Towards digital twins for understanding and managing underground thermal energy storage systems

Øystein Klemetsdal, Odd Andersen, Stein Krogstad, Applied Computational Sciences, SINTEF Digital

III International Conference on Computational Science and AI in Industry (CSAI 2023), Trondheim, Norway, 28-30 August, 2023.

Presentation outline

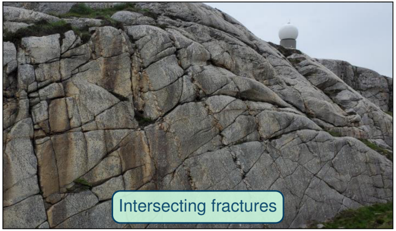
- Introduction
- Digital twins for UTES
- Numerical examples
- Conclusions

Presentation outline

Introduction
Digital twins for UTES
Numerical examples
Conclusions

Underground thermal energy storage (UTES)

- The subsurface is an excellent candidate for scalable energy storage
  - Circulate water through fractured bedrock (fractures $\approx$ fins of a heat exchanger)
  - Charge with excess heat from e.g., industrial processes/waste incineration
  - Constant discharge of base heat, rapid discharge of heat in periods of high demand
- Complex geology (horizons, faults, intertwined fracture networks, ...)
- Complex operation (multiple wells, heaters, heat pumps, heat exchangers, ...)



Underground thermal energy storage (UTES)

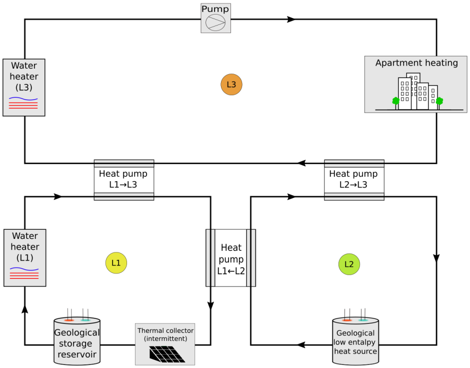
- The subsurface is an excellent candidate for scalable energy storage
  - Circulate water through fractured bedrock (fractures $\approx$ fins of a heat exchanger)
  - Charge with excess heat from e.g., industrial processes/waste incineration
  - Constant discharge of base heat, rapid discharge of heat in periods of high demand
- Complex geology (horizons, faults, intertwined fracture networks, ...)
- Complex operation (multiple wells, heaters, heat pumps, heat exchangers, ...)
- To justify investments and fully utilize potential of underground thermal energy storage, **numerical simulation and optimization** is imperative.

Here: Show how a fully differentiable geoenergy simulator can be integrated in a larger system model, and practically used for operational support and iterative model tuning
$\rightarrow$ **Towards digital twin system for underground thermal energy storage**

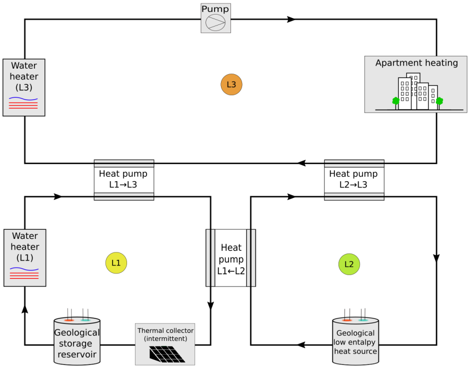
Presentation outline

- Introduction
- Digital twins for UTES
- Numerical examples
- Conclusions

Underground thermal energy storage system



Underground thermal energy storage system

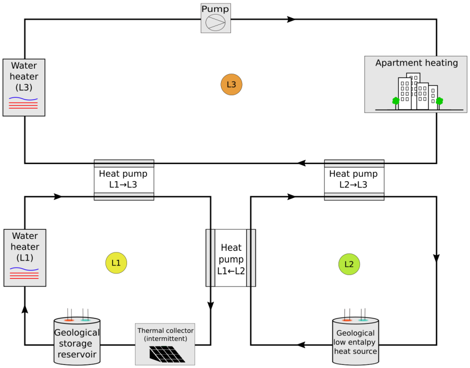


PDE-constrained optimization

Minimize C (e.g., cost of delivering heat to apartment complex), while ensuring conservation of mass/thermal energy

$$\min_u C(x(u)) \quad \text{such that} \quad \mathcal{S}(x(u), u, \theta) = 0$$

Underground thermal energy storage system



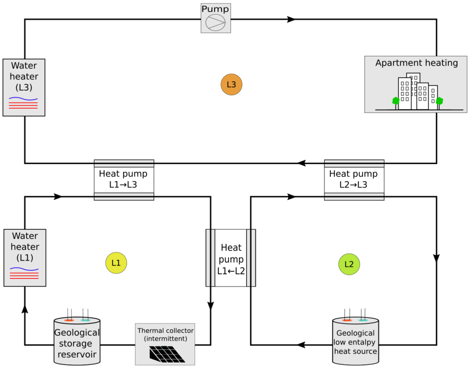
PDE-constrained optimization

Minimize C (e.g., cost of delivering heat to apartment complex), while ensuring conservation of mass/thermal energy

$$\min_u C(x(u)) \quad \text{such that} \quad \mathcal{S}(x(u), u, \theta) = 0$$

Controls (pump rates, energy input, ...)

Underground thermal energy storage system



PDE-constrained optimization

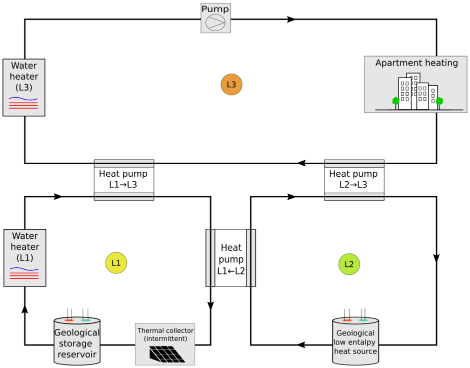
Minimize C (e.g., cost of delivering heat to apartment complex), while ensuring conservation of mass/thermal energy

$$\min_u C(x(u)) \quad \text{such that} \quad \mathcal{S}(x(u), u, \theta) = 0$$

Controls (pump rates, energy input, ...)

Parameters (geology, COP, dissipation, ...)

Underground thermal energy storage system



PDE-constrained optimization

Minimize C (e.g., cost of delivering heat to apartment complex), while ensuring conservation of mass/thermal energy

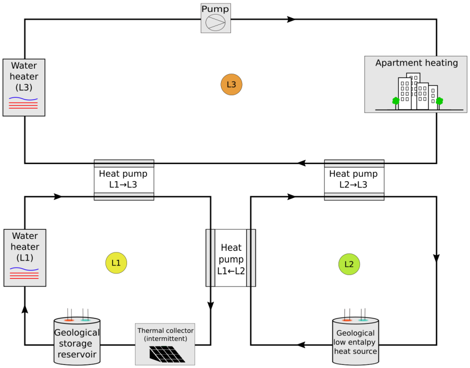
$$\min_u C(x(u)) \quad \text{such that} \quad \mathcal{S}(x(u), u, \theta) = 0$$

Controls (pump rates, energy input, ...)

Parameters (geology, COP, dissipation, ...)

Q1 what parameters θ give output x that matches observed data?

Underground thermal energy storage system



PDE-constrained optimization

Minimize C (e.g., cost of delivering heat to apartment complex), while ensuring conservation of mass/thermal energy

$$\min_u C(x(u)) \quad \text{such that} \quad \mathcal{S}(x(u), u, \theta) = 0$$

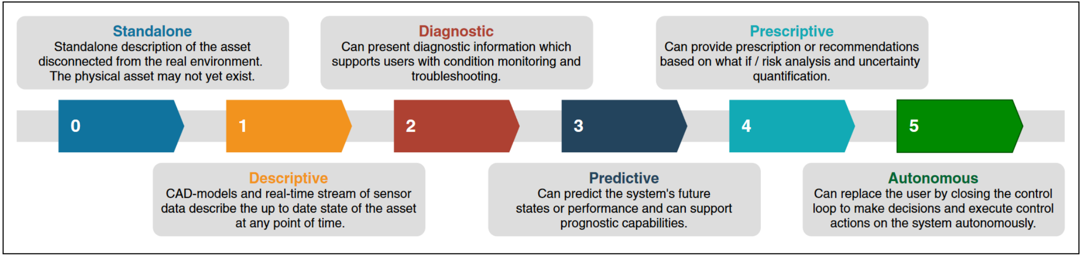
Controls (pump rates, energy input, ...)

Parameters (geology, COP, dissipation, ...)

Q1 what parameters θ give output x that matches observed data?

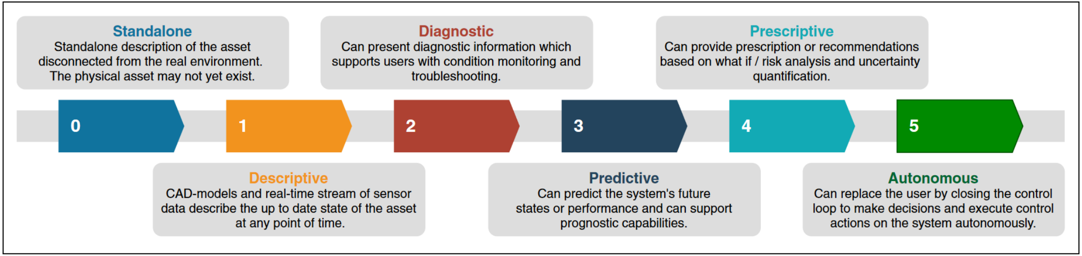
Q2 what are the optimal controls u that minimize cost?

Digital twin system enablers



San, Rasheed, and Kvamsdal 2021

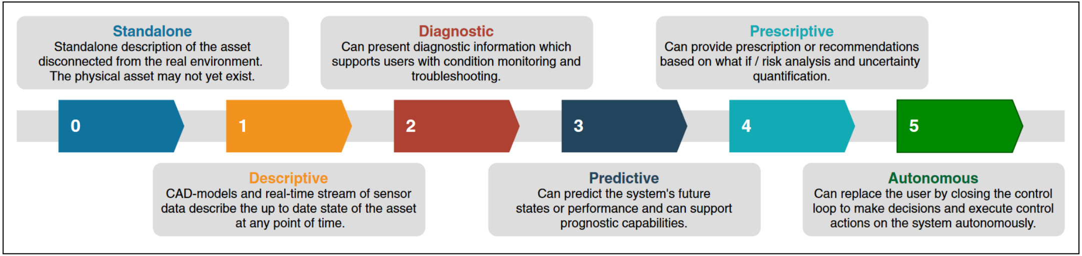
Digital twin system enablers



San, Rasheed, and Kvamsdal 2021

1. Composable and modular in the design of system components
2. Able to integrate and adapt to external data streams (real-time/forecast)
3. Fully differentiable, i.e., able to provide sensitivities/gradients
4. Able to quantify uncertainty

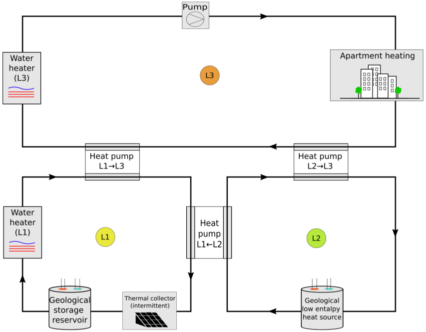
Digital twin system enablers



San, Rasheed, and Kvamsdal 2021

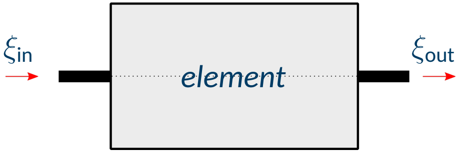
1. **Composable and modular in the design of system components**
2. Able to integrate and adapt to external data streams (real-time/forecast)
3. Fully differentiable, i.e., able to provide sensitivities/gradients
4. Able to quantify uncertainty

Loop-based modeling approach

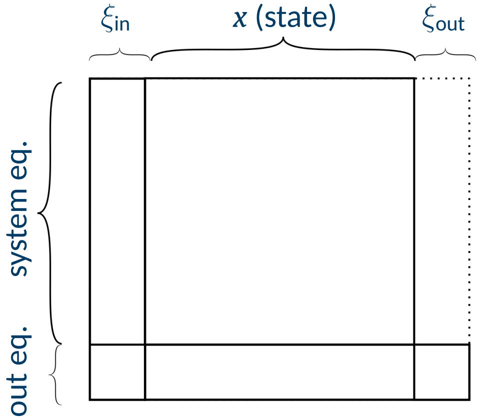


- System conceptualized as set of closed loops
- Heat moved between loops by heat pumps, exchangers or common components
- Each system compoent is a *differentiable* physical model with its own internal state
- Through loops, all system components are coupled together and simulated as a single, large model

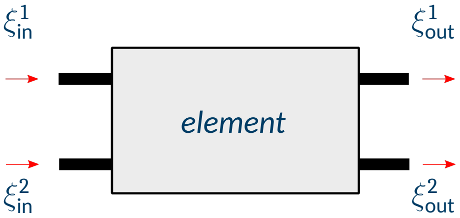
Single-loop element



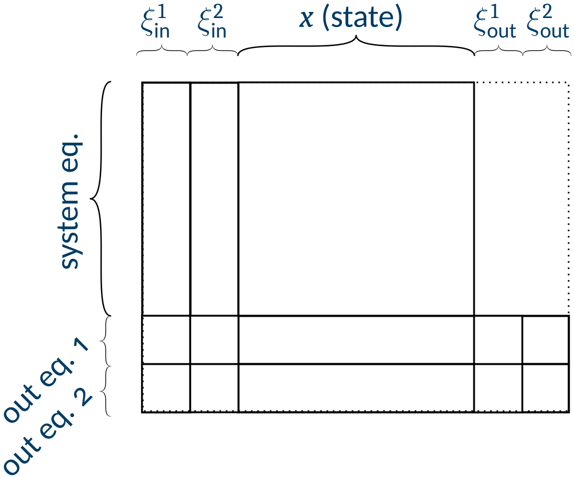
ξ_α = (rate$_\alpha$, T_α)



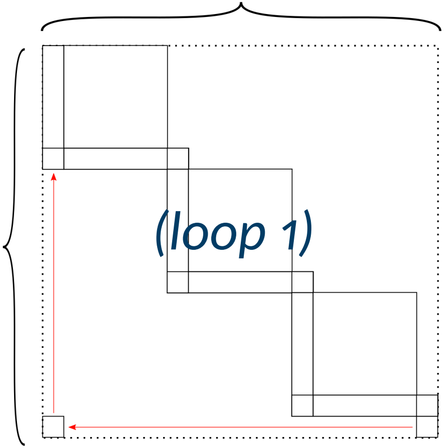
Double/multi-loop element

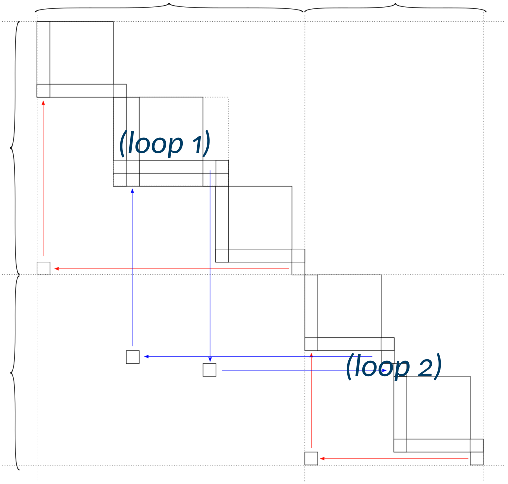


ξ_α^l = (rate$_\alpha^l$, T_α^l)

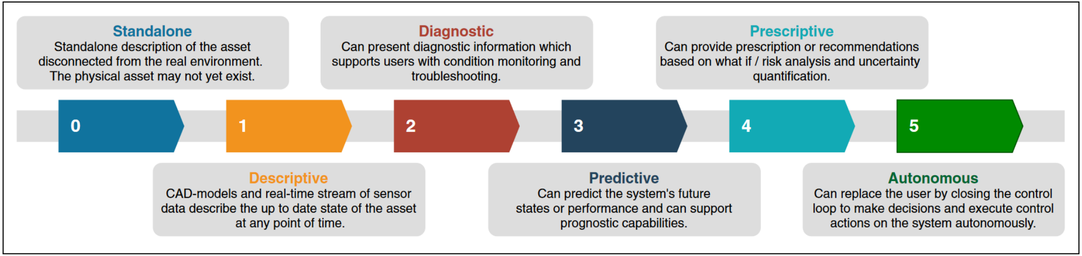


Assembling the loops





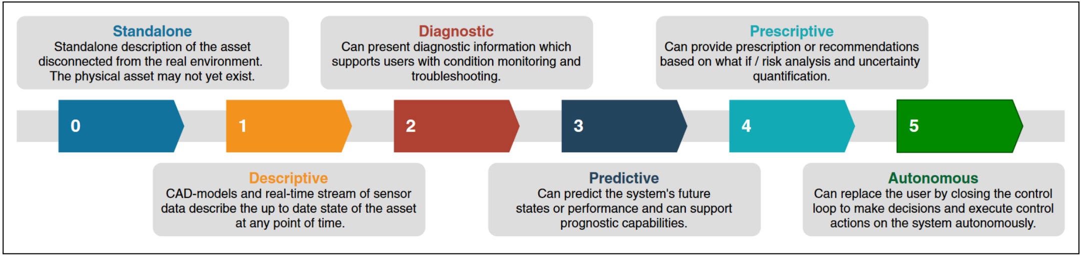
Digital twin system requirements



San, Rasheed, and Kvamsdal 2021

1. Composable and modular in the design of system components
2. Able to integrate and adapt to external data streams (real-time/forecast)
3. Fully differentiable, i.e., able to provide sensitivities/gradients
4. Able to quantify uncertainty

Digital twin system requirements



San, Rasheed, and Kvamsdal 2021

1. Composable and modular in the design of system components
2. Able to integrate and adapt to external data streams (real-time/forecast)
3. **Fully differentiable, i.e., able to provide sensitivities/gradients**
4. Able to quantify uncertainty

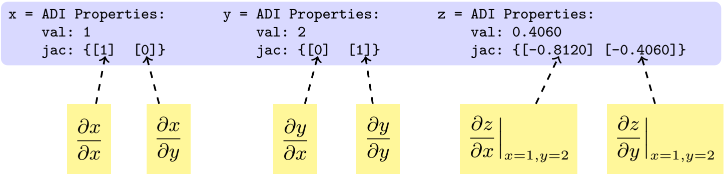
Enabling technology: automatic differentiation

- Introduce extended pair, $\langle v, v_x \rangle$, to represent the value v and its derivative v_x
- Combine chain rule and elementary derivative rules
  - mechanically accumulate derivatives *at specific values* of x

Elementary:	$v = \sin(x)$	$\longrightarrow$	$\langle v \rangle = \langle \sin x, \cos x \rangle$
Arithmetic:	$v = f * g$	$\longrightarrow$	$\langle v \rangle = \langle f * g, f * g_x + f_x * g \rangle$
Chain rule:	$v = \exp(f(x))$	$\longrightarrow$	$\langle v \rangle = \langle \exp(f(x)), \exp(f(x)) f'(x) \rangle$

- Use operator overloading to avoid messing up code

```
[x,y] = initVariablesADI(1,2);
z = 3*exp(-x*y)
```



Adjoint-based optimization

Define a Lagrange function (cost function $\mathcal{C}$ penalized by simulator residual)

$$J_\lambda = \mathcal{C}\big(\boldsymbol{x}(\boldsymbol{u}), \boldsymbol{\theta}\big) + \boldsymbol{\lambda}^\top \mathcal{S}\big(\boldsymbol{x}(\boldsymbol{u}), \boldsymbol{u}, \boldsymbol{\theta}\big)$$

Gradient: differentiate with respect to $\boldsymbol{u}$

$$\frac{dJ_\lambda}{d\boldsymbol{u}} = \left(\frac{\partial \mathcal{C}}{\partial \boldsymbol{x}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \boldsymbol{x}}\right) \frac{d\boldsymbol{x}}{d\boldsymbol{u}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \boldsymbol{u}} + \mathcal{S}^\top \frac{d\boldsymbol{\lambda}}{d\boldsymbol{u}}$$

Adjoint-based optimization

Define a Lagrange function (cost function $\mathcal{C}$ penalized by simulator residual)

$$J_\lambda = \mathcal{C}\big(\mathbf{x}(\boldsymbol{u}), \boldsymbol{\theta}\big) + \boldsymbol{\lambda}^\top \mathcal{S}\big(\mathbf{x}(\boldsymbol{u}), \boldsymbol{u}, \boldsymbol{\theta}\big)$$

Gradient: differentiate with respect to $\boldsymbol{u}$

$$\frac{dJ_\lambda}{d\boldsymbol{u}} = \left(\frac{\partial \mathcal{C}}{\partial \mathbf{x}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \mathbf{x}}\right)\frac{d\mathbf{x}}{d\boldsymbol{u}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \boldsymbol{u}} + \underbrace{\mathcal{S}^\top}_{=0} \frac{d\boldsymbol{\lambda}}{d\boldsymbol{u}}$$

Forward simulation:
$\mathcal{S}\big(\mathbf{x}(\boldsymbol{u}), \boldsymbol{u}, \boldsymbol{\theta}\big) = 0$
Solved with a standard simulator

Adjoint-based optimization

Define a Lagrange function (cost function $\mathcal{C}$ penalized by simulator residual)

$$J_\lambda = \mathcal{C}\big(\mathbf{x}(\mathbf{u}), \boldsymbol{\theta}\big) + \boldsymbol{\lambda}^\top \mathcal{S}\big(\mathbf{x}(\mathbf{u}), \mathbf{u}, \boldsymbol{\theta}\big)$$

Gradient: differentiate with respect to $\mathbf{u}$

$$\frac{dJ_\lambda}{d\mathbf{u}} = \left(\frac{\partial \mathcal{C}}{\partial \mathbf{x}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \mathbf{x}}\right)\frac{d\mathbf{x}}{d\mathbf{u}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \mathbf{u}} + \mathcal{S}^\top \frac{d\boldsymbol{\lambda}}{d\mathbf{u}}$$

=0

=0

Adjoint equations:
$(\partial \mathcal{S}/\partial \mathbf{x})^\top \boldsymbol{\lambda} = -(\partial \mathcal{C}/\partial \mathbf{x})^\top$
Solved backward for $\boldsymbol{\lambda}$
after solving forward for $\mathbf{x}$

Forward simulation:
$\mathcal{S}\big(\mathbf{x}(\mathbf{u}), \mathbf{u}, \boldsymbol{\theta}\big) = 0$
Solved with a
standard simulator

Adjoint-based optimization

Define a Lagrange function (cost function $\mathcal{C}$ penalized by simulator residual)

$$J_\lambda = \mathcal{C}(\mathbf{x}(\boldsymbol{u}), \boldsymbol{\theta}) + \boldsymbol{\lambda}^\top \mathcal{S}(\mathbf{x}(\boldsymbol{u}), \boldsymbol{u}, \boldsymbol{\theta})$$

Gradient: differentiate with respect to $\boldsymbol{u}$

$$\frac{dJ_\lambda}{d\boldsymbol{u}} = \left(\frac{\partial \mathcal{C}}{\partial \mathbf{x}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \mathbf{x}} \right) \frac{d\mathbf{x}}{d\boldsymbol{u}} + \boldsymbol{\lambda}^\top \frac{\partial \mathcal{S}}{\partial \boldsymbol{u}} + \mathcal{S}^\top \frac{d\boldsymbol{\lambda}}{d\boldsymbol{u}}$$

=0 =0

Adjoint equations:
$(\partial \mathcal{S}/\partial \mathbf{x})^\top \boldsymbol{\lambda} = -(\partial \mathcal{C}/\partial \mathbf{x})^\top$
Solved backward for $\boldsymbol{\lambda}$
after solving forward for $\boldsymbol{x}$

Automatic differentiation:
$\partial \mathcal{S}/\partial \boldsymbol{u}$ computed "behind the curtain"
by the code during the backward adjoint
solve (set $\boldsymbol{u}$ as independent variable)

Forward simulation:
$\mathcal{S}(\mathbf{x}(\boldsymbol{u}), \boldsymbol{u}, \boldsymbol{\theta}) = 0$
Solved with a
standard simulator

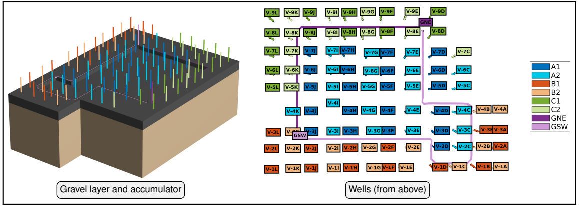
Presentation outline

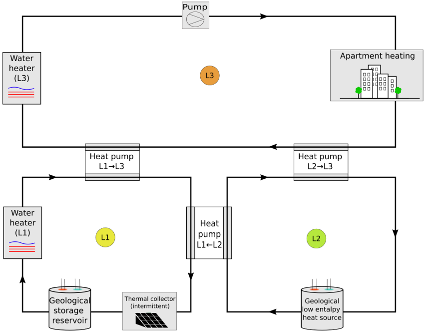
- Introduction
- Digital twins for UTES
- Numerical examples
- Conclusions

Model tuning

Wesselkvartalet

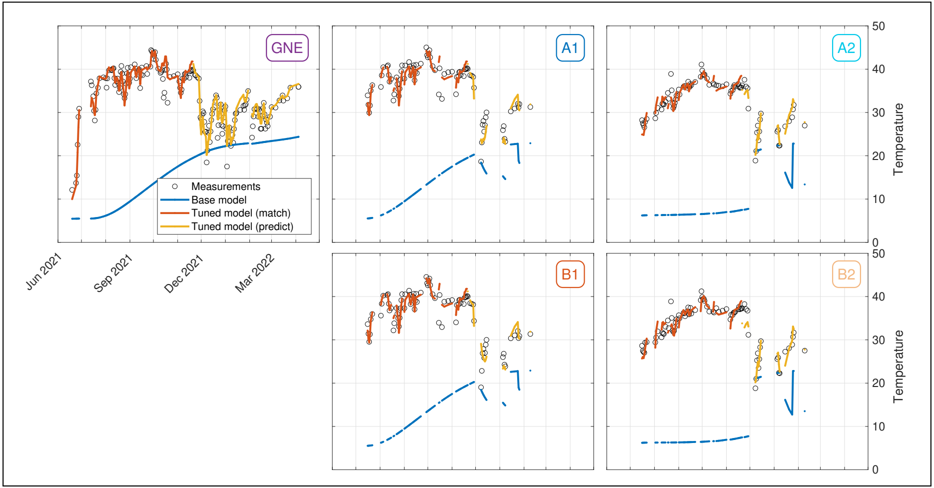
- Residential/commercial building in Asker (NO)
- Multi-reservoir, shallow geothermal storage
  - Three reservoirs at different depths
  - More than 100 wells, coupled in groups
  - Constant base load, rapid release at peak loads
  - Connected to deicing system for the city streets



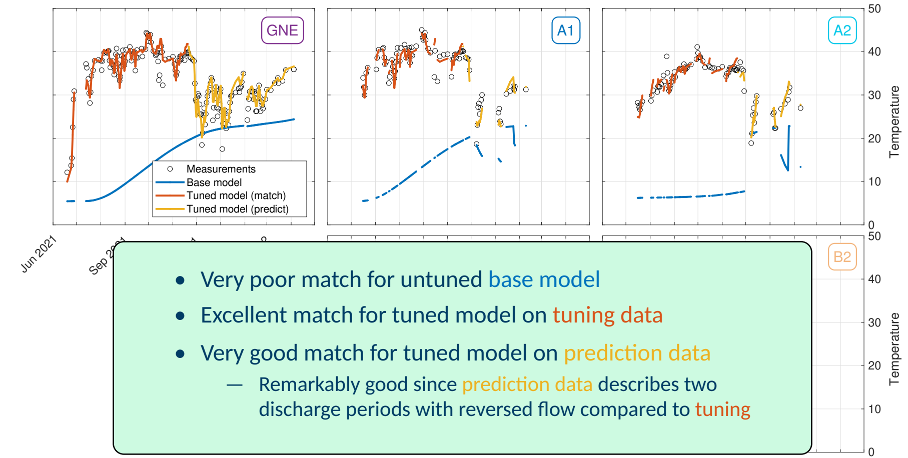


Here: tune shallow reservoir based on observed temperatures

Wesselkvartalet – model parameter tuning



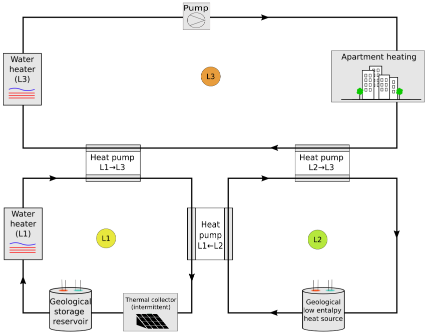
Wesselkvartalet – model parameter tuning



- Very poor match for untuned base model
- Excellent match for tuned model on tuning data
- Very good match for tuned model on prediction data
  - Remarkably good since prediction data describes two discharge periods with reversed flow compared to tuning

Control optimization

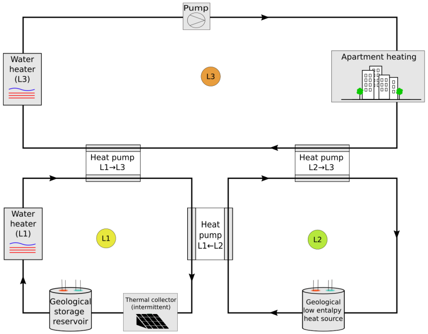
Wesselkvartalet – optimal control



Scenario

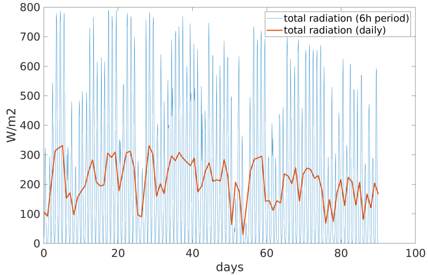
- Charge reservoir for 90 days
- Rest for 60 days
- Discharge for 65 days:
  - 60 d 6 l/s, 75°C, 5 d 8 l/s, 90°C
- Control variables:
  - rates for L1/L2, power of heater in L1
- Minimize total cost over whole cycle

Wesselkvartalet – optimal control

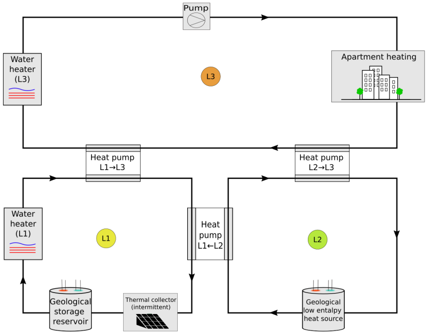


External factors

- Varying solar radiation

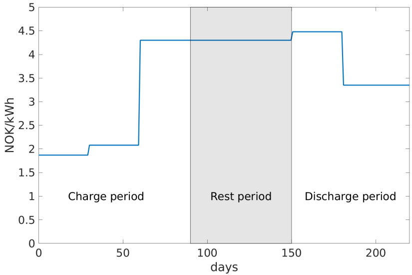


Wesselkvartalet – optimal control



External factors

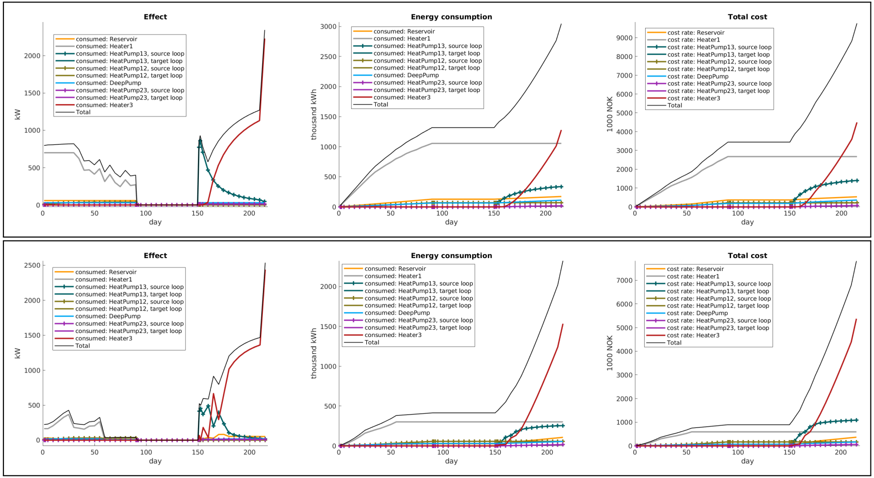
- Varying solar radiation
- Varying energy prices



Optimal control: energy consumption and cost

Base case

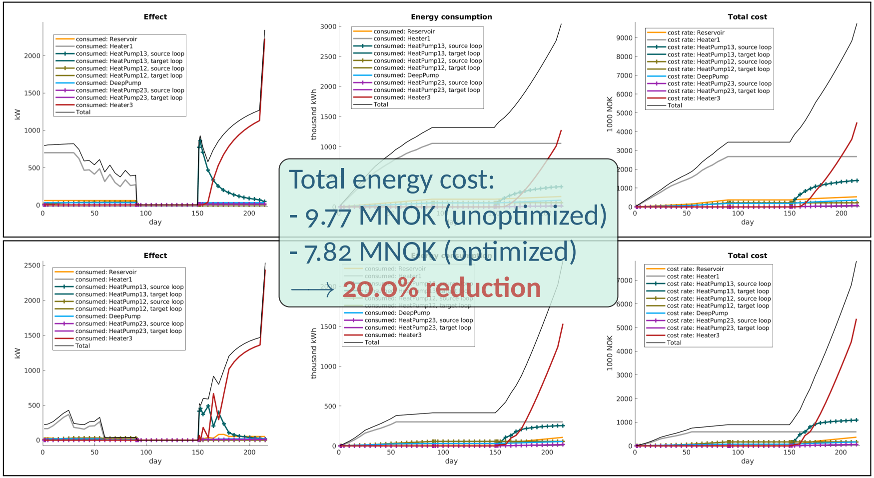
Optimized



Optimal control: energy consumption and cost

Base case

Optimized



Total energy cost:

- 9.77 MNOK (unoptimized)
- 7.82 MNOK (optimized)

⟶ **20.0% reduction**

Presentation outline

Introduction
Digital twins for UTES
Numerical examples
Conclusions

Concluding remarks

Towards digital twin system for underground thermal energy storage

- Composable and modular modelling framework
  - Coupling of complex simulation models into a single simulator
- Fully differentiable code
  - Compute sensitivities and perform adjoint-based optimization
- Two examples of important applications
  - Model parameter tuning – model output matches observations
  - Control optimization – minimize operational costs
- Next steps:
  - Integrate and adapt to external data streams (real-time/forecast)
  - Incorporate uncertainty – quantify, optimize under uncertainty

SINTEF

Concluding remarks

MRST
TRANSFORMING RESEARCH

All simulation code has been developed in the open-source MATLAB Reservoir Simulation Toolbox

- Source code, documentation, tutorials, etc.; `mrst.no`
- Book chapter on geothermal modelling with MRST (open-access):
  Collignon, M., Klemetsdal, Ø, Møyner, O. (2021). *Simulation of Geothermal Systems Using MRST*
  Cambridge University Press. doi: 10.1017/9781009019781.018
- Conference paper on modelling and optimization of geothermal energy systems:
  Klemetsdal, Ø., et al. (2022). *Modeling and Optimization of Shallow Geothermal Heat Storage*
  ECMOR 2022, Sep 2022. doi: 10.3997/2214-4609.202244109 – Journal version in press (Geoenergy, 2023)

Acknowledgements

The authors would like to thank Ruden AS and Wessel Energi AS
for allowing the presentation of this work

SINTEF

Technology for a better society

Governing equations and discretization

Finite volumes in space, implicit backward Euler in time

$$\boldsymbol{R}_w^{n+1} = \frac{1}{\Delta t^n}(\boldsymbol{M}_w^{n+1} - \boldsymbol{M}_w^n) + \texttt{div}(\boldsymbol{V}_w^{n+1}) - \boldsymbol{Q}_w^{n+1} = \boldsymbol{0}$$

$$\boldsymbol{V}_w = -\texttt{upw}(\boldsymbol{\rho}_w/\boldsymbol{\mu}_w)\boldsymbol{\Theta}[\texttt{grad}(\boldsymbol{p}) - g\texttt{favg}(\boldsymbol{\rho}_w)\texttt{grad}(\boldsymbol{z})]$$

- $\boldsymbol{\Theta}\texttt{grad}$: discrete representation of $\mathrm{K}\nabla$ (linear/nonlinear two-/multipoint, etc.)
  - In this work: linear two-point flux approximation (comparison: Klemetsdal et al. 2020)
  - $\boldsymbol{\Theta}$: vector of interface transmissibilities
- $\texttt{div}$: divergence, $\texttt{upw}$: upwind (single-point here), $\texttt{favg}$: face average

$\boldsymbol{M}$	Mass	$\boldsymbol{V}$	Flux	$\boldsymbol{Q}$	Sources/sinks	g	Gravity
$\boldsymbol{\rho}$	Density	$\boldsymbol{\mu}$	Viscosity	$\boldsymbol{u}$	Internal energy	$\boldsymbol{h}$	Enthalpy
$\boldsymbol{p}$	Pressure	$\boldsymbol{T}$	Temperature	$\boldsymbol{K}$	Permeability	$\boldsymbol{\Lambda}$	Thermal cond.

Governing equations and discretization

Finite volumes in space, implicit backward Euler in time

$$\boldsymbol{R}_w^{n+1} = \frac{1}{\Delta t^n}(\boldsymbol{M}_w^{n+1} - \boldsymbol{M}_w^n) + \mathtt{div}(\boldsymbol{V}_w^{n+1}) - \boldsymbol{Q}_w^{n+1} = \boldsymbol{0}$$

$$\boldsymbol{V}_w = -\mathtt{upw}(\boldsymbol{\rho}_w/\boldsymbol{\mu}_w)\boldsymbol{\Theta}[\mathtt{grad}(\boldsymbol{p}) - g\mathtt{favg}(\boldsymbol{\rho}_w)\mathtt{grad}(\boldsymbol{z})]$$

Conservation of mass

Darcy's law

- $\boldsymbol{\Theta}\mathtt{grad}$: discrete representation of K∇ (linear/nonlinear two-/multipoint, etc.)
  - In this work: linear two-point flux approximation (comparison: Klemetsdal et al. 2020)
  - $\boldsymbol{\Theta}$: vector of interface transmissibilities
- $\mathtt{div}$: divergence, $\mathtt{upw}$: upwind (single-point here), $\mathtt{favg}$: face average

$\boldsymbol{M}$	Mass	$\boldsymbol{V}$	Flux	$\boldsymbol{Q}$	Sources/sinks	g	Gravity
$\boldsymbol{\rho}$	Density	$\boldsymbol{\mu}$	Viscosity	$\boldsymbol{u}$	Internal energy	$\boldsymbol{h}$	Enthalpy
$\boldsymbol{p}$	Pressure	$\boldsymbol{T}$	Temperature	$\boldsymbol{K}$	Permeability	$\boldsymbol{\Lambda}$	Thermal cond.

Governing equations and discretization

Finite volumes in space, implicit backward Euler in time

$$\boldsymbol{R}_w^{n+1} = \frac{1}{\Delta t^n}(\boldsymbol{M}_w^{n+1} - \boldsymbol{M}_w^n) + \texttt{div}(\boldsymbol{V}_w^{n+1}) - \boldsymbol{Q}_w^{n+1} = \boldsymbol{0}$$

$$\boldsymbol{V}_w = -\texttt{upw}(\boldsymbol{\rho}_w/\boldsymbol{\mu}_w)\boldsymbol{\Theta}[\texttt{grad}(\boldsymbol{p}) - g\texttt{favg}(\boldsymbol{\rho}_w)\texttt{grad}(\boldsymbol{z})]$$

Conservation of mass

Darcy's law

- $\boldsymbol{\Theta}\texttt{grad}$: discrete representation of $K\nabla$ (linear/nonlinear two-/multipoint, etc.)
  - In this work: linear two-point flux approximation (comparison: Klemetsdal et al. 2020)
  - $\boldsymbol{\Theta}$: vector of interface transmissibilities
- $\texttt{div}$: divergence, $\texttt{upw}$: upwind (single-point here), $\texttt{favg}$: face average

$\boldsymbol{M}$	Mass	$\boldsymbol{V}$	Flux	$\boldsymbol{Q}$	Sources/sinks	g	Gravity
$\boldsymbol{\rho}$	Density	$\boldsymbol{\mu}$	Viscosity	$\boldsymbol{u}$	Internal energy	$\boldsymbol{h}$	Enthalpy
$\boldsymbol{p}$	Pressure	$\boldsymbol{T}$	Temperature	$\boldsymbol{K}$	Permeability	$\boldsymbol{\Lambda}$	Thermal cond.

Governing equations and discretization

Finite volumes in space, implicit backward Euler in time

$$\boldsymbol{R}_h^{n+1} = \frac{1}{\Delta t^n}\left([\boldsymbol{M}_w \boldsymbol{u}_w + \boldsymbol{M}_r \boldsymbol{u}_r]^{n+1} - [\boldsymbol{M}_w \boldsymbol{u}_w + \boldsymbol{M}_r \boldsymbol{u}_r]^{n}\right) + \texttt{div}\left([\boldsymbol{V}_w \boldsymbol{h}_w + \boldsymbol{H}_c]^{n+1}\right) - [\boldsymbol{Q}_w \boldsymbol{h}_w]^{n+1} - \boldsymbol{Q}_h^{n+1} = \boldsymbol{0}$$

$$\boldsymbol{H}_c = -(\boldsymbol{\Theta}_{hw} + \boldsymbol{\Theta}_{hr})\texttt{grad}(\boldsymbol{T})$$

- Conductive heat flux $\boldsymbol{H}_c$ discretized by two-point method (same as mass flux)
  - $(\boldsymbol{\Theta}_{hw} + \boldsymbol{\Theta}_{hr})\texttt{grad}$: discrete representation of $(\boldsymbol{\Lambda}_w + \boldsymbol{\Lambda}_r)\nabla$
  - $\boldsymbol{\Theta}_{hw}$, $\boldsymbol{\Theta}_{hr}$: vectors of interface heat transmissibilities

$\boldsymbol{M}$	Mass	$\boldsymbol{V}$	Flux	$\boldsymbol{Q}$	Sources/sinks	$\boldsymbol{g}$	Gravity
$\boldsymbol{\rho}$	Density	$\boldsymbol{\mu}$	Viscosity	$\boldsymbol{u}$	Internal energy	$\boldsymbol{h}$	Enthalpy
$\boldsymbol{p}$	Pressure	$\boldsymbol{T}$	Temperature	$\boldsymbol{K}$	Permeability	$\boldsymbol{\Lambda}$	Thermal cond.

Governing equations and discretization

Finite volumes in space, implicit backward Euler in time

$$\boldsymbol{R}_h^{n+1} = \frac{1}{\Delta t^n}\left([\boldsymbol{M}_w\boldsymbol{u}_w + \boldsymbol{M}_r\boldsymbol{u}_r]^{n+1} - [\boldsymbol{M}_w\boldsymbol{u}_w + \boldsymbol{M}_r\boldsymbol{u}_r]^{n}\right) + \texttt{div}\left([\boldsymbol{V}_w\boldsymbol{h}_w + \boldsymbol{H}_c]^{n+1}\right) - [\boldsymbol{Q}_w\boldsymbol{h}_w]^{n+1} - \boldsymbol{Q}_h^{n+1} = \boldsymbol{0}$$

Conservation of energy

$$\boldsymbol{H}_c = -(\boldsymbol{\Theta}_{hw} + \boldsymbol{\Theta}_{hr})\texttt{grad}(\boldsymbol{T})$$

Fourier's law

- Conductive heat flux $\boldsymbol{H}_c$ discretized by two-point method (same as mass flux)
  - $(\boldsymbol{\Theta}_{hw} + \boldsymbol{\Theta}_{hr})\texttt{grad}$: discrete representation of $(\boldsymbol{\Lambda}_w + \boldsymbol{\Lambda}_r)\nabla$
  - $\boldsymbol{\Theta}_{hw}$, $\boldsymbol{\Theta}_{hr}$: vectors of interface heat transmissibilities

$\boldsymbol{M}$	Mass	$\boldsymbol{V}$	Flux	$\boldsymbol{Q}$	Sources/sinks	g	Gravity
$\boldsymbol{\rho}$	Density	$\boldsymbol{\mu}$	Viscosity	$\boldsymbol{u}$	Internal energy	$\boldsymbol{h}$	Enthalpy
$\boldsymbol{p}$	Pressure	$\boldsymbol{T}$	Temperature	$\boldsymbol{K}$	Permeability	$\boldsymbol{\Lambda}$	Thermal cond.