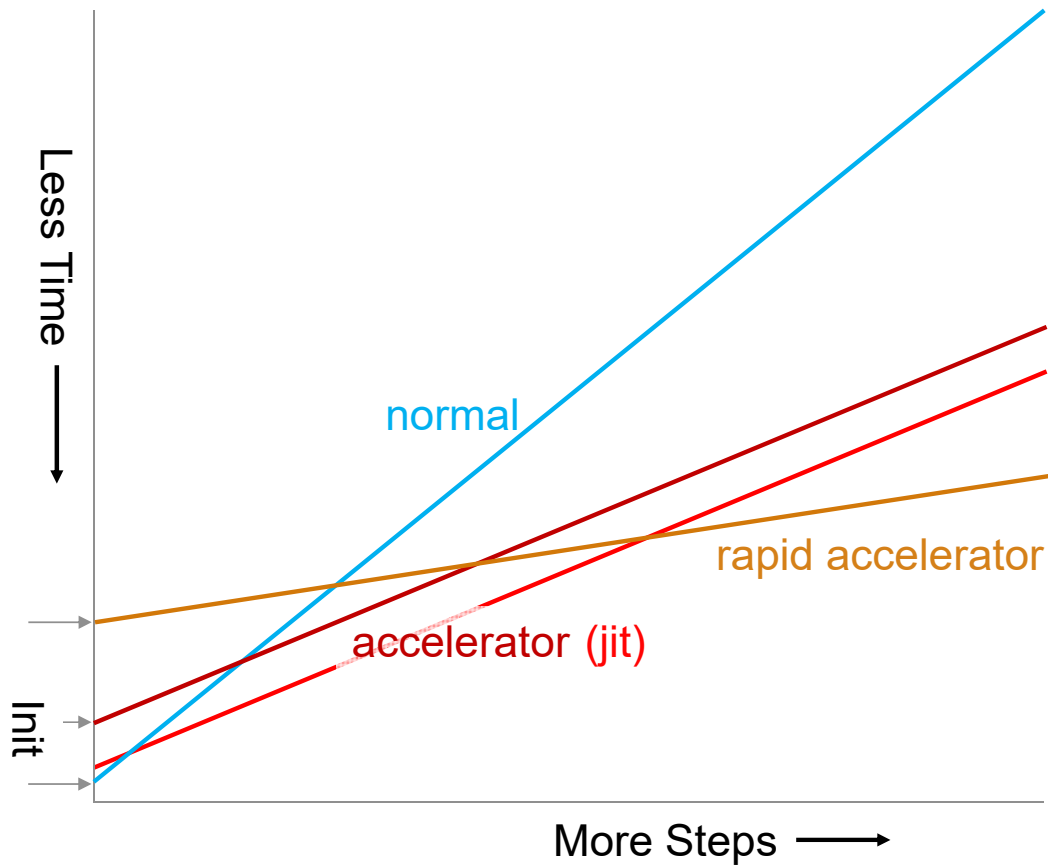
What's the tradeoff?

- Takes time run the analysis
- Not comprehensive
 - Trading off fidelity for speed is not part of performance advisor



Help Search: performance advisor

Rough Comparison of Simulation Modes



Accelerator is faster

- Unless your simulations are short
- With JIT, accelerator is faster than normal mode in many more cases

Rapid-accelerator has the least per-step overhead but the most init overhead

Just-In-Time Accelerator Mode
Introduced in R2016b

Questions

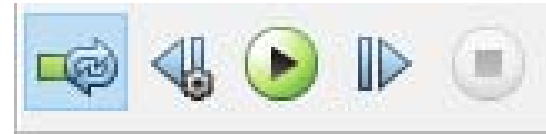
Agenda

- Typical use cases
- Accelerator mode
- Performance Advisor
- **Fast Restart and parsim**
- Incremental workflows
- Solver Profiler

Fast Restart

Why would Simulink speed up?

- Avoids recompile between simulation runs
- Works with Accelerator mode



What's the tradeoff?

- Cannot edit the model when in fast restart mode

Help Search: fast restart

parsim

R2017b
R2017a

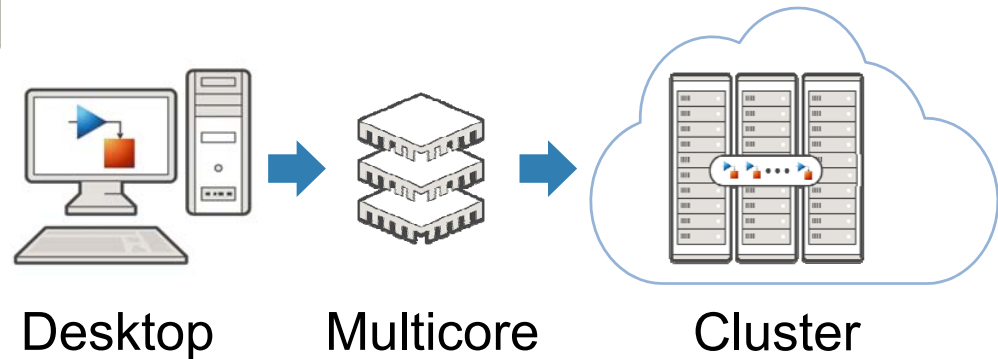
Why would Simulink speed up?

- Runs simulations in parallel using MATLAB Parallel Computing
- Parallelization details are automatically handled
 - if your model works with `sim` ...
... it works with `parsim`

```
for i = 10000:-1:1
    in(i) = Simulink.SimulationInput('my_model');
    in(i) = in(i).setVariable('my_var', i);
end
out = parsim(in);
```

What's the tradeoff?

- Overhead of setting up parallel pool
- Overhead of starting simulations on the workers
- Needs scripting in MATLAB

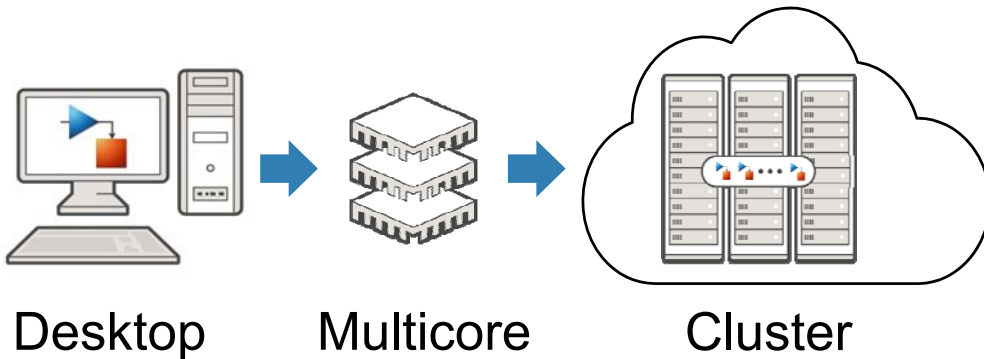


Help Search: [parsim](#)

parsim : Benefits

```

for i = 10000:-1:1
    in(i) = Simulink.SimulationInput('my_model');
    in(i) = in(i).setVariable('my_var', i);
end
out = parsim(in);
    
```

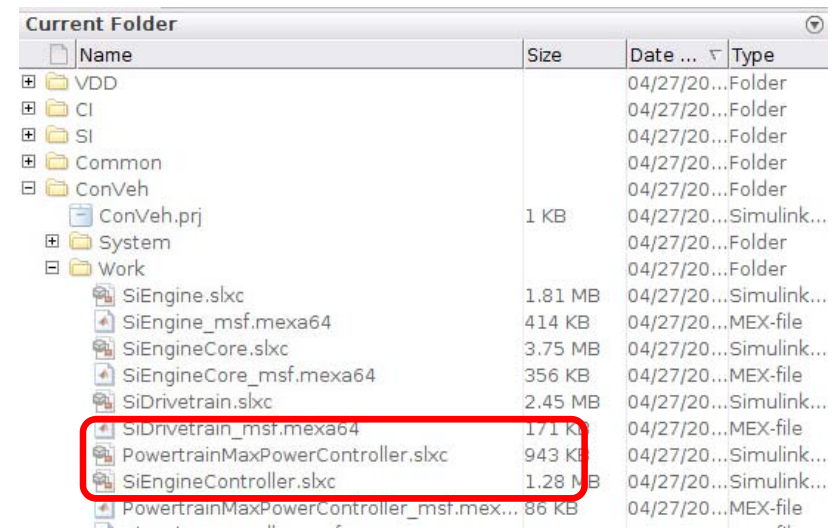


parsim manages the
details of running parallel
simulations

... so you can focus on
the design tasks

parsim: automates book-keeping details (1)

- Handles cross platform details
 - Use parsim from a Windows desktop to run simulations on Linux Cluster
- Handles model dependencies
 - MATLAB Code, Libraries, S-Functions, ...
- Integrated with Simulink Cache
- Leverages model reference parallel build



| Name | Size | Date ... | Type |
|---|---------|-------------|-------------|
| VDD | | 04/27/20... | Folder |
| CI | | 04/27/20... | Folder |
| SI | | 04/27/20... | Folder |
| Common | | 04/27/20... | Folder |
| ConVeh | | 04/27/20... | Folder |
| ConVeh.prj | 1 KB | 04/27/20... | Simulink... |
| System | | 04/27/20... | Folder |
| Work | | 04/27/20... | Folder |
| SiEngine.slxc | 1.81 MB | 04/27/20... | Simulink... |
| SiEngine_msf.mexa64 | 414 KB | 04/27/20... | MEX-file |
| SiEngineCore.slxc | 3.75 MB | 04/27/20... | Simulink... |
| SiEngineCore_msf.mexa64 | 356 KB | 04/27/20... | MEX-file |
| SiDrivetrain.slxc | 2.45 MB | 04/27/20... | Simulink... |
| SiDrivetrain_msf.mexa64 | 171 KB | 04/27/20... | MEX-file |
| PowertrainMaxPowerController.slxc | 943 KB | 04/27/20... | Simulink... |
| SiEngineController.slxc | 1.28 MB | 04/27/20... | Simulink... |
| PowertrainMaxPowerController_msf.mex... | 86 KB | 04/27/20... | MEX-file |

parsim: automates book-keeping details (2)

- Brings back log files from the workers
 - Appends run id to make them unique

| Current Folder | | | | |
|----------------|--------|----------------|----------|--|
| Name | Size | Date Modified | Type | |
| out_1.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_2.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_3.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_4.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_5.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_6.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_7.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_8.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_9.mat | 223 KB | 06/06/2017 ... | MAT-file | |
| out_10.mat | 223 KB | 06/06/2017 ... | MAT-file | |

```
>> out(198)

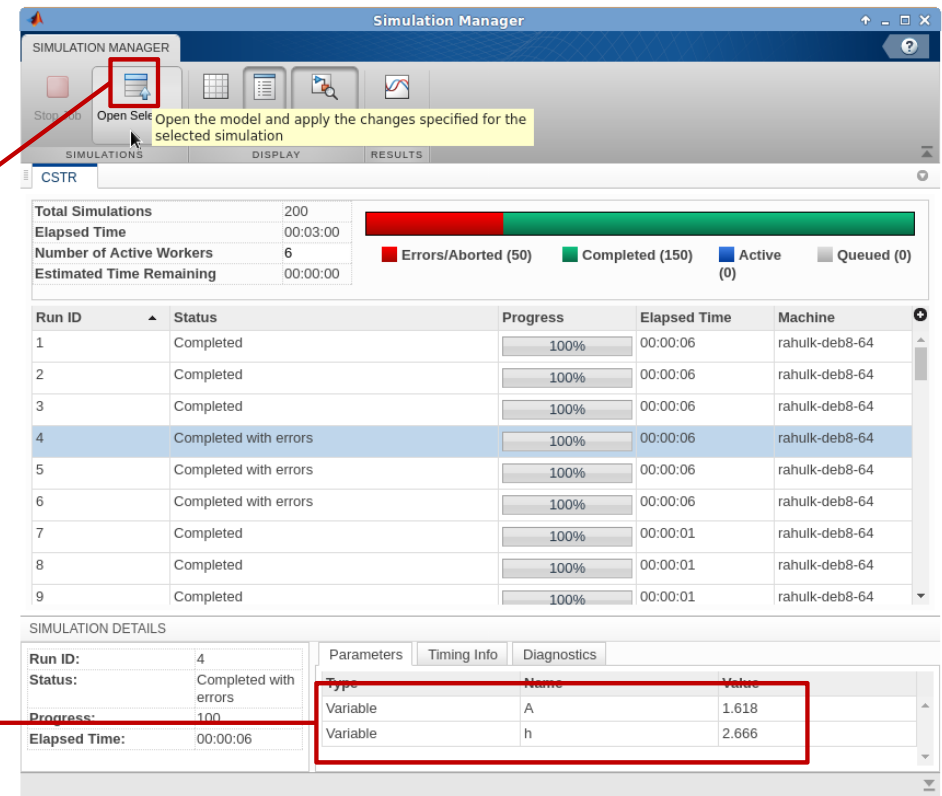
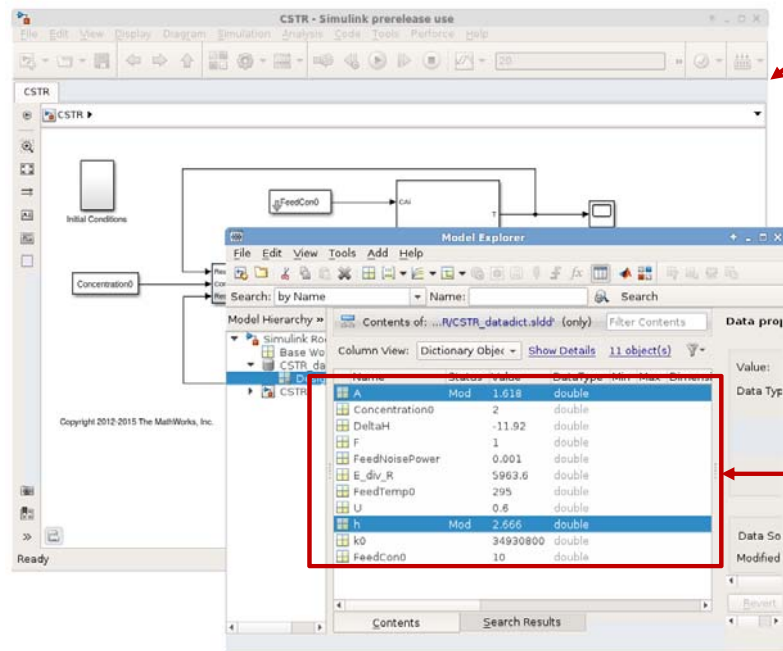
Simulink.SimulationOutput:
    tout: [141565x1 double]
    logout: [1x1 Simulink.SimulationData.DatasetRef]

SimulationMetadata: [1x1 Simulink.SimulationMetadata]
ErrorMessage: [1x15267 char]
```

Automatically get references to logged files

parsim: automates book-keeping details (3)

- Show progress and error diagnostics
 - Setups up model to run locally to debug



Visualizing Results

Simulation Manager

Stop Job Open Selected Grid List Simulation Details Show Results

View results of selected simulation(s) in Simulink Data Inspector

CSTR

Total Simulations: 200
 Elapsed Time: 00:03:00
 Number of Active Workers: 6
 Estimated Time Remaining: 00:00:00

Errors/Aborted (50) Completed (150) Active (0) Queued (0)

| Run ID | Status | Progress | Elapsed Time | Machine |
|--------|-----------------------|----------|--------------|----------------|
| 4 | Completed with errors | 100% | 00:00:06 | rahuik-deb8-b4 |
| 5 | Completed with errors | 100% | 00:00:06 | rahuik-deb8-b4 |
| 6 | Completed with errors | 100% | 00:00:06 | rahuik-deb8-b4 |
| 7 | Completed | 100% | 00:00:01 | rahuik-deb8-b4 |
| 8 | Completed | 100% | 00:00:01 | rahuik-deb8-b4 |
| 9 | Completed | 100% | 00:00:01 | rahuik-deb8-b4 |
| 10 | Completed | 100% | 00:00:01 | rahuik-deb8-b4 |
| 11 | Completed | 100% | 00:00:01 | rahuik-deb8-b4 |
| 12 | Completed | 100% | 00:00:01 | rahuik-deb8-b4 |

SIMULATION DETAILS

Run ID: 8
 Status: Completed
 Progress: 100
 Elapsed Time: 00:00:01

| Type | Name | Value |
|----------|------|-------|
| Variable | A | 0.613 |
| Variable | h | 0.931 |

Simulation Data Inspector - untitled*

DATA INSPECTOR

Inspect Compare

Filter Signals

Run 7: CSTR

- Concentration
- CoolantTemp

Run 8: CSTR

- Concentration
- CoolantTemp

PROPERTIES

| NAME | VALUES |
|-------------|---------------|
| Name | Concentration |
| Line | Concentration |
| Units | |
| Data Type | double |
| Sample Time | Continuous |
| Model | CSTR |
| Block Name | CSTR |
| Block Path | CSTR/CSTR |

Concentration Concentration

3.6
3.3
3.0
2.7
2.4
2.1
1.8

0 2 4 6 8 10 12 14 16 18 20

`parsim` : customization(1)

- `TransferBaseWorkspaceVariables`

```
outs = parsim(inps, 'TransferBaseWorkspaceVariables','on', ...)
```

- `UseFastRestart`

```
outs = parsim(inps, 'UseFastRestart','on', ...)
```

`parsim` : customization(2)

- SetupFcn

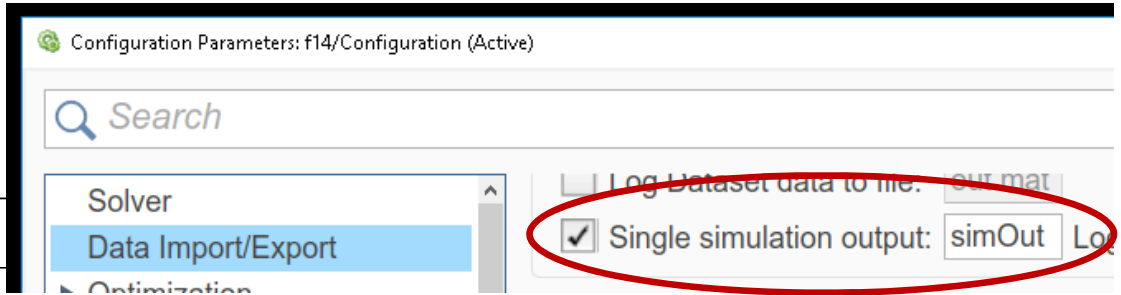
```
setupFcn = @()addpath('myProjectDir')  
outs = parsim(inps, 'SetupFcn',setupFcn, ...)
```

- CleanupFcn

```
cleanupFcn = @()rmpath('myProjectDir')  
outs = parsim(inps, 'CleanupFcn',cleanupFcn, ...)
```


SimulationOutput object

```
simOut = sim('model',
```



```
Trial>> simOut
```

```
Simulink.SimulationOutput:
```

```
ScopeData1: [1x1 Simulink.SimulationData.Dataset]
ScopeData2: [1x1 struct]
    tout: [1353x1 double]
    xout: [1x1 struct]
    yout: [1x1 struct]
```

```
SimulationMetadata: [1x1 Simulink.SimulationMetadata]
ErrorMessage: [0x0 char]
```

- Contains all logged simulation data
- Use dot notation to access the data
- Introduced in R2009a

SimulationInput object

A SimulationInput object '`simInp`' encapsulates all input to one simulation

```
simOut = sim(simInp)
```

Array of `simInps` encapsulate all inputs to multiple simulations

```
simOuts = sim(simInps)
```

* Simulations are run *sequentially*

```
simOuts = parsim(simInps)
```

Simulations are run in **parallel** if MATLAB parallel computing tools are available, *serially* otherwise

SimulationInput Object

SimulationInput with properties:

```

    ModelName: 'sldemo_suspn_3d
    InitialState: [0x0 Simulink.Si
    ExternalInput: []
    ModelParameters: [1x1 Simulink.Si
    BlockParameters: [0x0 Simulink.Si
    Variables: [1x2 Simulink.Si
    PreSimFcn: []
    PostSimFcn: []
    UserString: ''
  
```

Specify MATLAB functions to run before and after each simulation

Use the UserString property to add a brief UserString describing these changes for easy reference

as external messages

Change variables in base workspace, data dictionary, or model workspace

PreSimFcn

- Use PreSimFcn to offload parameter computations to parallel workers

```
for i = 10:-1:1
    in(i) = Simulink.SimulationInput(i);
    in(i).PreSimFcn = @(inp) myPreSimFcn(inp, i);
end
```

```
function simInp = myPreSimFcn(rawSimInp, runId)
    prmValue = expensiveComputation(runId);
    simInp = rawSimInp.setBlockParameter( ...
        [rawSimInp.ModelName, '/my_block'], 'prmName', prmValue);
end
```

PostSimFcn

- use PostSimFcn to post-process raw simulation outputs in parallel
- reduce data returned back from workers

```
>> inps = Simulink.SimulationInput('myModel');  
>> ...  
>> inps.PostSimFcn = @(out) myPostSimFcn(out);  
>> outs = parsim(inps);  
>> outs(i).result
```

```
function simOut = myPostSimFcn(rawSimOut)  
    simOut.result = expensivePostProc(rawSimOut.lotsOfLogsOut);  
end
```

Questions

Agenda

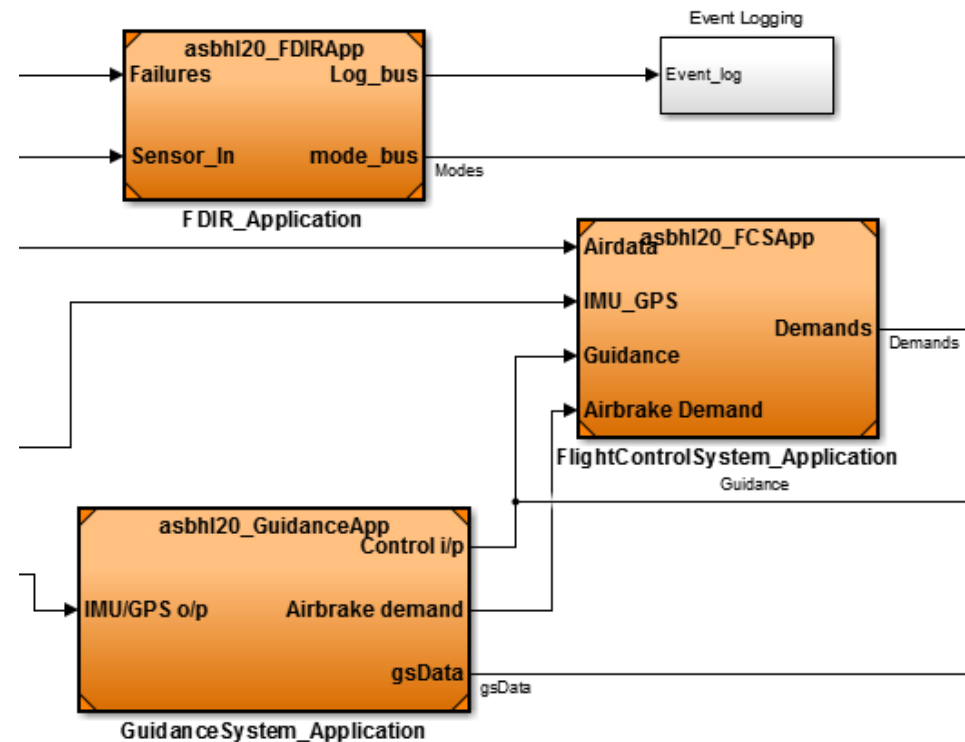
- Typical use cases
- Accelerator mode
- Performance Advisor
- Fast Restart and parsim
- **Incremental workflows**
- Solver Profiler

What is an incremental workflow?

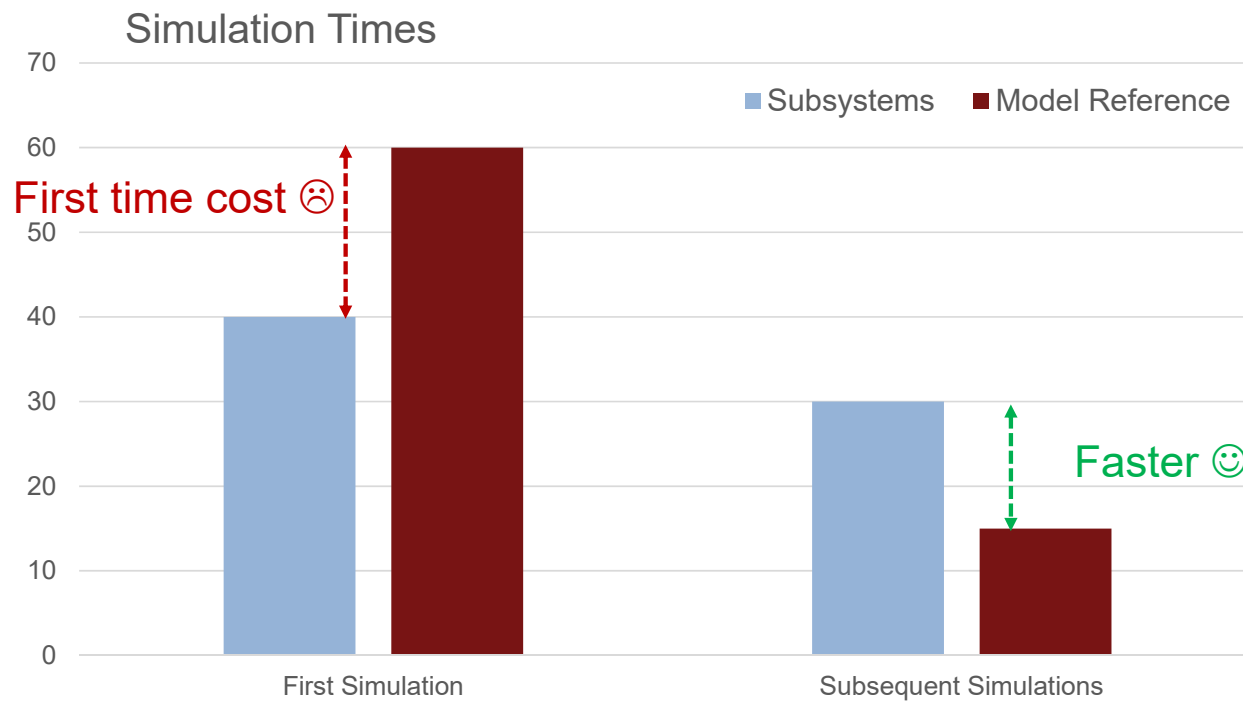
Only perform an action when necessary;
reuse and cache as much as possible

Model reference: incremental workflows

- Incremental Loading
- Incremental Update Diagram
- Incremental Code Generation
- Selective acceleration

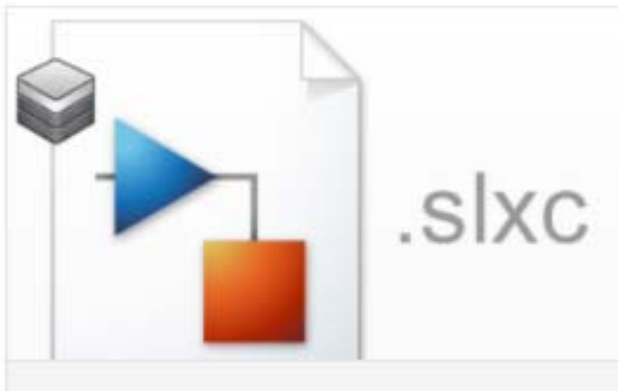


Model Reference: Performance

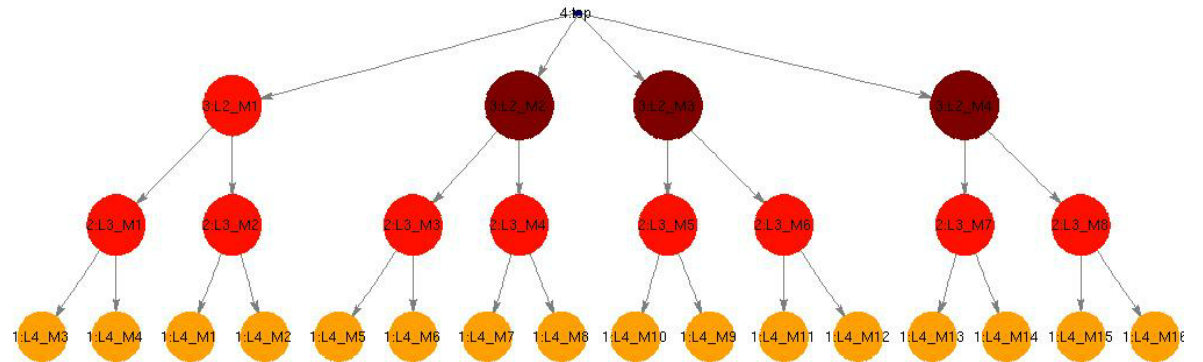


How to reduce first time cost?

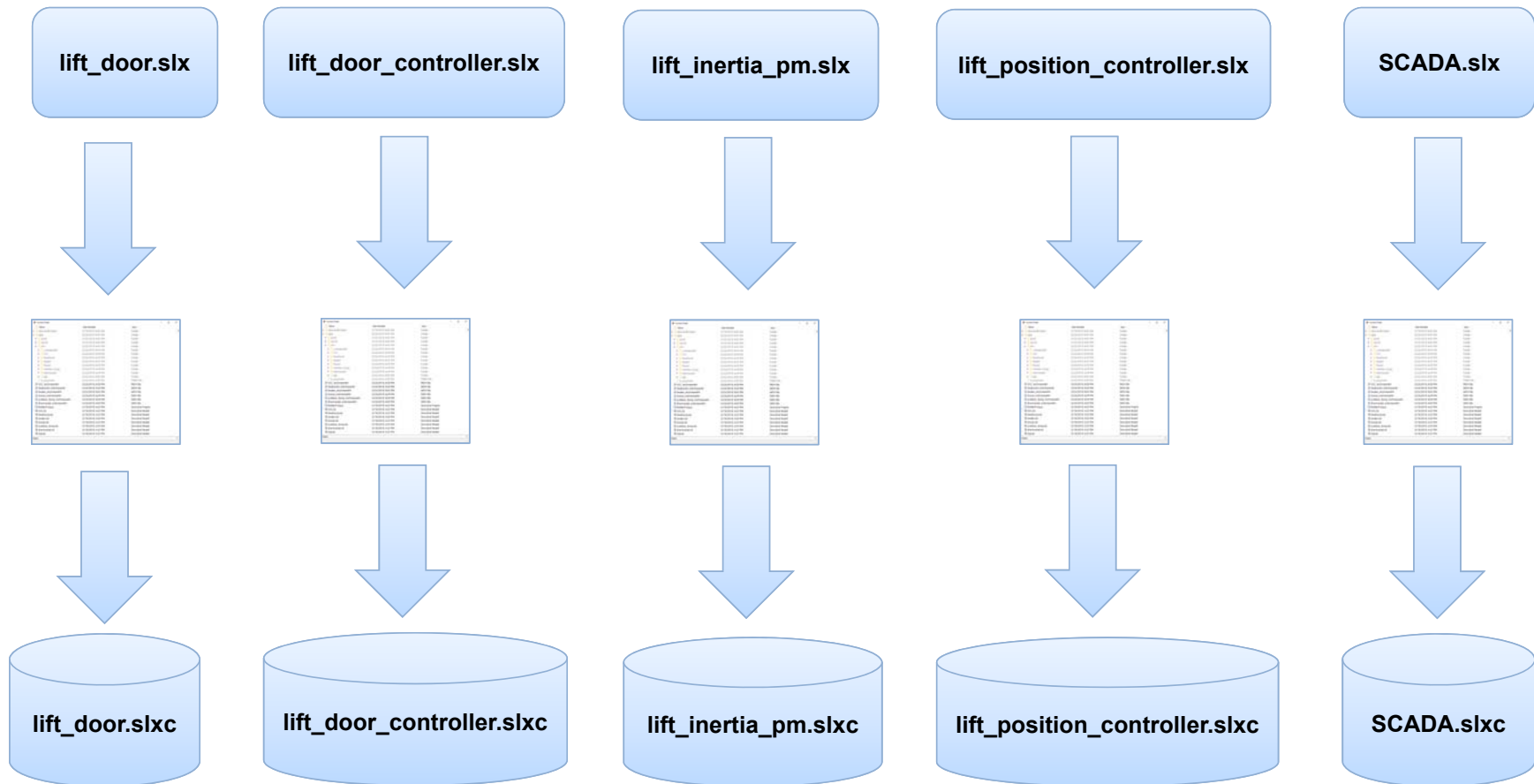
Simulink Cache



Parallel Model Reference Build



Simulink Cache



Simulink Cache

Current Folder

| Name | Type |
|---------------------------------|--------------------------|
| Simulink Data Dictionary | |
| lift_doors.sldd | Simulink Data Dictionary |
| lift_inertia.sldd | Simulink Data Dictionary |
| Simulink Library | |
| generic_motor.slx | Simulink Library |
| lift_inertia_utils.slx | Simulink Library |
| Simulink Model | |
| lift_door.slx | Simulink Model |
| lift_door_controller.slx | Simulink Model |
| lift_inertia.slx | Simulink Model |
| lift_inertia_pm.slx | Simulink Model |
| lift_position_controller.slx | Simulink Model |
| lift_system.slx | Simulink Model |
| SCADA.slx | Simulink Model |
| Simulink cache | |
| lift_door.slxc | Simulink cache |
| lift_door_controller.slxc | Simulink cache |
| lift_inertia_pm.slxc | Simulink cache |
| lift_position_controller.slxc | Simulink cache |
| SCADA.slxc | Simulink cache |

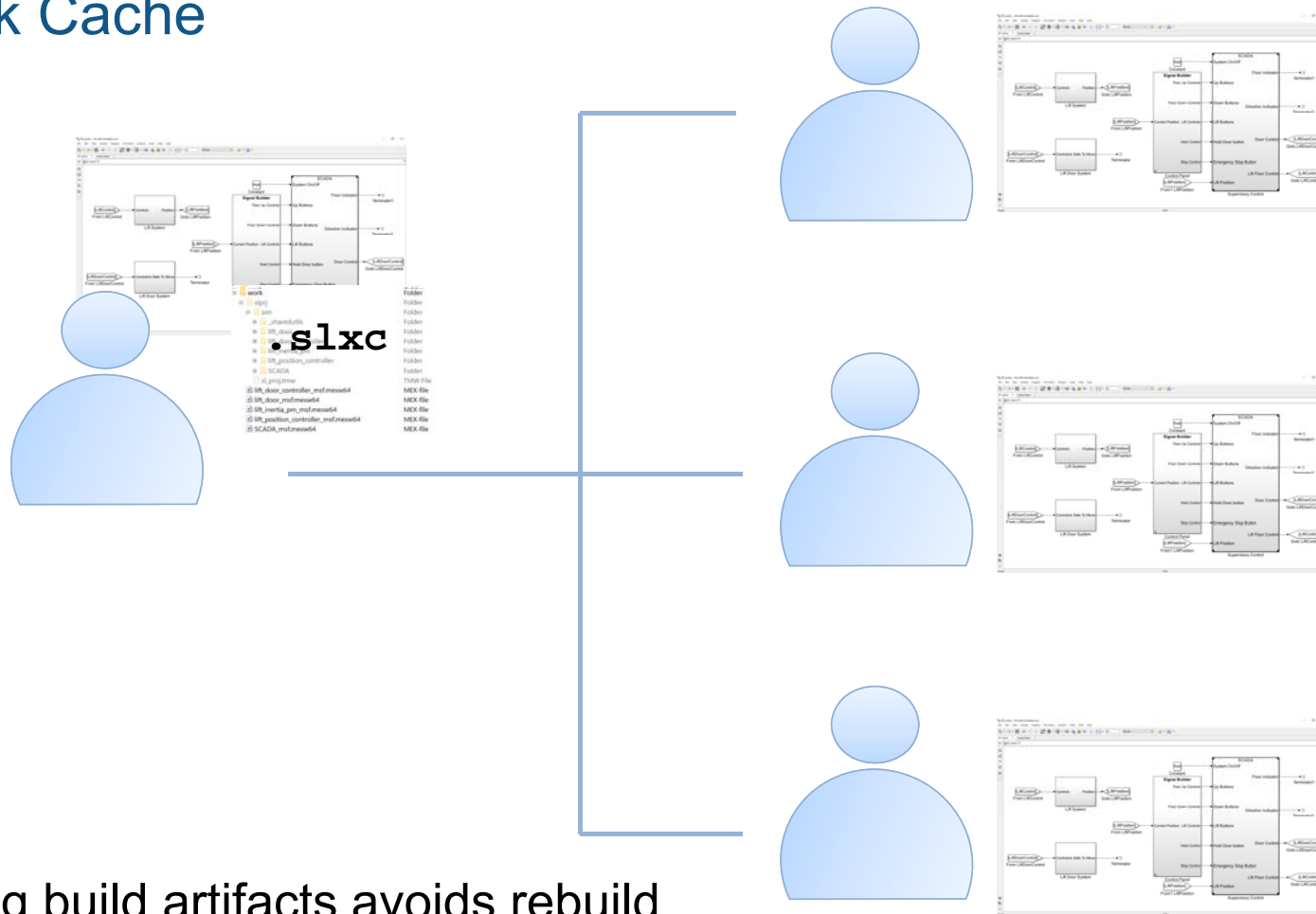
extract

repackage

Current Folder

| Name | Type |
|---------------------------------|--------------------------|
| Folder | |
| slprj | Folder |
| MEX-file | |
| lift_door_controller_msf... | MEX-file |
| lift_door_msf.mexw64 | MEX-file |
| lift_inertia_pm_msf.mex... | MEX-file |
| lift_position_controller_... | MEX-file |
| SCADA_msf.mexw64 | MEX-file |
| Simulink Data Dictionary | |
| lift_doors.sldd | Simulink Data Dictionary |
| lift_inertia.sldd | Simulink Data Dictionary |
| Simulink Library | |
| generic_motor.slx | Simulink Library |
| lift_inertia_utils.slx | Simulink Library |
| Simulink Model | |
| lift_door.slx | Simulink Model |
| lift_door_controller.slx | Simulink Model |
| lift_inertia.slx | Simulink Model |
| lift_inertia_pm.slx | Simulink Model |
| lift_position_controller.slx | Simulink Model |
| lift_system.slx | Simulink Model |
| SCADA.slx | Simulink Model |
| Simulink cache | |
| lift_door.slxc | Simulink cache |
| lift_door_controller.slxc | Simulink cache |
| lift_inertia_pm.slxc | Simulink cache |
| lift_position_controller.slxc | Simulink cache |
| SCADA.slxc | Simulink cache |

Simulink Cache



- Sharing build artifacts avoids rebuild

Simulink Cache

Why would Simulink speed up?

- Sharing build artifacts reduces first time cost
- Integrated into Simulink Projects and parsim

What's the tradeoff?

- Extra work needed to manage .slxc files
 - If Simulink Projects is not used

Simulink Cache

Get simulation results faster by using shared model artifacts

Managing Projects

Develop and share applications as code, executables, or software components.

Help Search: simulink cache

Parallel Model Reference Builds

Configuration Parameters: sldemo_mdref_basic/Configuration (Active)

- Solver
- Data Import/Export
- ▶ Optimization
- ▶ Diagnostics
- Hardware Implementation
- Model Referencing
- Simulation Target
- ▶ Code Generation
- ▶ Coverage

Options for all referenced models

Rebuild:

Parallel

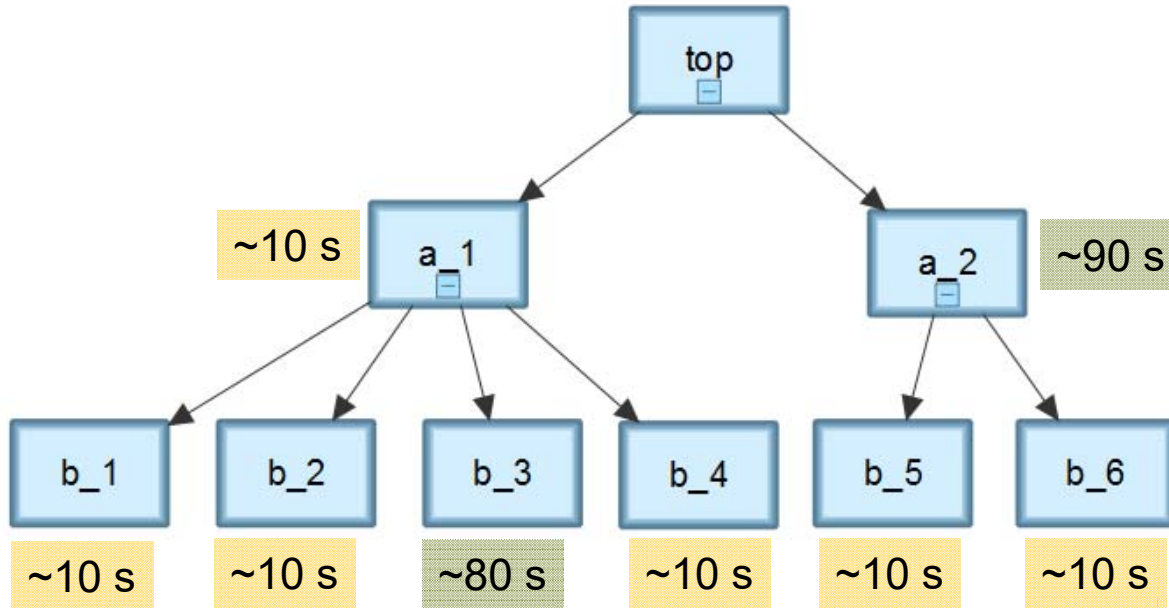
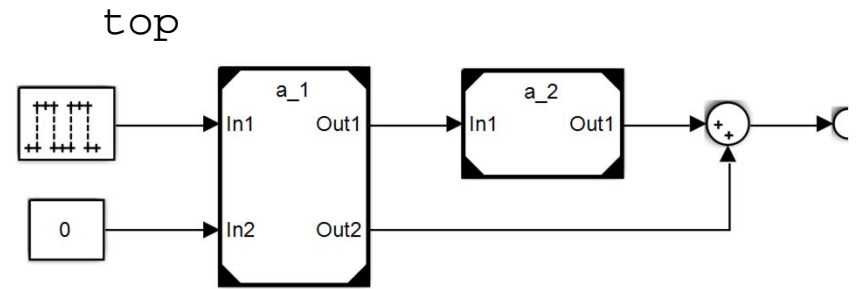
- Enable parallel model reference builds

MATLAB worker initialization for builds:

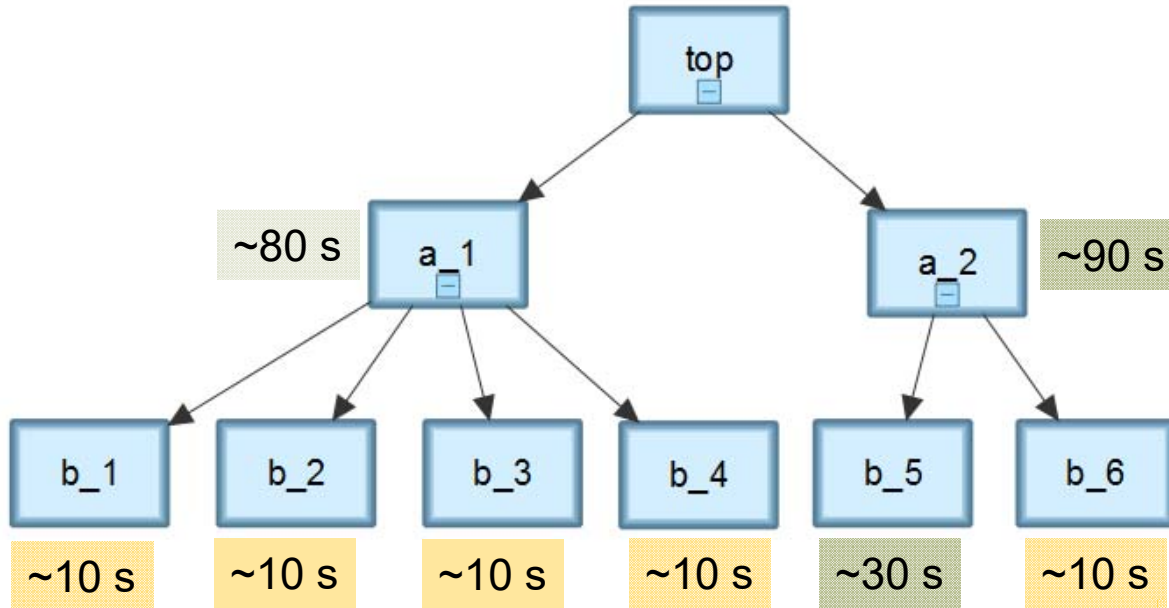
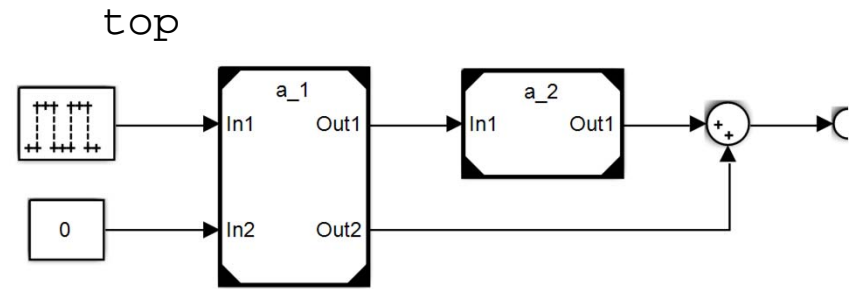
Enable strict scheduling checks for referenced

40

Parallel Model Reference Builds



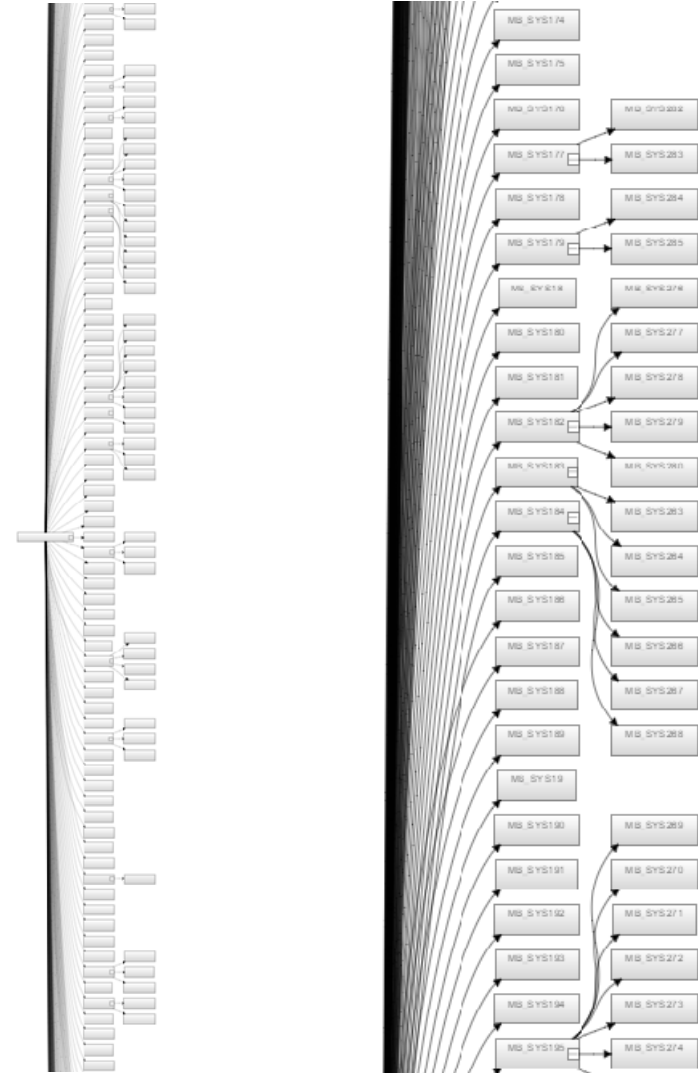
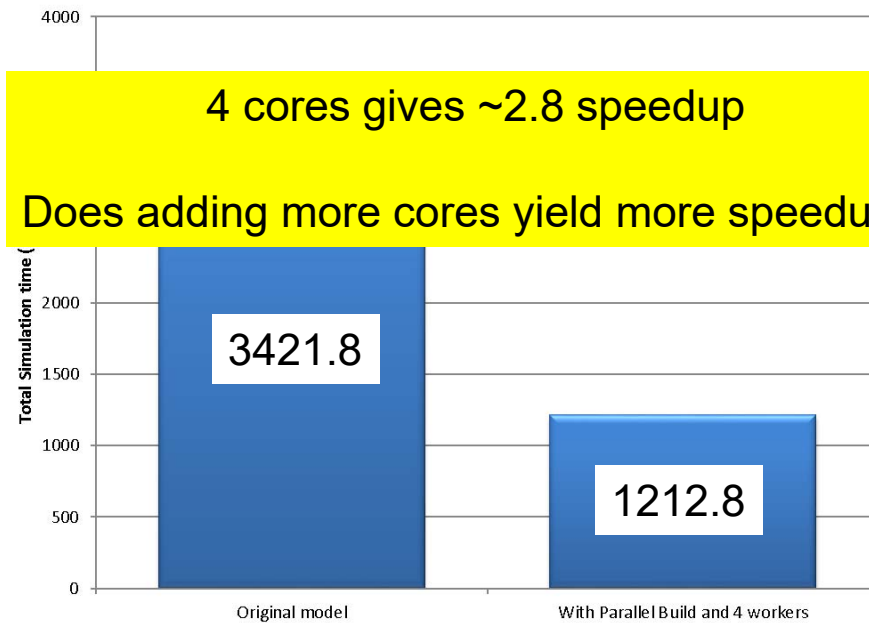
Parallel Model Reference Builds



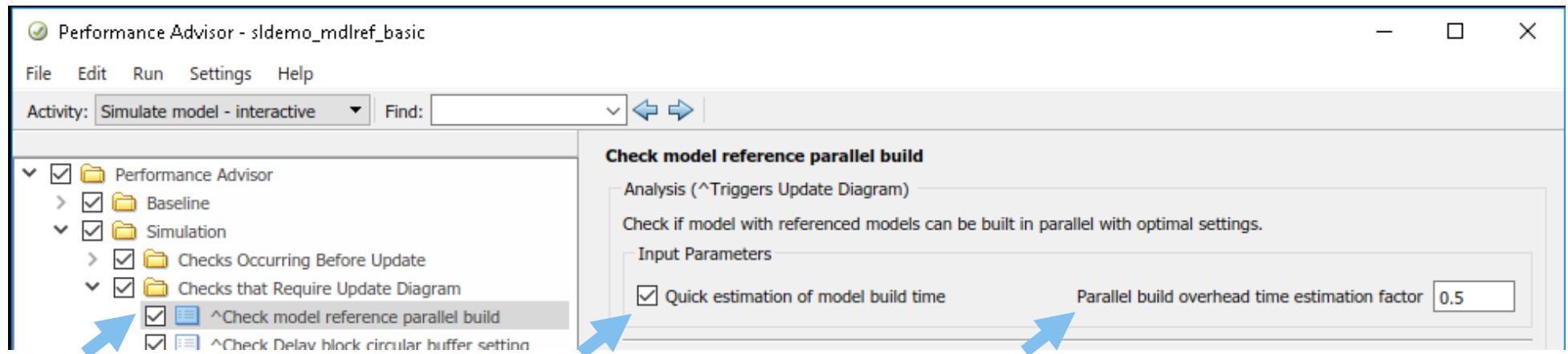
Model Reference Parallel Build User example

- Approximately 400 referenced models

Model Update Time comparison of first-time build with and without PCT



Performance Advisor: Check model reference parallel build



The screenshot shows the Performance Advisor window for a simulation. The left sidebar contains a tree view of checks, with the following items checked:

- Performance Advisor
 - Baseline
 - Simulation
 - Checks Occurring Before Update
 - Checks that Require Update Diagram
 - ^Check model reference parallel build** (highlighted with a blue arrow)
 - ^Check Delay block circular buffer setting

The main panel displays the configuration for the selected check, **Check model reference parallel build**. It includes the following settings:

- Analysis (^Triggers Update Diagram)**: Check if model with referenced models can be built in parallel with optimal settings.
- Input Parameters**:
 - Quick estimation of model build time
 - Parallel build overhead time estimation factor: (highlighted with a blue arrow)

Blue arrows point to the check name in the sidebar, the check's title in the main panel, and the overhead time estimation factor input field.

Performance Advisor: Check model reference parallel build

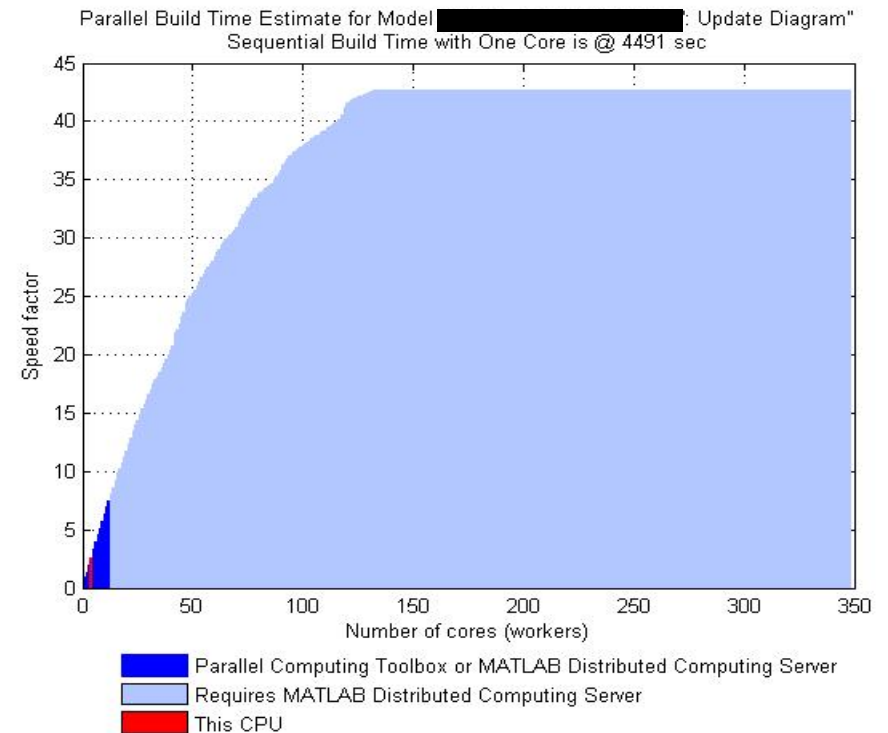
- Performance Advisor estimates the speedup with more cores
- The estimated speed up with 4 cores is ~2.6
 - Close to the measured value ~2.8
- Given ~120 cores, the estimated speed up is ~42
 - => Build time goes from ~3400s to ~80s

Analysis and Advice:

Estimate build time speedup using number of 4 cores of this CPU: ---- **--2.6268x**

Estimate build time speedup using 348 workers: ---- **--42.6475x**

Estimated build times for various worker counts are as follows:



Model Reference Parallel Build

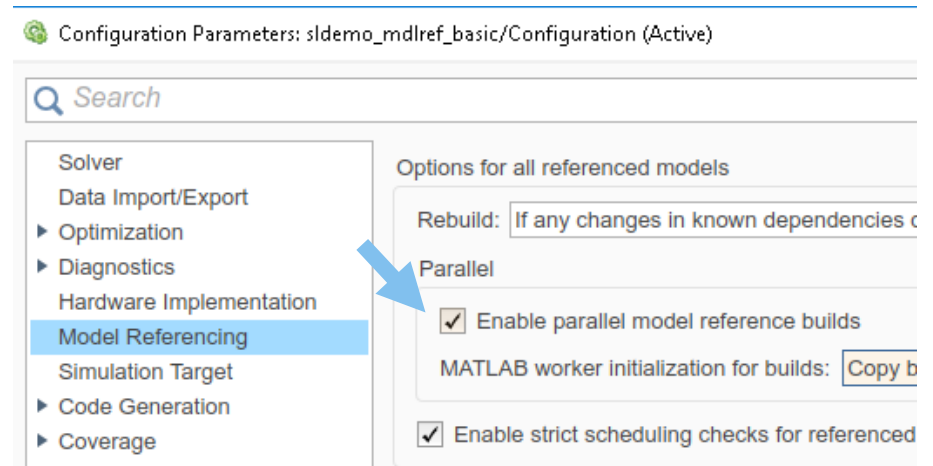
Why would Simulink speed up?

- Model reference targets are built in parallel
- Use Performance Advisor to check if your large models can benefit from this option

What's the tradeoff?

- Speedup is model dependent
- Requires MATLAB Parallel Computing

Help Search: model reference parallel build



Questions

Agenda

- Typical use cases
- Accelerator mode
- Performance Advisor
- Fast Restart and parsim
- Incremental workflows
- Solver Profiler

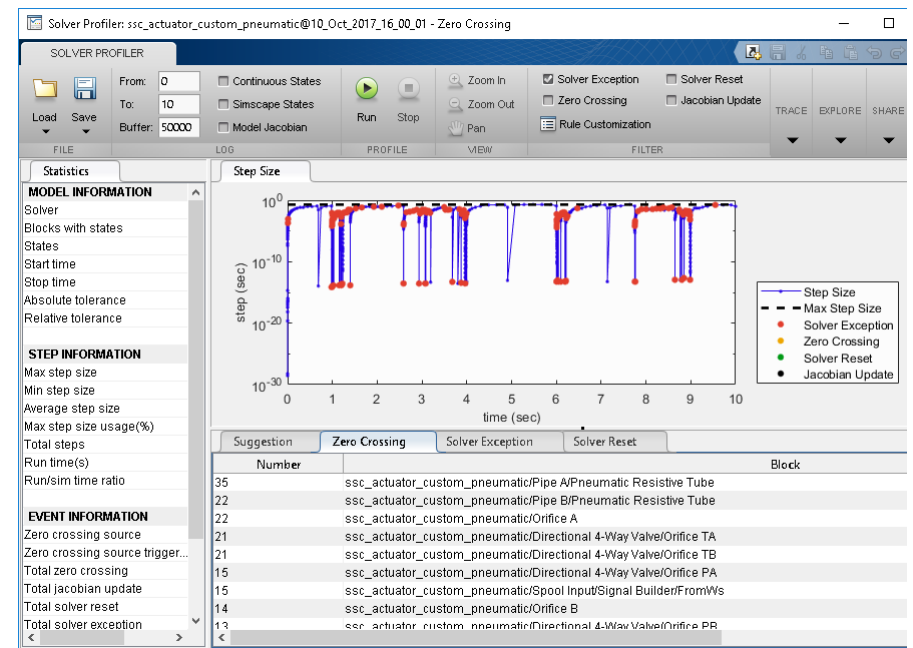
Solver Profiler

Why would Simulink speed up?

- Identifies parts of the model causing solver to slow down
 - too many resets
 - too many zero crossings etc.

What's the tradeoff?

- Profiling overhead
- Requires domain knowledge to optimally fix the issues identified by the Solver profiler.



Help Search: solver profiler

Summary

Use Case

Recommended Features

- Edit-Sim-Repeat

- Normal mode
- Accelerator mode

Smart Editing

Model Reference

- Tune-Sim-Repeat

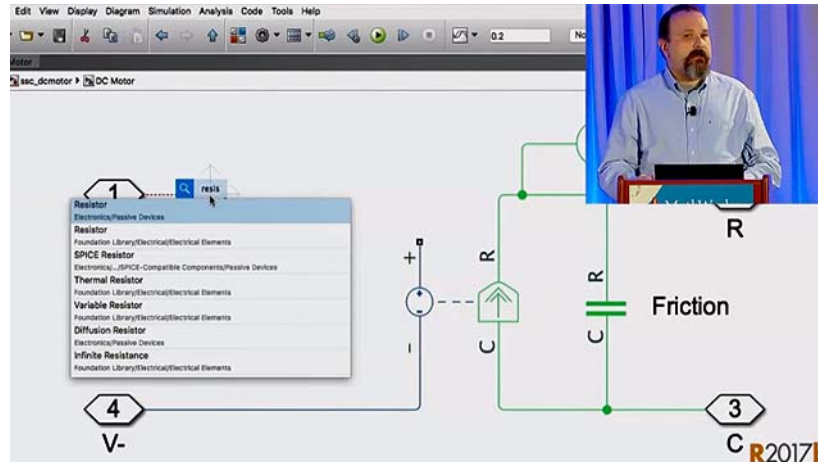
- Performance Advisor
- Fast Restart
- Accelerator + Fast Restart

Simulation Data Inspector

Solver Profiler

- Multi-Sim

- parsim
- parsim + Accelerator + Fast Restart
- parsim + Rapid Accelerator + Up-To-Date-Check-Off



[Editing at the Speed of Thought with Simulink](#)

Learn about the latest smart editing features that have been added to Simulink to increase your modeling speed.

Thank you