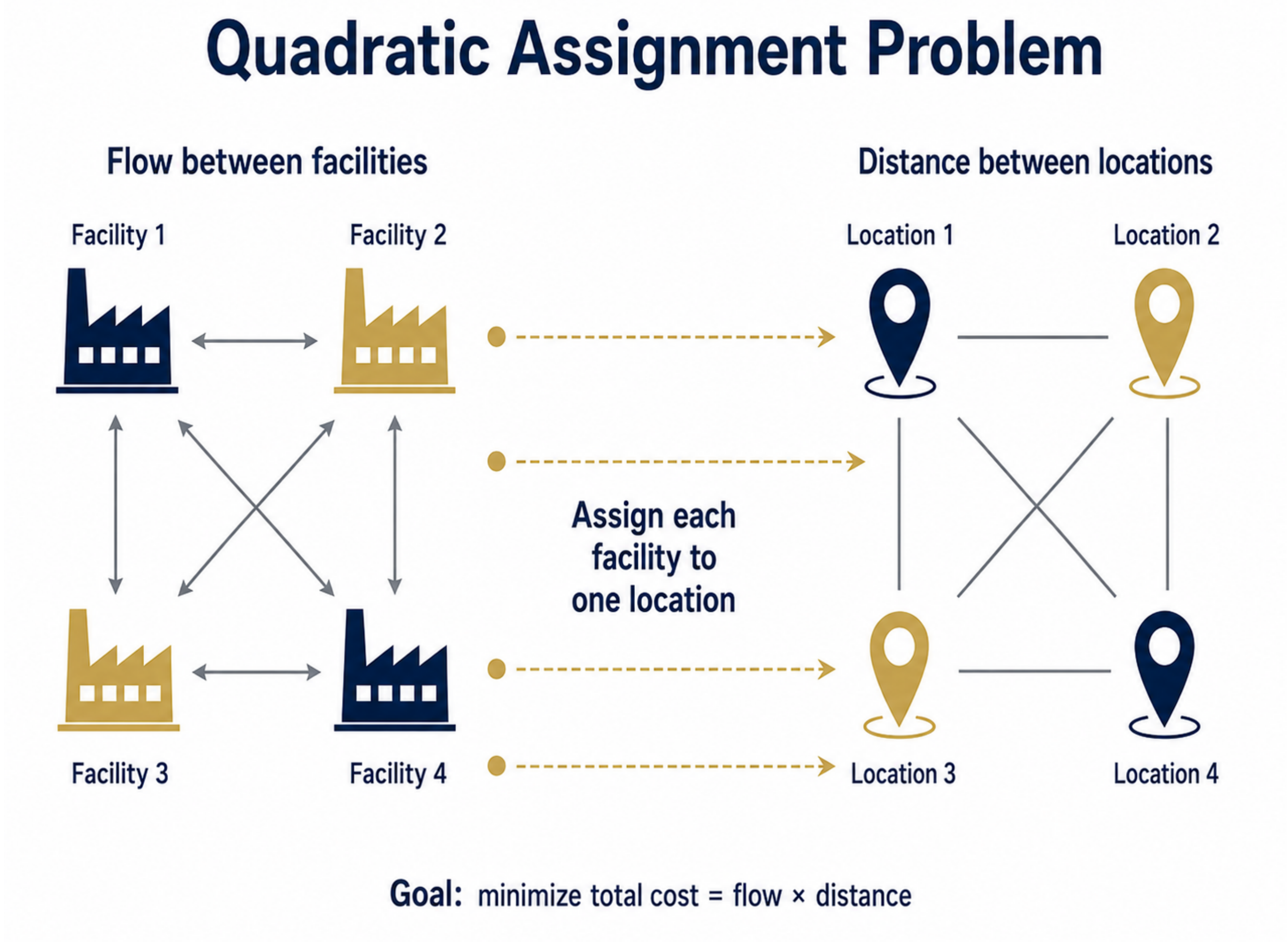
Rounding the Quadratic Assignment Problem from a Single SDP Solve

Rui Gong, Mohit Singh, Alejandro Toriello
H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology

Core idea: turn one SDP dual certificate into a cost-to-go approximation for rounding.

1. Solve once — obtain one dual SDP solution → **2. Interpret** — dual slack ⇒ cost-to-go surrogate → **3. Round** — fix facility-location pairs greedily

1. The quadratic assignment problem



- **QAP:** assign facilities to locations when pairwise *flows* interact with pairwise *distances*.
- In permutation-matrix form,

$$\min_{X \in \mathrm{Perm}(n)} \mathrm{tr}(AXBX^\top) + \mathrm{tr}(CX^\top).$$

- With $x := \mathrm{vec}(X)$, $Q := B \otimes A$, and $c := \mathrm{vec}(C)$,

$$\min_{x \in \mathrm{Perm}(n)} x^\top Q x + c^\top x.$$

2. The SDP certificate we exploit

One relaxation, two uses: it provides a lower bound and the dual structure used to score partial assignments.

Let

$$M := \begin{bmatrix} I_n \otimes e^\top \\ e^\top \otimes I_n \end{bmatrix}, \qquad b := \begin{pmatrix} e \\ e \end{pmatrix}.$$

A compact primal relaxation is

$$\begin{aligned} \min \quad & \langle Q, X \rangle + c^\top x \\ \text{s.t.} \quad & \begin{pmatrix} 1 & x^\top \\ x & X \end{pmatrix} \succeq 0, \\ & Mx = b, \qquad \mathcal{G}(X) = 0, \\ & \mathrm{diag}(X) = x, \qquad X \geq 0. \end{aligned}$$

Its dual certificate contains

$$\begin{pmatrix} t & \frac{1}{2}(c + d - M^\top y)^\top \\ \frac{1}{2}(c + d - M^\top y) & \underbrace{Q - \mathrm{Diag}(d) - \mathcal{G}(W) - P}_{H} \end{pmatrix} \succeq 0, \qquad P \geq 0.$$

Key shift: the lower-right dual slack matrix H is not discarded after bounding; it becomes the scoring engine for rounding.

3. Dual-guided VFA rounding

Given a dual solution (y^*, d^*, W^*, P^*), define

$$H := Q - \mathrm{Diag}(d^*) - \mathcal{G}(W^*) - P^*.$$

For each currently feasible facility-location pair (i, j), compute

$$\mathcal{C}_{ij}(S) = c_{ij} + 2\delta_S^\top H_{:,ij} + Q_{ij,ij} + \mathcal{C}(S \cup \{(i,j)\}).$$

Step 1. Start from the empty partial assignment $S = \emptyset$.

Step 2. Score every feasible pair (i, j) using the dual-induced VFA surrogate.

Step 3. Fix a minimizer, delete row/column conflicts, and repeat until a permutation is built.

Computational implication: after the single SDP solve, the rounding stage is lightweight and requires no repeated SDP optimization.

Main contributions

- Reuse a **single SDP dual solution** as a structured rounding guide.
- Interpret the dual slack through a **value-function approximation** over partial assignments.
- Obtain promising preliminary behavior on **structured QAPs** and **graph-isomorphism-inspired QAPs**.

4. Why does the score make sense?

Let $N = [n] \times [n]$, let δ_S be the indicator vector of a partial assignment S, and let $\hat{S} := S \cup \Gamma(S)$.

True dynamic-programming recursion.

$$\begin{aligned} \mathcal{C}^*(S) &:= \min_{x \in \mathrm{Perm}(|N \setminus \hat{S}|)} x^\top Q_{N\setminus\hat{S}} x + c_{N\setminus\hat{S}}^\top x + \delta_S^\top Q_{:,N\setminus\hat{S}} x \\ &= \min_{(i,j) \in N\setminus\hat{S}} \left\{ c_{ij} + Q_{ij,ij} + 2\delta_S^\top Q_{:,ij} + \mathcal{C}^*(S \cup \{(i,j)\}) \right\}. \end{aligned}$$

SDP-based surrogate. If the dual bound is tight, $b^\top y^* - t^* = \mathcal{C}^*(\emptyset)$, define

$$g^* := c + d^* - M^\top y^*, \qquad H := Q - \mathrm{Diag}(d^*) - \mathcal{G}(W^*) - P^*.$$

Then

$$\begin{aligned} \mathcal{C}(S) := b^\top y^* - \sum_{(i,j) \in S} [M^\top y^*]_{ij} \\ + \min_{\hat{z}} \hat{z}^\top H_{N\setminus\hat{S}} \hat{z} - 2{g^*}_{N\setminus\hat{S}}^\top \hat{z} - 2\delta_S^\top H_{:,N\setminus\hat{S}} \hat{z}. \end{aligned}$$

This yields the one-step bound

$$\mathcal{C}(S) \leq c_{ij} + 2\delta_S^\top H_{:,ij} + Q_{ij,ij} + \mathcal{C}(S \cup \{(i,j)\}).$$

Exact recursion along an optimal trajectory

Suppose that $\mathcal{C}(\emptyset) = \mathcal{C}^*(\emptyset)$ and that every partial assignment generated by the algorithm lies on an optimal trajectory. Then

$$\mathcal{C}(S) = c_{ij} + 2\delta_S^\top H_{:,ij} + Q_{ij,ij} + \mathcal{C}(S \cup \{(i,j)\}).$$

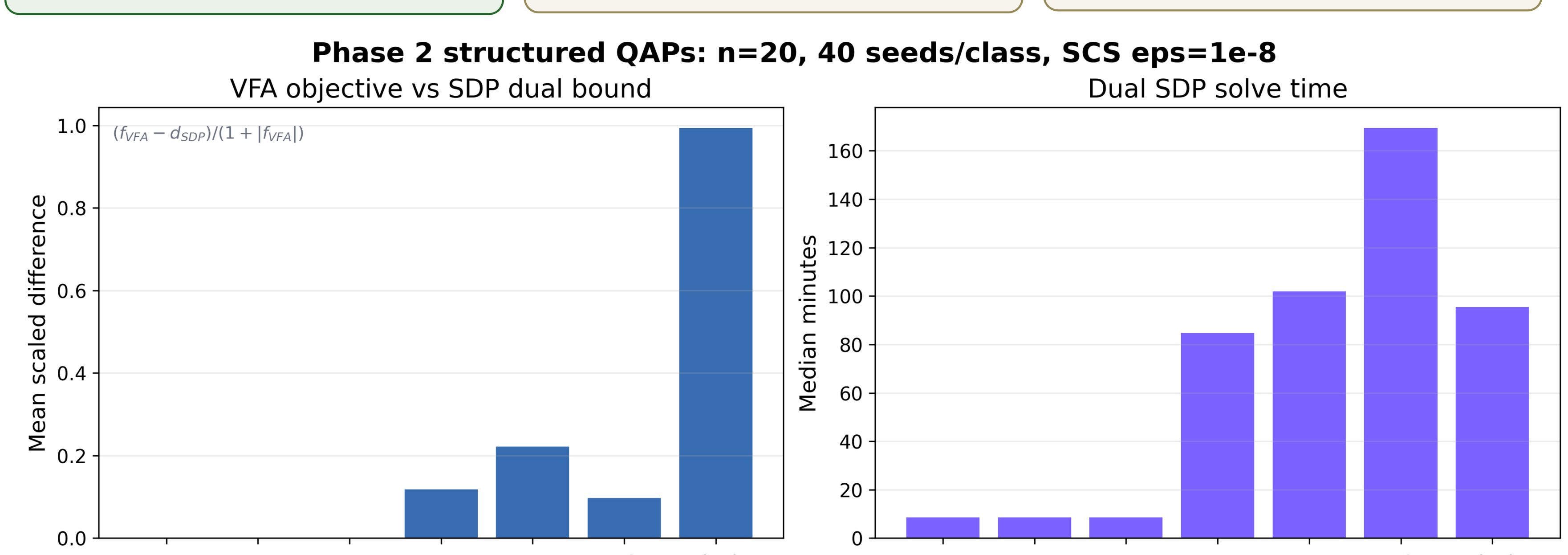
Thus, along an optimal trajectory, the SDP-based approximation recovers the correct one-step dynamic-programming recursion.

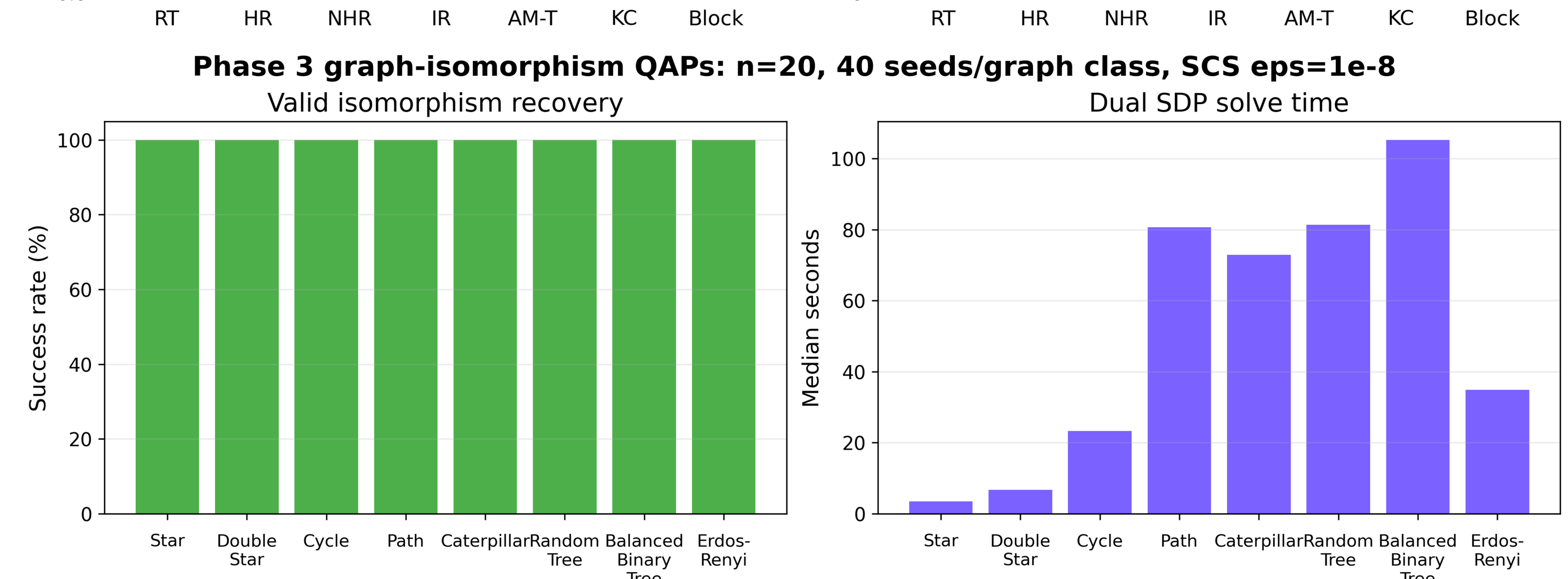
5. Preliminary computational evidence

100% — valid graph-isomorphism recovery

≈ 4 s — post-SDP rounding time

Tight gaps — on several structured families





- **Structured QAPs:** near-zero VFA-dual gaps for RT, HR, and NHR; wider gaps appear for harder structured families such as Block.
- **Graph-isomorphism QAPs:** valid isomorphism recovery is 100% across the shown $n = 20$ graph classes.
- **Runtime profile:** the dual SDP solve dominates, while the post-SDP rounding stage stays around four seconds.

Phase 2: structured QAPs

Class	Gap	SDP (s)	Round (s)
RT	1.488×10^{-10}	518.203	3.992
HR	1.697×10^{-10}	515.913	4.001
NHR	1.969×10^{-10}	519.220	4.004
IR	0.118	5088.124	3.996
AM-T	0.222	6111.294	3.983
KC	0.097	10167.422	4.194
Block	0.994	5727.644	4.246

Phase 3: graph-isomorphism QAPs

Class	Gap	SDP (s)	Round (s)
Star	6.034×10^{-10}	3.490	3.829
D-Star	2.867×10^{-11}	6.727	3.869
Cycle	-9.546×10^{-12}	23.212	3.818
Path	1.421×10^{-9}	80.658	3.918
Cat.	-1.063×10^{-10}	72.932	3.962
R-Tree	-7.762×10^{-10}	81.378	3.957
BBT	5.388×10^{-10}	105.251	3.941
ER	3.334×10^{-10}	34.859	3.931

All rows use 40 seeds per class. SDP times are median dual solve times using SCS with tolerance 10^{-8}.

RT: Robinson Toeplitz; HR: Hidden Robinsonian; NHR: Noisy Hidden Robinsonian; IR: Irregular Robinsonian; AM-T: Anti-Monge Toeplitz; KC: Kalmanson Circulant.

Poster-level takeaway: one dual SDP certificate appears to provide both a bound and a useful rounding policy on these preliminary test families.