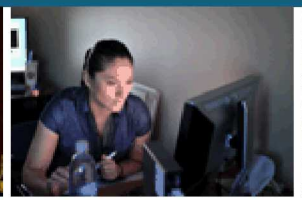
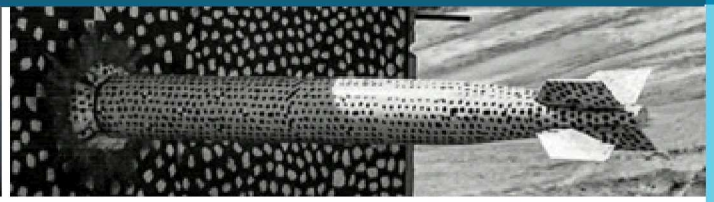
This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

SAND2019-11105C

EGRET: Unit Commitment and {Economic,…} Dispatch





PRESENTED BY

Bernard Knueven

Co-authors: Michael Bynum, Anya Castillo, Carl Laird, Jean-Paul Watson

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

EGRET: Electrical Grid Research and Engineering Tools

EGRET is a Python-based package for electrical grid optimization based on the Pyomo optimization modeling language. EGRET is designed to be friendly for performing high-level analysis (e.g., as an engine for solving different optimization formulations), while also providing flexibility for researchers to rapidly explore new optimization formulations.

Major features:

- Expression and solution of unit commitment problems, including full ancillary service stack
- Expression and solution of economic dispatch (optimal power flow) problems (e.g, DCOPF, ACOPF)
- Library of different problem formulations and approximations
- Generic handling of data across model formulations
- Declarative model representation to support formulation development

EGRET is available under the BSD License at https://github.com/grid-parity-exchange/Egret

EGRET: Commentary

Development of the commitment and dispatch models in EGRET started in 2012

- Funded by an ARPA-E GENI project (scalable stochastic unit commitment)

Predecessor models were validated against models in Alstom's Market Management System

- 2 months on-site effort at Alstom

C. Laird, M. Bynum, and A. Castillo (SNL) have led ACOPF-related efforts in EGRET

- Not the topic of today's talk

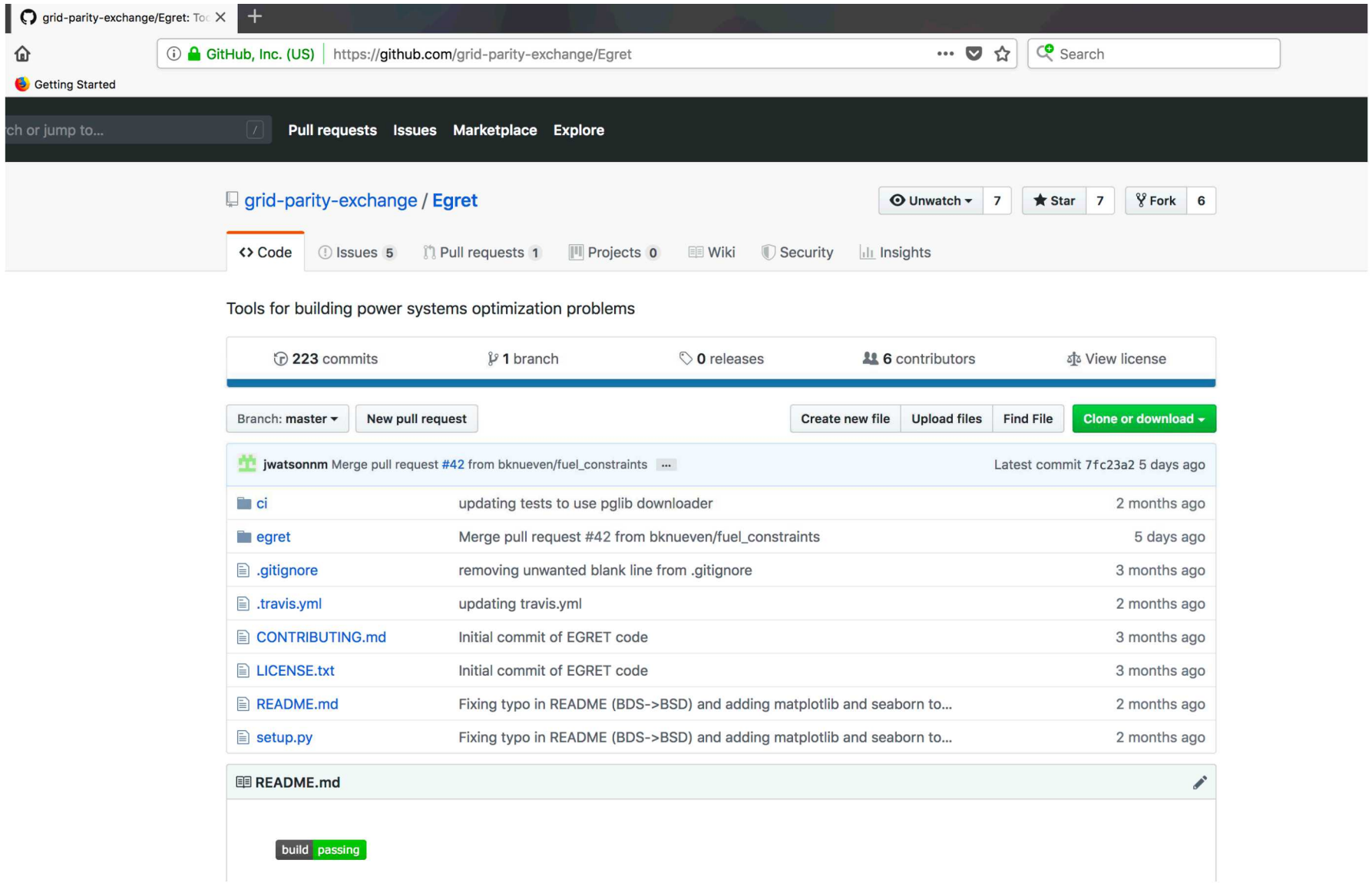
Documentation is lagging capability by a significant margin

- All function, no flash – yes, we are terrible at PR

Still migrating advanced models and capabilities into EGRET

- E.g., from ASU/Sandia/Nexant ARPA-E NODES project

EGRET: GitHub Home Page (Existence Proof)



EGRET Prerequisites

Python >= 3.6

Pyomo

- www.pyomo.org
- Version >= 5.6
- Use ”conda install” or ”pip install”

Solver

- Pyomo is an Algebraic Modeling Language (AML)
- EGRET expresses unit commitment and dispatch models using the Pyomo AML
- Need a mixed-integer linear (commitment) and linear (dispatch) solver
  - Commercial: Gurobi, CPLEX, Xpress
  - Academic: CBC, GLPK
  - In Between: SCIP

The Unit Commitment Model "Zoo"…

A Modular Framework for UC Formulations in EGRET

```
from egret.model_library.unit_commitment.uc_model_generator \
                import UCFormulation, generate_model

## get the formulation from Carrion and Arroyo (2006)
formulation = UCFormulation(status_vars = 'CA_1bin_vars',
                            power_vars = 'basic_power_vars',
                            reserve_vars = 'CA_power_avail_vars',
                            generation_limits = 'CA_generation_limits',
                            ramping_limits = 'CA_ramping_limits',
                            production_cost = 'CA_production_costs',
                            uptime_downtime = 'CA_UT_DT',
                            startup_costs = 'CA_startup_costs',
                            )

## construct the model based on the data md
model = generate_model(md, formulation)
```

Over 100,000 formulations

This instantiates a Pyomo `ConcreteModel` (`model`) based on the data provided in the object `md`, which can be used as part of a script.

The eights components of `UCFormulation` can be changed as easily as modifying a string in this file. Runtime checks to ensure incompatible components are not combined.

Number of implemented formulations per component:

- `status_vars`: 5
- `power_vars`: 3
- `reserve_vars`: 4
- `generation_limits`: 9
- `ramping_limits`: 8
- `production_costs`: 12
- `uptime_downtime`: 5
- `startup_costs`: 9

Paper

On Mixed Integer Programming Formulations for the Unit Commitment Problem

Bernard Knueven
Discrete Math & Optimization, Sandia National Laboratories, Albuquerque, NM 87185, bknueve@sandia.gov

James Ostrowski
Industrial and Systems Engineering, University of Tennessee, Knoxville, TN 37996, jostrows@utk.edu

Jean-Paul Watson
Data Science & Cyber Analytics, Sandia National Laboratories, Livermore, CA 94551, jwatson@sandia.gov

We provide a comprehensive overview of mixed integer programming formulations for the unit commitment problem (UC). UC formulations have been an especially active area of research over the past twelve years, due to their practical importance in power grid operations, and this paper serves as a capstone for this line of work. We additionally provide publicly available reference implementations of all formulations examined. We computationally test existing and novel UC formulations on a suite of instances drawn from both academic and real-world data sources. Driven by our computational experience from this and previous work, we contribute some additional formulations for both production upper bound and piecewise linear production costs. By composing new UC formulations using existing components found in the literature and new components introduced in this paper, we demonstrate that performance can be significantly improved – and in the process, we identify a new state-of-the-art UC formulation.

Key words: Unit commitment, mixed integer programming, mathematical programming formulations

Available on *Optimization Online*: http://www.optimization-online.org/DB_HTML/2018/11/6930.html

EGRET: Other Capabilities Included

Full ancillary service stack
- Seven products
  - Reg-up, reg-down, spinning, non-spinning, supplemental, flex-up, and flex-down reserves

Storage
- Generic, full-fidelity models

Zonal definition support
- Relevant for reserve modeling

Network
- B-theta
- PTDF, PTDF+Losses (full and lazy)

Fuel Constraints
- Dual-fuel generator model

Future Development Directions

Intellectual investment in commitment / dispatch libraries is massive

- Decades of person-years invested in aggregate
- Very intricate knowledge of mathematical programming required for performance advances
- Significant advances come from the lab community, not necessarily industry
- Screams for consolidation across the complex

Security Constraints

- N-1, sophisticated transmission constraint filtering

Drive toward open and transparent models is critical

- Necessary condition for V&V of power systems operations simulation models

Enhanced Fuel Supply Constrained Models

- Co-optimization of NG and BES
- Avoiding non-linearities of full NG system models