

THE UNIVERSITY OF  
**AUCKLAND**  
Te Whare Wānanga o Tāmaki Makaurau  
NEW ZEALAND

# GOCPI

A Scalable Energy Modelling Solution

Developed by Connor McDowall

Supervised by Rosalind Archer

# Facing the Energy Transition



- ✓ The imminence of the energy transition is clear after my experience at ExxonMobil Australia.
- ✓ Literature review informs the need for rapid transformation as economic models predict unfavourable consequences if no swift action.
- ✓ Sustainable investment is driven by Net Present Value (NPV) analysis and the ability to generate returns for investors.
- ✓ Sustainable technologies have seen significant cost reductions over the last decade improving the feasibility of the transition.
- ✓ However, there are educational disparities between policy makers, government, private companies, stakeholders and voters.

# Empower users to influence policy



- ✓ The reasons creating these issues is the sophistication and inaccessibility of energy modelling.
- ✓ Energy modelling usually requires:
  - Proprietary data.
  - The understanding of LP, Integer LP, MIP and/or NLP optimisation techniques.
  - Access to expensive commercial solvers.
  - A thorough understanding of energy systems, mathematics, economics and finance.
- ✓ This complexity creates difficulties in evaluating energy investment, policy and their alignment to the United Nation's Sustainable Development Goals and Paris Agreements.
- ✓ My proposed solution is to develop an accessible, scalable energy system modelling tool.
- ✓ The product will remove this sophistication and enable users to model their own energy systems to inform investment and policy.

# Methodology and Implementation



- ✓ A comprehensive literature review on energy, emissions, the economy, policy, obstacles, challenges and energy modelling framed the problem and addressed the needs the product will fulfil.
- ✓ A high-level overview of the methodology and implementation for the GOCPI project is as follows:
  - ✓ Adapt The Integrated Market E-Form System (TIMES) modelling methodology.
  - ✓ Adapt the OseMOSYS modelling methodology.
  - ✓ Understand and create energy systems.
  - ✓ Develop a Python-based open source scalable energy modelling tool.
  - ✓ Develop forecasting methodologies for energy systems.
  - ✓ Adapt IBM technologies to facilitate optimisation.
  - ✓ Develop an web-based user interface to both inform and distribute the product.

# The Incumbent's Limitations – TIMES

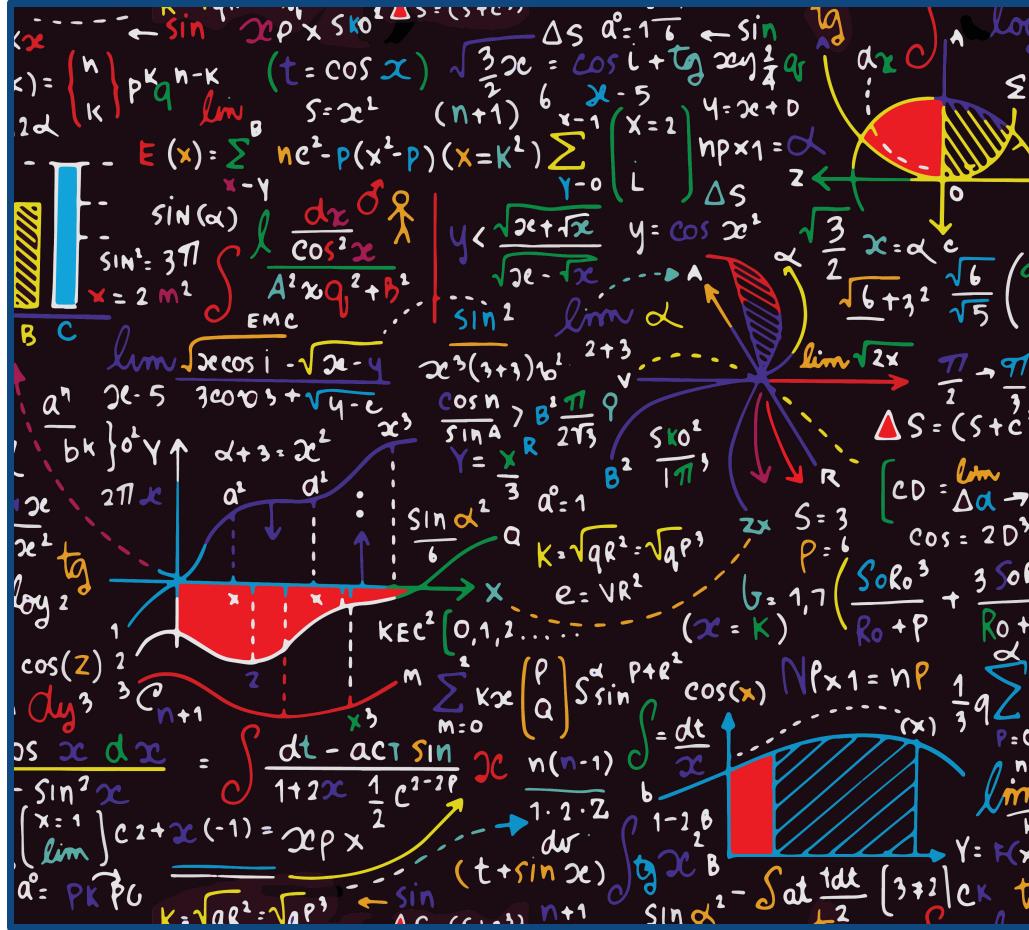


- ✓ The Integrated Market E-Form System (TIMES) is the legacy methodology used to inform the energy scenarios for both the World Energy Council and New Zealand Business Energy Council.



- ✓ A existing modelling system was adapted using enterprise versions of GAMS Studio and the Versatile Data Analyst (VEDA).
- ✓ Excel-based functions and custom macros were to be used to create a scalable template.
- ✓ Reproducibility, integration and complexity issues led to the abandonment of this approach in favour of an alternative.

# Discovering an Alternative – OseMOSYS



- ✓ The Open Source Energy Modelling System (OseMOSYS) was the chosen alternative for modelling energy systems.
- ✓ OseMOSYS is user development by a community of developers with contributions made directly to the OseMOSYS GitHub Repository. There are a couple versions available: GAMS, GNU Mathprog and Python (Pyomo).
- ✓ OseMOSYS is a linear optimisation problem describing an energy system. There are 11 sets, 52 parameters, 67 variables, 94 constraints and 1 objective function.
- ✓ Several sets underpin parameters, constraints and variables: Year, Technology, Timeslice, Fuel, Emission, Mode of Operation, Region, Season, Day Type, Daily Time Bracket and Storage.

$$\text{Min: } \sum_r \sum_y \text{TotalDiscountedCost}_{r,y}$$

# Breaking Down Energy Systems



- ✓ A reference energy system is the main concept underpinning the system. It describes the network flows amongst the production, conversion and consumption of different fuels using different technologies.
- ✓ The system models trade relationships between regions. Individuals, companies, towns, cities, countries and continents are examples of all regions you can represent using the OseMOSYS methodology.
- ✓ The objective function minimises total discounted costs derived by each regions' discount rate. Salvage, operating and capital expenditure are considered with the discount rate depending a region's mix of equity, debt and financing costs.
- ✓ The OseMOSYS model uses emissions and renewable technology constraints to drive sustainable outcomes.

# A Powerful Package



- ✓ The GOCPI prototype was developed and distributed as a Python-based open source scalable energy modelling tool.
- ✓ GOCPI required best practise software development. The project made use of several technologies: Version control using Git and GitHub, Python 3.7.6, Anaconda, PyPI, IBM ILOG CPLEX Optimization Studio (CPLEX Python APIs), IBM Watson Machine Learning and Yapf.
- ✓ The model is an adaptation of the OseMOSYS methodology, formulated in GNU Mathprog and integrated into Excel and Python. The GNU Mathprog structure is stored within an Excel spreadsheet. A user may toggle constraints or adapt the objective function to drive different outcomes.
- ✓ The prototype enables the user to formulate their own energy systems in python, perform forecasting functionalities, generate linear programmes and solve them using commercial solvers.

# Intersecting Technology, Energy & Finance



- ✓ The GOCPI saw the intersection of technology, energy and finance in the construction of energy systems.
- ✓ Deconstructed the Utopian energy system exemplar from OseMOSYS to formulate standardised modelling approach.
- ✓ Partially developed an Australia/New Zealand energy system with a bi-lateral trade relationship.
- ✓ Developed a reference energy system for both countries using data available from the Ministry of Business, Innovation and Education (MBIE, NZ) and the Department of Environment and Energy (AUS).
- ✓ Determined discount rates for both New Zealand and Australia using treasury reports and financial statements.

# Programming the Foundations

```
document.getElementById(div).innerHTML = errEmail;
else if (i==2)
{
    var atpos=inputs[i].indexOf('@');
    var dotpos=inputs[i].lastIndexOf('.');
    if (atpos<1 || dotpos<atpos+2 || dotpos>atpos+5)
        document.getElementById('errEmail').innerHTML = errEmail;
    else
        document.getElementById(div).innerHTML = errEmail;
}
else if (i==5)
    document.getElementById(div).innerHTML = errEmail;
```

- ✓ The GOCPI prototype is built with several classes.
- ✓ EnergySystems: A class containing modules to load in existing energy systems. Additionally, this class creates energy model files from the OseMOSYS structure and energy data files to form LP files.
- ✓ CreateCases: A class containing modules to enable the user to build their own energy systems.
- ✓ Forecasting: A class containing modules to formulate base year energy balances from the International Energy Agency's (IEA) energy balances and forecast both energy and financial values.
- ✓ Optimisation: A class containing modules to solve energy system optimisation problems locally using CPLEX or remotely using the IBM Watson Machine Learning service.
- ✓ Navigation: A class containing modules to enable the user to navigate their local directory to access files.

# Driving Optimisation with IBM



- ✓ CPLEX was selected as the commercial solver of choice for the GOCPI prototype. It is readily available for educational institutions and individuals using IBM's Academic Initiative.
- ✓ IBM ILOG Optimisation Studio was installed locally using the IBM Academic Initiative.
- ✓ LP files are created using the glpsol terminal command, from the GLPK package, in a custom python environment.
- ✓ Small LP files are solved locally using the CPLEX APIs accessible from the Optimisation class.
- ✓ Large LP files must be solved using a Python-based OseMOSYS formulation and the IBM Decision Optimisation on the IBM Watson Machine Learning service.
- ✓ Created a standardised method to access IBM cloud services, create deployments, request jobs and solve requests.

# Distributable on the GOCPI Interface



- ✓ The package is distributed on PyPI and GitHub, accessible through a web-based interface.
- ✓ Developed a GitHub repository to version control all code and resources.
- ✓ Developed a custom Python package for deployment. New distribution are uploaded using Twine and downloaded using Pip, Python's package management software.
- ✓ Developed a website to act as an interface for the GOCPI prototype. The website is built using Jekyll, a simple static site generator. Jekyll is written in Ruby and converts Markdown files to HTML. The website is hosted using GitHub pages
- ✓ Connected a Google domain to direct a user to the interface. The website displays this presentation and links to the GOCPI prototype.

# Success So Far



- ✓ The GOCPI Prototype is readily available and distributable.
- ✓ Created a standardised modelling process to create user-defined energy systems.
- ✓ Partially developed an Australia and New Zealand energy system with a bi-directional trade relationship.
- ✓ Created 80+ new distributions for the GOCPI Prototype as at Monday 19<sup>th</sup> of October.
- ✓ Created a Web-based interface on my [personal website](#). I used my personal website as purchased a domain name for additional uses outside the scope of the project.
- ✓ Created a pipeline to continue developing the GOCPI prototype.

# Pursuing GOCPI's Full Potential



- ✓ Continue developing Australia and New Zealand energy system example.
- ✓ Adopt new forecasting methodologies to project the needs of energy systems in the future. These include methods common in financial services and data science.
- ✓ Make adjustments to the objective function to account for emissions from technologies and carbon taxes.
- ✓ Continue to develop GOCPI classes to improve usability and convert the OseMOSYS model to Python to utilise IBM Watson Machine Learning and Cloud services.
- ✓ Create user interfaces to display the outputs of energy systems and make comparisons to align with the United Nation's Sustainable Development Goals (UNSDG) and Paris Agreements.

