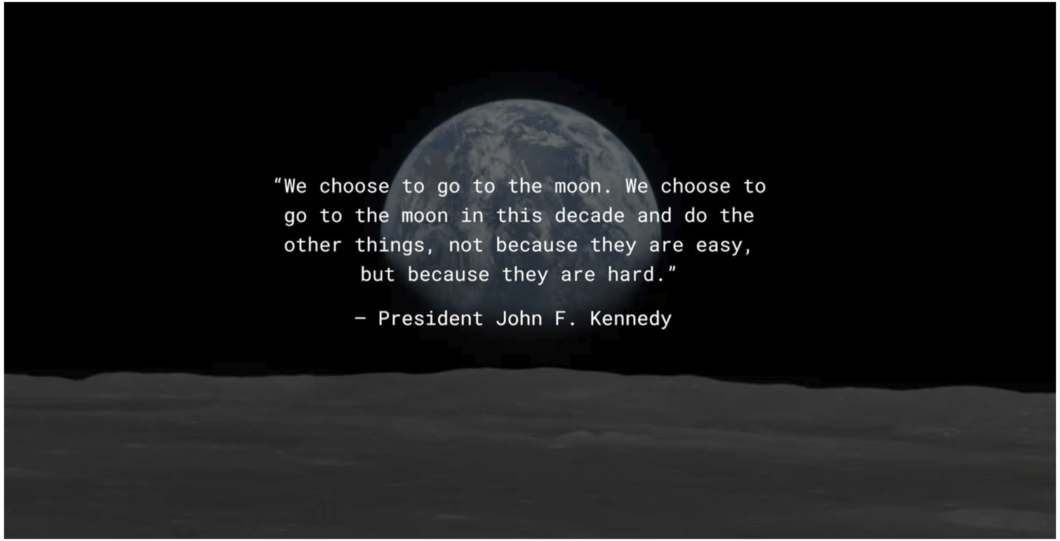
EnergyPlus 10X



LBNL: Tianzhen Hong, PhD; Xuan Luo
ORNL: Mark Adams
thong@lbl.gov , adamsmb@ornl.gov

Project Summary

Timeline:

Start date: 10/1/2018

Planned end date: 9/30/2021

Key Milestones:

1. EnergyPlus 9.4, 9/30/2020
2. EnergyPlus 9.5, 4/30/2021
3. EnergyPlus 9.6, 9/30/2021

Budget:

Total Project $ to Date:

- DOE: $1.3M
- Cost Share: N/A

Total Project $:

- DOE: $1.6M
- Cost Share: N/A

Key Partners:

NREL	TRANE
GARD Analytics	AUTODESK
UCF FSEC Energy Research Center, University of Central Florida	TESS, DesignBuilder

Project Outcome:

This project applied an integrated set of computing techniques to improve runtime performance of EnergyPlus. Results show runtime reductions range from 3.2% to 93% (mean 23%) with the same settings between EnergyPlus 9.6 and 9.0. If running at a high speed mode with some accuracy tradeoffs, the reductions range from 20% to 98% (mean 83%). Such speed improvements enable EnergyPlus for large scale simulations and increase user productivity. Changes to the data structure, algorithms, and code also improve EnergyPlus continuous development and maintenance.

Team - A diverse project team spans expertise in building science, EnergyPlus, and computer science


Tianzhen Hong
Senior Scientist
PI, LBNL


Xuan Luo
EnergyPlus Developer
LBNL

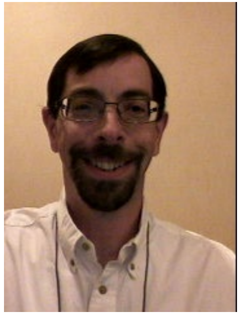

Wanni Zhang
Programmer
LBNL


Mark Adams
EnergyPlus Developer
PI, ORNL

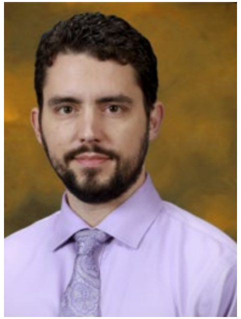

Jason DeGraw
EnergyPlus Developer
ORNL


Edwin Lee
EnergyPlus Developer,
NREL


Mike Witte
EnergyPlus Developer,
GARD Analytics


Jason Glazer
EnergyPlus Developer,
GARD Analytics


Lixing Gu
EnergyPlus Developer
FSEC

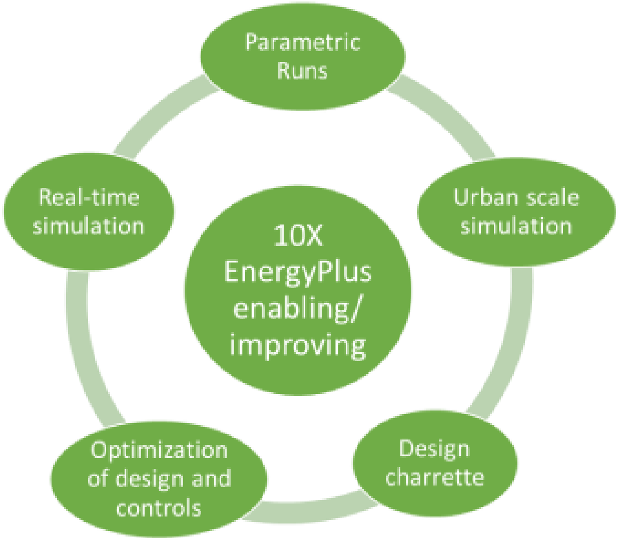

Joshua New
Computer Scientist
ORNL

Challenge

Compared with other commonly used building energy modeling programs, EnergyPlus runs slower by an order of magnitude. This hinders EnergyPlus for use cases that need many simulations.

EnergyPlus code base is big and complex: translated from FORTRAN to C in 2014 (version 8.2), about 1M lines of code written by many developers across 25 years .

Past efforts focus on new feature developments to model technologies, systems and controls, rather than runtime performance.



Approaches

This project employs an integrated approach to significantly improve EnergyPlus' annual simulation run-time performance from the baseline EnergyPlus 9.0. The approaches include but are not restricted to:

1. **Baseline**: Develop a suite of models covering diverse use cases and complexity, and establish an overall baseline run time of the suite that the projected improvement will be based on.
2. **Analysis**: Profile models to identify performance hotspots, and use correlation techniques to identify problematic component models or structures.
3. **Refactoring**: Enhance code in major routines of surface and zone heat balance, the plant components and loops, the nested HVAC simulations, the calculations of CTF, solar and window heat transfer using new data structures, improved algorithms, and optimized convergence criteria.
4. **Threading**: Test task parallelism to take advantage of multi-core processors using OpenMP
5. **GPU**: Parallel computing for solar shading calculations
6. **Speed up workflow**: Cache intermediate results (e.g., CTF, solar shading) and reuse them between simulations.

63 Models for Performance Benchmark

- 16 DOE Reference Building models
  - Chicago climate, ASHRAE 90.1-2004, annual run, 10-minute timestep, as-is reporting
- 16 large models
  - Large scale building with radiant underfloor heating
  - Natural ventilation with airflow network
  - 12 large models with many zones or surfaces
  - 2 residential buildings
- 31 diagnostics models
  - From shoebox no HVAC to complex geometry and HVAC systems
  - From no shading to lots of shading
  - From no window to many windows
  - From 5 zones to 500 zones

20 Hot Spots of the Benchmark Models in EnergyPlus 9.0

Functions	Percentage CPU time over benchmark suite	Functions	Percentage CPU time over benchmark suite
__pow	4.582	SolarShading::CLIPPOLY	1.852
CalcHeatBalanceInsideSurf	4.020	CalcZoneSums	0.437
CalcInteriorRadExchange	3.772	PsyTsatFnHPb	0.683
Psychrometrics::cached_psat_t	2.561	PierceSurface	1.597
UpdateThermalHistories	1.919	ReportSurfaceHeatBaance	0.407
InitSolarHeatGains	1.505	InitIntSolarDistribution	0.335
__GI___exp	1.283	CalcInteriorSolarDistribution	0.379
InitSurfaceHeatBalance	0.692	__ieee754_log_fma	0.316
__memset_avx2_erms	0.623	CalcHeatBalanceOutsideSurf	0.379
UpdateDataandReport	0.471	SetZoneEquipSimOrder	0.279

Algorithms - Caching & Optimization of Utility Functions

➢ Caching of utility functions

- **Subroutine functionality:**
  - PsyTsatFnPb(pb) - Calculate the saturation temperature from barometric pressure
  - PsyTsatFnHPb(h,pb) - Calculate the saturation temperature from the enthalpy and barometric pressure.
  - GetSpecificHeatGlycol(T, GlycolIndex) - Calculate the Glycol specific heat by interpolation
- **Performance:**
  - Each caching implementation achieved an average of 1-3% speed up of the total run time of all 623 EnergyPlus routine test files

➢ Cubic interpolation of psychrometric functions

- PsyTsatFnPb(pb) - Up to 10% speed up to selected test files
- A new performance precision trade off option in EnergyPlus 9.6 release

* Psychrometric functions: compute properties of moist air from other properties

Algorithms - Optimized Subroutines

➢ Polygon clipping for shadow calculation

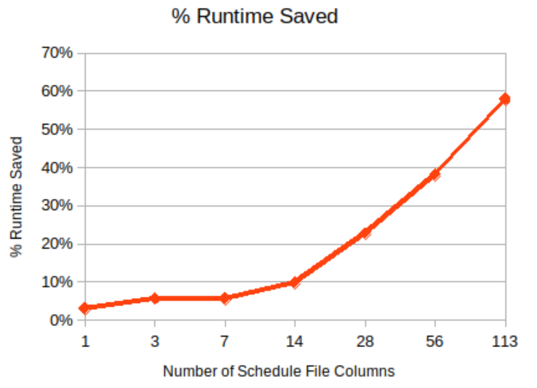
- **Methods:** Use *Liang-Barsky* algorithm for clipping rectangular polygons
- **Performance gain:**
  - 1.5X speed up for the subroutine itself
  - 6% speed up for HospitalLowEnergy.idf, a shading calculation intensive test file

➢ Direct solution to get part load ratio (PLR) of coils

- **Method:** Use linear relationship between part load ratio and system/coil output with given inlet conditions.
- **Performance**: 2x speed is achieved for coil subroutine

➢ Speedier schedule manager

- **Method:** more efficiently open and parse external schedule file data, reducing runtime in proportion to the number of columns of data in a file
- **Performance**: 92% run time saving on Mac for a schedule importing test file

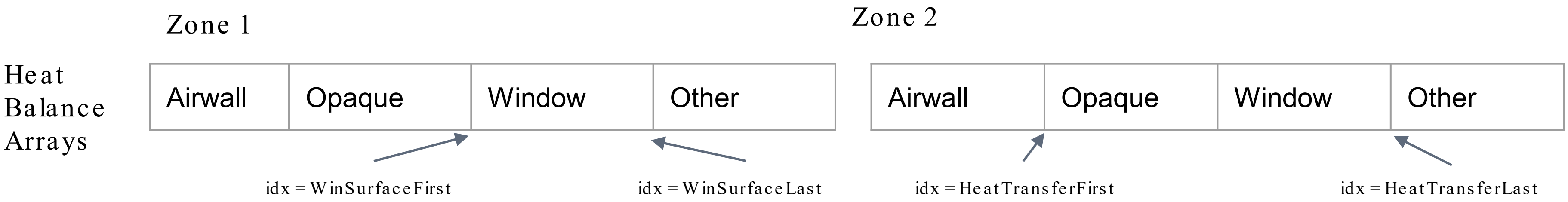


For a file with ~100 columns of data the runtime is cut in half.

Logic - Make the Common Case Faster

➢ Zone surface grouping and reordering

- Surfaces are grouped by zone
- Within each zone opaque surfaces are grouped separately from window surfaces
- Main loops executed by zone and then by surface
- These allow -
  - Heat balance loops to operate on a continuous set of surfaces without if conditions within the loops
  - Future parallelization by zone



Logic - Make the Common Case Faster

- ## Inside surface heat balance
  - Two functions, one for simple Conduction Transfer Function (CTF) - only simulations and another for more complex simulations
  - Eliminate if conditions within the loops

- ## Move uncommon conditional calculations out of the main loop
  - Tubular daylighting device
  - Movable insulation for opaque surfaces
  - Movable slats for window blinds
  - Storm windows
  - Source or sink surface

✓ Allows easier branch prediction and vectorization, and better cache performance for compilers

```
For each surface:
  If storm window: Do stuff A
  Else: Do stuff B
```

→

```
For each surface: Do stuff B
If any storm window exists:
  For each storm window: Do stuff A
```

Data Structure - Heat Balance Array Refactoring

➢ Array of objects to array of scalars

- Convert large object arrays (e.g. SurfaceWindow with 200+ fields and Surface with 100+ fields) to arrays of scalars for cache locality and future vectorization

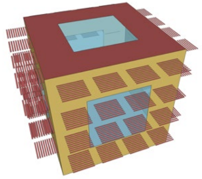
➢ Efficient initialization of data structure

- Remove third-party initialization function calls for vectorization
- Loop-wise initialization with small loops to make the maximum use of vector registers
- Reduce re-initializations (e.g. unnecessary solar arrays reset when sun is down at each time step)

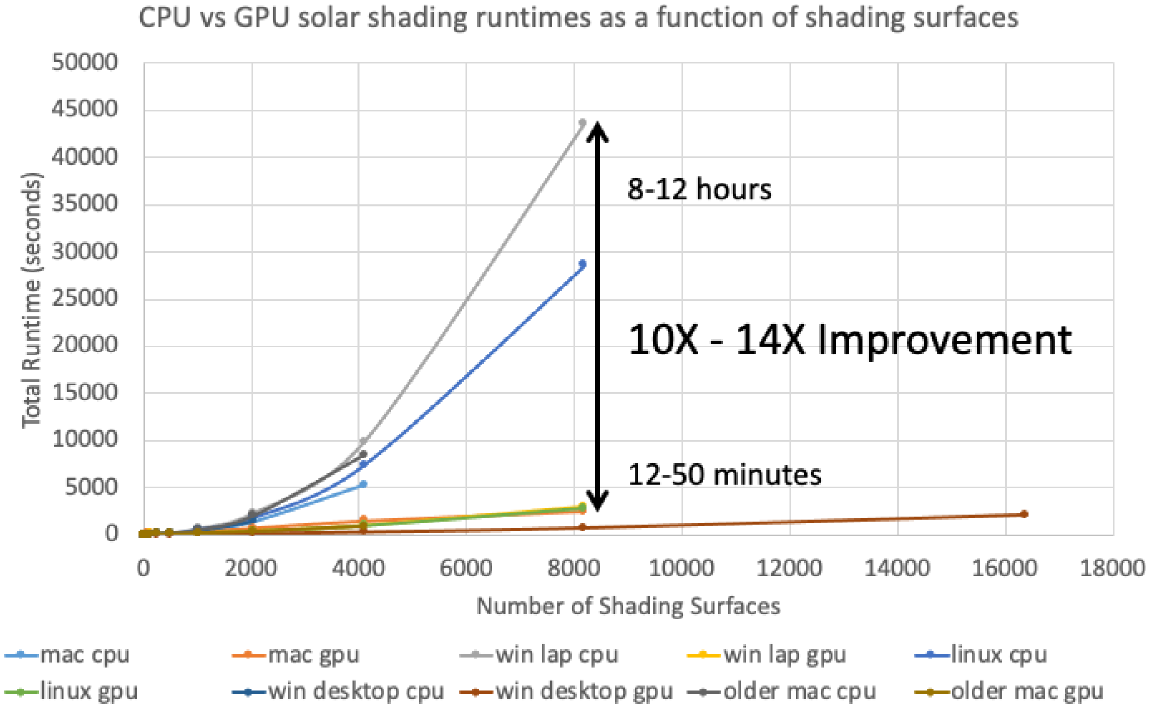
➢ Optimized multi-dimensional arrays

- Reorder multi-dimensional arrays to loop execution order for cache locality
- Reduce array slicing and reconstruction (2D array -> array of array)
- Reduce dimension of arrays

GPU Accelerate Solar Shading

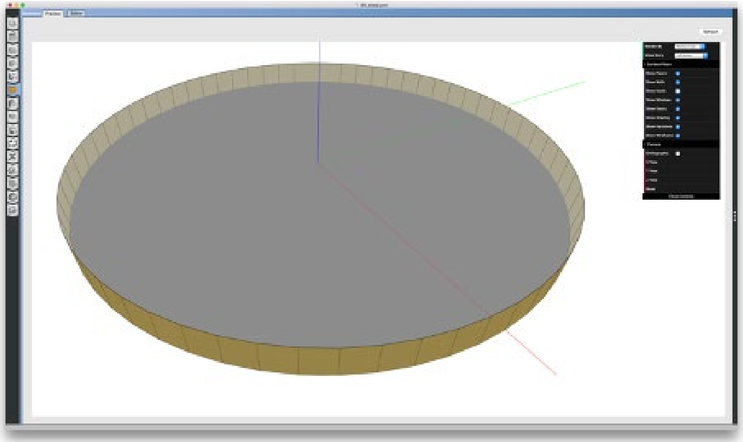


- **Performance improvements:**
  - Up to 10 -14 x improvement
  - Scales with number of shading surfaces
- **Penumbra:**
  - Open-source library for GPU solar shading
  - OpenGL 2.1 (cross platform, widely support)
- **Benefits:**
  - Enables modeling of large buildings with many shading surfaces
  - Scales with GPU power



Profiling Guided Performance Improvements

- **CarrolMRT:**
  - Alternate interior radiant heat exchange model
  - O(n) vs O(n^2) algorithm
  - Up to 95% runtime reduction (scales with number of interior surfaces in a zone)

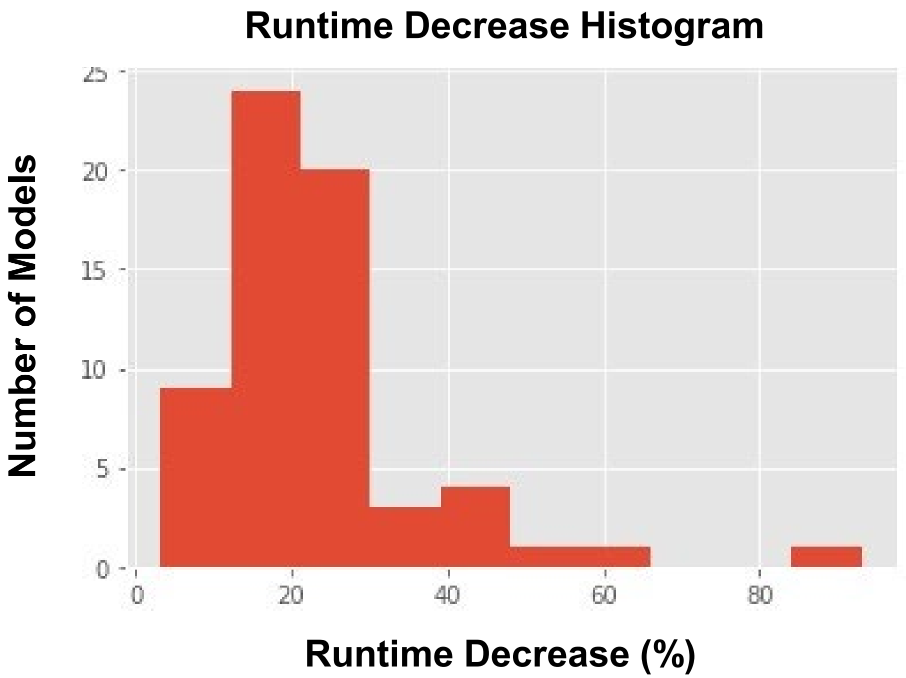


- **Native C++ CSV output / conditional output:**
  - Alternative to ReadVarsESO
  - 25-82+% runtime reduction, scaling on number of outputs
  - Conditionally turn on/off output files
    - Up to 42% runtime reduction

- **Function approximation:**
  - Min/max and some pow functions to use new, faster approximations
  - Up to 13% runtime reduction, average 4%

Runtime: EnergyPlus 9.6 (Sep21) vs. 9.0 (Sep18)

Machine: Intel Xeon W-2123 CPU @ 3.60GHz, Ubuntu 18.04.2 LTS
Models: 63 EnergyPlus IDFs



Mean: 23%, median: 20%, min: 3.2%, **max: 93%**

Same default settings: 9.6 vs 9.0

Parallelization - OpenMP

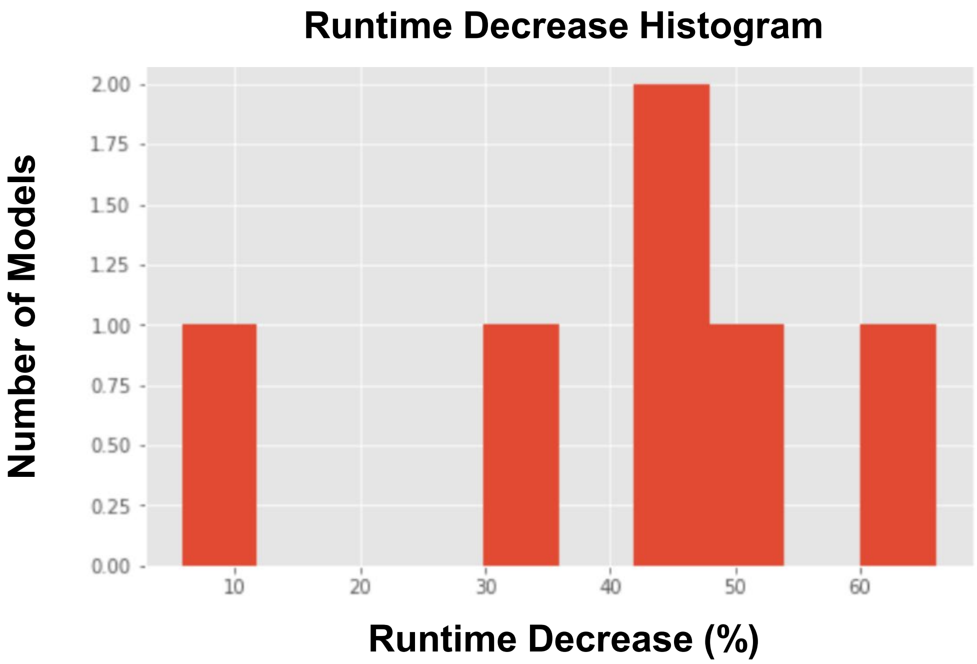
➢ Applied OpenMP parallel in major heat balance code blocks

- Major functionalities:
  - Init surface heat balance (radiant exchange, convective coefficients, etc.)
  - Calculate inside and outside surface temperatures
  - Update thermal histories and reporting
- Generate identical results with single thread

# Threads	2	4	8
10x_incr_100Zones100Surfaces (4-CPU desktop)	1.4x	1.7x	-
10x_incr_100Zones100Surfaces (24-CPU cluster)	1.5x	1.9x	2.1x
10x_large_82zones_ASHRAE_HQ (24-CPU cluster)	1.6x	1.8x	2.2x

Runtime: Parallelization with OpenMP

9.6 with OpenMP: 8 vs 1 CPU for 6 "large" (> 20 surfaces/zone) models



Performance Precision Tradeoffs

- **Successive modes to trade speed for precision:**
  - Use existing simulation parameters
  - Expose existing internal convergence variables to the user
  - New optimizations

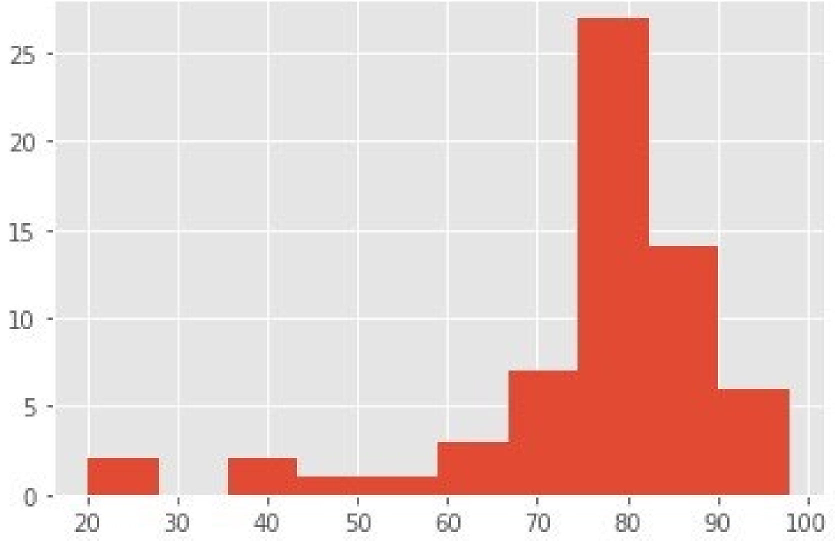
Mode	Description
Normal	No overrides
Mode01	Zone time step (TimeStep object) will be set to one timestep per hour
Mode02	Mode01 plus ZoneAirHeatBalanceAlgorithm will be set to Euler
Mode03	Mode02 plus Minimum Number of Warmup Days will be set to 1
Mode04	Mode03 plus Begin Environment Reset Mode will be set to SuppressAllBeginEnvironmentResets
Mode05	Mode04 plus minimum system timestep length will be 1 hour
Mode06	Mode05 MaxZoneTempDiff will be set to 1.00
Mode07	Mode06 MaxAllowedDelTemp will be set to 0.1
Advanced	Allow direct input of convergence field values

- The simulation time can be dramatically reduced
- Impacts simulation results
- "_perflog.csv" file provides user the impact of changing modes on annual energy, runtime, warnings, and temperature oscillations
- Other settings: Use Coil Direct Solutions (Y/N), Zone Radiant Exchange Algorithm (ScriptF, CarrollMRT)
- Additional modes in EnergyPlus 9.6 (e.g. Psychrometric)

Runtime: Speed Mode4 + CarrollMRT

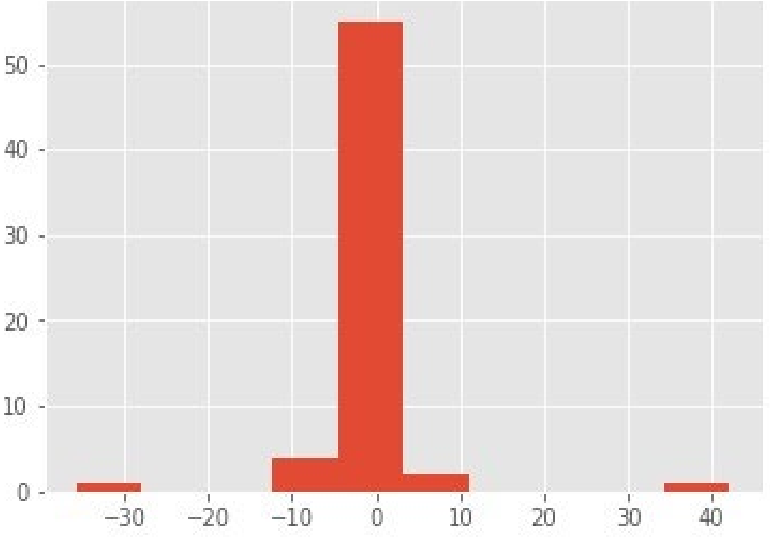
Runtime Decrease Histogram

Number of Models



Runtime Decrease (%)

Total Site Energy Variance Histogram

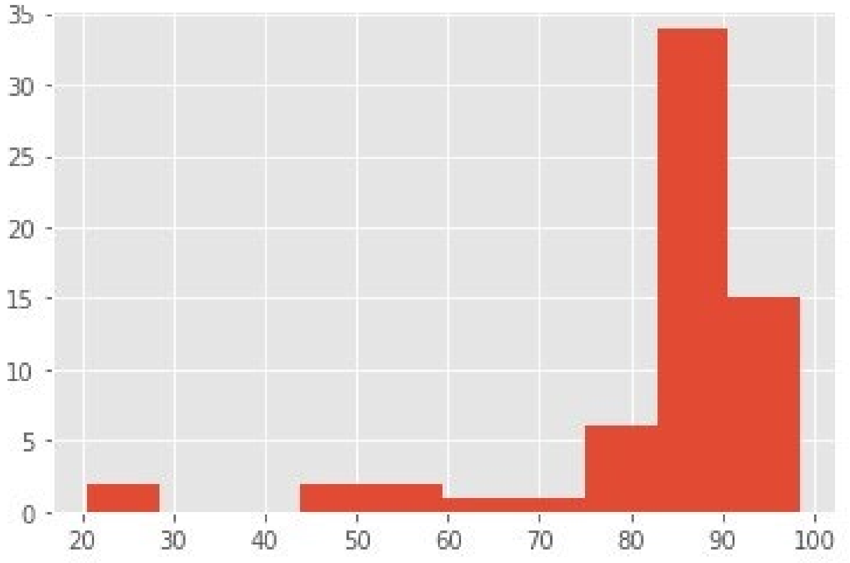


Total Site Energy Variance (%)

Mean: **76%,** min: 20%, median: 78%, max: 98%

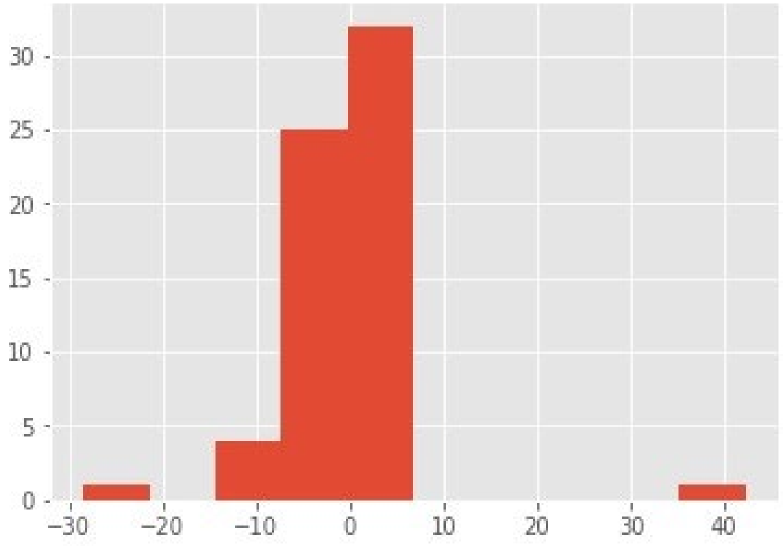
Runtime: Speed Mode7 + CarrollMRT

Runtime Decrease Histogram



Runtime Decrease (%)

Total Site Energy Variance Histogram



Total Site Energy Variance (%)

Mean: **83%,** min: 21%, median: 88%, max: 98%

Performance Precision Tradeoff Characterization

10x_large_705_zones_radiant_heating.idf
10x_large_123_zones_VRF.idf
10x_RefBldgOutPatientNew2004_Chicago.idf
10x_RefBldgWarehouseNew2004_Chicago.idf
10x_large_98_zones_HVAC_London.idf

- ➢ Performance Precision tradeoffs introduce variance in results which can be large
- ➢ Five models show especially high variance
- ➢ Future work: understand which modeling features lead to high variance and provide guidance on when Precision-Performance Tradeoffs are safe to use

Impacts

- The 3-year project applied an integrated set of computing techniques to reduce runtime of EnergyPlus. Performance improvements were included in EnergyPlus releases.
- Compared the runtimes of 63 benchmark models between EnergyPlus 9.0 and 9.6, runtime reductions range from 3% to 93% (i.e., 1X to 14X, median 1.25X) with default settings, and from 20% to 98% (i.e., 1.3X to 50X, median 7.7X) in a high speed mode with some accuracy tradeoffs (excluding OpenMP).
- Reducing runtime of complex EnergyPlus simulation models improves user productivity and supports quick decision making (e.g., design charrette).

Stakeholder Engagement and Outreach

- Annual stakeholder briefing
- Biweekly meetings to provide timely feedback
- A hackathon session with NERSC on OpenMP
- Engaged the broader EnergyPlus development team and community
- Presentation at ASHRAE BPA Conference, September 25, 2019

TRANE
AUTODESK
TESS DesignBuilder
Carrier

Remaining Project Work

- Contribute to the EnergyPlus 9.6 release
- Re-run final performance benchmark with the official EnergyPlus 9.6
- Project wrap up meeting with stakeholders
- After the project, continue the performance work as part of the EnergyPlus core project in areas, e.g.:
  - Refactor data structure and code
  - Vectorization and parallelization
  - Reduce unnecessary calls
  - Optimize looping and iterations
  - Calculate output variables only when required

Thank You

LBNL and ORNL
Tianzhen Hong and Mark Adams
thong@lbl.gov , adamsmb@ornl.gov

REFERENCE SLIDES

Project Budget

Project Budget : $1.6M
Variances : N/A
Cost to Date : $1.3M
Additional Funding : N/A

Budget History					
FY 2019 (past)		FY 2020 (past)		FY 2021 (current)	
DOE	Cost-share	DOE	Cost-share	DOE	Cost-share
$500k	N/A	$500k	N/A	$600k	N/A

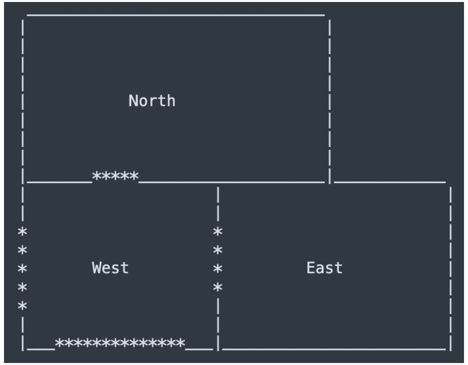
Project Plan and Schedule

Project Schedule

Project Start: October 1, 2018 (FY19) — Completed Work
Project End: September 30, 2021 (FY21) — Active Task (in progress work)
◆ (red) Milestone/Deliverable (Originally Planned)
◆ (black) Milestone/Deliverable (Actual)

Task	FY2019 Q1 (Oct-Dec)	FY2019 Q2 (Jan-Mar)	FY2019 Q3 (Apr-Jun)	FY2019 Q4 (Jul-Sep)	FY2020 Q1 (Oct-Dec)	FY2020 Q2 (Jan-Mar)	FY2020 Q3 (Apr-Jun)	FY2020 Q4 (Jul-Sep)	FY2021 Q1 (Oct-Dec)	FY2021 Q2 (Jan-Mar)	FY2021 Q3 (Apr-Jun)	FY2021 Q4 (Jul-Sep)
Past Work												
FY19 Q2 Milestone: EnergyPlus 9.1 release	▇	▇ ◆										
FY19 Q4 Milestone: EnergyPlus 9.2 release			▇	▇ ◆								
FY20 Q2 Milestone: EnergyPlus 9.3 release					▇	▇ ◆						
FY20 Q4 Milestone: EnergyPlus 9.4 release							▇	▇ ◆				
FY21 Q2 Milestone: EnergyPlus 9.5 release									▇	▇ ◆		
Current/Future Work												
FY21 Q4 Milestone: EnergyPlus 9.6 release											▇	▇ ◆ (planned)

Logic - Remove Redundant Subroutines



➢ Skip unnecessary interzone window calculation

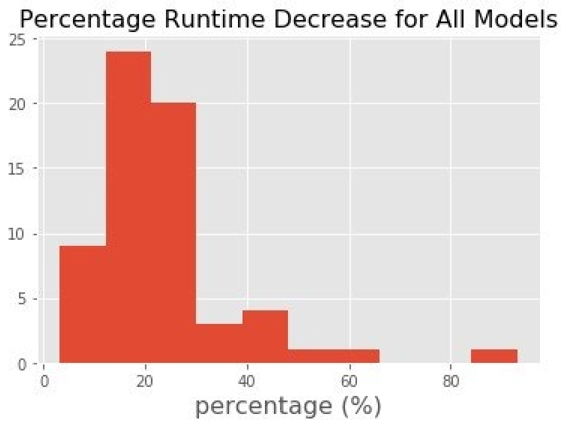
- **Methods:**
  - Skip calculating the diffuse solar exchange factors between zones if no interzone window exists
- **Performance gain:**
  - Reduced 6+% total E+ run time in large models (500 zones and up) in calculating the interzone solar exchange factor matrix

➢ Reduce run time solar and thermal absorptance calculation

- **Methods:**
  - Reduce unnecessary loop iterations to calculate solar and thermal absorptance factors at each time step by pre-checking if any shading or glazing status change
- **Performance gain:**
  - Reduced 90+% function calls and run time of subroutine itself

Runtime benchmarking 9.6 vs. 9.0

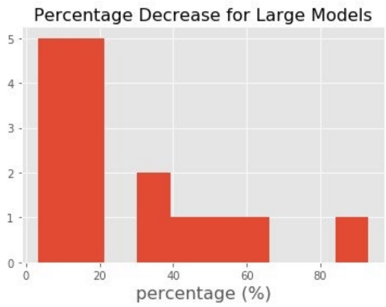
All default settings (Machine: Intel Xeon W-2123 CPU @ 3.60GHz, Ubuntu 18.04.2 LTS)

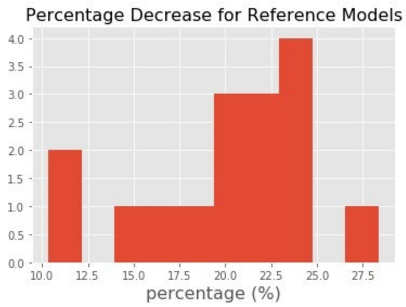


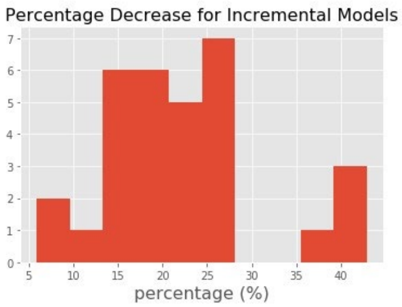
Models with largest runtime decrease:

file	time_901	time_960	percent_improve(%)	absolute_improve(s)
10x_large_9_zone_solar_shading_external.idf	105.48	7.38	93.003413	98.10
10x_large_6_zones_residential.idf	149.20	55.14	63.042895	94.06
10x_large_82_zones_ASHRAE_HQ.idf	916.74	402.55	56.088967	514.19
10x_large_25_zones_AFN.idf	874.31	496.49	43.213506	377.82
10x_incr_500zones.idf	509.96	291.19	42.899443	218.77

count	63
mean	23.113265
min	3.245094
median	20.465755
max	93.003413

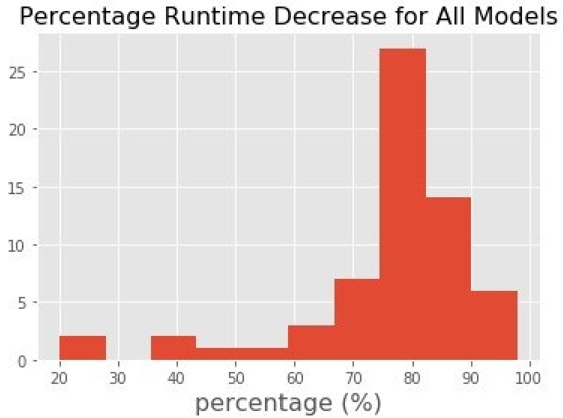






Runtime benchmarking 9.6 vs. 9.0

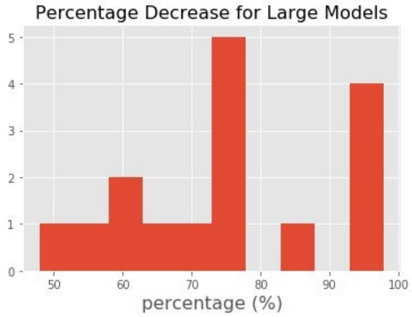
Version 9.6 with **fast mode 04** and radiant exchange algorithm CarrollMRT

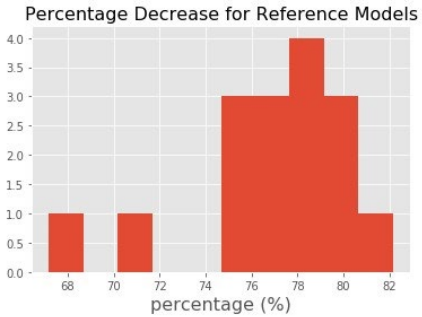


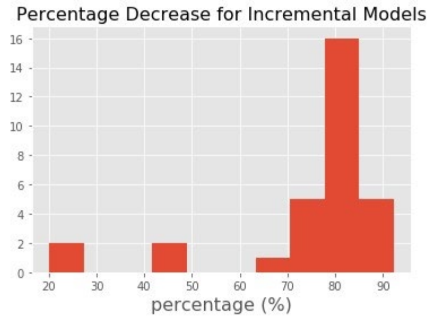
Models with largest runtime decrease:

file	time_901	time_960_fast04	percent_improve(%)	absolute_improve(s)
10x_incr_100zones1win10overhangsFullExInRefl.idf	1453.80	1161.89	20.079103	291.91
10x_incr_100zones1win10overhangsFullExRefl.idf	1465.19	1159.60	20.856681	305.59
10x_incr_100zones10shadingsFullExRefl.idf	340.68	196.43	42.341787	144.25
10x_incr_100zones10shadingsFullExInRefl.idf	343.44	196.66	42.738178	146.78
10x_large_705_zones_radiant_heating.idf	13619.32	7091.71	47.929045	6527.61

count	63
mean	75.602455
min	20.079103
median	77.948718
max	97.884805

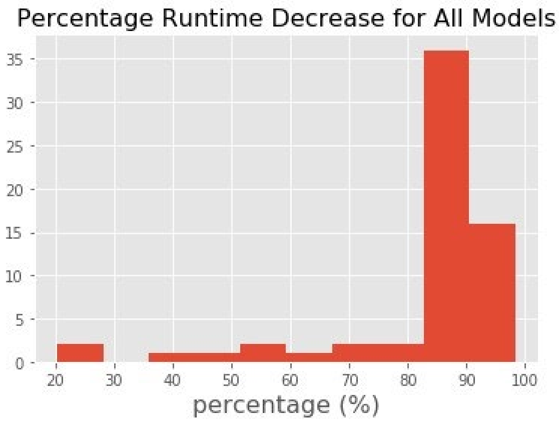






Runtime benchmarking 9.6 vs. 9.0

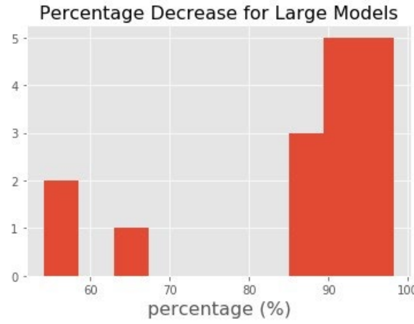
Version 9.6 with **fast mode 07** and radiant exchange algorithm CarrollMRT

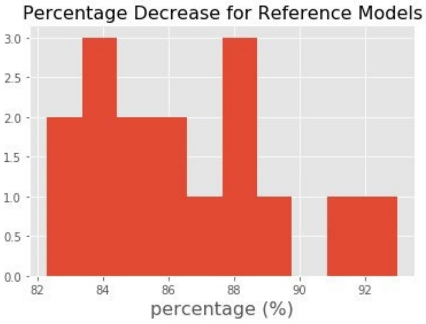


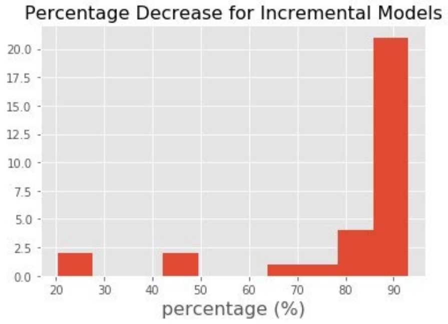
Models with largest runtime decrease:

file	time_901	time_960_fast07	percent_improve(%)	absolute_improve(s)
10x_incr_100zones1win10overhangsFullExInRefl.idf	1453.80	1157.54	20.378319	296.26
10x_incr_100zones1win10overhangsFullExRefl.idf	1465.19	1159.36	20.873061	305.83
10x_incr_100zones10shadingsFullExRefl.idf	340.68	192.65	43.451333	148.03
10x_incr_100zones10shadingsFullExInRefl.idf	343.44	192.91	43.830072	150.53
10x_large_95_zones_multiple_AHUs.idf	1373.64	629.99	54.137183	743.65

count	63
mean	82.878219
min	20.378319
median	86.796048
max	98.241773

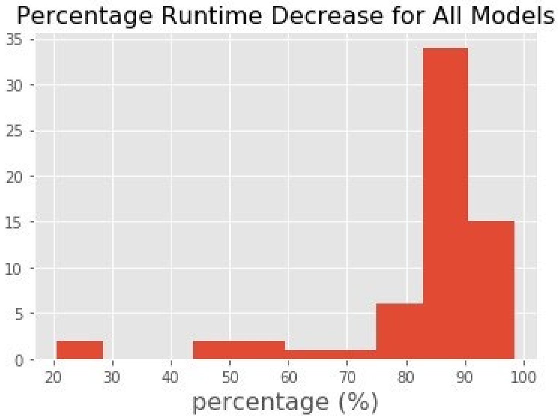






Runtime benchmarking 9.6 vs. 9.0

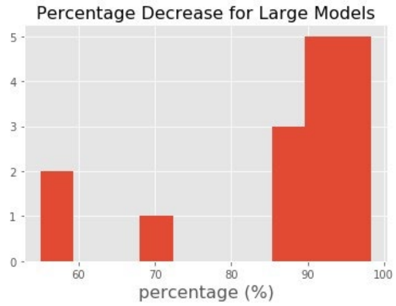
Version 9.6 with **fast mode 07** and radiant exchange algorithm CarrollMRT + OpenMP **4 threads**

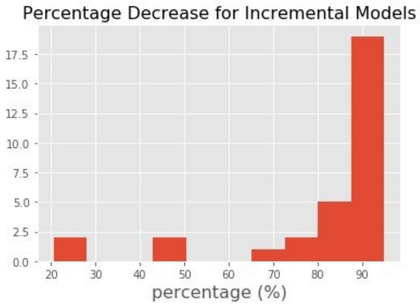


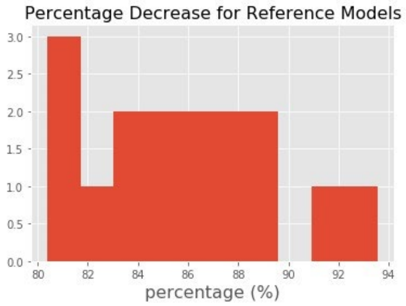
Models with largest runtime decrease:

file	time_901	time_960_fast07_openmp	percent_improve(%)	absolute_improve(s)
10x_incr_100zones1win10overhangsFullExInRefl.idf	1453.80	1153.59	20.650021	300.21
10x_incr_100zones1win10overhangsFullExRefl.idf	1465.19	1151.95	21.378797	313.24
10x_incr_100zones10shadingsFullExRefl.idf	340.68	188.07	44.795703	152.61
10x_incr_100zones10shadingsFullExInRefl.idf	343.44	188.28	45.178197	155.16
10x_large_95_zones_multiple_AHUs.idf	1373.64	618.40	54.980927	755.24

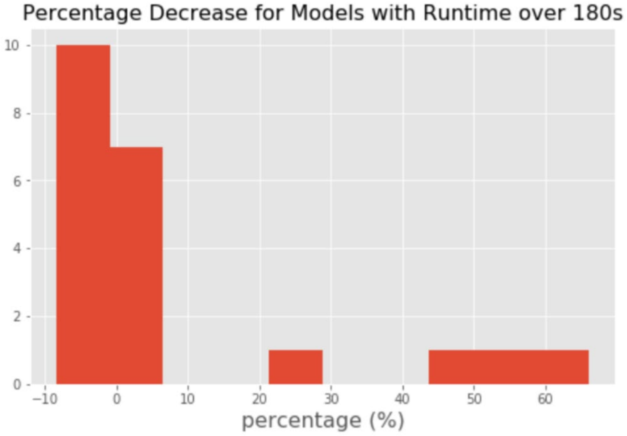
count	63
mean	83.286181
min	20.650021
median	88.107236
max	98.355650



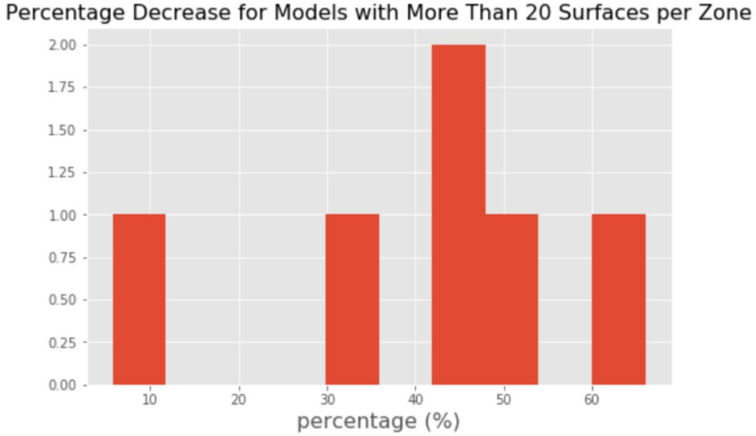




9.6: OpenMP 8 threads vs. w/o OpenMP

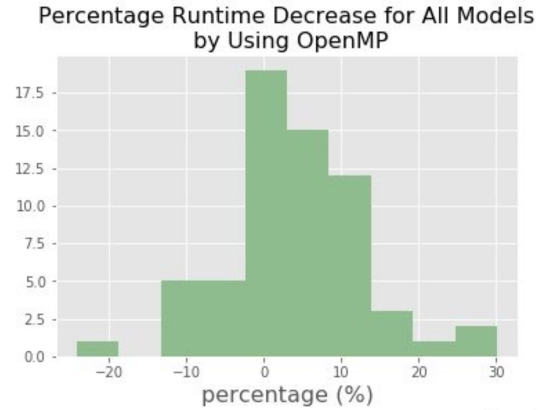


count	21
mean	8.1%
min	-8.4%
meidan	-0.4%
max	66.0%

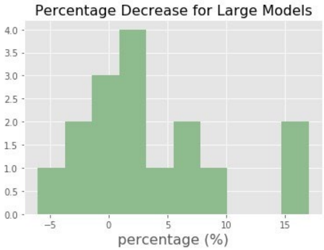


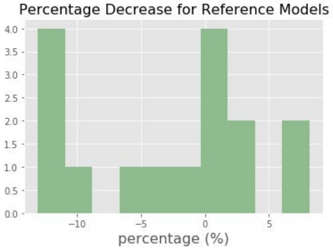
count	6
mean	42.8%
min	5.7%
median	47.9%
max	66.0%

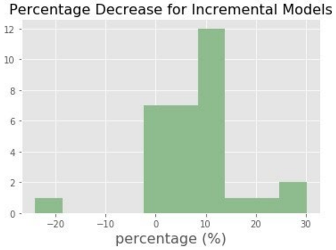
9.6 fast mode 07: w/ OpenMP vs. w/o OpenMP



count 63
mean 4.186573
min -24.137931
median 3.316062
max 30.159744







Models with runtime increase:

file	time_fast07	time_fast07_openmp	percent_improve(%)	absolute_improve(s)
10x_incr_10zones.idf	0.87	1.08	-24.137931	-0.21
10x_RefBldgStand-aloneRetailNew2004_Chicago.idf	1.99	2.25	-13.065327	-0.26
10x_RefBldgQuickServiceRestaurantNew2004_Chicago.idf	1.84	2.05	-11.413043	-0.21
10x_RefBldgSuperMarketNew2004_Chicago.idf	2.63	2.92	-11.026616	-0.29
10x_RefBldgWarehouseNew2004_Chicago.idf	1.46	1.62	-10.958904	-0.16
10x_RefBldgFullServiceRestaurantNew2004_Chicago.idf	2.05	2.27	-10.731707	-0.22
10x_large_6_zones_residential.idf	13.80	14.63	-6.014493	-0.83
10x_RefBldgSmallOfficeNew2004_Chicago.idf	2.63	2.78	-5.703422	-0.15
10x_large_9_zone_solar_shading_external.idf	2.59	2.68	-3.474903	-0.09
10x_large_7_zones_residential.idf	11.57	11.90	-2.852204	-0.33
10x_RefBldgStripMallNew2004_Chicago.idf	4.11	4.22	-2.676399	-0.11
10x_RefBldgMidriseApartmentNew2004_Chicago.idf	10.30	10.45	-1.456311	-0.15
10x_large_705_zones_radiant_heating.idf	5728.32	5770.82	-0.741928	-42.50
10x_large_98_zones_HVAC_London.idf	150.62	151.73	-0.736954	-1.11

Models with largest runtime decrease:

file	time_fast07	time_fast07_openmp	percent_improve(%)	absolute_improve(s)
10x_incr_100zones20surfaces.idf	15.65	10.93	30.159744	4.72
10x_incr_100zones100surfaces.idf	113.28	82.59	27.092161	30.69
10x_incr_500zones.idf	46.38	35.92	22.552824	10.46
10x_large_25_zones_AFN.idf	27.87	23.13	17.007535	4.74
10x_large_82_zones_ASHRAE_HQ.idf	27.52	23.23	15.588663	4.29