Solving the Weapon Target Assignment Problem with Decision Diagrams

INFORMS 2024

🚧 This work is ongoing

Ryan O'Neil

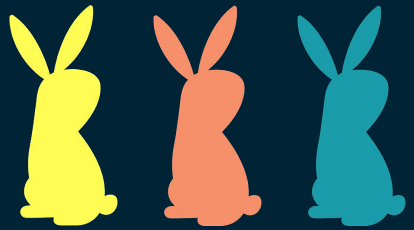
CTO at Nextmv

Integer scientist, cat and early music enthusiast, Go programmer

🤷 Background
A bit about the (static) weapon target assignment problem

🛠️ Models
A few models and solvers we can throw at it

📊 Results
How'd we do so far?



background

🤔 This inquiry is the result of a sales call.

A prospect wondered if we could use decision diagrams to allocate millions of drones in real time during combat operations.

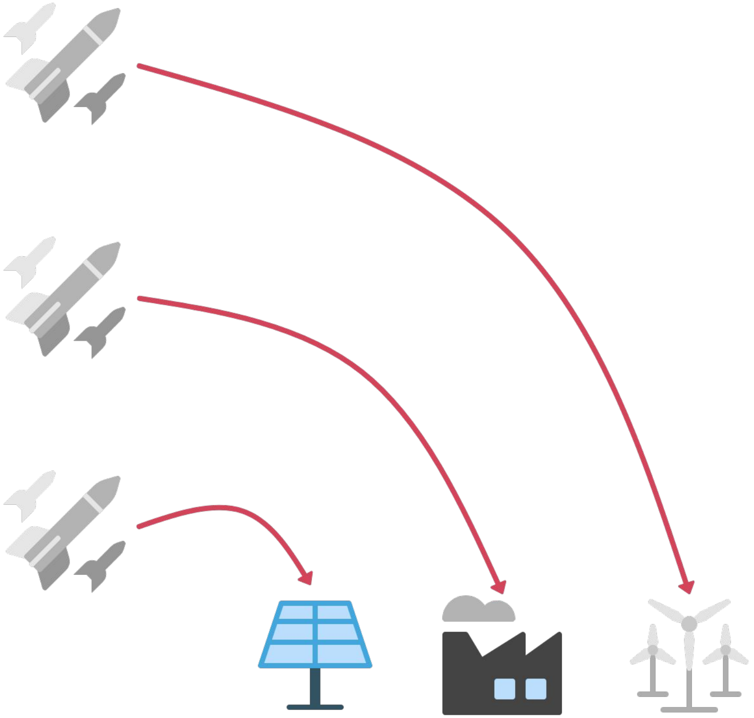
I said "probably not."

Origins in missile defense

Say an adversary is **attacking our power grid**.

We want to **disable incoming missiles** and preserve our infrastructure.

Each missile has a value. This could be the value of the infrastructure it is aimed at.

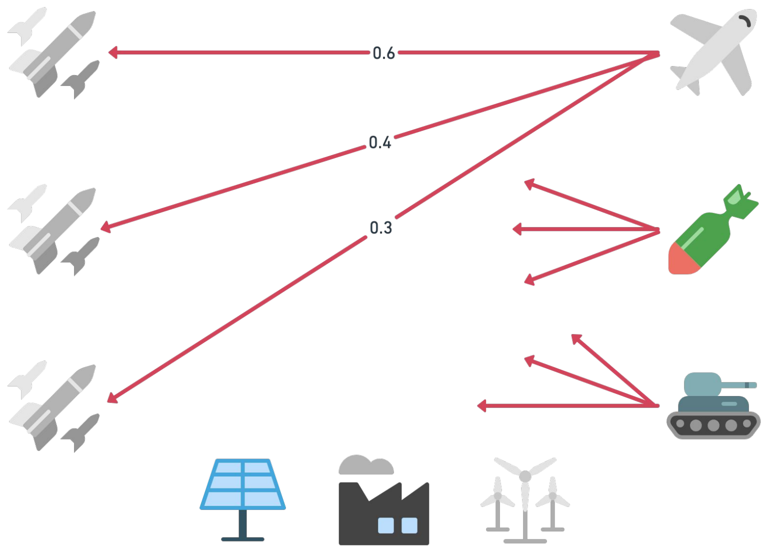


It may look like a simple assignment problem

We have a **number of countermeasures** we can deploy against incoming targets.

Each countermeasure **can attack one target**.

Countermeasures have a **probability for disabling each target**, if assigned to them.



[Manne-1958] The canonical (static) form

$$
\begin{aligned}
\min \quad & \sum_t v_t \prod_w (1 - p_{wt})^{x_{wt}} & \\
\text{s.t.} \quad & \sum_t x_{wt} = 1 & \forall \quad w \in W \\
& x_{wt} \in \{0, 1\} & \forall \quad w \in W, t \in T
\end{aligned}
$$

- W is the set of weapons
- T is the set of targets
- v_t is the value of disabling target t
- $p_{w,t}$ is the probability target t survives weapon w, if assigned
- $x_{w,t}$ is 1 if w is assigned to t, 0 otherwise

[Manne-1958] The canonical formulation

$$\min \quad \sum_t v_t \prod_w (1 - p_{wt})^{x_{wt}}$$

$$\text{s.t.} \quad \sum_t x_{wt} = 1 \qquad \forall \quad w \in W$$

$$x_{wt} \in \{0,1\} \qquad \forall \quad w \in W, t \in T$$

😬 **This is the hard part.**

- W is the set of weapons
- T is the set of targets
- v_t is the value of disabling target t
- $p_{w,t}$ is the probability target t survives weapon w, if assigned
- $x_{w,t}$ is 1 if w is assigned to t, 0 otherwise

[Kline-2019] Current approaches

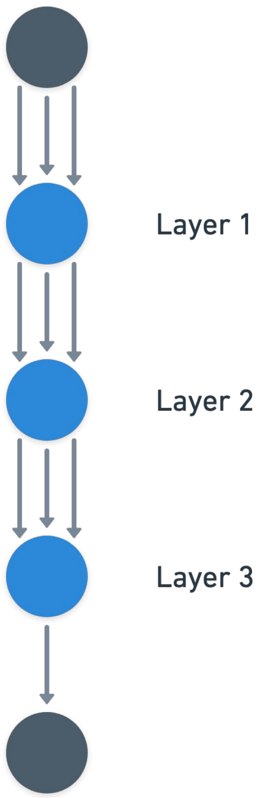
- **Most approaches use heuristics.**
- **There's very little in the way of exact methods.**
- **Many of the exact methods either vastly simplify the problem assumptions (all weapons are the same) or approximate the canonical model (piecewise linearity).**

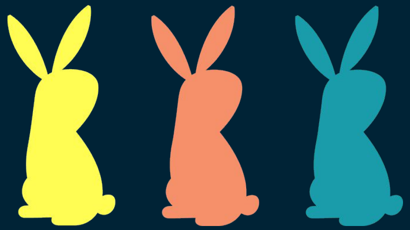
🤔 Why a DD formulation should work…

- [Bergman-2018] Decision diagrams easily encode nonlinear objectives and constraints.
- [Bergman-2014] Restricted decision diagrams can be used as an effective primal heuristic.

🤔 Why a DD formulation shouldn't work...

- There's very little structure to this problem.
- Useful dual bounds can require lots of work.
- It's challenging to find dominant nodes.





models

[Hammond-2016] DP formulation is a bit simpler

$$
\begin{aligned}
\min \quad & \sum_t z_{|W|t} & & \\
\text{s.t.} \quad & z_{0t} = v_t & \forall \quad & t \in T \\
& z_{wt} = \begin{cases} z_{w-1,t}(1 - p_{wt}) & \text{if } x_w = t \\ z_{w-1,t} & \text{else} \end{cases} & \forall \quad & w \in W, t \in T \\
& x_w \in T & \forall \quad & w \in W
\end{aligned}
$$

[Hammond-2016] DP formulation is a bit simpler

No exponents!

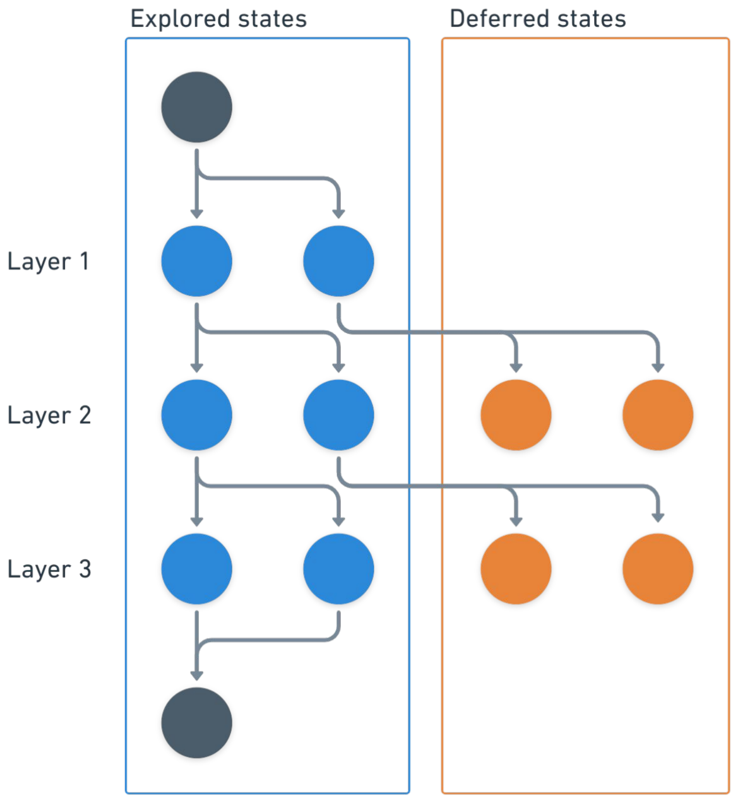
$$
\begin{aligned}
\min \quad & \sum_t z_{|W|t} \\
\text{s.t.} \quad & z_{0t} = v_t & \forall \quad t \in T \\
& z_{wt} = \begin{cases} z_{w-1,t}(1 - p_{wt}) & \text{if } x_w = t \\ z_{w-1,t} & \text{else} \end{cases} & \forall \quad w \in W, t \in T \\
& x_w \in T & \forall \quad w \in W
\end{aligned}
$$

🐰 Hop explores rectangles using Best-First Search

SEARCH

Restricted diagram

Selectively explore some states now and some states later

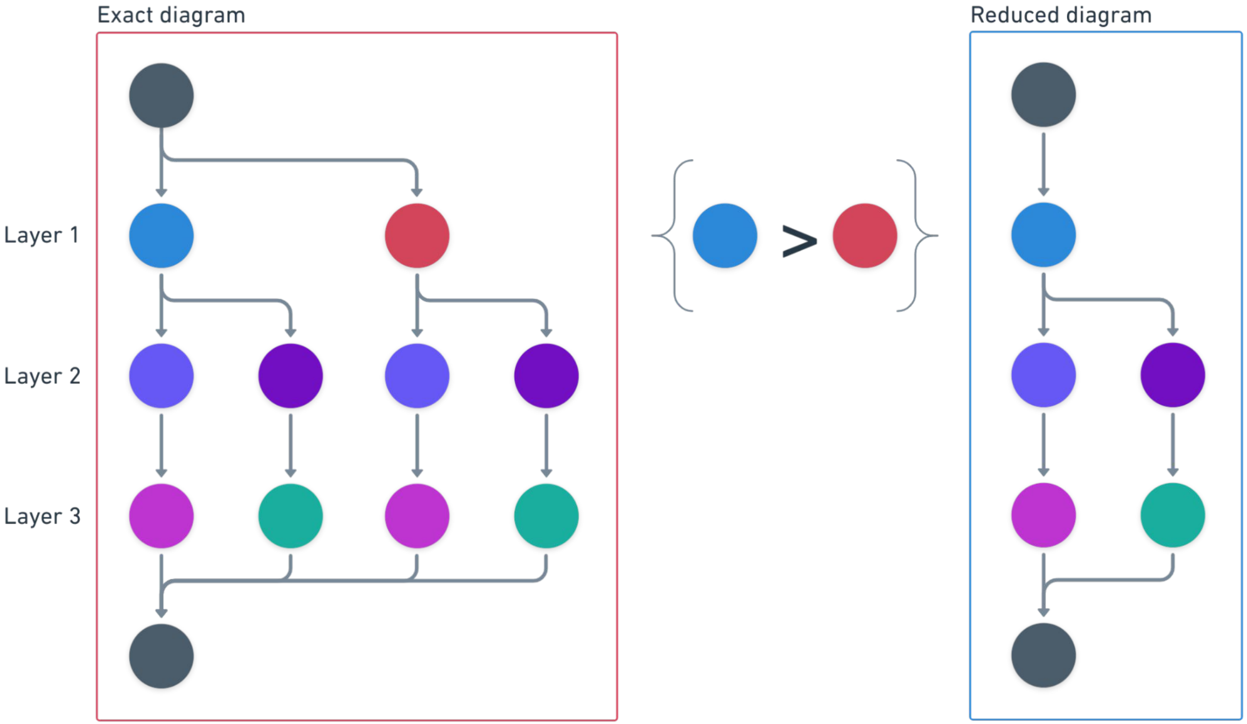


🐰 Hop reduces diagrams during search

INFERENCE

Reduced diagram

Learn as we explore to avoid unproductive branches of the search tree.

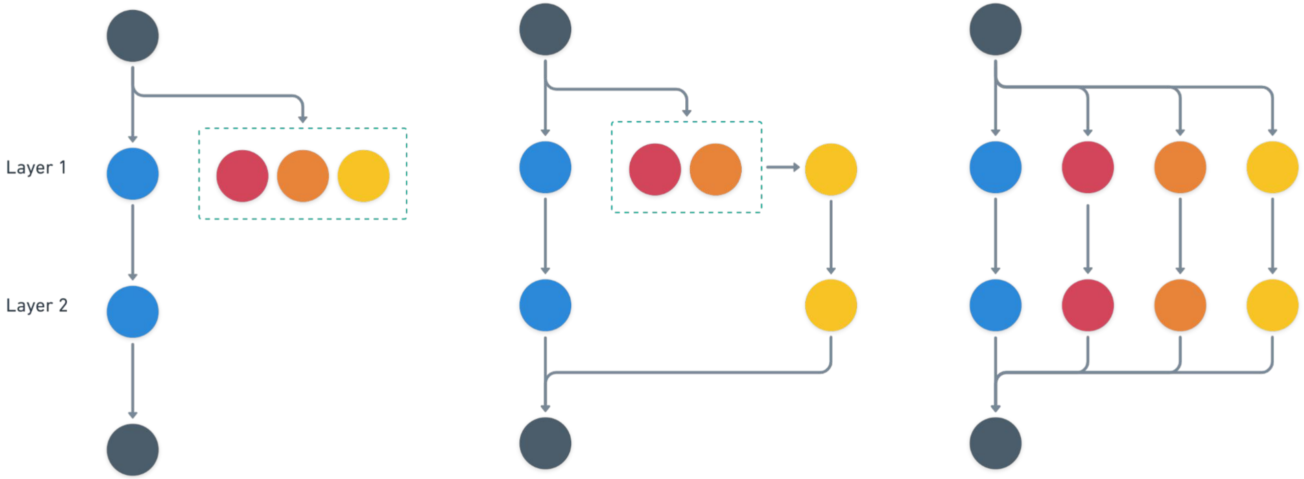


🐰 Hop expands states on demand

RELAXATION

State expansion

Create new exact states as requested by the search. This avoids merging wide layers.



❌ Dual bounds from DP formulation

$$
\begin{aligned}
\max \quad & \sum_t \lambda_{|W|t} \\
\text{s.t.} \quad & \lambda_{0t} = v_t \prod_w (1 - p_{wt}) && \forall \quad t \in T \\
& \lambda_{wt} = \begin{cases} \lambda_{w-1,t} & \text{if } x_w = t \\ \frac{\lambda_{w-1,t}}{(1-p_{wt})} & \text{else} \end{cases} && \forall \quad w \in W, t \in T \\
& x_w \in T && \forall \quad w \in W
\end{aligned}
$$

Can be updated alongside primal bounds during search.

❌ Dual bounds from MMR [Ahuja-2007]

$$\begin{aligned}
\min \quad & \sum_t v_t (1 - p_{wt}^{\max})^{x_{wt}} & & \\
\text{s.t.} \quad & \sum_t x_{wt} = 1 & \forall \quad & w \in W \\
& x_{wt} \in \{0, 1\} & \forall \quad & w \in W, t \in T
\end{aligned}$$

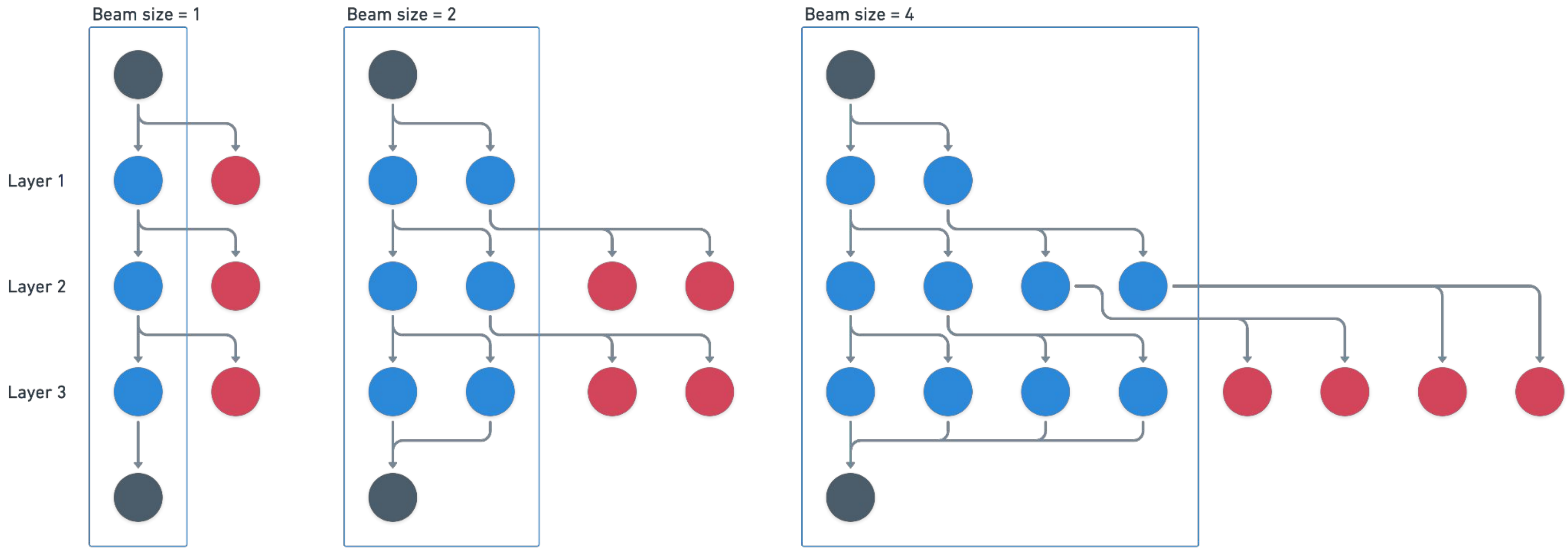
Can be solved with a greedy algorithm.

❌ Dominance detection

- At a given layer n, if all of the z values are better for node a than node b, then node a dominates node b.
- This can be done globally across layers, too.
- The list of dominant nodes grows very large.

✅ [Kuroiwa-2023] Domain Independent DP

Complete Anytime Beam Search



results

🧪 Test instances

Most papers use instances randomly generated based on [Ahuja-2007].

- 5-80 weapons
- 5-160 targets
- $\mu = 1$ of each weapon

[Andersen-2022] provides bigger instances with $\mu > 1$ and lower bounds.

- 50-500 weapons
- 100-1000 targets
- $\mu = 1, 2,$ and 3 per weapon

🧪 Solvers & models

DP formulation

- Hop using BFS and CABS
- DIDP using CABS and LNBS

DP and canonical formulations

- Hexaly (commercial heuristic solver)

Objective after 10 min for [Andersen-2022] - μ = 1

Weapons	Targets	DIDPPY (CABS)	DIDPPY (LNBS)	Hop (BFS)	Hop (CABS)
50	100	1,833	1,838	1,857	1,792
100	200	3,065	3,098	3,107	3,028
150	300	4,074	4,159	4,112	4,043
200	400	5,373	5,398	5,468	5,372
250	500	7,173	7,228	7,295	7,227
300	600	8,790	8,848	8,729	8,640
350	700	9,696	9,785	9,723	9,712
400	800	12,198	12,238	12,190	12,170
450	900	13,272	13,373	13,270	13,250
500	1,000	14,017	14,100	14,046	13,984

Objective after 10 min for [Andersen-2022] - μ = 1

Weapons	Targets	Hexaly (Canonical)	Hexaly (DP)	DIDPPY (CABS)	Hop (CABS)	Lower Bound
50	100	1,778	1,826	1,833	1,792	NA
100	200	2,970	2,989	3,065	3,028	NA
150	300	3,884	3,979	4,074	4,043	NA
200	400	5,171	5,393	5,373	5,372	4,970
250	500	6,974	7,573	7,173	7,227	6,689
300	600	8,438	9,813	8,790	8,640	8,136
350	700	9,381	11,311	9,696	9,712	9,049
400	800	11,882	14,820	12,198	12,170	11,489
450	900	12,933	16,979	13,272	13,250	12,520
500	1,000	13,565	18,699	14,017	13,984	13,139

Conclusions & next steps

- Beam search can be very effective with DD restriction.
- Gaps after 10 minutes were all within 3.2% to 4% for Hexaly (Canonical) and in 5.8% to 8% for Hop (CABS).
- What does it take to close the gap? Time? Heuristics? Merge nodes? Threads?
- Extend the model to $\mu = 2$ and $\mu = 3$ weapons per type. This likely requires additional modeling.
- Can we combine beam search with diagram relaxation?

References - WTAP

- Ravindra K. Ahuja, Arvind Kumar, Krishna C. Jha, and James B. Orlin. "Exact and heuristic algorithms for the weapon-target assignment problem." Operations research 55, no. 6 (2007): 1136-1146.

- Alexandre Colaers Andersen, Konstantin Pavlikov, and Túlio AM Toffolo. "Weapon-target assignment problem: Exact and approximate solution algorithms." Annals of Operations Research 312, no. 2 (2022): 581-606.

- Lloyd Hammond. "Application of a dynamic programming algorithm for weapon target assignment." Weapons and Combat Systems Division (2016).

- Alexander Kline, Darryl Ahner, and Raymond Hill. "The weapon-target assignment problem." Computers & Operations Research 105 (2019): 226-236.

- Alan S. Manne. "A Target-Assignment Problem." Operations Research 6.3 (1958): 346-351.

References - DDs and DIDP

- David Bergman and Andre A. Cire. "Discrete nonlinear optimization by state-space decompositions." Management Science 64, no. 10 (2018): 4700-4720.

- David Bergman, Andre A. Cire, Willem-Jan van Hoeve, and Tallys Yunes. "BDD-based heuristics for binary optimization." Journal of Heuristics 20 (2014): 211-234.

- Ryo Kuroiwa, and J. Christopher Beck. "Domain-independent dynamic programming: Generic state space search for combinatorial optimization." In Proceedings of the International Conference on Automated Planning and Scheduling, vol. 33, no. 1, pp. 236-244. 2023.

- Ryo Kuroiwa, and J. Christopher Beck. "Solving domain-independent dynamic programming problems with anytime heuristic search." In Proceedings of the International Conference on Automated Planning and Scheduling, vol. 33, no. 1, pp. 245-253. 2023.

Thank you!