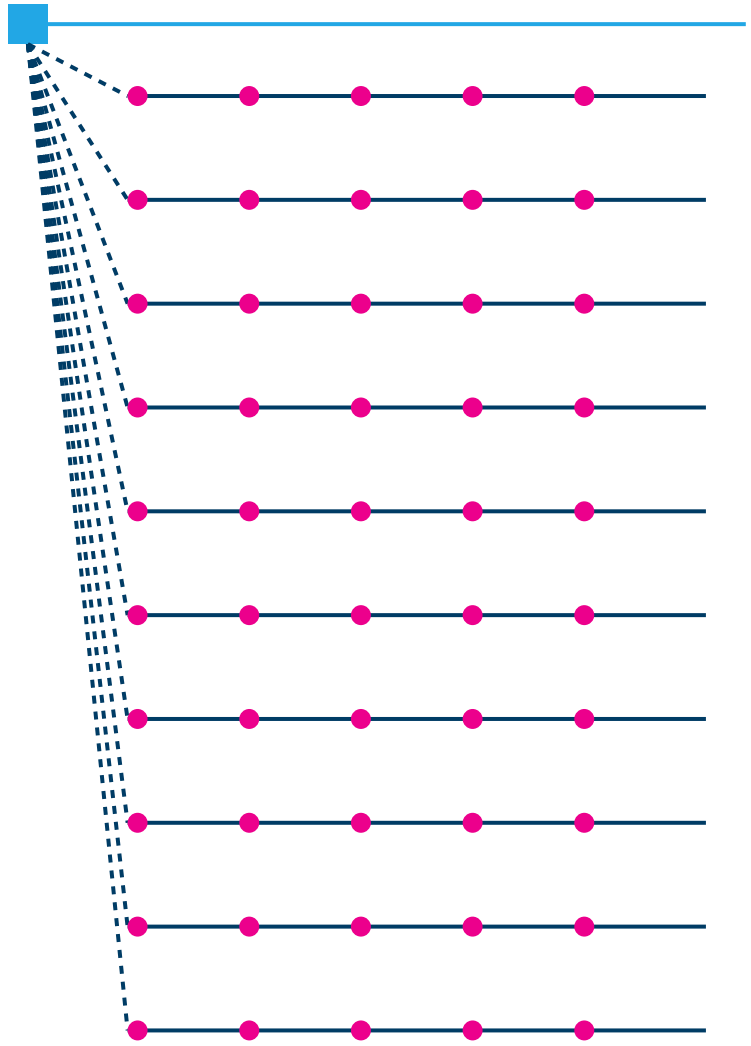
SINTEF
75 years

Stochastic Programming for the Green Transition: Model Composition and Decomposition

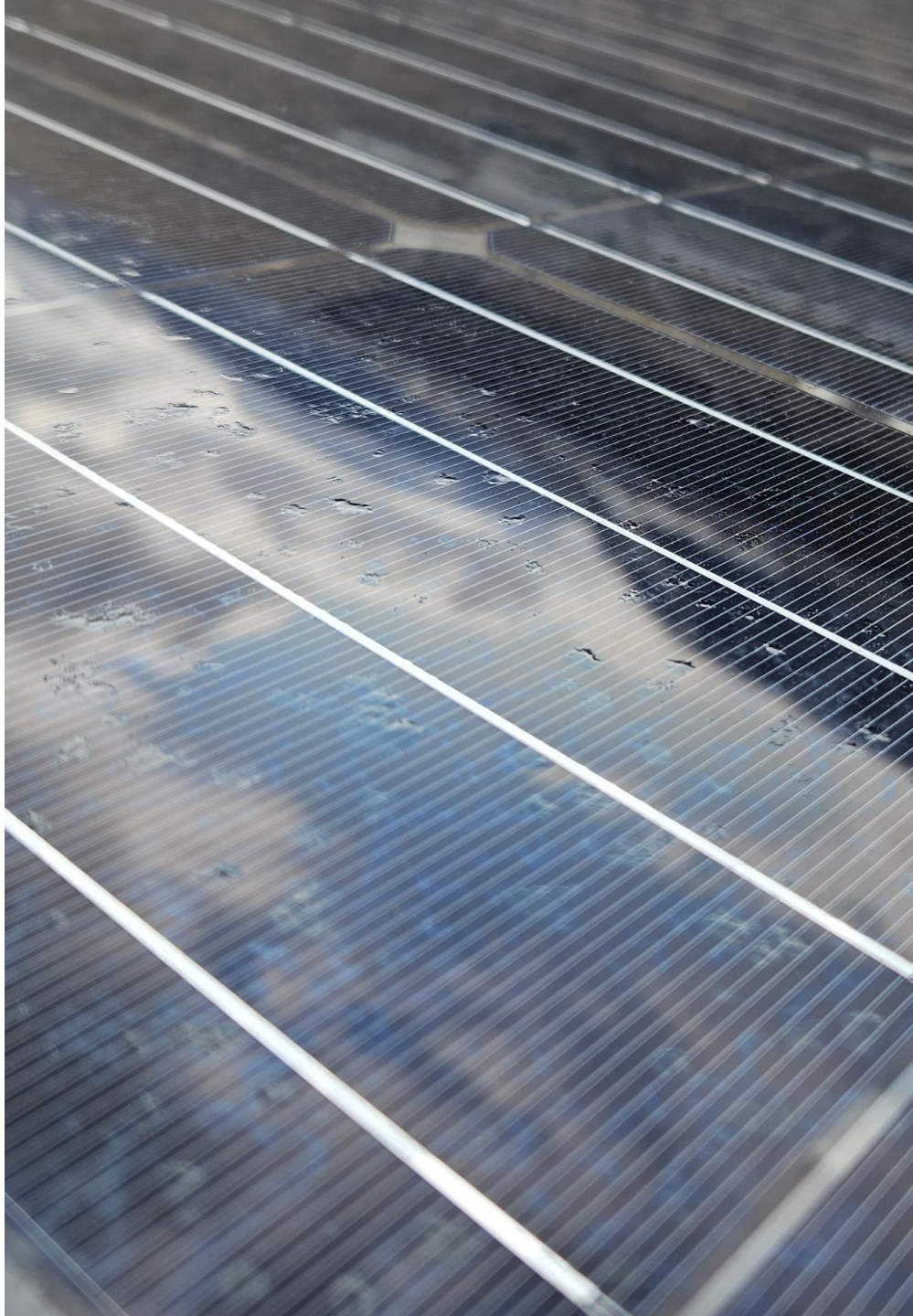
ICSP 2025-07-30

Lars Hellemo and Truls Flatberg
SINTEF



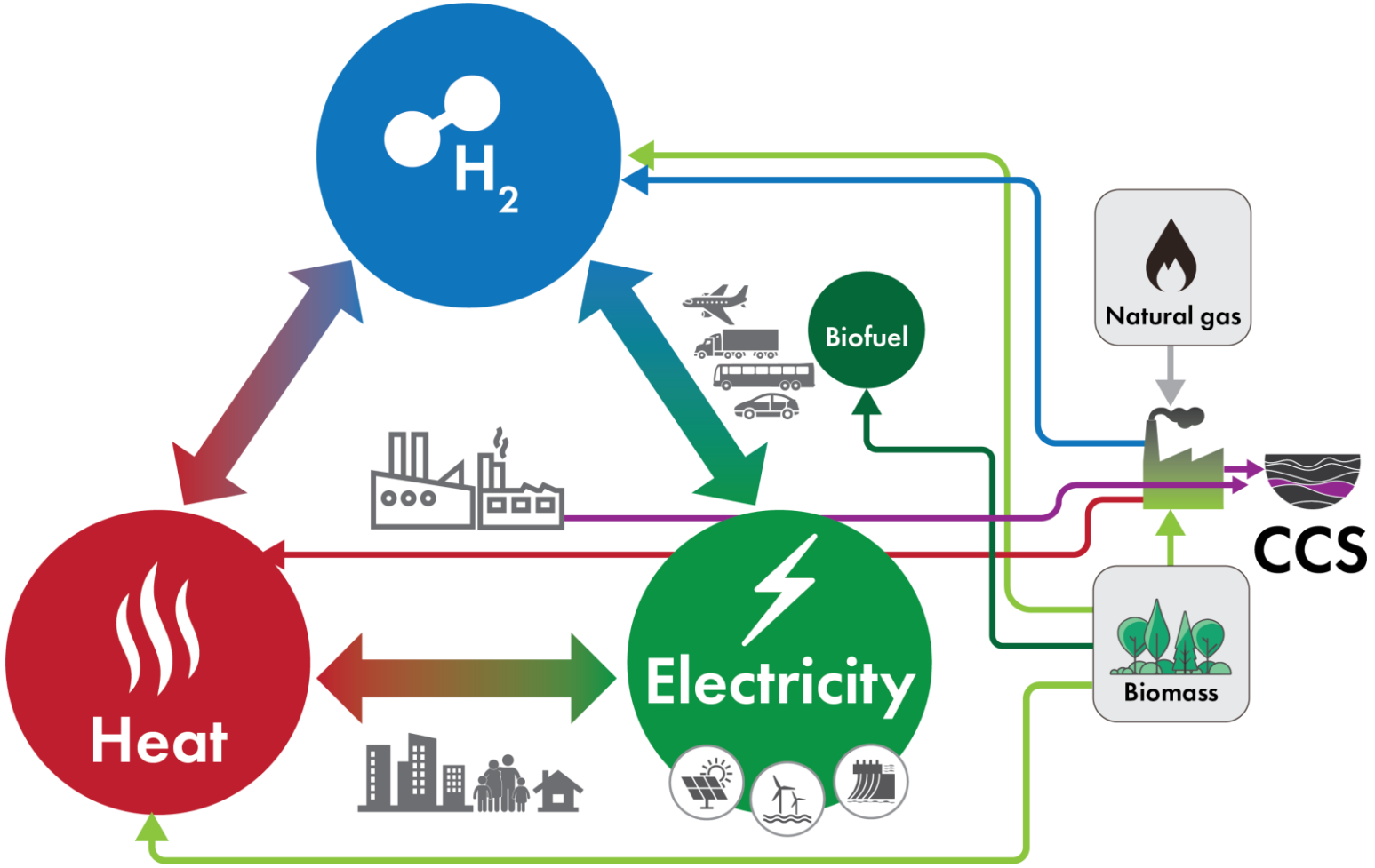
Motivation

- The green transition poses many challenges:
- Renewable Energy Sources (RES)
  - Intermittency
  - Stochasticity
  - Need for system flexibility
- Energy system integration
  - Increasing model complexity
  - Linking models
  - Need for model flexibility



Energy System Integration

A coordinated planning and operation of the energy system "as a whole" across multiple energy carriers, infrastructures and consumption sectors.



Source: *European Commission: An EU strategy for Energy System Integration.*

Pain Points

- Need model flexibility
- Reimplementing is costly and error-prone
- Stochastic models are often specialized; model tailored to solution method or vice versa
- Too little reuse
- Hard to apply state-of-the art algorithms
- Too few stochastic models are used in practice

Composition and Decomposition

- Compose:
  - to form by putting together
- Decompose
  - to separate into constituent parts or elements or into simpler compounds
- Can facilitating composition enable decomposition?

Composition

- Model components
  - Reuse
  - Flexibility
  - Correctness
  - Collaboration
  - Can be combined
- Julia and JuMP
  - Easy to create and publish modules/packages
  - Easy to extend functionality
- Open Source
  - Important in many projects to publish models and assumptions (EU, Research Council of Norway)

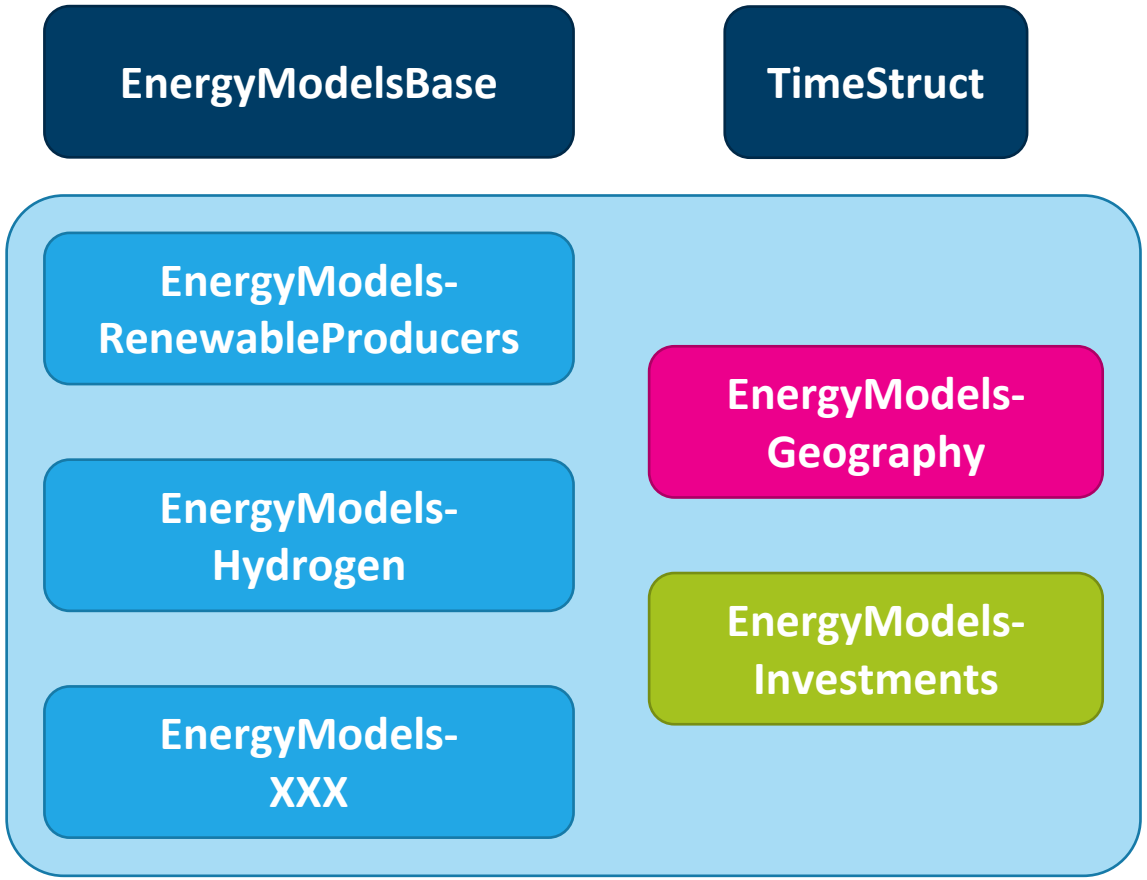
Example: EnergyModelsX

Modular structure:

- Flexible time resolution
- Multiple energy carriers
- Flexible geographic representation
- Easy to combine
- Easy to extend

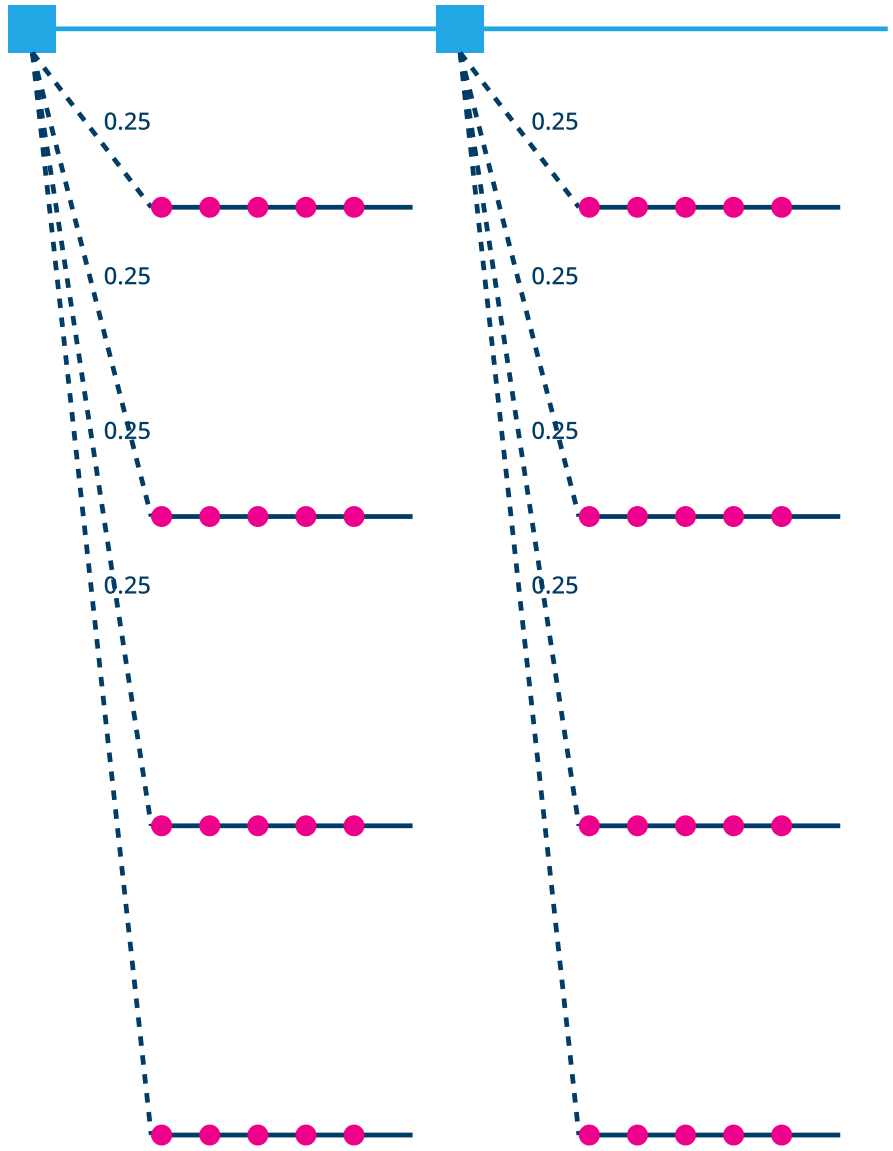
Hellemo et al. 2024

EnergyModelsX @ GitHub



TimeStruct.jl

- Abstraction of time structures
- Facilitate supporting time structures of different complexity
- Well-suited for multi-horizon modelling
- Semantically encode time structure



Flatberg et al. 2025

TimeStruct.jl @ GitHub

SINTEF
75 years

Simplify Modelling

```
using JuMP, TimeStruct

function create_model(periods::TimeStructure, cost::TimeProfile, max_cost)

    model = Model()

    @variable(model, x[periods])

    for sp in strat_periods(periods)
        @constraint(model, sum(cost[t] * x[t] for t in sp) <= max_cost)
    end

    return model
end
```

Strategic Periods and Profiles

```
latex_formulation(create_model(SimpleTimes([1, 1, 1, 5, 5]), FixedProfile(3), 10))
```

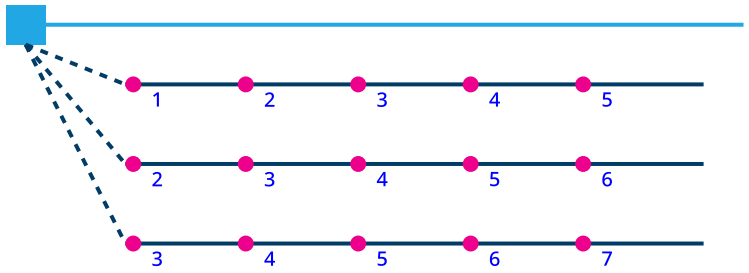
$$3x_{t1} + 3x_{t2} + 3x_{t3} + 3x_{t4} + 3x_{t5} \leq 10$$

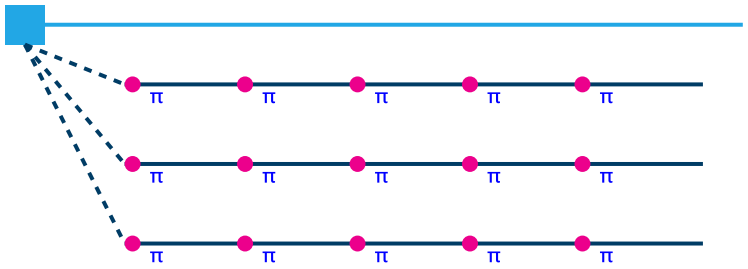
```
latex_formulation(
      create_model(TwoLevel(3, SimpleTimes(5, 1)), StrategicProfile([3, 4, 5]), 10),
   )
```

$$3x_{sp1-t1} + 3x_{sp1-t2} + 3x_{sp1-t3} + 3x_{sp1-t4} + 3x_{sp1-t5} \leq 10$$
$$4x_{sp2-t1} + 4x_{sp2-t2} + 4x_{sp2-t3} + 4x_{sp2-t4} + 4x_{sp2-t5} \leq 10$$
$$5x_{sp3-t1} + 5x_{sp3-t2} + 5x_{sp3-t3} + 5x_{sp3-t4} + 5x_{sp3-t5} \leq 10$$

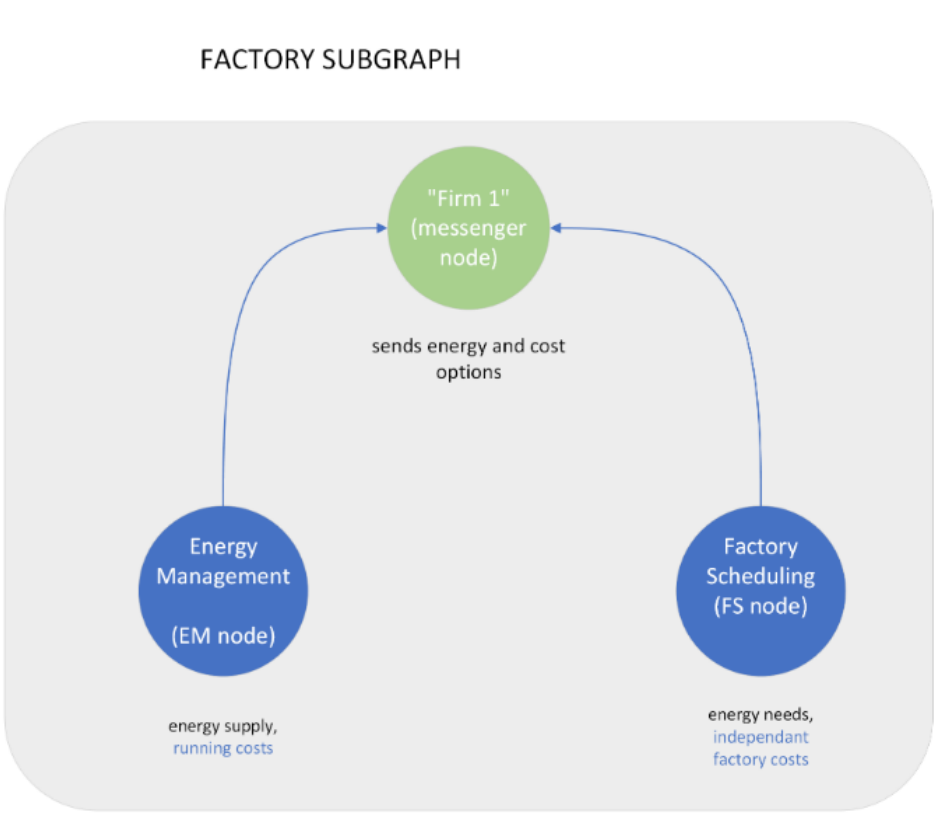
Operational Scenarios and Profiles

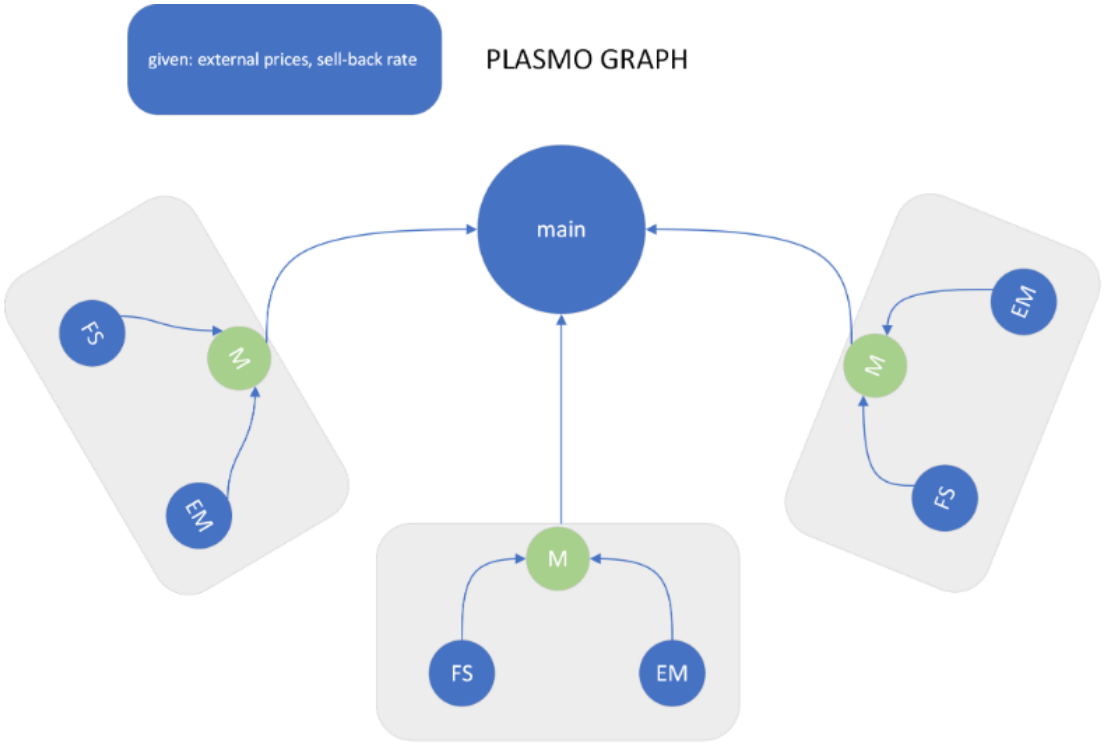
```
ts = TwoLevel(1, 52, OperationalScenarios(3, SimpleTimes(5, 1)))

cost = StrategicProfile([
    ScenarioProfile([
        OperationalProfile([1, 2, 3, 4, 5]),
        OperationalProfile([2, 3, 4, 5, 6]),
        OperationalProfile([3, 4, 5, 6, 7]),
    ]),
])
demand = FixedProfile(π)
```





Simplify Linking

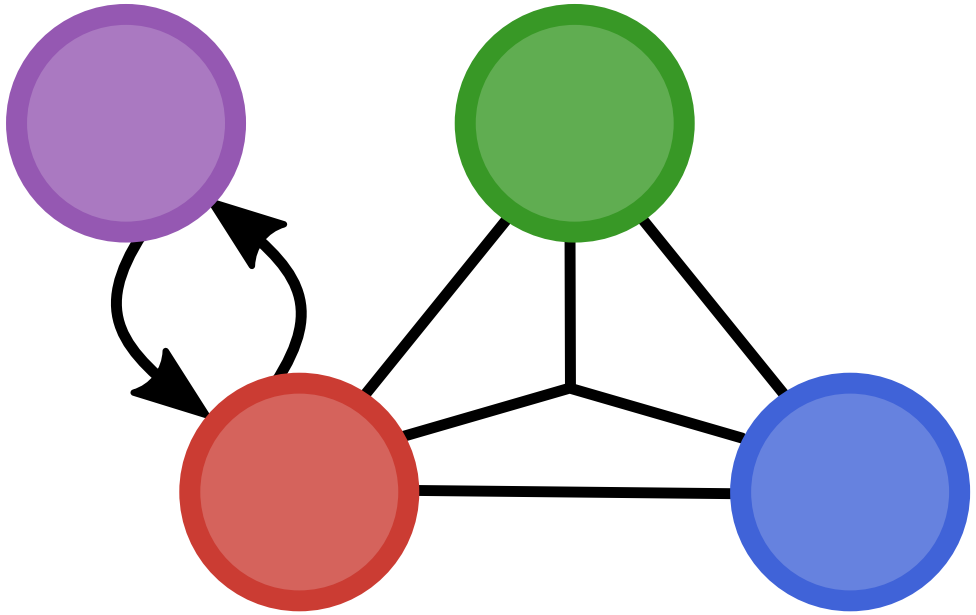




[M. Kitch et al. 2024]

Plasmo.jl

- Graph-based optimization framework
  - Define model for each node
  - Add linking constraints
- Apply graph analysis and decomposition
- Adds a layer “outside” the models



Jalving et al. 2022

Plasmo.jl @ GitHub

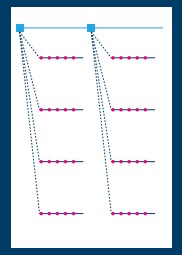
TimeStructDecomposition.jl

- Automatic decomposition
- Algorithms that can be applied across different models based on TimeStruct.jl
- Use semantic encoding of time structure for automatic decomposition (“inside out”)

SINTEF

75 years

JuMP model



scenario

strategic

submodel

submodel

submodel

submodel

submodel

submodel

submodel

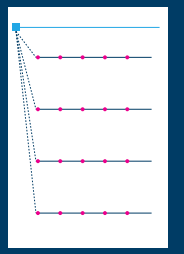
submodel

submodel

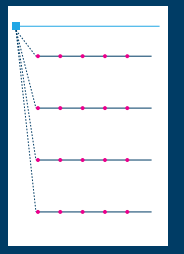
submodel

submodel

submodel



submodel



Apply Benders Decomposition

```
# Build model
model, case = IsfjordRadio.main_fc_case()
# Wrap model as TimeStructProblem
ts_instance = TSD.TimeStructProblem(
                                    model,
                                    TSD.ScenarioDecomposition()
                                );
# Decompose and solve with Benders
m, v, xm, xv = TSD.basic_benders(
                                   ts_instance,    # wrapped model
                                   case[:T],       # TimeStruct
                                );
```

Developing Zero Emission Energy Systems in the Arctic

Lars Hellemo, Sigrid Aunsmo, Saket Adhau, Brage Rugstad Knudsen


Greenland Sea
Baffin Bay
Barents Sea
Kara Sea
Norwegian Sea
Northwestern Passages
Greenland
Iceland
Canada
Hudson Bay
Labrador Sea
Sweden
Finland
Norway
Russia
United Kingdom
Denmark
Ireland
Belarus
Poland
Germany
Ukraine
Austria
Kazakhstan
Mongolia
France
Romania
Italy
Spain
Kyrgyzstan
Portugal
Greece
Türkiye
Turkmenistan
China
Syria
Tunisia
Iraq
Afghanistan
Morocco
Iran
Pakistan
United States
North Atlantic Ocean

Case Studies

Isfjord Radio

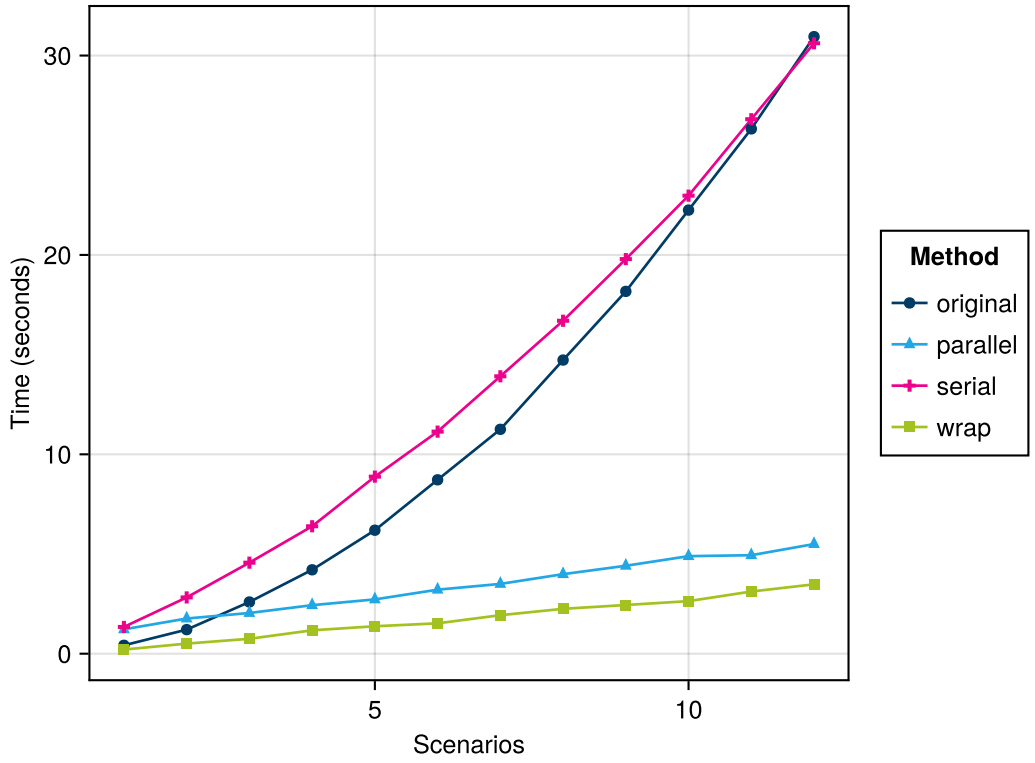
Photo: A. Garreau/UNIS

Longyearbyen

Photo: M. Juel/SINTEF

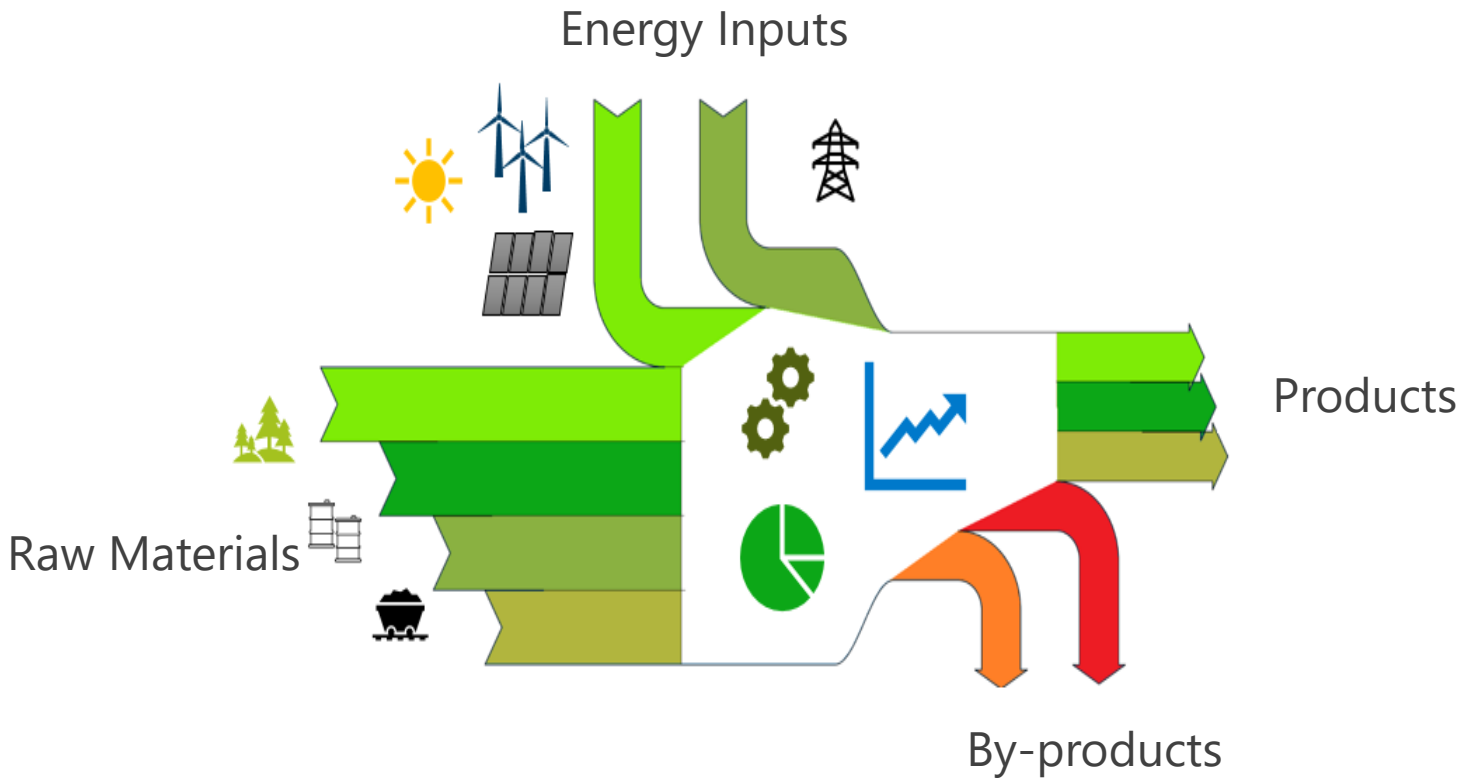
Preliminary Performance Results

- Local energy system design @ Isfjord Radio
  - Battery
  - Thermal storage
- 12 scenarios for renewable energy production
- Benders decomposition
  - Single-threaded (serial)
  - Multi-threaded (parallel)



Composing Green Product Portfolios in the Process Industry

- Sourcing of raw materials with different emission footprints
- Producing premium products with limits on the CO_2 content
- Find the optimal portfolio of products and raw material contracts



Case study: Fertilizer Production

- Single fertilizer plant producing various type of fertilizer
- Options for sourcing electricity and ammonia with lower emission footprints
- Uncertainty considered for raw material prices
- 3 strategic periods, multiple scenarios for each strategic period
- Considerable speed-up observed with Benders' (4 threads):

	Scenarios		
	20	**40**	**60**
Original	7.8 s	18.3 s	49.7 s
Strategic decomposition	3.6 s	9.2 s	25.8 s
Scenario decomposition	2.2 s	4.8 s	7.7 s

Future Work

- Multi-process parallelisation
- API design to facilitate adding more algorithms
- Performance enhancements
- Testing on more models and real-world cases
- On-board InterPlay PhD-students

SINTEF

75 years

1950 – 2025
Technology for a better society

sintef.no/75