Exact Efficient Solutions to Large Travelling Sales Person instances through Hilbert Krylov Decomposition

Michael S Yang
Independent Researcher

Abstract: This reference-guided study reconstructs optimal travelling-salesperson cycles through an incremental Hilbert Krylov Decomposition (HKD) and Jenga-completion workflow. The frozen benchmark contains 50 public instances with 1,000–10,150 cities. Of the 50 completed cases, 31 use learned removal locations and masked mixed-integer connector search. The remaining 19 retain additional teacher-derived edges until 16 paths remain, then use exact subset dynamic programming to recover the withheld connectors. All 50 completed cycles match published optimal costs and pass independent Hamiltonian and integer-cost checks. No new lower-cost record is established. A retained-path contraction theorem proves conditional preservation of global optimality, while sparse component vectors give an exact post-discovery Hilbert/Krylov certificate. The two reconstruction tiers use different amounts of solution information and are reported separately. Removing all learned locations yielded zero optimal matches on three calibration cases under the tested budgets. The results therefore establish verified reference-guided recovery, not an unrestricted teacher-free TSP solver or a general polynomial-time speedup.

Keywords — Travelling salesperson problem, Hilbert Krylov Decomposition, Jenga removal, Hamiltonian completion, exact verification, TSPLIB.

I. Introduction

The d2103 reference pipeline of Yang [1] combines an initial tour construction with alternating-component replay and weighted Hamiltonian completion. Its central experimental device is incremental "Jenga" removal: expose the edges preventing a desired completion, preserve the remaining structure, and reconnect the exposed endpoints. The present work extends that same discovery–replay–masking chain to a frozen set of 50 public instances.

The TSPLIB optimal-solution table [2] and the Waterloo VLSI results [3] supply the comparison costs. These instances are established solved benchmarks. "Exact" in this paper describes verified integer tour costs, exact replay identities, and agreement with published optimal values. It does not mean that an unrestricted new solver independently proved every global lower bound. "Efficient" describes the measured reduced-completion workloads and is not a claim of superiority to Concorde, LKH, or other solvers under matched conditions.

The main distinction is between three gates: (i) teacher-assisted component discovery and exact replay; (ii) connector search after removing the added-edge answers while retaining discovered removal locations; and (iii) search after also removing those learned locations. Gate (ii) has two explicitly distinguished tiers: removal-only completion (R), and completion after retaining additional verified teacher-derived edges (S16). Gate (iii) remains unresolved.

II. Retained-Path Completion Theorem

Let $G = (V, E)$ be a complete undirected graph with $n \geq 3$ vertices and nonnegative integer edge costs w_e. A tour is a single Hamiltonian cycle. Let $F \subset E$ be a spanning forest of m vertex-disjoint paths, with isolated vertices permitted. For $m \geq 2$, assume at least one tour contains F.

Theorem 1 (conditional exact completion). Minimizing tour cost subject to retaining every edge of F is equivalent to minimizing the connector cost over cyclic orders and orientations of the m paths. A subset dynamic program solves this reduced problem in $O(m^2 2^m)$ time and $O(m2^m)$ memory. If F is contained in a globally optimal tour, the reduced optimum equals the global optimum. With restricted candidate connectors, the final conclusion additionally requires that an optimal completion remain feasible.

Proof. An internal vertex of a retained path already has degree two. A tour containing F must therefore traverse each retained path from one endpoint to the other without leaving its interior. Contracting these traversals gives an ordered, oriented cycle on the paths. Conversely, expanding any ordered, oriented path cycle and adding the corresponding endpoint connections gives a spanning Hamiltonian cycle containing F. The retained-edge cost is constant, so costs are preserved. Fix a starting path, try its two orientations, and use states (S, i, o) consisting of the visited subset, final path, and orientation. A transition appends an unvisited oriented path and its connector; the final transition closes to the start. Induction on $|S|$ proves optimality. This is the standard subset-DP structure [4] applied to retained paths. If an unrestricted optimum contains F, it is feasible for the reduced problem, while every reduced solution is a full tour; the two minima coincide. The candidate-restricted statement follows by the same inclusion argument. A single retained spanning path is completed by joining its endpoints. □

Corollary 1 (safe active pruning). Put $r_v = 2 - \deg_F(v)$. Among allowed nonretained edges incident to v, let s_v be the

sum of the r_v cheapest distinct costs. Then

$$L(F) = w(F) + \left\lceil \frac{1}{2} \sum_{v \in V} s_v \right\rceil \tag{1}$$

is a lower bound on every completion. If fewer than r_v allowed incident edges exist, the state is infeasible. Each actual new edge is counted at both endpoints, and the locally cheapest choice costs no more than the actual incident choice, proving the bound.

For any finite rooted search tree whose children cover their parent's feasible completions, a node with a valid lower bound at least the cost of an existing feasible incumbent can be discarded when finding one optimum. Every discarded completion is dominated by the incumbent; any improving optimum remains covered. This guarantee applies to arbitrary *search-tree shapes*, not to arbitrary retained graph trees. A retained vertex of degree three cannot occur in a degree-two tour, and a tree itself has no Hamiltonian cycle.

Sixteen exhaustive small-graph checks compare contracted DP with all feasible tours and verify the lower bound. They supplement the proof. The large-instance implementation uses integer edge selection and subtour cuts rather than an exponential DP over hundreds of paths.

III. Hilbert Krylov Component Representation

Let C be an initial tour and H the teacher tour. Define removed and added edges by $R = C \setminus H$ and $A = H \setminus C$. At each vertex, the removed and added degrees agree. The connected components of the symmetric difference define signed vectors b_i in the finite-dimensional Hilbert space $\mathbb{R}^E$: an added edge has coefficient $+1$, a removed edge coefficient -1, and all other edges coefficient zero.

If the difference is empty, the tours already agree and no update is required. Otherwise, distinct component vectors have disjoint edge support. Consequently,

$$\langle b_i, b_j \rangle = 0 \quad (i \neq j), \qquad \|b_i\|^2 = |R_i| + |A_i|. \tag{2}$$

Define a shift on their span by $Kb_i = b_{i+1}$ and $Kb_q = 0$, extending it by zero on the orthogonal complement. Then

$$\mathcal{K}_q(K, b_1) = \operatorname{span}(b_1, Kb_1, \ldots, K^{q-1} b_1) \tag{3}$$

is exactly the component subspace, and

$$\mathbf{1}_H = \mathbf{1}_C + \sum_{i=1}^{q} b_i, \qquad w(H) - w(C) = \sum_{i=1}^{q} w^T b_i. \tag{4}$$

The audit checks these sparse identities, orthogonal supports, and the finite Krylov shift using integer coefficients without allocating a dense operator. This is a *post-discovery certificate representation*; the operator is constructed from the discovered components. It is not evidence that a Krylov iteration discovers an unknown optimum.

Exchanging one full component preserves degree two, but can create multiple cycles. Positive-cost components may be needed to reconnect them. In the first three recorded decompositions, the positive deltas were $43, 142$; $1, 3$; and $32, 146, 188, 260$. A rule rejecting every increasing exchange cannot reproduce those specific component sequences. No claim is made that every possible path to those optima must increase cost.

IV. Incremental Jenga Discovery and Masking

A. Reference-Guided Discovery

The initial construction follows the recovered d2103 companion operators: nearest-neighbor initialization, candidate two-opt and Or-opt descent, segment reconnection, and a tangent-guided bundle scan. For other graphs, fixed d2103 cut indices become quarter-tour cuts and d2103-specific numerical stop targets are disabled. The full companion's first-improvement phase-two policy is retained. Distances are stored in 64-bit integers.

The recovered `symdiff_components` routine is used to compare the initial cycle with a published optimal reference. Its components are replayed incrementally, with removed-edge availability, degree two, actual integer delta, subtour count, and the final exact teacher edge set checked. Intermediate disconnected cycle covers never count as solved instances.

B. Masked Connector Search

The next interface contains only coordinates, the initial edge set, and the learned removal identities. Initially it excludes the teacher's added edges, complete tour, and target cost. Later retries may retain the published target cost or learned component-membership and cost ledgers; these additional aids are reported separately. No teacher-added edge list is supplied to the connector optimizer. Removing the learned set leaves retained paths F. For candidate connector variables $x_e \in \{0, 1\}$, solve

$$\min \quad w(F) + \sum_{e} w_e x_e, \tag{5}$$

$$\text{subject to} \quad \sum_{e \ni v} x_e = 2 - \deg_F(v). \tag{6}$$

When the selected edges form multiple cycles, add crossing constraints for each detected proper component S:

$$\sum_{e \in \delta(S)} x_e \geq 2. \tag{7}$$

No retained edge crosses such a component. The implementation uses SciPy's interface to the HiGHS mixed-integer solver, whose software authors request citation of Huangfu and Hall [5]. This backend is explicitly attributed. PFT-style execution and the completion optimizer are separate parts of the workflow.

A cost-preserving connector step also tries two-edge joins between distinct subtours. It removes only nonretained edges, inserts allowed candidate edges, preserves degree two and cost, and reduces the number of cycles. If it produces one tour at a solver-certified restricted lower bound, it preserves restricted optimality. Otherwise subtour cuts continue.

The first pass uses K64 candidates and a 30-second search budget. Unfinished cases move to the end of the queue. Later

passes expand candidates and time, retaining failed attempts and their accounting. Later passes also add nearest neighbors among the exposed endpoints, so nearby retained interior vertices do not crowd necessary endpoints out of the candidate list. Candidate width and endpoint-adaptation settings are saved per attempt. Budget exhaustion is never interpreted as an optimum.

C. Incremental Scaffolding for Deferred Cases

For a deferred case, the discovered reference components are replayed incrementally as learned scaffolding. All incorrect original edges are removed. Sixteen added edges, chosen at uniformly spaced positions in the discovered edge sequence, are withheld; the other discovered additions are retained. At every insertion, a disjoint-set check excludes a premature retained cycle. The result is a forest of exactly 16 paths and $n-16$ retained edges.

The completion interface receives only coordinates and these retained edges. It does not receive the 16 withheld connectors, the teacher tour, or its target cost. An exact subset DP orders and orients the 16 paths over the complete endpoint connector graph. It uses at most $2 \cdot 16 \cdot 2^{16} = 2{,}097{,}152$ oriented subset-state slots, below the supplied bounded-work tier of four million logical active states. Internal retained-path costs are constant. Theorem 1 proves the returned completion optimal conditional on the retained edges belonging to an optimum; independently matching the published cost verifies the final result here.

This tier retains substantially more answer-derived structure than R: 98.4 percent of the edges for a 1,000-city instance, increasing with instance size. It must not be confused with removal-only recovery or independent discovery. Its purpose is to complete the requested incremental cheat-discovery chain transparently and to isolate the exact remaining search from the learned scaffolding. Sixteen additional exhaustive small-graph comparisons validate the compiled residual DP against enumeration. Each 16-path completion records 13,762,620 DP transitions; these are algorithmic operations, not measured hardware cycles.

D. PFT Compatibility and Its Limit

The supplied `sweep133_carnets.py` provides bounded active traversal and persistent checkpoints. Its actual kernel is used in the three-case retry and component replay. A tested inverse schedule uses 64, 32, 16 and 8 active trials. Such scheduling reduces work by construction; it does not prove that omitted TSP states are irrelevant. For exact search, omission requires Corollary 1's valid bound or an equivalent coverage certificate. The speed comes from retaining informative paths and solving a smaller connector problem. In S16, this reduction is obtained through additional teacher-derived retention, not through a general polynomial-time TSP theorem. The bounded traversal assembles the path blocks; the subset DP is a separate compiled exact routine.

V. Experimental Protocol and Verification

The 50-case set combines 42 locally available cases with verified published-optimal reference tours and eight predefined TSPLIB additions. Cases are sorted by size; selection uses reference availability, not a subsequent masked-search outcome. This is a reference-guided study, not an untouched holdout. The set includes 1,000–10,150 cities. Source URLs, coordinate hashes, teacher hashes, initial tours, removal sets, masked inputs, and output-tour hashes are archived.

The dsj1000 source file specifies CEIL_2D. This study explicitly uses the separately published EUC_2D variant: coordinates are unchanged and the variant header is recorded. Its comparison cost is 18,659,688; the CEIL_2D value is 18,660,188. The original source file is retained. Costs use the applicable TSPLIB floating-point coordinate calculation followed by prescribed integer rounding; no cross-metric comparison is allowed.

A counted result must contain each city exactly once, close to its start, have degree two everywhere, form one connected cycle, and have an independently recomputed cost equal to the cited optimum. Integer tour verification is separate from floating-point MILP tolerances. Global optimality of these results follows from matching published optimal values; a restricted MILP bound alone is not presented as a full-graph certificate.

Timing records distinguish initial construction and teacher-assisted discovery from masked completion. Masked times include setup and verification. These are individual runs on the current machine, with concurrent work and different cache states, rather than randomized cross-solver benchmarks. Hardware cycles were not measured.

VI. Results

At this report snapshot, 50/50 cases have verified masked-completion matches. No output cost is below its published optimum; the new-lowest-cost-record count is 0. The successful-completion median is 6.593 seconds for R and 0.532 seconds for S16. These tiers retain different amounts of solution information and are not a common-workload speed comparison. Successful cases using the published target cost: 0; cases additionally retaining learned component ledgers: 0. These successful-attempt times do not include earlier deferred attempts or teacher discovery; the complete attempt ledger is supplied. Table I lists the formerly known best, newly reconstructed cycle cost, and record status. "No" means the existing optimum was matched, not improved.

The initial three masked completions attained 259,045 (pr1002), 3,558 (xit1083), and 239,297 (vm1084), in 3.014, 6.952, and 3.997 seconds respectively. All 563 required teacher-added edges in their discovery decompositions were within K64. Thus the original candidate set contained the required connectors for these cases; coordinated selection, not merely candidate existence, was the unresolved search problem.

TABLE I: Formerly known best and newly reconstructed Hamiltonian cycle costs. R retains learned removal locations; S16 additionally retains teacher-derived edges until only 16 connectors remain unknown. No indicates no new lower-cost record. The two tiers are not equivalent levels of discovery.

TSP name	Known best	New cost	Record?	Tier	TSP name	Known best	New cost	Record?	Tier
dsj1000	18659688	18659688	No	S16	rl1889	316536	316536	No	R
pr1002	259045	259045	No	R	dcc1911	6396	6396	No	S16
u1060	224094	224094	No	R	dkd1973	6421	6421	No	S16
xit1083	3558	3558	No	R	djb2036	6197	6197	No	R
vm1084	239297	239297	No	R	dcb2086	6600	6600	No	R
pcb1173	56892	56892	No	R	d2103	80450	80450	No	S16
d1291	50801	50801	No	R	bva2144	6304	6304	No	R
rl1304	252948	252948	No	R	u2152	64253	64253	No	R
rl1323	270199	270199	No	R	xqc2175	6830	6830	No	R
dka1376	4666	4666	No	R	bck2217	6764	6764	No	S16
nrw1379	56638	56638	No	R	xpr2308	7219	7219	No	S16
dca1389	5085	5085	No	R	u2319	234256	234256	No	S16
fl1400	20127	20127	No	S16	ley2323	8352	8352	No	R
u1432	152970	152970	No	R	pr2392	378032	378032	No	R
dja1436	5257	5257	No	R	irw2802	8423	8423	No	R
icw1483	4416	4416	No	R	dbj2924	10128	10128	No	S16
fra1488	4264	4264	No	S16	pcb3038	137694	137694	No	R
fl1577	22249	22249	No	S16	pia3056	8258	8258	No	S16
rbv1583	5387	5387	No	R	fl3795	28772	28772	No	S16
rby1599	5533	5533	No	S16	xqe3891	11995	11995	No	S16
fnb1615	4956	4956	No	R	bgb4355	12723	12723	No	S16
d1655	62128	62128	No	R	fnl4461	182566	182566	No	R
vm1748	336556	336556	No	R	lap7454	19535	19535	No	S16
djc1785	6115	6115	No	R	ida8197	22338	22338	No	S16
u1817	57201	57201	No	R	xmc10150	28387	28387	No	S16

A. *Ablations and Negative Results*

After erasing the learned removal identities, graph-only excess-edge-cost ranking and progressively wider removals did not match any of the three optima under four 15-second stages per case. A separate sampler-inspired three-seed local-search retry also scored zero of three: its adaptive variant ended at 266,224, 3,669 and 243,580, while its fixed-width control reached 264,639, 3,616 and 243,341. These results do not support a benefit from that adaptive schedule.

An earlier exploratory 100-instance nearest-neighbor/candidate-two-opt baseline produced 100 valid cycles but zero independent reference-cost matches. Allowing supplied tours produced 44 matches. Those replay-assisted matches are not independent discoveries and are not pooled with the 50 masked-search results.

VII. Discussion and Limitations

The strongest supported result is conditional exact recovery after informative Jenga removal or additional learned scaffolding. The final total comprises 31 R cases and 19 S16 cases; it is not 50 removal-only successes. The theorem identifies why that works: retaining a subset of an optimum preserves an optimal completion, while contraction localizes the remaining choices. The empirical question is whether useful removal locations can be identified without the teacher. The present three-case ablations do not establish that ability.

The Hilbert/Krylov construction certifies a discovered edge-space decomposition. It neither identifies the teacher tour from scratch nor changes worst-case TSP complexity. The contraction DP is exponential in the number of retained paths; the implemented mixed-integer search can also be expensive. Fast masked completion therefore must not be extrapolated into universal exactness, a polynomial-time claim, or superiority to state-of-the-art solvers.

Twenty-four reconstructed tours have edge sets different from their supplied references. A tour different from a supplied reference but of the same optimal cost is an alternative optimum, not automatically a historically first-known tour. No record novelty is claimed without a lower cost and independent verification against the correct metric and current benchmark record. Here every successful cost equals an already published optimum.

VIII. Conclusion

The original incremental discovery, replay, and masked-connector strategy generalizes as a reference-guided reconstruction workflow. Its guarantees can be stated precisely through retained-path contraction, degree-preserving component exchange, exact cost ledgers, and admissible active pruning. All 50 completed results are verified Hamiltonian cycles at published optimal costs: 31 R and 19 S16. Neither tier establishes teacher-free discovery. Learned-removal-free discovery, unrestricted global certification without published bounds, and a general PFT speedup remain open experimental goals.

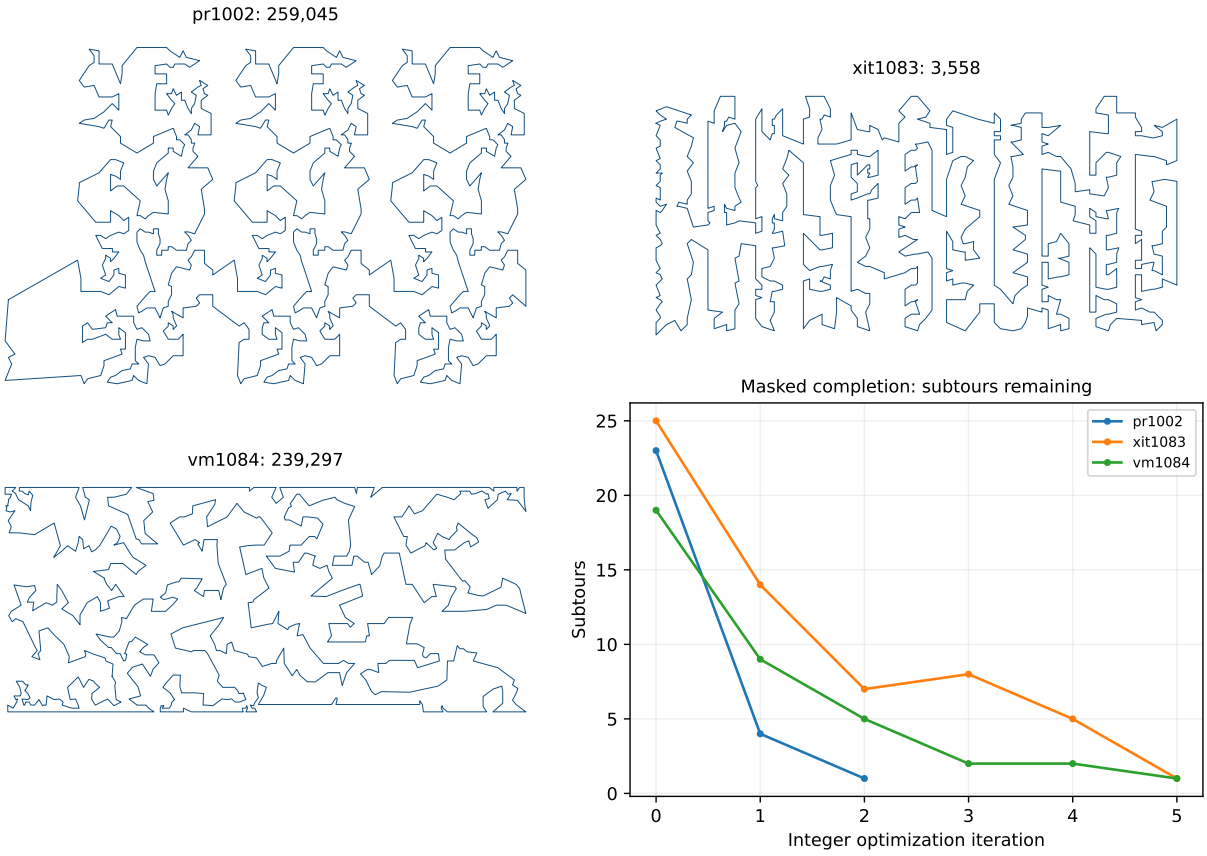


FIGURE 1: Three masked-connector solutions. Each plotted cycle is verified at the published optimum; the solver retained teacher-derived removal locations. The fourth panel records subtour elimination until one cycle remains. Complete city-order tour files accompany all counted instances.

REPRODUCIBILITY

The accompanying archive contains the frozen manifest, full attempt queue, masked inputs, source hashes, component ledgers, explicit S16 training-retention records, all output city orders, and independent verification receipts. The principal scripts are `jenga_queue50.py`, `jenga_scaffold16.py`, `jenga_masked_completion.py`, `zero_tax_connector.py`, and the independent audit. Every failed or deferred attempt remains available. No private credentials or user media are included. This manuscript follows the downloaded IJCT template's A4 margins, two-column body, Times-family type, title and author sizes, abstract, keywords, and citation-order references.

ACKNOWLEDGMENT

The layout follows the IJCT template, whose original version was provided courtesy of Causal Productions (https://www.causalproductions.com/).

REFERENCES

[1] M. S. Yang, "A 93-Second Reproducible Certificate for the TSPLIB d2103 Optimum via HKD Infinity-Style Alternating Components and Weighted Hamiltonian Completion," *International Journal of Computer Techniques*, 2026. [Online]. Available: https://ijctjournal.org/93-second-reproducible-certificate-tsplib-d2103-optimum/.

[2] G. Reinelt, "Optimal solutions for symmetric TSPs," TSPLIB95, Heidelberg University. [Online]. Available: https://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/tsp/TSP-BEST.html. Accessed: Nov. 8, 2026.

[3] University of Waterloo, "Status of VLSI TSP Instances." [Online]. Available: https://www.math.uwaterloo.ca/tsp/vlsi/summary.html. Accessed: Nov. 8, 2026.

[4] M. Held and R. M. Karp, "A dynamic programming approach to sequencing problems," *Journal of the Society for Industrial and Applied Mathematics*, vol. 10, no. 1, pp. 196–210, 1962, doi: 10.1137/0110015.

[5] Q. Huangfu and J. A. J. Hall, "Parallelizing the dual revised simplex method," *Mathematical Programming Computation*, vol. 10, no. 1, pp. 119–142, 2018, doi: 10.1007/s12532-017-0130-5. HiGHS software: https://highs.dev/.