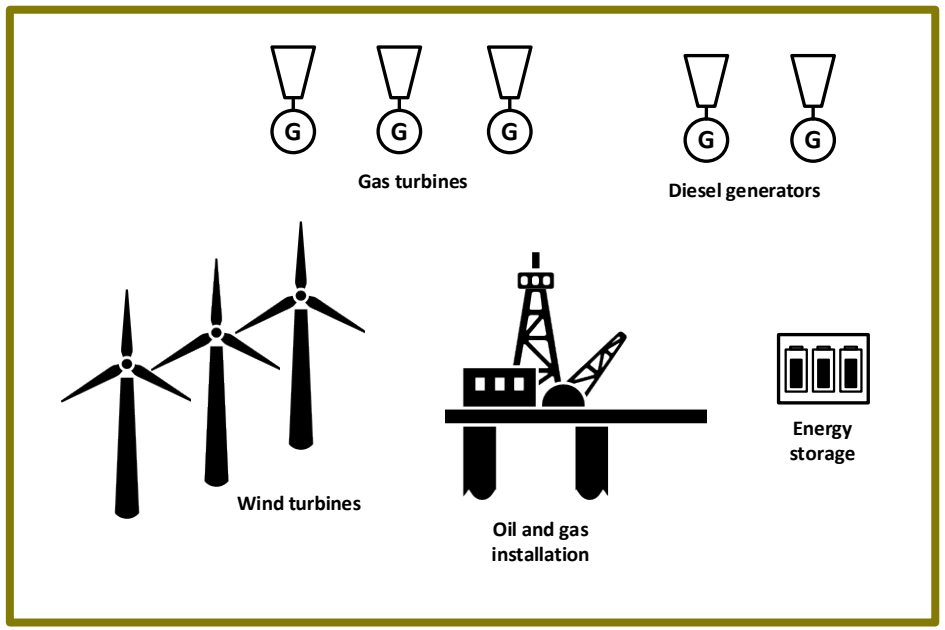
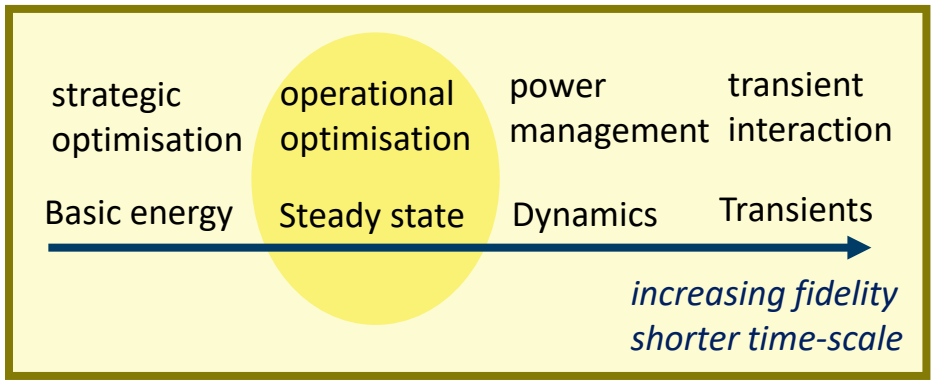
Wind power integration in offshore energy systems – operational optimisation

Harald G Svendsen

SINTEF Energy Research, Trondheim, Norway
harald.svendsen@sintef.no





Motivation:

- Urgent need to remove carbon emissions from oil and gas extraction activities, today mainly coming from the use of gas turbine generators using fossil fuels
- Power from shore is costly
- Local supply from offshore wind turbines is attractive alternative

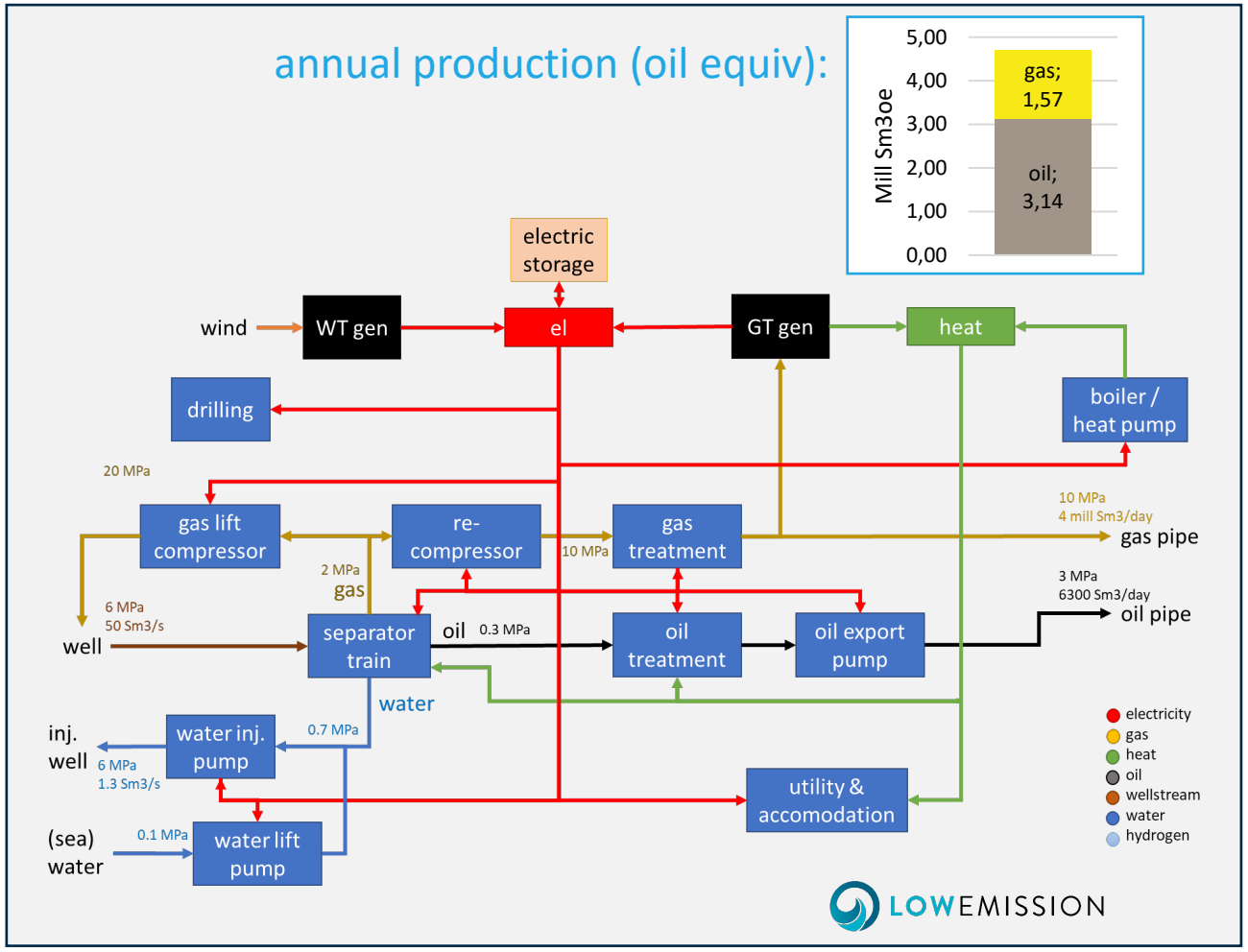
→ Isolated energy systems with variable energy supply and with different sources of flexibility
→ New ways to operate, need new tools for energy system planning

Objective:

Identify the potential for carbon emission reductions with different offshore energy system configurations and operating strategies respecting relevant constraints

SINTEF

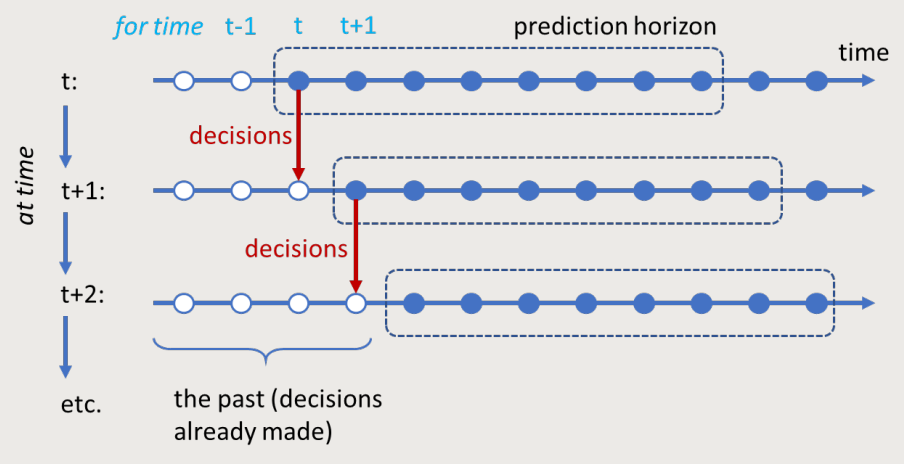
LowEmission Oil and Gas Open (*LEOGO*) Reference Platform



- Oil and gas platform
  - gas lift, water injection
  - produced oil and gas is exported in pipes to another platform
  - "typical" pressure levels
  - gas/oil ratio (GOR) = 500, water cut = 0.6
- Energy system
  - Gas turbines, (wind power, battery, ...)
- Data
  - Load and wind time-series
  - Component parameters

SINTEF

Optimisation in operation



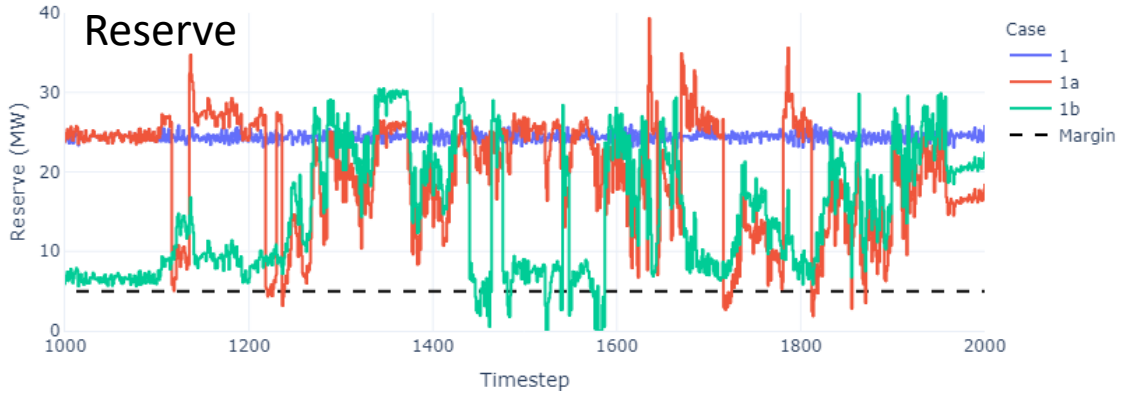

```
def rule_devmodel_compressor_el(model,dev,t,i):
    if model.paramDevice[dev]['model'] != 'compressor_el':
        return pyo.Constraint.Skip
    if i==1:
        ''' power demand depends on gas pressure difference'''
        node = model.paramDevice[dev]['node']
        lhs = model.varDevicePower[dev,t]
        k = model.paramDevice[dev]['eta']
        rhs = k*(model.varGasPressure[(node,'out',t)]
                 -model.varGasPressure[(node,'in',t)])
        return (lhs==rhs)
    elif i==2:
        '''gas flow in equals gas flow out'''
        lhs = model.varDeviceFlow[dev,'gas','in',t]
        rhs = model.varDeviceFlow[dev,'gas','out',t]
        return (lhs==rhs)
    elif i==3:
        '''el in equals power demand'''
        lhs = model.varDeviceFlow[dev,'el','in',t]
        rhs = model.varDevicePower[dev,t]
        return (lhs==rhs)
model.constrDevice_compressor_el = pyo.Constraint(model.setDevice,
          model.setHorizon,pyo.RangeSet(1,3),
          rule=rule_devmodel_compressor_el)
```

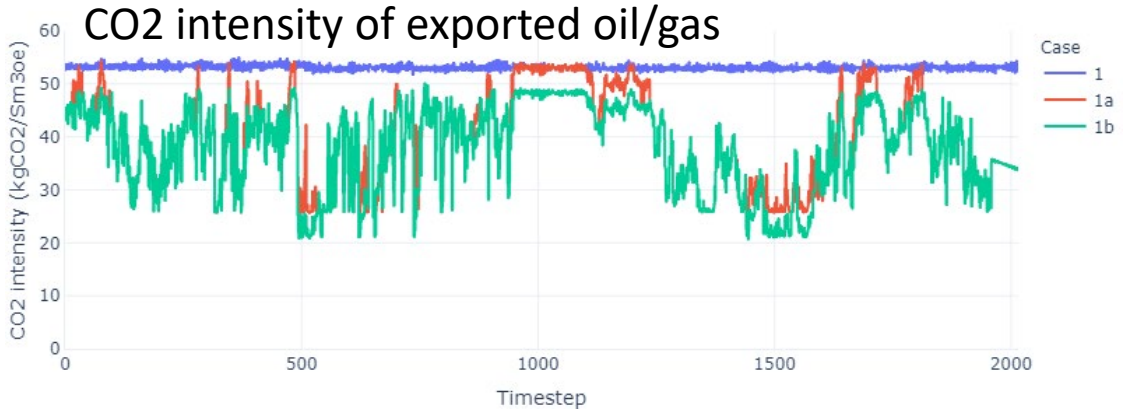
Offshore Oil and Gas Energy System Operational Optimisation Tool (Oogeso)

https://bitbucket.org/harald_g_svendsen/oogeso

- Rolling horizon, considering wind predictions for the next hours
- Integrated energy system description linking energy demand and oil/gas production
- Output:
  - Operational set-points
  - start/stop signals
- Implemented as open-source Python package (Oogeso)

SINTEF





Three simulation cases:

1: Base case without wind
1a: Wind turbines (24 MW)
1b: Wind turbines (24 MW) and batteries (4 MW/4 MWh)

Results

- Quantified CO2 reduction with wind integration
- Batteries provide reserve, allowing gas turbine shutdown, giving additional emission reduction
- Higher emission reduction requires
  - Higher wind power capacity
  - Energy storage

Ongoing work

- Include longer-term storage (hydrogen)
- Flexible energy demand

SINTEF