Advanced Tools for Conceptual Design

Qi Chen[a], Michael Bynum[b], Jaffer Ghouse[c], Carl Laird[b], Ignacio Grossmann[a], David Miller[c]
[a]Carnegie Mellon University, [b]Sandia National Laboratories, [c]National Energy Technology Laboratory

MOTIVATION: design advanced energy processes

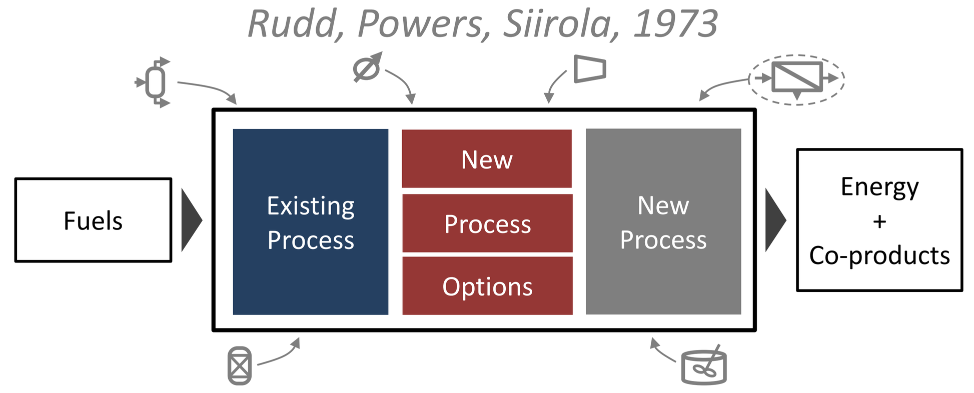
Goal

- New tools to **automate** design of **flexible**, **efficient** combustion energy systems
- **Eliminate** need for **manual simulation** of several varied process **alternatives**

Process synthesis

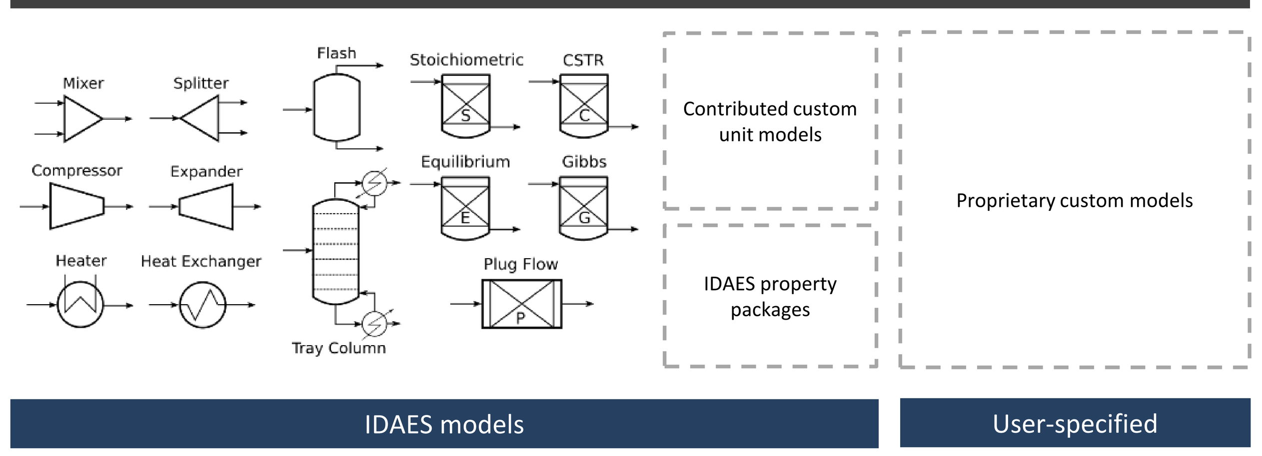
The **generation** of process flowsheet **alternatives** to find best design

Rudd, Powers, Siirola, 1973

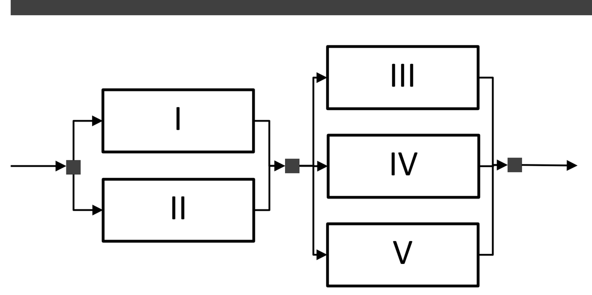


PYOSYN: synthesis workflow

Define design problem

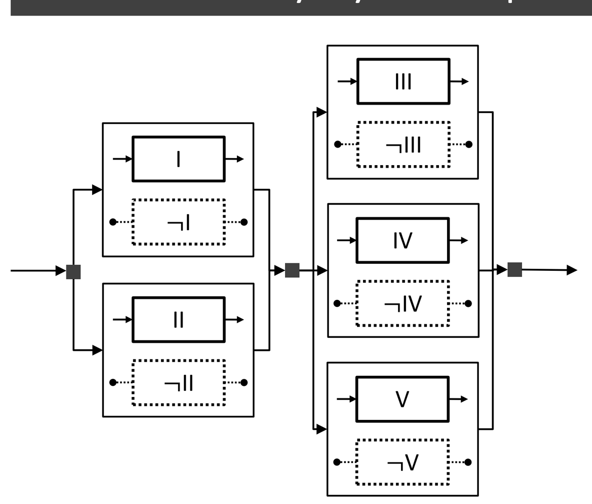


Generate alternatives



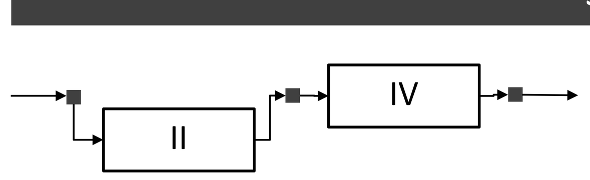
- Define all potential process units
- Define all possible interconnections

Pyosyn conceptual design transformation



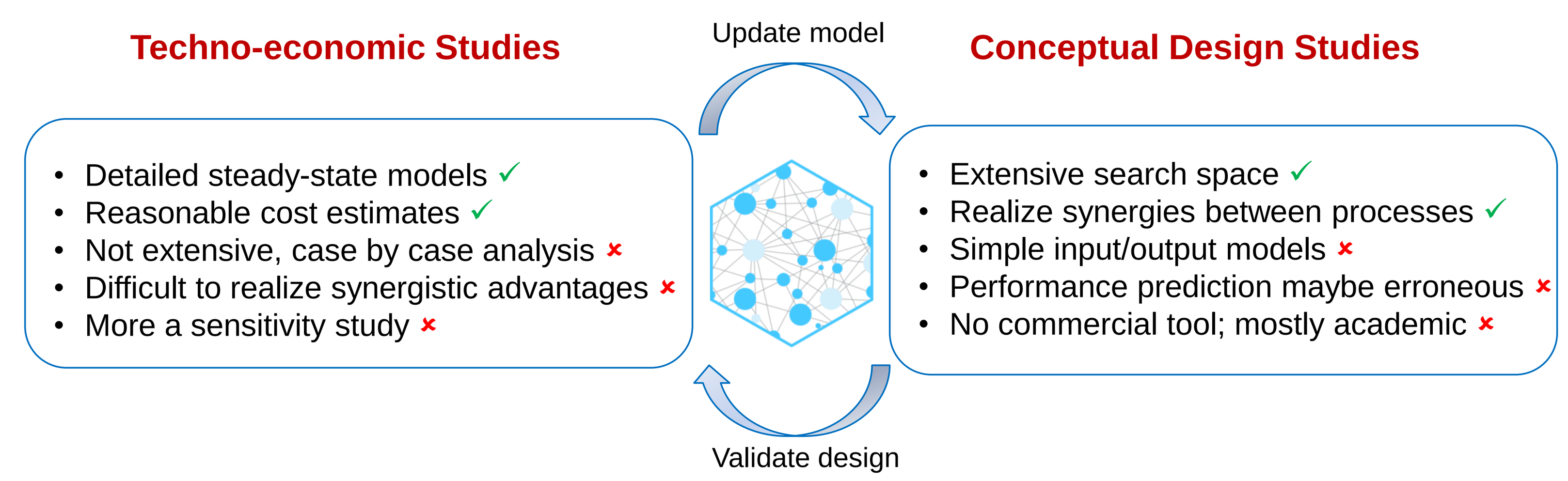
- Automatic model transformation
  - Inserts "unit exists" or "unit absent and no flow" logic
- Define custom logic:
  - Must have *I* or *II*
  - Must have *III* or *IV* or *V*

Solve

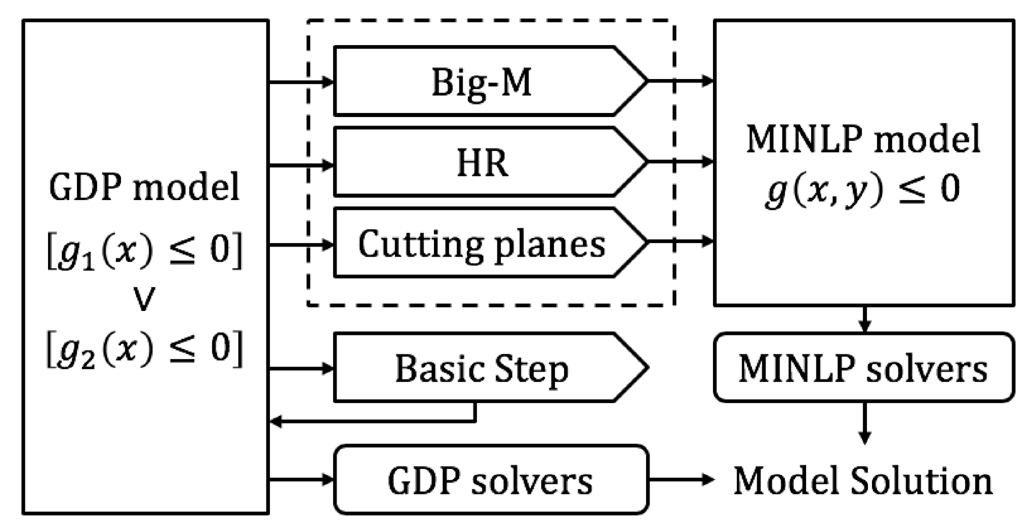


- Solution gives optimal configuration

CONCEPTUAL DESIGN: guide techno-economic analysis

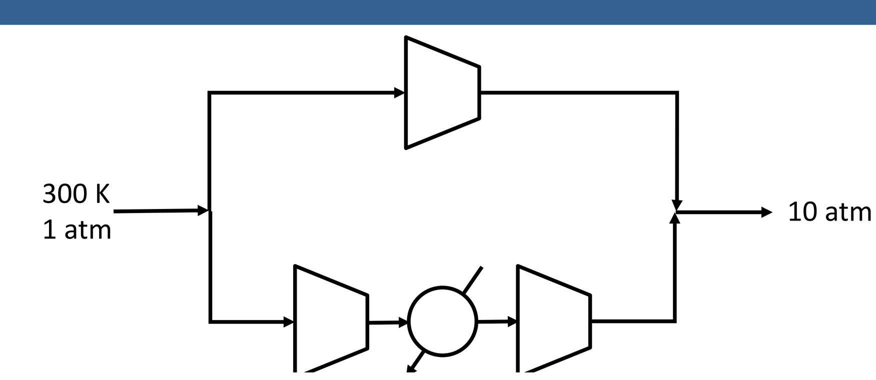


FLEXIBLE SOLUTION STRATEGIES: advanced algorithms



- Reformulate to MINLP
  - Utilize any MINLP solver (e.g. MindtPy)
- Direct solution of GDP
  - GDPopt (logic-based outer approximation)
  - GDPbb (nonlinear disjunctive branch and bound)
- Logic-based relaxation tightening
  - Basic steps
  - Disjunctive domain reduction

IDAES EXAMPLE: compression of an ideal gas




```
m = pe.ConcreteModel()
m.fs = fs = FlowsheetBlock(default={"dynamic": False})
fs.properties = props = PhysicalParameterBlock(default={"valid_phase": 'Vap'})

fs.feed = feed = Feed(default={"property_package": props})
feed.flow_mol.fix(1)
feed.pressure.fix(0.101325)
feed.temperature.fix(3)

fs.product = product = Product(default={"property_package": props})
product.pressure[0.0].fix(1.01325)
```

Feed and product specs

```
fs.single_stage_compressor_disjunct = sscd = gdp.Disjunct(concrete=True)
sscd.compressor = IdealGasIsentropicCompressor(default={"property_package": m.fs.properties,
                                                        "has_phase_equilibrium": False})

sscd.stream1 = network.Arc(source=feed.outlet, destination=sscd.compressor.inlet)
sscd.stream2 = network.Arc(source=sscd.compressor.outlet, destination=product.inlet)
```

Disjunct for single-stage compressor

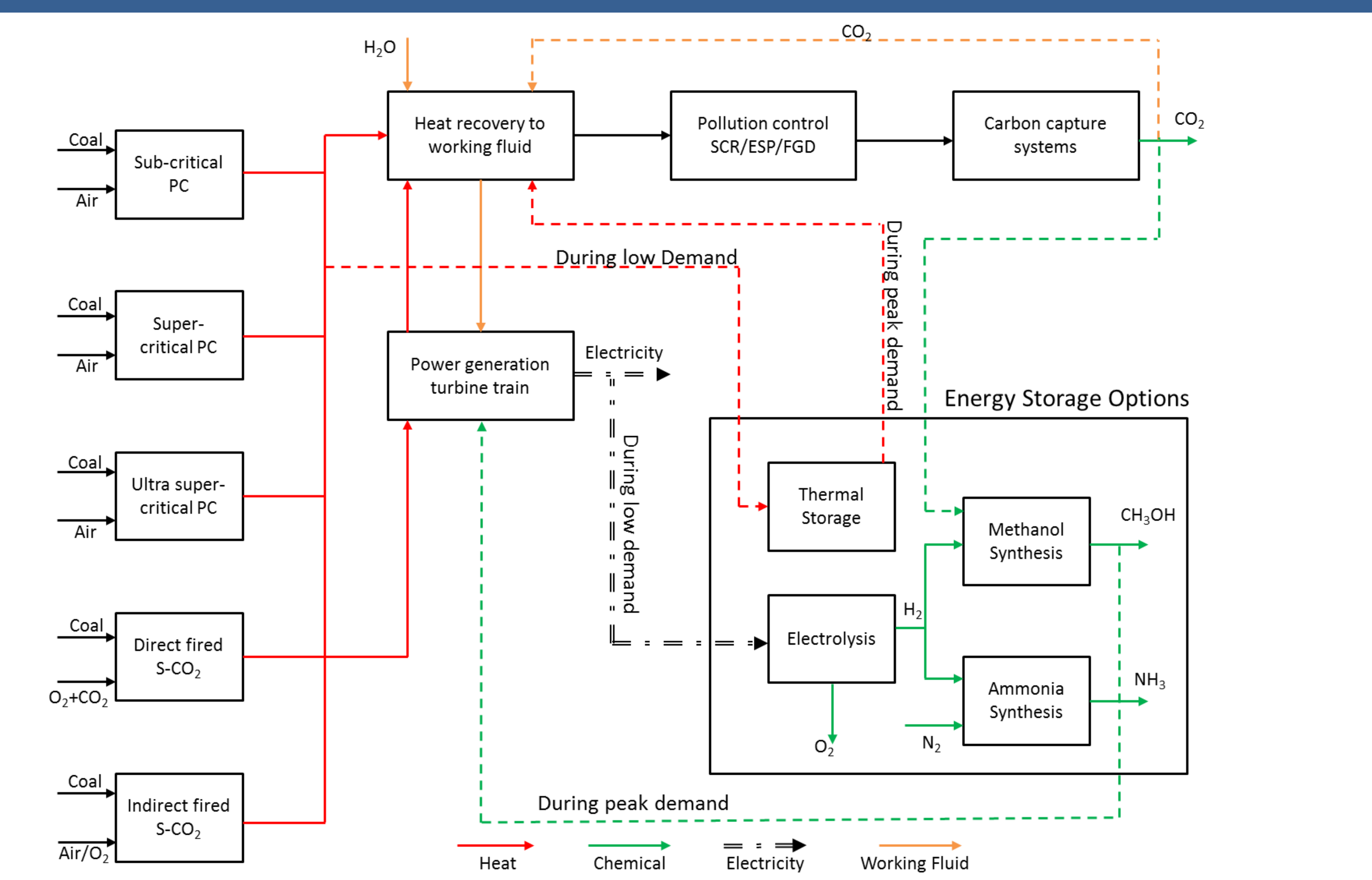
```
fs.two_stage_compressor_disjunct = tscd = gdp.Disjunct(concrete=True)
tscd.compressor1 = IdealGasIsentropicCompressor(default={"property_package": m.fs.properties,
                                                         "has_phase_equilibrium": False})
tscd.compressor2 = IdealGasIsentropicCompressor(default={"property_package": m.fs.properties,
                                                         "has_phase_equilibrium": False})
tscd.cooler = Heater(default={"property_package": props, "has_phase_equilibrium": False})
tscd.cooler.heat_duty[0.0].setub(0)  # it is a cooler
tscd.cooler.outlet.temperature[0.0].setlb(3)

tscd.stream1 = network.Arc(source=feed.outlet, destination=tscd.compressor1.inlet)
tscd.stream2 = network.Arc(source=tscd.compressor1.outlet, destination=tscd.cooler.inlet)
tscd.stream3 = network.Arc(source=tscd.cooler.outlet, destination=tscd.compressor2.inlet)
tscd.stream4 = network.Arc(source=tscd.compressor2.outlet, destination=product.inlet)
```

Disjunct for two-stage compressor

SOLVE!

SUPERSTRUCTURE: advanced energy process with storage



TARGET: conceptual design for coal-fired power plants of the future

- Market competitiveness of new and existing coal-fired power plants of the future
  - Capital cost, operating cost, efficiency
  - Flexibility: reliable, efficient part-load operation, rapid ramping, short up/down-time
  - Stretch performance without cost-prohibitive maintenance side-effects (e.g., stress / fatigue)
- Requires:
  - Reliable, accurate models of existing facilities (steady-state and dynamic) [Another Poster]
  - Exploration of modern alternatives for plant retrofit and new design to increase capabilities in terms of efficiency, flexibility, and product opportunities
  - **Conceptual design tools for Coal-Fired Power Plants of the Future**
- Key elements:
  - Advance the capabilities of the conceptual design framework (decomposition strategies, modularity, uncertainty, flexibility, and multi-objective)
  - Develop standardized conceptual design models (keeping flexibility for specialized equipment)
  - Execute framework for retrofit and new design options

Future Targets:

- Initial focus on conceptual design, retrofit of subsystems of the coal-fired power plant (power and energy storage options, heat integration and design of turbine train for improved efficiency under part-load)
- Exploration of flowsheet options: **Firing options**: Super/Ultra-super critical, direct and indirect-fired super-critical CO2 cycles. **Storage options**: chemical / thermal, **Emissions and pollution control**: carbon capture, alternative flowsheets, desulphurization, NOX, **Upstream**: gasification, poly-generation
- Overall conceptual design of new, modular coal-fired power plants with improved efficiency and flexibility

CONCLUSIONS AND CONTACT

- Developing **computational tools** to support **conceptual design**
- IDAES framework enables **integrated workflow** for generating **novel processes**, from developing custom models to specifying equipment

Qi Chen, qichen@andrew.cmu.edu; Ignacio Grossmann, grossmann@cmu.edu; Michael Bynum, mlbynum@sandia.gov

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and 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-NA-0003525.

National Energy Technology Laboratory · Berkeley Lab · Sandia National Laboratories · Carnegie Mellon · West Virginia University · U.S. Department of Energy