IOWA STATE UNIVERSITY
Department of Economics, Department of Electrical & Computer Engineering

Report on:

ERCOT PNNL Contract 401882: *Start Date* 3/19/2018

Development of an Integrated Transmission and Distribution Test System to Evaluate Transactive Energy Systems

ISU Project Team:

PI Leigh Tesfatsion & Co-PI Zhaoyu Wang

Grad Research Assistant: Swathi Battula

PNNL/ISU Web Conference, 05 April 2019

ERCOT Contract: Presentation Outline

- Original Task/Milestone Schedule: M1-M3
- Updated Task/Milestone Schedule: M1, M2, M3.1, M3.2, M3.3
- M3.2: Completed Work on AMES V5.0
- M3.2: Work in Progress on AMES V5.0 and ERCOT 200-Bus Test Case
  - ✓ ***Key Milestone:*** ERCOT 200-Bus Test Case now runs for multiple successive simulated days with no compilation or run-time errors
- Updated all test case data files to be able to run with latest code
- Preparation for AMES V5.0 J-Unit Testing: Event Sequencing

Original Task & Milestone Schedule

Milestone	Date Due	Original Description
M1	May 31, 2018	5-zone model of the old ERCOT system, posted to a web repository.
M2	Sep 30, 2018	Nodal model of the new ERCOT system, posted to a web repository.
M3	Sep 30, 2018	Submitted conference or journal paper on this work.

Updated Task & Milestone Schedule

Milestone	Date Due	Date Delivered	Fuller Descriptions of Actual Work
M1* DONE	May 31, 2018	June 5, 2018	Development of 8-Bus ERCOT model (with nodal locational marginal pricing); grid/load/gen data posted at PNNL repository
M2.1 DONE	Sept 30, 2018	August 1, 2018	Basic 8-Bus ERCOT Test System, implemented via AMES V3.1, posted at https://github.com/ITDProject/ERCOTTestSystem
M2.2 DONE	Sept 30, 2018	August 24, 2018	8-Bus ERCOT Test System (with wind power), implemented via AMES V3.2, posted at https://github.com/ITDProject/ERCOTTestSystem
M3.1 DONE	Sept 30, 2018	August 31, 2018	200-Bus ERCOT Test System (with wind power), implemented via AMES V3.2, posted at https://github.com/ITDProject/ERCOTTestSystem/tree/master/ERCOT_Test_Systems/The_200Bus_ERCOT_Test_System
M3.2**	July 31, 2019		200-Bus ERCOT Test System (with Non-Dispatchable Generation), implemented via AMES V5.0, to be posted at PNNL/ISU repositories.
M3.3**	July 31, 2019		Paper to be submitted that focuses on the development of the ERCOT Test Systems

- * **M1 Modification (Ok'd by PNNL):** For M1 we have skipped the modeling of the old (zonal) ERCOT system and instead directly worked to develop an 8-bus model of the new (nodal) ERCOT system.
- ** **M3 Modification: Contract extension through July 31, 2019 received from PNNL on March 4, 2019, for completion of task M3**

Summary of AMES V5.0 Work to Date for M3.2

- **Extension of AMES V5.0 Capabilities for Milestone M3.2**
  - [DONE] Coding for Daily DAM SCUC optimization
  - [DONE] Coding for RTM SCED optimization every M minutes with a user-specified M
  - [DONE] Coding for FNCS integration to enable network co-simulation
  - [DONE] Detailed documentation for analytical DAM SCUC/SCED optimization in AMES V5.0
  - [DONE] Basic documentation for AMES V5.0, including a detailed list for all parameters/flags and initial state variables that need user configuration.

Summary of AMES V5.0 Work to Date for M3.2 … Continued

- **[DONE]**

— **Modified `PSST' Code**

- To ensure correct refreshing of initial DAM/RTM conditions for multiple-day runs.
- To report DAM LMPs and GenCo Commitments back to the user.
- To read 'startup' and 'shutdown' cost components from AMES
- To produce output messages related to solver, e.g. status of the solver, termination condition of the solver
- To include the parameter 'Maximum Time Limit' – to allow the solver to terminate after the prescribed time has elapsed

— **Verification Tests Done**

- Verified 'DAM SCUC' outcomes for their correctness for simple test cases
- Verified 'RTM SCED' outcomes for their correctness for simple test cases with RTM running every five minutes (i.e., M=5)
- Verified that AMES V5.0 runs for multiple days
- Verified that all the cost components from AMES are read correctly into the SCUC formulation

Summary of AMES V5.0 Work to Date for M3.2…Continued

- VerTestCaseBaseCase **DONE**
  - This test case produces SCUC/SCED outcomes under the following conditions:
    - ✓ Transmission congestion is absent
    - ✓ Minimum power generation limits are taken to be zero
    - ✓ Start up, shut down and no-load costs are taken to be zero
    - ✓ Minimum up-time and down-time values are taken to be 0 (hr)
    - ✓ No ramping limits
    - ✓ Day-ahead and real-time load forecasts are set equal
  - This test case provides a base case for later comparison purposes.
- VerTestCaseGenMinPowerLevel **DONE**
  - This test case verifies a generator's minimum power level is maintained when it is committed, given the above-stated conditions (i) and (iii)-(vi).
- VerTestCaseUpTimeDownTime **DONE**
  - This test case verifies a generator's minimum up time and down time are maintained when it is committed, given the above-stated conditions (i) and (iii)-(vi).
- VerTestCaseMultiDayRun **DONE**
  - This test case verifies DAM/RTM initial conditions are refreshed appropriately when AMES V5.0 is run for multiple successive days.

Note: Files for the above test cases are uploaded at https://github.com/ITDProject/ERCOTTestSystem/tree/dev-source-code/AMES-V5.0/DATA/VerificationTestCases

Summary of Additional Completed and Ongoing AMES V5.0 Verification Test Cases

- VerTestCaseCostComponents

The purpose of this test case is to verify cost component aspects of the SCUC formulation under the above-stated conditions (i)-(ii) and (iv)-(vi) – i.e., to verify that the SCUC formulation correctly includes no load, start-up, dispatch, and shut-down cost components.

✓ **DONE**

- VerTestCaseCostComponentsNoLoad

  This test case verifies that no load cost is appropriately taken into account in SCUC/SCED formulation given the above-stated conditions (i), (ii) and (iv)-(vi).

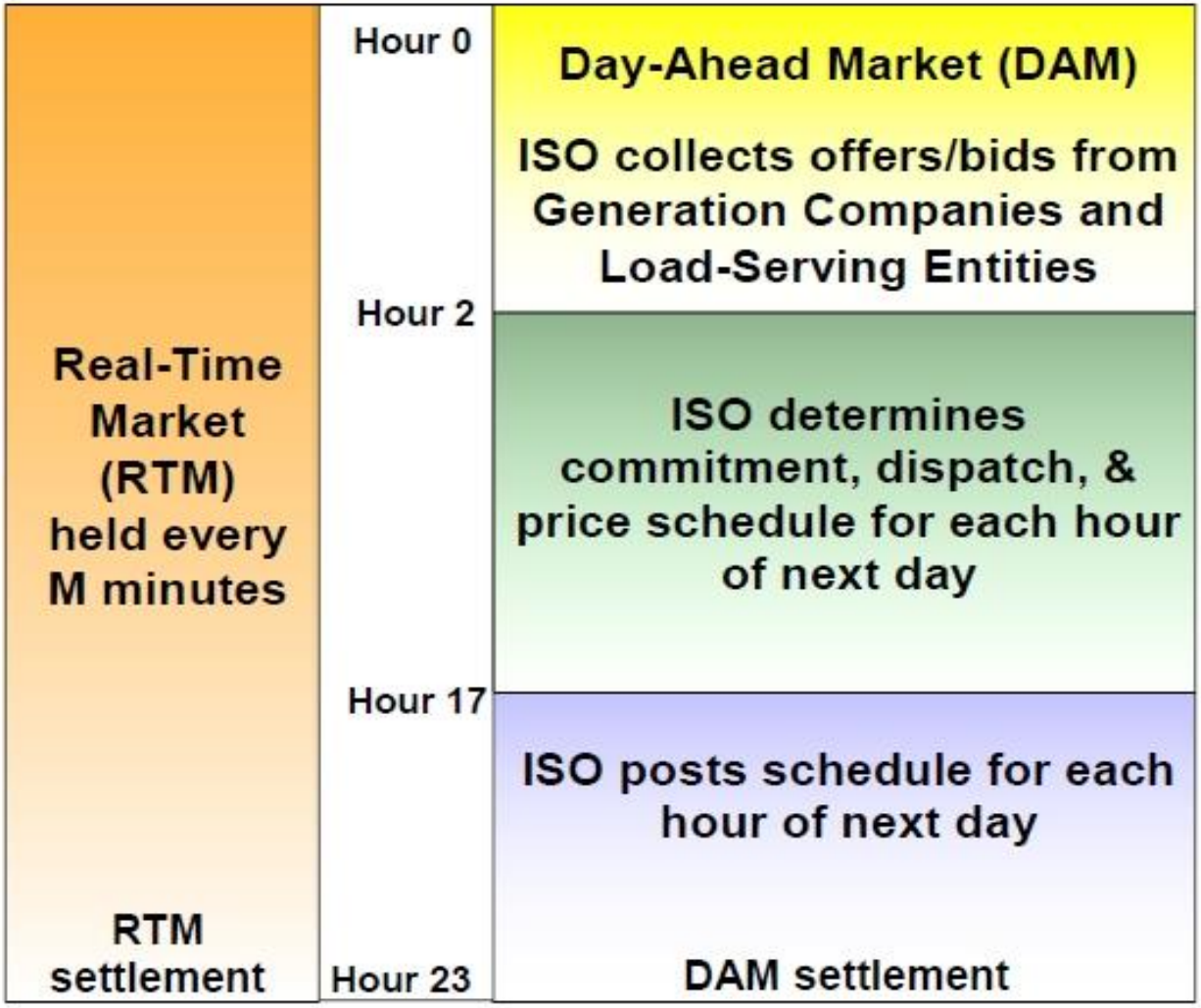
✓ To be completed

- VerTestCaseCostComponentsStartUP
- VerTestCaseCostComponentsShutDown

Latest Work on AMES V5.0 … The 200-Bus ERCOT Test Case

- This past week, run-time errors were observed for reading of SCED data while running the 200-bus test case for multiple days.
- Modified AMES V5.0 code to ensure correct reading of SCED data.
- With the latest modifications, AMES V5.0 is now able to run test case 'AMES_ercot_200bus.dat' without run-time errors for multiple days.
- Uploaded the modified 'AMES_ercot_200bus.dat' file at https://github.com/ITDProject/ERCOTTestSystem/tree/dev-source-code/AMES-V5.0/DATA/ERCOT
- Uploaded the latest code at https://github.com/ITDProject/ERCOTTestSystem/tree/dev-source-code/AMES-V5.0

Preparation for AMES V5.0 J-Unit Testing: Event Sequencing



AMES V5.0: Sequence of Events During a Typical Simulated Day

AMES – PSST : Sequence of Events

Step 0: Set D = 1

Step 1: Set H = 0 and I = 0

Step 2: 'AMESMarket' agent initiates market operations through ISO agent. (AMESMarket has an ISO)

Step 2: 'ISO' agent performs DAM operation on day 'D' to plan for next-day operations on day 'D+1'.

Step 3: 'ISO' agent has a 'PSSTSCUC' agent that writes 'ReferenceModel.dat' file and makes an external call to PSST to solve a Security-Constrained Unit Commitment (SCUC) optimization.

Step 4: PSST reads the input file 'ReferenceModel.dat' and performs SCUC.

Step 5: Status of each unit is set to 0/1 based on the SCUC outcomes in the previous step and another call to SCUC is made to obtain dual solutions – i.e LMP at each bus.

Step 6: PSST writes SCUC outcomes from Step 3 and Step 4 into 'xfertoames.dat' and 'DAMLMP.dat' files.

Step 7: 'PSSTSCUC' agent reads commitment and LMP data from 'DAMLMP.dat' and 'xfertoames.dat', and updates DAM outcomes.

AMES – PSST : Sequence of Events … Continued

Step 10: 'ISO' agent has a 'RTMarket' agent that writes 'rt-unitcommitments.dat' (contains generator unit commitments of day 'D') and 'RTReferenceModel.dat'.

Step 11: 'RTMarket' agent has a 'PSSTSCED' agent that makes an external call to PSST to solve Security-Constrained Economic Dispatch (SCED) optimization.

Step 12: PSST reads the input files 'rt-unitcommitments.dat', 'RTReferenceModel.dat' and performs SCED.

Step 13: PSST writes SCED outcomes into 'RTSCED.dat'.

Step 14: 'PSSTSCED' agent reads dispatch and LMP data from 'RTSCED.dat' and updates RTM outcomes.

Step 15: 'AMESMarket' agent posts RTM LMPs.

Step 16:

```
Increment I;
 If (I*M % 60 == 0) {
                    H++;
                    Reset I = 0;
    }
    If (H% 24 == 0) {
```