

Draft for ESCAPE Paper:

ACTUAL TEXT IN PAPER (DRAFT)

UNRESOLVED ISSUES/DOUTTS
IN MY HEAD THAT MAKE
ME UNREAS.

WHAT I'M TRYING TO
CONVEY TO THE READER

Multi-scenario Design Optimization using ADMM
of a Thermal Energy Storage System

(expt) Abstract →

TBC

(dep) SEC 1 → INTRODUCTION:

TBC

(2 pg) SEC 2 - METHODOLOGY / DESCRIBING TWO-STAGE DYNAMIC OPTIMIZATION PROBLEMS USING ADMM

* Explain how 2-stage DO problems
can be decomposed as needed
in our approach.

Explain the general notation
for the central problem

→ The optimal design problem for a dynamic system under time varying uncertainty can be represented as a 2-stage dynamic optimization problem.

The design decisions using the first stage variables (w^{des})

the operations decisions on the second stage variables (w^{oper})

In this representation it is assumed that the uncertainty is fully realized after the first stage decisions

(design decisions) are made and hence the operations are optimal for the realized uncertainty.

We can represent the uncertainty by a set of discrete scenarios $S = \{1, 2, \dots, S\}$ with the cost weight w_k

To represent the likelihood of scenarios being realized,

the dynamics of the system can be discretized into N equally spaced sampling intervals represented by the set $K = \{0, 1, \dots, N-1\}$.

The optimal design problem can be then cast as a nonlinear programming problem (NLP) of the form

$$\min_{w^{des}} \quad d^{des}(w^{des}) + \sum_{k \in K} w_k [\sum_{s \in S} L(x_{k,s}, u_{k,s}, w^{des})] \quad (1)$$

$$\text{s.t.} \quad g(x_{k,s}, u_{k,s}, \theta_{k,s}, w^{des}) \leq 0 \quad \forall k \in K, s \in S$$

$$x_{0,s} = h(x_{0,s}, u_{0,s}, \theta_{0,s}, w^{des}) \quad \forall s \in S$$

$$x_{k,s} = x_0 \quad \forall k \in K, s \in S$$

$$w^{oper} = [x_0, x_1, \dots, x_N] \quad \forall k \in K, s \in S \rightarrow \text{How to make stacking more easy to read?}$$

where the $(x_{k,s})$ represents the s^{th} scenario at time step k .

Vectors x and u represent the differential states and the control inputs applied to the plant during operations respectively.

The initial condition during operations is given by x_0 for all the scenarios

In the interest of simplifying notation we collect these

second stage variables for all the scenarios

together into a single vector (w^{oper}).
The time varying parameters are represented by $\theta_{k,s}$, for time k and scenario s

and $\theta_{0,s}$

The function d^{des} used to represent the capital cost

and $L(\cdot)$ represent the operating cost.

The objective of the design problem is then simply defined as the sum of capital costs and the expected costs during operating.

Function $g(\cdot)$ represent the nonlinear equality

constraints and $h(\cdot)$ used to represent the dynamics of the system.

Sec 2.1 → Reformulating the design problem as a general form
consensus optimization problem.

Solving the optimal design problem as formed in (1) can become computationally intractable due to memory constraints when having to consider many scenarios for uncertainty and long horizon times.

partition problems which can then be solved separately
as a way to handle the memory limitations.

We can divide the two stage dynamic optimization problem into P partitions denoted by the set $P = \{1, 2, \dots, P\}$ in a very flexible manner.

An illustrative example of a design problem with 2 scenarios divided into 6 partition problems is shown in Figure 1.

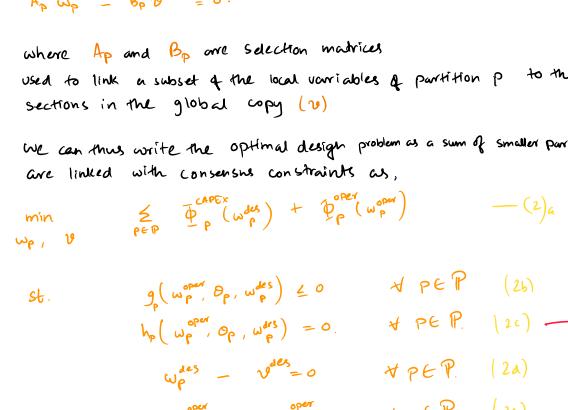


Fig 1 → An illustration of dividing the optimal design problem with 2 scenarios into 6 partitions

In the partition problems, each partition P has its own local variables for design (w_P^{des}) and operations (w_P^{oper})

The sum of the individual partition problems together is similar to the central problem in (1) when,

i) All the partitions are in consensus of the value of their local design variables,

ii) Adjacent partitions are in consensus of the differential state variable shared between them (at the edge of the partitions)

We can introduce a global copy of all the variables that must reach consensus into a vector denoted as v .

The consensus requirements can then be imposed as an additional constraint in each partition P as

$$\begin{aligned} w_P^{des} - v^{des} &= 0 \\ A_P w_P^{des} - B_P v^{oper} &= 0. \end{aligned}$$

where A_P and B_P are selection matrices used to link a subset of the local variables of partition P to the corresponding sections in the global copy (v)

We can thus write the optimal design problem as a sum of smaller partition problems that are linked with consensus constraints as,

$$\min_{w_P, v} \quad \sum_{P \in P} \Phi_P^{\text{CAP}}(w_P^{des}) + \Phi_P^{\text{OP}}(w_P^{oper}) \quad -(2a)$$

$$\text{s.t.} \quad \begin{aligned} g_P(w_P^{des}, \theta_P, u_P) &\leq 0 && \forall P \in P \\ h_P(w_P^{des}, \theta_P, w_P^{oper}) &= 0. && \forall P \in P. \end{aligned} \quad -(2b) \quad \rightarrow \text{cannot derive } x_{k+1} \text{ from } x_k \text{ without creating a mess}$$

$$w_P^{des} - v^{des} = 0 \quad \forall P \in P. \quad -(2c)$$

$$A_P w_P^{des} - B_P v^{oper} = 0 \quad \forall P \in P. \quad -(2d)$$

$$w_P = [w_P^{des}, w_P^{oper}] \quad \forall P \in P. \quad -(2e)$$

The objective function terms in the partition problems are chosen appropriately to add up to the original objective in (1)

For the sake of compact notation, we collect the local variables in the partition as w_P and can represent the constraints (2b) and (2c) which only involve the local variables (w_P)

mentioned above!

Sec 2.2 → APPLYING ADMM to get a distributed algorithm.

The individual partition problems are not trivially separable due to the presence of constraint

the constraint (2c) and (2e) to the objective as

$$\mathcal{L}_P(w_P, v, \lambda_P) = \sum_{P \in P} [\Phi_P^{\text{CAP}}(w_P^{des}) + \Phi_P^{\text{OP}}(w_P^{oper})] + (\lambda_P^{\text{des}})^T (w_P^{des} - v^{des}) + \frac{1}{2} \|w_P^{des} - v^{des}\|^2 + (\lambda_P^{\text{oper}})^T (A_P w_P^{des} - B_P v^{oper}) + \frac{1}{2} \|A_P w_P^{des} - B_P v^{oper}\|^2 \quad -(3)$$

where $\lambda_P^{\text{des}} = [(\lambda_P^{\text{des}})^T, (\lambda_P^{\text{oper}})^T]^T$ are the lagrange multipliers associated with the consensus constraints (2c) and (2e)

$\lambda_P^{\text{oper}} = [(\lambda_P^{\text{des}})^T, (\lambda_P^{\text{oper}})^T]^T$ is a vector of chosen penalty parameters.

It can be seen that (3) is additively separable except for the quadratic penalty terms.

The iterations of ADMM algorithm involves solving the partition problems while keeping v and λ fixed and then updating them by keeping the local variables w_P fixed

in an alternating fashion until convergence

The iteration i of the ADMM algorithm thus takes the form

$$(w_P)^{i+1} = \underset{w_P \in W_P}{\text{argmin}} \quad \mathcal{L}_P(w_P, v^i, \lambda_P^i) \quad \forall P \in P \quad -(4a)$$

$$(v)^{i+1} = \underset{v \in V}{\text{argmin}} \quad \mathcal{L}_P(w_P^i, v, \lambda_P^i) \quad -(4b)$$

$$(\lambda_P^{\text{des}})^{i+1} = (\lambda_P^{\text{des}})^i + \rho_{\text{des}} ((w_P^i)^{i+1} - (w_P^i)^{i+1}) \quad \forall P \in P \quad -(4c)$$

$$(\lambda_P^{\text{oper}})^{i+1} = (\lambda_P^{\text{oper}})^i + \rho_{\text{oper}} ((A_P w_P^i)^{i+1} - B_P v^{i+1}) \quad \forall P \in P \quad -(4d)$$

we can see that steps (4c) involve solving an optimization problem for each partition separately while the global variable is kept constant.

Thus partition problem can be solved using its own separate machine

Step (4b) is the minimization of the AL function while the local variables in the partitions are kept constant.

for consensus optimization problems reduces to finding the minimum of a quadratic function and can be shown to be the averaging operator

$$v^{i+1}(j) = \frac{1}{|P_j|} \sum_{P \in P_j} w_P^{i+1}(j)$$

where $P_j \in P$ denotes the set of partitions connected to the global

$v(j)$

Steps (4c) and (4d) are the updates to the lagrange multipliers of the consensus constraints and can also be carried out in each partition separately.

Explain termination criterion for ADMM →

termination criteria used for the ADMM iterations are that the primal and dual residuals are reasonably close to zero as explained in (page)

$$\|v^{i+1}\|_2 \leq \epsilon^{\text{primal}}$$

$$\|\lambda^{i+1}\|_2 \leq \epsilon$$

The primal and dual residuals in our case are defined as

$$\gamma_P^{i+1} = [(w_P^{i+1})^{i+1} - (w_P^{i+1})^{i+1}] \quad \forall P \in P$$

$$\delta_P^{i+1} = [(A_P w_P^{i+1})^{i+1} - B_P v^{i+1}] \quad \forall P \in P$$

where $P_j \in P$ denotes the set of partitions connected to the global

$v(j)$

(1P) SEC-3 → The TES design Problem.

Describe the process we have.

→ we use the flowsheet in Fig 2 to represent the heating section of a district heating network where water is used in a closed loop to satisfy the heating requirements of households in an area.

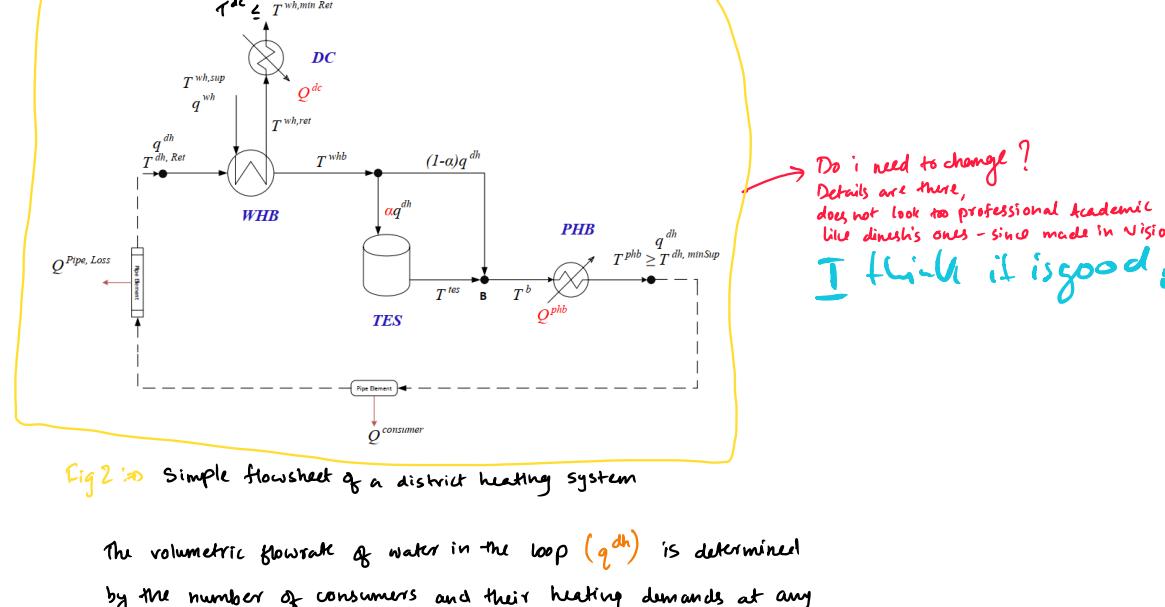


Fig 2 → Simple flowsheet of a district heating system

The volumetric flowrate of water in the loop (q^{dh}) is determined by the number of consumers and their heating demands at any particular time.

The temperature of the water returned to the heating section ($T^{dh,ret}$) is determined by the heat losses in the system and is assumed to be correlated to the ambient weather conditions.

Do I need to change?
Details are there,
does not look too professional academic
like dinish's ones - since made in Visio
I think it is good!

Need to explain more/better about the suppliers?

The operational objective of the heating system is to heat this water up to a temperature (T^{phb}) which must be above a contractually specified temperature ($T^{dh,minRet}$).

A cheap source of heat is available in the form of a process stream from an industrial process which requires to be cooled down below a specified temperature ($T^{wh,minRet}$).

This stream can loose heat to the district heating system using the heat exchanger WHB, and any additional cooling required in the process stream satisfied by the utility heat exchanger DC which uses some external cooling utility.

In a similar manner, any additional heating required in the district heating side is satisfied by using the peak heat boiler (PHB) which burns fossil fuels.

There is a temporal mismatch in the supply of heat from the process stream and the demand for heating from the consumers.

In order to better manage this mismatch and decrease the reliance on external utilities, a thermal energy storage system in the form of a simple buffer tank is being considered to be installed.

This simple TES system can charge/dischARGE thermal energy by raising/lowering the temperature of the tank by manipulating the flow split α .

Explain model simplification used

The TES tank is modelled as a well mixed tank with perfect level control with the volume maintained at V^{tes} .

The WHB is modelled as a purely countercurrent heat exchanger with the hot and cold sides discretized into a series of N_{cw} cells. Each cell is assumed well mixed with the heat transfer occurring only across the cell walls.

Heat Exchangers PHB and DC are modelled simply as a well mixed tank with the heat duty added or removed directly.

you can refer to your thesis for these details if you need to save space.

How the equations are obtained

Mass and energy balance equations can thus be written out for the system to model the dynamics of the system.

The differential states thus consist of $x^T = [T^{whb}, T^{dh,minRet}, T^{tes}, T^{phb}]$

while the control inputs are $u^T = [\alpha, q^{phb}, q^{dc}]$

The scenarios → what are the θ_v ?

In our model, we consider the uncertain parameters at each time step K as $\Theta_K = [q^{dh}_K, T^{dh,ret}_K, q^{wh,sup}_K, T^{wh,sup}_K]$

The uncertainties of the future profile can then be represented by building representative scenarios of these uncertain parameters.

but here is our too

→ The design problem is then to find the optimal volume of the TES tank (V^{tes}) and the area of heat exchanger (A^{whb}) that must be installed, given the uncertainties in future supply and demand of thermal energy.

Could also be pert in beginning of section,

we can thus form it as the large two-stage dynamic optimization problem of the form in eqn (1), partition it into smaller problems of the form shown in eqn (2), which can then be solved in a distributed manner using ADMM as described in Sec 2.2

(1 pg) SEC-4 → RESULTS AND DISCUSSIONS

Exact numbers used
in this chart I year
Since → what I need
for that numbers
to make my life easier
↓ A lot of numbers and
not enough space.

We present a simple simulation study to demonstrate the approach described in section 2 on the TES model described in section 3 and compare the solution and some convergence metrics against solving the design problem as a single NLP (1).

We consider a simple case with 2 equally probable scenarios with $N=60$ discretizations to represent the future operations.

The profiles of q^{opt} and T^{opt} both have a step change at time steps $k=20$ and $k=40$, while all the other parameters are held at their steady state values.

The design variables (w^{opt}) and operations variables (u^{opt}) were all scaled to be within zero and one while formulating the optimization problem.

Quadratic functions are used to represent the capital cost ($\phi^{cap} = 10^3 (V^{opt})^2 + 10^3 (A^{opt})^2$) and the operating costs ($L = (a^{opt})^2 + (a^{opt})^2$)

Scaling facilitated setting the AL penalty parameter ρ to two times the convergence speed of our distributed approach more easily.

The penalty parameters chosen as $\rho^{des} = 10^{-3}$ and $\rho^{oper} = 10^{-1}$ in the partition problems to roughly balance the magnitude of the objective and penalization terms.

can be removed if you add space.

Have an urge to make into
more realistic units and
get results.
But → we have to do
→ no space to explain
Hopping reviewers not too critical
in this

We solve the problem using the distributed algorithm in Sec 2.2 and show the results against the optimal design solution by solving the problem centrally.

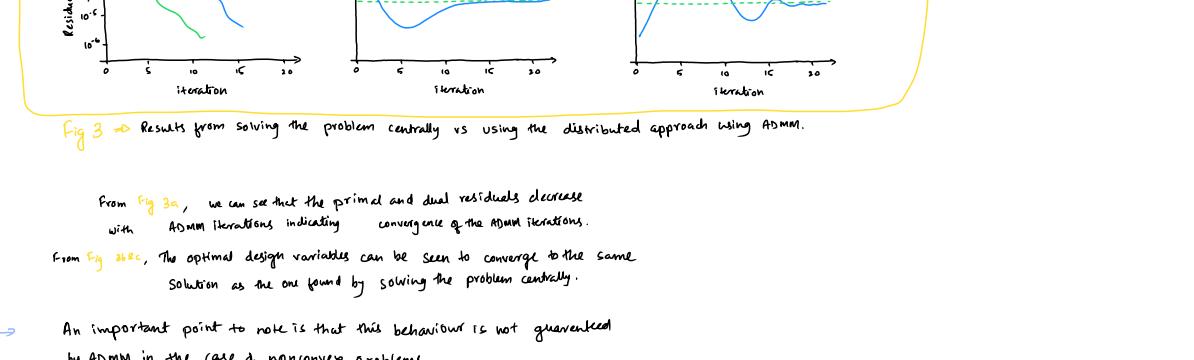


Fig 3 → Results from solving the problem centrally vs using the distributed approach using ADMM.

From Fig 3a, we can see that the primal and dual residuals decrease with ADMM iterations indicating convergence of the ADMM iterations.

From Fig 3bc, the optimal design variables can be seen to converge to the same solution as the one found by solving the problem centrally.

Discussions about
Algorithm in
case of nonconvex
problems.

An important point to note is that this behaviour is not guaranteed by ADMM in the case of nonconvex problems.

When applied to nonconvex problems, ADMM need not converge, and even when it does converge, it need not converge to an optimal point and must be hence considered just as another local optimization method (Boyd 2004)

Although the convergence guarantees for ADMM in the case of complex nonconvex NLPs are poorly understood, it has been shown to perform satisfactorily in practice (Rodriguez)

Three snapshots of the optimal differential state variables in the partition problems at iterations $i=1, 10$ and 20 of the ADMM algorithm are shown in Fig 4.

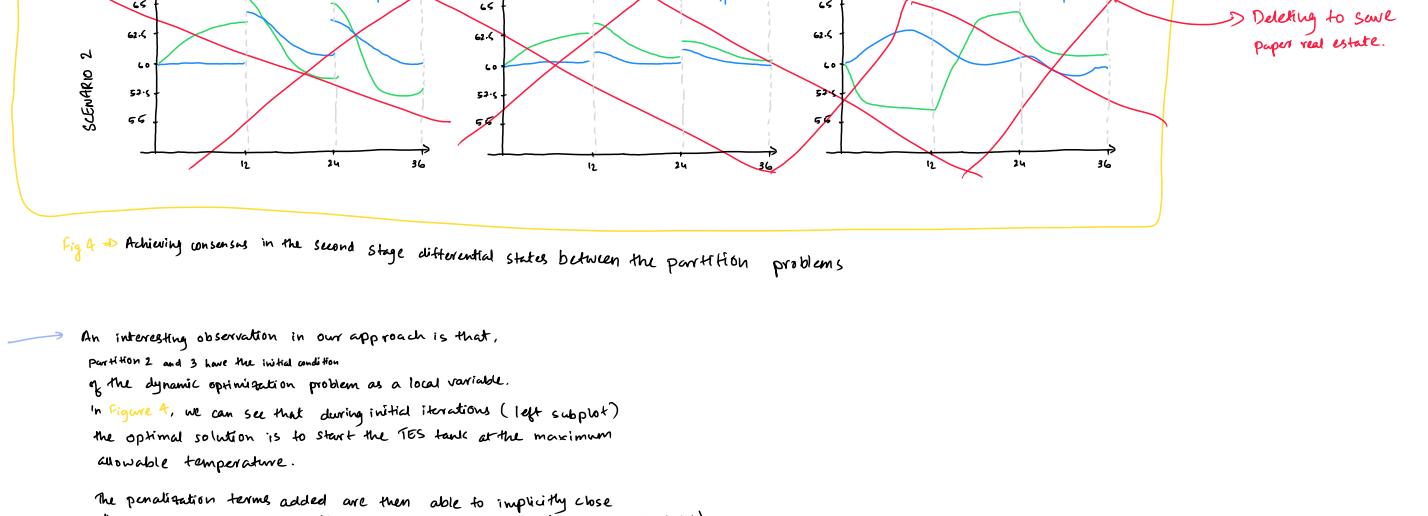


Fig 4 → Achieving consensus in the second stage differential states between the partition problems

Discussion on
simulations with MS

An interesting observation in our approach is that, partition 2 and 3 have the initial condition of the dynamic optimization problem as a local variable. In Figure 4, we can see that during initial iterations (left subplot) the optimal solution is to start the TES tank at the maximum allowable temperature.

The penalization terms added are then able to implicitly close this gap between the partitions and reach consensus (middle and right subplot).

In this aspect our approach shares similarities to multiple shooting (MS) approach in dynamic optimization.

The key difference is that the state continuity (consensus constraint) is enforced explicitly by the solver in MS, while in the ADMM approach it is enforced implicitly by minimizing the AL.

Discussion about
speed

The distributed approach converges in 20 iterations of the ADMM algorithm, where smaller NLPs are solved in each iteration.

When comparing against the centralized solution, we do observe that our distributed approach takes more computation time, even if perfect parallelization is assumed.

Thinking of removing
this whole section
because → real estate in paper
↳ speed CPU on a parallel box
I do not want to open now.

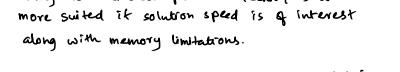


Fig 5 → Comparison of solution times of the distributed approach against solving it centrally.

This is due to the fact that the optimizer does not have access to the full information of all the other partition problems and hence cannot take advantage of it.

Using internal decomposition techniques would be more suited if solution speed is of interest along with memory limitations.

The use of internal decomposition methods for forming general partitions for similar two-stage dynamic optimization problems would be an interesting area for future work.

CPU - time could also be put in one sentence in the text. or a table with 2 rows.

(0-25 pg) SEC-5 → CONCLUSIONS

In this paper, we presented the optimal design of a simple TES system under uncertainty as a two-stage nonconvex dynamic optimization problem.

Due to limitations in memory of solving the problem centrally in a single machine, an approach for forming smaller partition problems in a general fashion was shown.

The ADMM algorithm was applied to coordinate between the subproblems which could be solved separately and in parallel.

A simple simulation exercise was used to demonstrate the approach.

REFERENCES

→ Boyd → Statistical learning and ADMM

→ Rodriguez → ADMM in nonconvex NLPs

→ Mandar → (Escape) - optimizing capacity of TES