EFFICIENT LEARNING OF MECHANISM DESIGNS AND COMPOSITIONALITY

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning mechanism behaviour requires a representation of how connected parts jointly constrain motion. We study whether exposing reusable constraint subassemblies improves prediction from limited data. Assur-KB represents planar mechanisms through irreducible modules, boundary interfaces, and solve dependencies while retaining source geometry. Our theoretical results characterize transfer through module interfaces and establish physical limits arising from missing interface information and accumulated sensitivity. To test the representation claim directly, the experimental design compares matched mechanism-curve pairs across training budgets within the same multilayer-perceptron family. This controlled comparison establishes the utility of the complete representation in the evaluated learning setting. Assur-derived features provide a geometric-mean $1.8\times$ advantage in sample efficiency over flattened adjacency and position inputs across budgets of 50-10,000 mechanisms. Separate experiments establish invariance under tested joint relabelings, higher valid-generation rates under dyadic composition, and a coarser structural index over 10.02 million mechanisms. In summary, the theory and experiments identify how physical constraint structure supports limited-data prediction and which interface conditions govern compositional reuse.

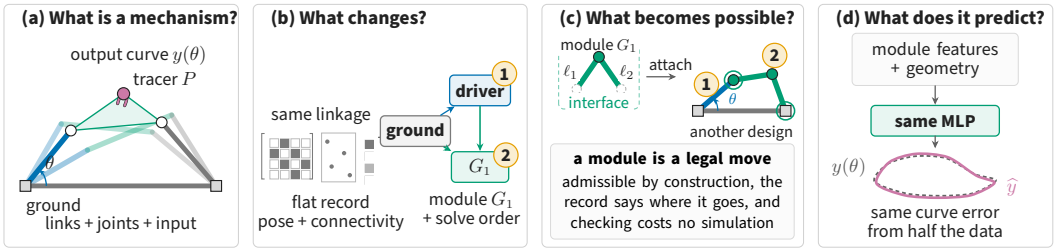


Figure 1: **Same mechanism and motion, different computational structure made explicit. (a)** A driver angle moves the links and a tracer point P draws $y(\theta)$. **(b)** The solve graph exposes an irreducible module G_1, its interface, and the order the local solves run in. **(c)** That module is then a move an agent can make, stored with the two bar lengths and the interface it needs, landing on any pair of already-solved joints, admissible by construction, and extending the solve order by one step. **(d)** The same record is what a predictor reads, so with the multilayer perceptron held fixed, module features and geometry reach a given error on the curve $y(\theta)$ from close to half the training data (§5.1).

1 INTRODUCTION

Learning the behaviour of physical systems raises a fundamental question about representation: which parts of a system provide useful units for learning and reuse? Compositional representations offer a way to share knowledge across systems by making their components and relationships explicit. Their effectiveness, however, depends on identifying units that capture the structure relevant to the behaviour being predicted. This question connects representation learning with relational inductive biases and modular learning, where the organization of a problem shapes how knowledge can transfer across examples (Battaglia et al., 2018; Alet et al., 2018).

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Mechanical linkages provide a concrete setting for studying this question. A linkage consists of connected rigid links and joints whose motion transforms a driving input into an output trajectory. Learning this relationship is difficult because the components satisfy coupled geometric constraints, where the movement of one joint depends on the configurations of other connected parts. Moreover, mechanisms with the same connectivity can produce different trajectories as their geometry or assembly configuration changes. Across mechanisms, the combinations of topology, geometry, and attachment patterns create a large space of behaviours that training examples can only partially cover.

The representational challenge is therefore to expose reusable structure at an appropriate level of granularity. Joint-level graphs describe elementary connections, but leave implicit the subassemblies whose constraints must be solved together and the dependencies between those subassemblies. A learner must recover this organization insofar as it is useful for prediction. Making it explicit could help the learner recognize recurring constraint patterns across mechanisms, reducing the amount of data needed to learn the relationship between structure and motion.

Planar mechanism theory provides a principled candidate for these representational units. Assur groups are irreducible subassemblies whose internal configurations can be determined, subject to geometric feasibility and assembly choices, once their boundary configurations are supplied (Servatius et al., 2010). Each group thus defines a conditional constraint problem with explicit interfaces. Connecting these groups according to their dependencies produces a *directed acyclic solve graph*, which specifies how local constraint solutions combine to determine the mechanism's configuration (Shai et al., 2013). Repeated across the driving input, this computation determines its motion.

We investigate whether exposing this *compositional structure* improves learning of mechanism behaviour. We represent mechanisms through their Assur groups, interfaces, and solve dependencies, while retaining the underlying geometry and assembly information. Our hypothesis is that this organization supplies a useful inductive bias by making recurring constraint structure accessible across examples. We instantiate the representation at corpus scale in Assur-KB and evaluate its value for predicting output trajectories, alongside its usefulness for structural composition and indexing. To our knowledge, this is the first large-scale evaluation of Assur-based representations for learning mechanism behaviour from limited data.

Our experiments provide evidence for improved learning when data is limited. Using the same multilayer perceptron, Assur-derived features reach comparable curve-prediction error with roughly half the training examples required by a flattened adjacency-and-position representation over the evaluated low-data range (Figure 1d). More broadly, this approach suggests that physical constraint structure can guide mechanism representation design, potentially reducing simulation requirements for predictive modelling and supporting modular reuse across related physical systems.

2 From Physical Parts to Constraint Modules

The prediction problem. A planar linkage consists of rigid links connected by revolute joints (pins) or prismatic joints (sliders). Ground fixes part of the linkage, while a driver supplies an input, such as a crank angle θ. Solving the geometric constraints determines the other joint positions and a tracer trajectory $y(\theta)$. We study learning this kinematic behaviour from mechanism examples. The regression experiment predicts the shape of the resulting curve after normalization and arc-length resampling.

A common source representation is (A, x_0, n), consisting of joint adjacency, an initial configuration, and a fixed-joint mask. Together with the designated driver and tracer, it supplies the information needed for the source simulation. A link-incidence representation additionally records which joint sets belong to the same rigid body. Both forms preserve physical connectivity, but neither explicitly identifies which constraints form a local solve problem.

A concrete module. Consider an unknown joint q connected by two bars of lengths ℓ_1, ℓ_2 to known joints p_1, p_2 (Figure 3a). Its position satisfies

$$\|q - p_1\|^2 = \ell_1^2, \qquad \|q - p_2\|^2 = \ell_2^2. \tag{1}$$

The bars must be jointly solved as for intersecting, non-tangent circles there are two solutions, while an assembly branch selects one. The pair of bars is therefore a useful computational unit, called

a *dyad*. Reusing it means reusing a conditional constraint relationship, where different boundary positions and lengths generally produce different motions.

Definition 1 (Assur group)**.** An Assur group is a subassembly that, with its boundary joints pinned, is generically rigid and independent, and has no proper subassembly with these properties (Servatius et al., 2010).

The boundary joints are the module's inputs, where its internal joints are the unknowns solved together. Irreducibility specifies the granularity when a group cannot be split into smaller sequentially solvable units under the same pinning. The dyad is the simplest example, while larger groups contain several coupled internal joints.

From modules to a solve graph. Draw an edge $G_i \to G_j$ whenever group G_j uses a boundary joint determined by G_i. For a valid Assur decomposition this gives a directed acyclic graph (DAG), and a topological traversal gives a dependency-respecting solve order (Shai et al., 2013). This graph exposes the organization of the constraint computation while preserving the mechanism's geometry. It provides the compositional structure evaluated in this paper.

Structure and geometric feasibility. Structural mobility and successful motion are distinct concepts. For L links including ground and J_1 one-degree-of-freedom lower pairs, the nominal Chebychev–Grübler–Kutzbach count is

$$\mathcal{F} = 3(L-1) - 2J_1. \tag{2}$$

This count assumes independent constraints, where special geometries or redundant constraints can change the actual mobility (Müller, 2009). An Assur decomposition describes generic constraint structure. Assembly at particular lengths and continuation through a full input cycle are geometric questions: e.g. the dyad in Eq. 1 has no real solution when its circles do not intersect. Appendix B develops these ideas and the *local solves* in detail.

3 A Compositional Representation of Mechanisms

Each record lifts one joint graph into a *solve graph* whose nodes are irreducible groups, whose edges are the dependencies between them, and whose statuses record what was verified and what was left open. The construction has two stages, a proposal stage belonging to the source format and an acceptance stage both sources share, which is what makes a structural fingerprint mean the same thing in either (Figure 2). Appendix E provides a descriptive account of the build and Appendix **??** states the five requirements behind the schema.

Write a source mechanism as $R = (J, \mathcal{L}, \tau, \Gamma, (u, v), x_0)$, with joints J, rigid links $\mathcal{L} \subseteq 2^J$ held as joint sets so that a binary bar is a link ℓ with $|\ell| = 2$, a map $\tau : J \to \{\mathrm{R}, \mathrm{P}\}$ marking each joint revolute or prismatic, grounded joints Γ, a driver turning about fixed pivot u and carrying joint v, and an optional pose x_0. Joint adjacency is the case $\mathcal{L} = \{\{i, j\} : A_{ij} = 1\}$ with $\tau \equiv \mathrm{R}$, and link incidence reads $\mathcal{L}$ off the rows of an incidence matrix B, one row per link, with τ from the slider specification, so one schema carries both corpora. The build turns R into a sequence of groups $\mathcal{G} = (G_0, \ldots, G_m)$ headed by the ground and driver G_0, where $G_k = (I_k, \partial_k, \mathcal{L}_k, T_k, \beta_k, s_k)$ for $k \geq 1$ holds the joints group k solves, its boundary, the links it owns, the tracer points it carries, its assembly branch, and its status.

Definition 2 (Valid decomposition)**.** Let $I_0 = \Gamma \cup \{v\}$ and let $K_k = I_0 \cup \cdots \cup I_{k-1}$ be the joints known before step k. The sequence $\mathcal{G}$ is a *valid decomposition* of R when the blocks $I_k \cup T_k$ partition J, the $\mathcal{L}_k$ partition $\mathcal{L}$, every boundary is already known, $\partial_k \subseteq K_k$, and every G_k with $k \geq 1$ is an Assur group on $(I_k, \partial_k, \mathcal{L}_k)$ in the sense of Definition 1.

Ownership then induces the edges $E = \{(j, k) : \partial_k \cap I_j \neq \emptyset\}$ between groups j and k, which precedence makes acyclic, so every topological order σ of $(\mathcal{G}, E)$ is a solve order. Because the $\mathcal{L}_k$ partition $\mathcal{L}$ while Γ, (u, v) and x_0 are carried through, R is recoverable from its decomposition.

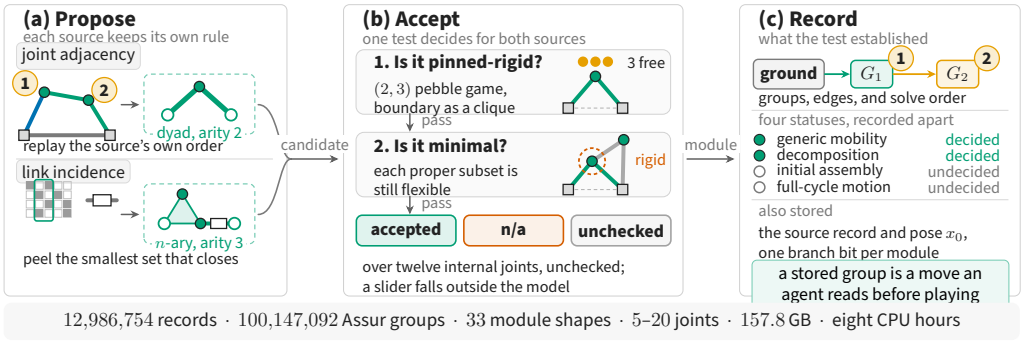


Figure 2: **Source formats(2x), acceptance test (1x) and record (1x). (a)** Each corpus proposes modules through the oracle it already ships, so joint adjacency yields the classical dyad and link incidence reaches n-ary bodies, prismatic pairs, and boundary arity above two. **(b)** Both proposals then meet the same test (Eq. 3), a module counting as pinned-rigid exactly when three pebbles remain free and as minimal exactly when every proper subset of its internal joints is still flexible. **(c)** The record keeps the solve graph beside four statuses, so a structural pass never reads as motion.

Algorithm 1 (a) The peeling loop, shared by both sources and specialized only through its proposal oracle Π, which returns a set S of links or $\perp$. **(b)** The test scoring every proposed module, where F is the pinned frame the pebble game runs on and r the rigidity outcome. Both per step in Appendix **??**.

procedure PEEL(R, Π)
  $K \leftarrow I_0 = \Gamma \cup \{v\}$; $k \leftarrow 0$
  $U \leftarrow \{j \in J : \deg j \geq 2\} \setminus K$ ▷ tracers held out
  while $U \neq \emptyset$ **do**
    $S \leftarrow \Pi(K, U, \mathcal{L})$; **if** $S = \perp$ **then fail**
    $k \leftarrow k + 1$; $\mathcal{L}_k \leftarrow S$
    $I_k \leftarrow \bigcup S \setminus K$; $\partial_k \leftarrow \bigcup S \cap K$
    $\beta_k \leftarrow$ branch of G_k realized by x_0
    $K \leftarrow K \cup I_k$; $U \leftarrow U \setminus I_k$
  carry tracers T_k; give each ℓ to its latest k
  $s_k \leftarrow$ ACCEPT$(I_k, \partial_k, \mathcal{L}_k)$ for every k
  return $\mathcal{G}$, $E = \{(j, k) : \partial_k \cap I_j \neq \emptyset\}$, σ

procedure ACCEPT$(I_k, \partial_k, \mathcal{L}_k)$
  if $\tau(j) =$ P for some $j \in I_k \cup \partial_k$ **then return** N/A
  $F \leftarrow$ complete graph on ∂_k ▷ ground
  $F \leftarrow F \cup \{\text{clique on } \ell \cap (I_k \cup \partial_k)\}_{\ell \in \mathcal{L}_k}$
  $p \leftarrow$ free pebbles of the $(2, 3)$ game on F
  $r \leftarrow [\, p = 3 \,]$ ▷ Eq. 3, left
  if $|I_k| > 12$ **then return** $(r,$ unchecked, $p-3)$
  for $\emptyset \neq I' \subsetneq I_k$ **do**
    if Rigid$(I', \partial_k, \mathcal{L}_k|_{I'})$ **then return** $(r,$ FALSE, $p-3)$
  return $(r,$ TRUE, $p-3)$

3.1 MODULES, INTERFACES, AND DEPENDENCIES

A mechanism corpus already entails a routine that orders the solves its own simulator performs, so the build reuses that routine as a proposal oracle and leaves the peeling rule with the source that defined it. PEEL (Algorithm 1a) is the loop both sources run, growing a solved set from ground and the driven joint and reading each proposed set of links as internal, where it is unknown and, boundary, where it is known. Degree-one tracer points are held out, since a link carries them rigidly, and each is reattached to the group owning its link, a step only link incidence exercises. An oracle with no next set reports a failure, so no module is inferred from a neighbour count, though no record in the build takes that path. What does occur is a peel that resolves no new joint, as in the 153,124 link-incidence records whose source incidence leaves every structural joint on the ground link, each keeping its curve and its ground group with no module below it.

Joint adjacency. For LINKS (Heyrani Nobari et al., 2022) (Appendix B.7) the oracle replays the corpus routine and returns, at each step, the two bars joining one unsolved joint to two already-known neighbours (Figure 3a). Every record of this form is ordered this way, and all of them yield the same module shape, the classical dyad.

Link incidence. For the Purwar Lab corpus (Nurizada et al., 2024) (Appendix B.8), where each rigid link retains its complete joint set, prismatic coupling is folded into the incidence first, which makes a slider an ordinary structural coupling and keeps its sliding joint from stalling the reduction at degree one. The oracle then returns the smallest set of two or more unsolved links whose Chebychev–Grübler–Kutzbach mobility against the solved part is non-positive, widening the set when none closes. This is the form that reaches n-ary bodies, prismatic pairs, and boundary arity above two.

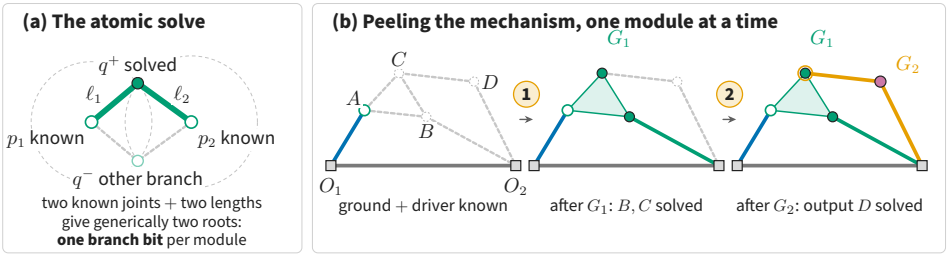


Figure 3: **Peeling proposes one module at a time against the part already solved. (a)** Two known joints and two lengths place the unknown joint at a circle–circle intersection with generically two real roots, so a mechanism with m dyads stores m branch bits. **(b)** Two steps on a six-bar, where the ternary link ABC and the bar from B to ground form one module because the ternary body enters once and fixes both B and C, and the remaining dyad closes at D.

3.2 Structural tests and geometric information

A proposal is a candidate, and Accept (Algorithm 1b) decides what it may be recorded as. Pinned rigidity is decided with the $(2,3)$ *pebble game* (Jacobs & Hendrickson, 1997; Sljoka et al., 2011), exact for generic bar-joint frameworks and near-linear in the number of constraints (in essence, two freedom tokens per joint, one token claimed by each independent bar, and the module is rigid once only the plane's three are left; Appendix B.4 describes the game from scratch). The boundary enters as a complete graph on the pinned joints, which keeps the plane's three residual freedoms explicit, and each link enters as one clique-expanded rigid body reduced to its independent constraints, so an n-ary link contributes once. Writing p for the free pebbles that remain on module k and $\mathcal{L}_k|_{I'}$ for the links meeting I' and contained in $I' \cup \partial_k$,

$$\mathsf{Rigid}(I_k, \partial_k, \mathcal{L}_k) \iff p = 3, \qquad \mathsf{Minimal} \iff \neg\,\mathsf{Rigid}(I', \partial_k, \mathcal{L}_k|_{I'}) \ \text{ for every } \emptyset \neq I' \subsetneq I_k, \tag{3}$$

and the module is accepted as Assur when both hold, with the surplus $p - 3$ stored as its residual freedoms.

Neither conjunct is spare, and a mobility count supplies neither, because the pinned count $|\mathcal{L}_k| = 2|I_k|$ is necessary for generic isostaticity and blind to minimality, since two stacked dyads satisfy it, while the first dyad alone is already pinned-rigid. Acceptance therefore rests on the pebble game, with two further failure modes in Appendix **??**. Minimality is exponential in $|I_k|$, which is affordable because Assur groups are small, and beyond twelve internal joints it is recorded as *unchecked*; no module in the build reaches that size. A module touching a prismatic pair falls outside the model, since a slider supplies no length for a bar-joint test to read.

3.3 What the record stores

Alongside R and the groups, each record names the source chunk it came from. Each physical link is given to the module resolving its latest joint, which keeps a ternary or quaternary body distinguishable from several binary bars, and each dyad stores the branch β_k its source pose realizes, since a mechanism with m dyads has 2^m assembly modes and one of them reproduces x_0, so the stored bit replays a record without a search. Modules are keyed by boundary arity $|\partial_k|$, which agrees with classical Assur order only under a one-to-one boundary-pair convention, so the schema reports arity and leaves classical order uninferred. The status s_k keeps the four outcomes of Figure 2c apart, each passed, failed, or not evaluated, and Appendix **??** gives the field-level schema.

Records are built in independent index partitions, where geometry is defined per record, so where a source supplies no coordinate array the design frame is recovered and accepted only once the traced point maps through it onto the stored curve, and 94.6% of those records end up with a validated frame. Every module the model covers passes the structural test, $286{,}467$ fall outside it and are stored as undecided, and initial assembly and full-cycle motion are undecided build-wide, since a decomposition never sees the geometry that would settle them (Appendix E).

3.4 USING THE REPRESENTATION FOR LEARNING

The learning experiment converts the decomposition into a fixed feature vector containing summaries of the modules, their interface structure, the dependency DAG, and available branch and geometric information. A three-layer MLP predicts the normalized output curve from this vector. The baseline uses the same network with flattened adjacency and position inputs. A fixed-width map is appropriate because both inputs are fixed-length and the target is resampled to a fixed number of points. Using the same network also avoids introducing relational biases from graph networks or transformers, which would confound the effect of the representation. The experiment therefore isolates the value of exposing constraint structure in the input.

The same representation supports two structural operations. A dyad can be attached to a compatible pair of existing boundary joints to construct a new candidate, after which its geometry is evaluated. A coarse fingerprint of the module counts and dependency graph can retrieve mechanisms with related constraint structure. These uses motivate the sampling and indexing experiments, while curve prediction remains the central learning test.

The combined build contains 12,986,754 mechanism records. Indexing is evaluated on 10,020,000 records, and the learning study uses 88,910 matched mechanism-curve pairs.

4 WHEN CONSTRAINT MODULES SUPPORT PREDICTION

An Assur group exposes a recurring conditional problem: given its boundary configuration, geometry, and assembly branch, determine its internal configuration. Reusing a learned solution in a new mechanism depends on three questions: are the local inputs familiar, how do local errors propagate, and which distinctions must the representation retain? Appendix D develops these questions through bounds and explicit mechanism constructions.

Local reuse and global error. A new assembly can present familiar local prediction problems. Consider conditional predictors shared by module type t and composed along a solve DAG. Let ϵ_t denote their population RMS error under training input distribution μ_t, and ν_v the true input distribution at occurrence v in the new assembly. Under the domain and Lipschitz assumptions of Theorem 2,

$$\|\widehat{y} - y\|_{L^2} \le \sum_v w_v \left[\epsilon_{t(v)} + K_{t(v)} W_2(\nu_v, \mu_{t(v)})\right]. \tag{4}$$

The bracket combines error on familiar inputs with the cost of changing local conditions: W_2 measures input-distribution displacement and K_t bounds residual variation. The weight w_v measures how strongly that error can affect the output along downstream paths. Thus local coverage and controlled propagation support reuse across assemblies. This bound concerns composed predictors; our experiments test the representational benefit within a global MLP.

Why small modules can still be difficult. Successive modules can direct motion into directions that amplify it. Two unit-bar dyad chains with identical dependency graphs and local Jacobian singular values can consequently have endpoint sensitivities 1 and 2^m to the same unit-speed crank (Theorem 7). At tangencies, a different effect appears: each solve can take a square root of the incoming displacement, reducing configuration regularity to exponent 2^{-m} after m stages (Theorem 8). These constructions explain why interface geometry matters alongside modular structure. The appendix connects trajectory-error bounds to normalized curve readouts through explicit normalization and sampling conditions.

What the representation must preserve. Even perfect local prediction cannot recover a distinction lost in its inputs. Exchanging which boundary port receives each dyad length changes output shape, including after 128-point arc-length sampling (Propositions 9–10). More generally, continuous features need enough dimensions to retain independently observable variations (Theorem 11). Finally, Assur component size and undirected graph width do not bound one another (Theorem 14): conditional solve structure provides a distinct description of complexity. Together, these results identify what useful modular representations expose: complete interfaces, recurring local problems, and their influence on the output.

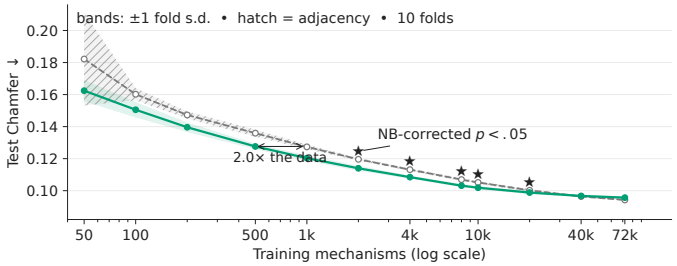


Figure 4: **Solve-graph features are worth about a factor of two in training data.** Mean symmetric Chamfer distance over ten stratified fold units, with the arrow reading the gain on the data axis and stars the sizes significant under the corrected paired test.

5 Experimental Evaluation

We first test the central hypothesis, i.e. *does making constraint structure explicit improve prediction from limited data?* Three further experiments separate that benefit from the representation's other properties, with the full tables in Appendix C.

5.1 Learning output trajectories with fewer examples

We test whether features organized by constraint structure help a learner where data is scarce, on the 88,910 matched mechanism-curve pairs spanning $N = 8, \ldots, 20$, none of which failed to decompose. Both representations feed the same three-layer MLP. The target is the output joint's normalized coupler curve, arc-length resampled to 128 points and scored by symmetric Chamfer distance, because a scalar summary would let a model match perimeter without matching shape. Ten joint-count-stratified folds give paired evaluation units, unconfounding mechanism size from fold.

The gain reads most directly on the data axis, where at 500 training mechanisms the Assur model reaches a Chamfer error of 0.1276 that the adjacency baseline does not reach until roughly 980 examples, and that ratio stays between $1.6\times$ and $2.0\times$ for every budget from 50 through 10,000, with a geometric mean of $1.8\times$ (Figure 4). On the error axis the same effect is an 11% reduction at 50 examples. The one-sided Nadeau-Bengio-corrected paired test over the ten folds falls below 0.05 at every size from 2,000 through 20,000, with p between 0.0078 and 7.4×10^{-5}.

The advantage then closes, since at 40,000 examples adjacency is lower by 0.0004, inside the fold-to-fold spread, and at the full 72,013-example pool the reversal resolves with adjacency lower by 0.0014 at 6.6×10^{-6}. The gain therefore belongs to the low-data regime and to the adjacency-plus-position baseline.

5.2 Constructive sampling improves validity within the tested family

The *compositional sampler* attaches every new joint as a two-boundary dyad. The *matched adjacency sampler* draws its edges at random at the same expected edge count and forces the same motor edge, holding density fixed so that only the construction rule varies. A theoretical branch evaluates $N = 4, \ldots, 8$ at $\bar{E}_N = 2N - 3$ and an empirical branch $N = 8, \ldots, 12$ at statistics from LINKS itself. The oracle scores connectedness, $\mathcal{F} = 1$ by Eq. 2, a solve path, and geometric simulation separately, and overall validity is their conjunction over 50,000 samples per cell.

Composition has higher overall validity at *every* evaluated size. In the theoretical branch its rate is 1.0 while the adjacency rate falls from 0.405 at $N = 4$ to 0.035 at $N = 8$. In the empirical branch composition falls from 0.284 at $N = 8$ to 0.121 at $N = 12$ and stays above adjacency sampling, whose rounded rates are at most 0.001. Over pairs where both are nonzero the geometric mean of the per-pair ratio is 54.5 (Figure 5, left).

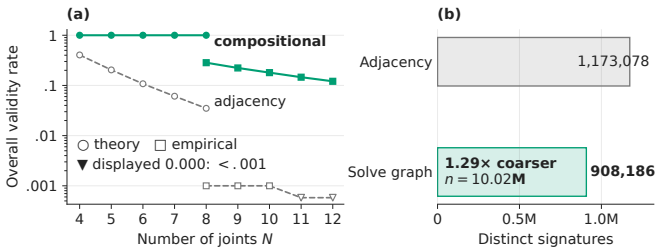


Figure 5: **Solve graphs expose useful structure before any learned model is introduced. (a)** Overall validity for compositional and matched adjacency sampling. **(b)** 10.02 million mechanisms yield fewer solve-graph fingerprints than bare adjacency hashes.

Composition reaches everything adjacency sampling reaches. A 10,000-sample-per-cell coverage control rebuilds by composition every valid mechanism the matched adjacency sampler produced, succeeding in 100% of cases at every branch and size, because the oracle's solve-path condition already establishes the dyadic stack composition assembles.

5.3 Grouping mechanisms by coarse constraint structure

Streaming 10,020,000 mechanisms, we compute for each a Weisfeiler-Lehman hash of the bare joint graph and a solve-graph fingerprint formed from group-order counts and the order-labelled condensation DAG. These yield 1,173,078 distinct hashes against 908,186 fingerprints, a $1.29\times$ coarsening averaging 11.03 mechanisms per fingerprint (Figure 5b), because mechanisms a joint-indexed hash separates are pulled together when they decompose the same way.

Which feature blocks earn their place. Leave-one-out importance at $n = 5{,}000$ over the same ten folds does not flatter the default vector, since the condensation-DAG block is the only clear contributor, the branch block the only one that costs anything, and the remaining six sit inside the fold-to-fold spread (Table 5). The two largest gains belong to a Grashof indicator and a transmission-angle summary, both describing motion feasibility, which the solve graph leaves to simulation.

5.4 Relabelling joints leaves the solve graph identical and the baseline unrecognizable

Permuting joint indices leaves the machine unchanged, so features that change under a permutation depend on the corpus's numbering convention. We generate 300 mechanisms at $N = 4, \ldots, 12$ with the compositional sampler, decompose each once, then re-decompose it under twenty random index permutations, carrying the motor pair and coupler joint through each.

Our representation is invariant under every relabelling and the baseline under almost none. Across all 6,000 trials the fingerprint, the full Assur feature vector, and its structural sub-block match their pre-permutation values exactly at every size, and every relabelled mechanism still decomposed. The adjacency-plus-position baseline is unchanged in 0.68% of the same trials, scoring 0.050 at $N = 4$ and exactly zero at every $N \geq 6$ (Table 6). That $N = 4$ residue is a small-graph artifact, since on four joints a random permutation can still be the identity or an automorphism.

Invariance is earned by sorting dyads into a canonical order before any feature is written, since the solve-order search would otherwise break ties on the lowest joint index, and 242 of the 300 mechanisms contain at least one such tie. No joint index enters the group-order counts or the condensation-DAG hash.

6 Related Work

Mechanism corpora and retrieval. LINKS delivered a large joint-level corpus of planar revolute mechanisms with simulated trajectories (Heyrani Nobari et al., 2022), and the Purwar Lab dataset

adds four-, six-, and eight-bar families carrying higher-arity links and sliders (Nurizada et al., 2024). Surveys document the shift toward data-driven synthesis (Purwar & Ge, 2025; Han et al., 2025), and a related line assembles linkages from available standard parts (Escande & Shea, 2025). LInK learns a joint representation over ten million mechanisms and warm-starts a numerical optimizer (Nobari et al., 2024), with contrastive extensions to mixed joint types (Park & Jang, 2026). We build on those corpora, but our contribution is the abstraction above them. Learned embeddings abstract instances by similarity; our decomposition abstracts mechanisms by structure, reducing them to irreducible modules with compositional interfaces. This abstraction makes admissibility closed under composition, specifies the conditions for transfer (§4), makes acceptance decidable for designs absent from the corpora (§5.2), and reduces the number of examples required at low budgets by half (§5.1).

Keeping the discrete part valid. Published generators handle discrete validity by restricting topology to a library of types (Bolanos et al., 2026), by appending Assur dyads under a grammar so that mobility holds inductively (Fogelson et al., 2023; Lipson, 2008; Vermeer et al., 2018), or by generating freely and filtering afterwards (Jadhav & Farimani, 2026). We apply the classical construction rule at a different point, since each of these systems constrains a graph it builds itself. The solve graph decomposes mechanisms that are given, which turns the rule into an acceptance test that every corpus record commits to, and §5.2 measures what construction alone delivers.

Decomposition and rigidity. The irreducible decomposition of pinned planar mechanisms and its relation to rigidity are established (Servatius et al., 2010; Shai et al., 2013), Laman's theorem characterizes generic planar bar-joint rigidity (Laman, 1970), and the pebble game decides it exactly (Jacobs & Hendrickson, 1997; Sljoka et al., 2011). This literature asks structural questions of the modules it identifies, so what a learner gains from being handed them stays open, and it is what we measure. Our structural baseline in §5.3 is Weisfeiler-Lehman hashing of the bare joint graph (Shervashidze et al., 2011). Design systems that pair a language model with a simulator read a symbolic layer describing what the simulation already produced (Gandarela et al., 2026c;b;a), while a decomposition is readable before any simulation runs.

Compositional structure and data efficiency. Exposing parts and their relations is a standard route to reuse and to learning from fewer examples, through relational inductive biases in graph networks (Battaglia et al., 2018), through modules recomposed across tasks (Alet et al., 2018), and, for molecules and programs, through decoders that assemble objects from individually valid pieces (Kusner et al., 2017; Jin et al., 2018). In those settings the parts are designed, mined, or searched for, because the domain does not settle what they are. Planar mechanism theory settles it, so modules, interfaces, and solve order enter the input of a network whose architecture is held fixed, which separates what compositional structure contributes from what a relational architecture contributes. Read on the data axis, constraint-organized features hold a geometric-mean $1.8\times$ sample-efficiency advantage from 50 through 10,000 mechanisms, and the advantage closes by 40,000 (§5.1), so the gain belongs to the regime where every label costs a simulation. Our contribution is this compositional representation at corpus scale, the interface conditions under which its modules support reuse (§4), and a measurement of what the compositionality is worth in training data.

7 Conclusion

Solve graphs change what is explicit without changing the mechanism. They replace a flat view of joints with irreducible modules, typed interfaces, and a dataflow order for local constraint solves, which is what lets a design agent read the corpus as an action set whose moves carry checkable preconditions. Knowledge representation has long held that a formalism is judged by the commitments it makes explicit and by the inferences those commitments make cheap (Brachman & Levesque, 2004), and for planar mechanisms the Assur decomposition supplies commitments that a stored adjacency matrix leaves implicit. Three consequences of that change are measured, with a fourth experiment controlling the third. In this context, composition stays inside a substantially more productive structural search space, with 10.02 million mechanisms admitting a more abstract structural representation, leading to a measurable low-data advantage over a flattened MLP. Structural validity and solve order are addressed before a simulator runs, while motion remains emergent.

REFERENCES

Ferran Alet, Tomás Lozano-Pérez, and Leslie P. Kaelbling. Modular meta-learning. In *Proceedings of the 2nd Conference on Robot Learning*, volume 87 of *Proceedings of Machine Learning Research*, pp. 856–868, 2018. URL https://proceedings.mlr.press/v87/alet18a.html.

Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Caglar Gulcehre, Francis Song, Andrew Ballard, Justin Gilmer, George Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018. URL https://arxiv.org/abs/1806.01261.

Diana Bolanos, Mohammadmehdi Ataei, and Pradeep Kumar Jayaraman. Mechaformer: sequence learning for kinematic mechanism design automation. In *Proceedings of the Fortieth AAAI Conference on Artificial Intelligence and Thirty-Eighth Conference on Innovative Applications of Artificial Intelligence and Sixteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'26/IAAI'26/EAAI'26. AAAI Press, 2026. ISBN 978-1-57735-906-7. doi: 10.1609/aaai.v40i24.39059. URL https://doi.org/10.1609/aaai.v40i24.39059.

Ronald J. Brachman and Hector J. Levesque. *Knowledge Representation and Reasoning*. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann, San Francisco, 2004. ISBN 978-1-55860-932-7. doi: https://doi.org/10.1016/B978-155860932-7/50095-9. URL https://www.sciencedirect.com/science/article/pii/B9781558609327500959.

Maxime Escande and Kristina Shea. Computational circular design of planar linkage mechanisms using available standard parts. *Proceedings of the 6th Product Lifetimes and the Environment Conference (PLATE2025)*, (6), Jun. 2025. doi: 10.54337/plate2025-10309. URL https://journals.aau.dk/index.php/plate2025/article/view/10309.

Mitchell B. Fogelson, Conrad Tucker, and Jonathan Cagan. Gcp-holo: Generating high-order linkage graphs for path synthesis. *Journal of Mechanical Design*, 145(7):073303, 04 2023. ISSN 1050-0472. doi: 10.1115/1.4062147. URL https://doi.org/10.1115/1.4062147.

João Pedro Gandarela, Thiago Rios, Stefan Menzel, and André Freitas. R-APS: Compositional reasoning and in-context meta-learning for constrained design via reflective adversarial pareto search, 2026a. URL https://arxiv.org/abs/2606.04823.

João Pedro Gandarela, Thiago Rios, Stefan Menzel, and André Freitas. Language models refine mechanical linkage designs through symbolic reflection and modular optimisation, 2026b. URL https://arxiv.org/abs/2604.27962.

João Pedro Gandarela, Thiago Rios, Stefan Menzel, and André Freitas. Symbolic intermediaries as a linguistic-numerical interface for LLM-driven geometric reasoning, 2026c. URL https://arxiv.org/abs/2505.17607.

Jack E Graver, Brigitte Servatius, and Herman Servatius. *Combinatorial Rigidity*. Number 2. American Mathematical Society, 1993. ISBN 9780821838013. doi: 10.1090/gsm/002. URL http://dx.doi.org/10.1090/gsm/002.

Xu Han, Ping Zhao, Xiran Zhao, and Bin Zi. Review on machine learning-based approaches for the kinematic analysis and synthesis of mechanisms. *Frontiers of Mechanical Engineering*, 20(2):11, Apr 2025. ISSN 2095-0241. doi: 10.1007/s11465-025-0827-5. URL https://doi.org/10.1007/s11465-025-0827-5.

Amin Heyrani Nobari, Akash Srivastava, Dan Gutfreund, and Faez Ahmed. Links: A dataset of a hundred million planar linkage mechanisms for data-driven kinematic design. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 3A: 48th Design Automation Conference (DAC), pp. V03AT03A013, 08 2022. doi: 10.1115/DETC2022-89798. URL https://doi.org/10.1115/DETC2022-89798.

Donald J. Jacobs and Bruce Hendrickson. An algorithm for two-dimensional rigidity percolation: The pebble game. *Journal of Computational Physics*, 137(2):346–365, 1997. ISSN 0021-9991. doi: https://doi.org/10.1006/jcph.1997.5809. URL https://www.sciencedirect.com/science/article/pii/S0021999197958095.

Yayati Jadhav and Amir Barati Farimani. LinkD: Autoregressive diffusion model for mechanical linkage synthesis. *ArXiv*, abs/2601.04054, 2026. URL https://api.semanticscholar.org/CorpusID:284532652.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2323–2332. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/jin18a.html.

Matt J. Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1945–1954. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/kusner17a.html.

G. Laman. On graphs and rigidity of plane skeletal structures. *Journal of Engineering Mathematics*, 4(4):331–340, Oct 1970. ISSN 1573-2703. doi: 10.1007/BF01534980. URL https://doi.org/10.1007/BF01534980.

Hod Lipson. Evolutionary synthesis of kinematic mechanisms. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, 22(3):195–205, 2008. doi: 10.1017/S0890060408000139.

Andreas Müller. Generic mobility of rigid body mechanisms. *Mechanism and Machine Theory*, 44(6):1240–1255, 2009. ISSN 0094-114X. doi: https://doi.org/10.1016/j.mechmachtheory.2008.08.002. URL https://www.sciencedirect.com/science/article/pii/S0094114X08001602.

Amin Heyrani Nobari, Akash Srivastava, Dan Gutfreund, Kai Xu, and Faez Ahmed. LInk: Learning joint representations of design and performance spaces through contrastive learning for mechanism synthesis. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=a1MRjOL6WJ.

Anar Nurizada, Rohit Dhaipule, Zhijie Lyu, and Anurag Purwar. A dataset of 3m single-dof planar 4-, 6-, and 8-bar linkage mechanisms with open and closed coupler curves for machine learning-driven path synthesis. *Journal of Mechanical Design*, 147(4):041702, 11 2024. ISSN 1050-0472. doi: 10.1115/1.4067014. URL https://doi.org/10.1115/1.4067014.

Chan-Yeong Park and Gang-Won Jang. Automated synthesis of planar linkage mechanisms with diverse joint types via spring-connected link models and contrastive graph learning. *Journal of Computational Design and Engineering*, 13(4):252–272, 04 2026. ISSN 2288-5048. doi: 10.1093/jcde/qwag034. URL https://doi.org/10.1093/jcde/qwag034.

Anurag Purwar and Qiaode Jeffrey Ge. Advancements in kinematic synthesis: Data-driven methods for mechanism design. *Journal of Computing and Information Science in Engineering*, 25(12):120810, 11 2025. ISSN 1530-9827. doi: 10.1115/1.4070204. URL https://doi.org/10.1115/1.4070204.

Brigitte Servatius, Offer Shai, and Walter Whiteley. Combinatorial characterization of the assur graphs from engineering. *European Journal of Combinatorics*, 31(4):1091–1104, 2010. ISSN 0195-6698. doi: https://doi.org/10.1016/j.ejc.2009.11.019. URL https://www.sciencedirect.com/science/article/pii/S0195669809002406. Rigidity and Related Topics in Geometry.

Offer Shai, Adnan Sljoka, and Walter Whiteley. Directed graphs, decompositions, and spatial linkages. *Discrete Applied Mathematics*, 161(18):3028–3047, 2013. ISSN 0166-218X. doi: https://doi.org/10.1016/j.dam.2013.06.004. URL https://www.sciencedirect.com/science/article/pii/S0166218X13002813.

Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn, and Karsten M. Borgwardt. Weisfeiler-lehman graph kernels. *J. Mach. Learn. Res.*, 12(null):2539–2561, November 2011. ISSN 1532-4435.

Adnan Sljoka, Offer Shai, and Walter Whiteley. Checking mobility and decomposition of linkages via pebble game algorithm. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 6: 35th Mechanisms and Robotics Conference, Parts A and B, pp. 493–502, 08 2011. doi: 10.1115/DETC2011-48340. URL https://doi.org/10.1115/DETC2011-48340.

Kaz Vermeer, Reinier Kuppens, and Justus Herder. Kinematic synthesis using reinforcement learning. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 2A: 44th Design Automation Conference, pp. V02AT03A009, 08 2018. doi: 10.1115/DETC2018-85529. URL https://doi.org/10.1115/DETC2018-85529.

A Generative AI Use Disclosure

In this work, we used generative AI tools to draft parts of the manuscript, to edit the text for readability, and to create and edit software code, and to assist with drafting informal proof sketches under manual supervision. The authors reviewed and revised all such material.

B Background: Full Tutorial Treatment

This appendix preserves the tutorial-style derivation of the kinematic and graph-theoretic background that motivates the Assur knowledge base. The main body (§2) presents only the condensed version.

B.1 Mechanisms as constraint systems

A planar rigid body has three *degrees of freedom* (DOF): two translational (x, y) and one rotational (θ). When two bodies are connected by a *revolute joint* (a pin), two relative DOF are removed (the joint fixes relative x and y but leaves θ free).

This maps cleanly onto a constraint-satisfaction problem. Given a mechanism with L links (including the fixed *ground* link) and J revolute joints, the configuration space has dimension

$$\boxed{\mathcal{F} = 3(L-1) - 2J} \tag{5}$$

This is the planar Chebychev–Grübler count (Müller, 2009). Under the regularity and independence assumptions of the generic model, $\mathcal{F} = 1$ means that one generalized coordinate (e.g. the input crank angle) locally drives all others. The count is structural: by itself it neither certifies that a particular geometry assembles nor that an assembled mechanism completes a full input cycle.

Mechanism	L	J	$\mathcal{F}$	Notes
4-bar linkage	4	4	1	standard crank–rocker
Slider-crank	4	4	1	piston–crank
Pantograph (6-bar)	6	7	1	copying mechanism
Rigid triangle	3	3	0	structure, not mechanism

B.2 The 4-bar linkage

Figure 6 shows the canonical planar mechanism. Links are named by their function: a *crank* (driver) rotates continuously; the *coupler* transmits motion; the *rocker* oscillates.

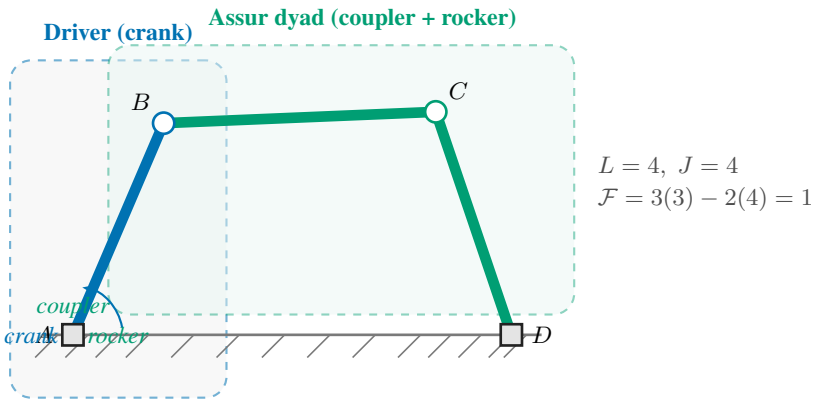


Figure 6: A planar 4-bar linkage decomposed into its driver link and an Assur dyad. Fixed pivots A and D are grounded. The dyad's two *external* joints are B (position known from the driver) and D (fixed); its one *internal* joint is C.

B.3 Generic mobility and minimal pinned components

For a pinned subassembly of n moving links and j lower-pair constraints, the equality

$$3n = 2j. \tag{6}$$

is the zero-mobility *count*. It is necessary for generic isostaticity, but equality alone is not a rigidity certificate: the corresponding sparsity/minimality conditions must also hold (Graver et al., 1993). We therefore keep the generic count, the decomposition result, assembly at the stored geometry, and full-cycle motion as four distinct checks.

For the 4-bar as a whole: $3 \times 3 = 9 \neq 8 = 2 \times 4$, so the full mechanism is *not* isostatic: it has one DOF, as required. But consider only the dyad sub-chain (coupler + rocker):

$$n = 2 \text{ links}, \quad j = 3 \text{ joints (one internal + two external)} \implies 3(2) = 6 = 2(3). \checkmark$$

B.4 The pebble game

Equation 6 totals freedoms and constraints over a whole subassembly, so it remains satisfied by a subassembly that spends surplus constraints in one region while leaving another region undercon-strained. Deciding generic rigidity therefore means verifying a count on every subset of joints, and the *pebble game* of Jacobs & Hendrickson (1997) performs all of those checks in one incremental pass. We describe it from scratch here, since the acceptance test of §3.2 is stated in its terms.

Each joint of a planar bar–joint framework receives two pebbles, one for each coordinate it could move along. A pebble resting on a joint is a freedom still available there, and a pebble moved onto a bar is a freedom that bar has consumed. Bars are inserted one at a time. A bar between joints u and v may enter only when four pebbles can be gathered on u and v, two at each endpoint, because a bar in the plane removes two freedoms and the game requires the two remaining ones to be visible. When an endpoint is short, the game follows the bars already inserted outward in search of a free pebble at some other joint and slides it back along the path on which it was found. This step, path reversal, re-charges which bar is paid for by which freedom and alters no total. If four pebbles are gathered, the new bar claims one of them and counts as independent. If the search exhausts everything reachable from u and v, every freedom in that region is already spent, so the new bar is implied by the bars already present and counts as redundant.

(In essence, the game is degree-of-freedom bookkeeping on the graph. Two tokens per joint, one token spent by each independent bar, and a failed search for a token proves that the region reached is already rigid.)

The pebbles still free when insertion finishes are the freedoms no bar removed, which is the framework's generic degree-of-freedom count. With every joint free to move, three pebbles remain exactly when the only motions left are the three rigid motions of the plane, and Laman's theorem identifies that state with the counting condition holding on every subset of joints (Laman, 1970). The name $(2, 3)$ records those two numbers, two pebbles per joint and the three a rigid framework keeps. In the pinned form the acceptance test uses, the boundary joints enter first as a complete graph, so the plane's three freedoms stay visible in the same tally, a module counts as pinned-rigid when exactly three pebbles are free, and any surplus over three is its residual freedoms. On the dyad of Figure 7 the three joints supply six pebbles, the boundary edge p_1p_2 and the two links p_1q and qp_2 each enter as independent, and three pebbles are left over, so the dyad is pinned-rigid. The cost stays close to linear in the number of bars at these sizes, which is what makes the test affordable on every module of the build, and it is the algorithm the rigidity-theoretic treatment of Assur decomposition runs as well (Sljoka et al., 2011).

B.5 Assur groups: definition

> An **Assur group** is a minimal pinned isostatic component: it is generically rigid once its interface joints are pinned, and no proper admissible subcomponent has that property (Servatius et al., 2010).

The simplest non-trivial Assur group is the **dyad**: two links sharing one internal revolute joint, with two external joints that attach to the rest of the mechanism (Figure 7).

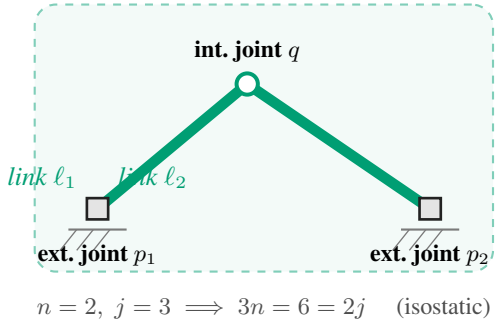


Figure 7: A dyad with interface arity two (and classical order two under the one-pair-per-boundary convention). Two links, three joints (one internal, two external). Given p_1 and p_2, the position of q is found by intersecting two circles, typically yielding two assembly modes.

Stored interface arity	Links n	Joints j	Int. joints	Boundary IDs
2	2	3	1	2
3	4	6	3	3
4	6	9	5	4

Table 1: Illustrative interface arities stored by this work. The value counts distinct boundary-joint identifiers and agrees with classical Assur order only under a one-to-one boundary-pair convention. It is not classical Assur class, which is recorded only with a named convention and independent validation. The equality $3n = 2j$ is a generic count, not an independence, assembly, or full-cycle certificate.

Once a mechanism has been peeled into groups (§3.1), contracting each group to a node and drawing an edge wherever one group supplies a boundary joint to another gives the solve graph of Figure 8, whose topological traversals are the dependency-respecting solve orders.

The mobility equation above uses the body–joint view (rigid links as bodies, revolute joints as lower pairs). The binary-link LINKS source admits an equivalent pinned bar–joint view (joint coordinates as vertices and binary links as distance constraints); there the necessary component count is $|E| = 2|I|$

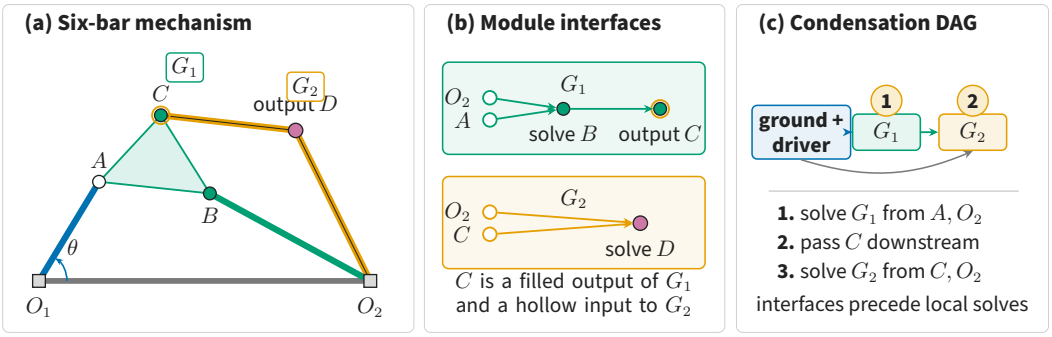
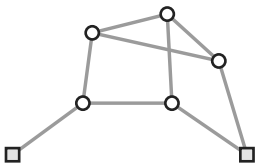


Figure 8: **A solve graph localizes the coupled computation in a multi-group mechanism.** Hollow joints are inputs and filled joints are outputs, and ground meets two moving links at the compound pin O_2. Numbers give one valid topological solve order. This is the six-bar of Figure 3b carried through to its condensation DAG.

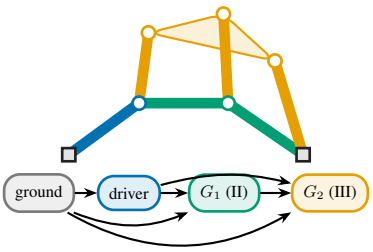
for internal vertices I. We do not substitute that bar–joint count for the link-incidence model used by higher-arity or prismatic sources.

B.6 A HIGHER-CLASS EXAMPLE

Figure 9 repeats the change of representation of Figure 3 outside the dyadic regime, on a Stephenson-style mechanism whose second group has three mutually coupled internal joints. What the group makes visible that adjacency hides is the *location* of the coupling. The rest of the mechanism factors into one small local solve per group, the driver and G_1 each fixing a single joint, and only the three internals of G_2 are coupled to each other. Adjacency alone gives a flat set of joints and bars: it neither names the dyad nor isolates the three joints whose coordinates must be solved together.



(a) Adjacency view: a flat set of joints and bars; the dyad and the class-III group are not visible, and the upper triangle is indistinguishable from three braced bars or one ternary rigid body.



(b) Assur view: driver → dyad G_1 (class II) → class-III group G_2. Solving G_1 writes one joint (circle intersection); solving G_2 writes three joints *coupled to each other*, surfacing exactly where the local solve stops factoring.

Figure 9: The change of representation in the higher-class regime, on a Stephenson-style mechanism. **(a)** The adjacency view: seven joints and a set of bars; no group structure, and the upper triangle cannot be told apart from one ternary rigid body. **(b)** The Assur view: the driver determines the joint above j_0, the class-II dyad G_1 writes the joint between the two ground pivots, and the class-III group G_2 (amber rigid body, three coupled internal joints) writes the upper three joints at once. The condensation DAG (below right) makes the cascade and the location of the coupling explicit: the rest of the mechanism factors into one small local solve per group, and only the three internals of G_2 are coupled to each other.

Table 2: Experiment 1 validity rates from 50,000 samples per procedure per size. Procedure A is the compositional dyad sampler; procedure B is the matched adjacency sampler.

Branch	N	A overall	B overall	A geom.	B geom.	A conn	B conn	A path	B path	A dof	B dof
theoretical	4	1.000	0.405	1.000	0.405	1.000	0.990	1.000	0.405	1.000	0.411
theoretical	5	1.000	0.204	1.000	0.204	1.000	0.974	1.000	0.204	1.000	0.277
theoretical	6	1.000	0.108	1.000	0.108	1.000	0.958	1.000	0.108	1.000	0.215
theoretical	7	1.000	0.061	1.000	0.061	1.000	0.940	1.000	0.061	1.000	0.178
theoretical	8	1.000	0.035	1.000	0.035	1.000	0.924	1.000	0.035	1.000	0.155
empirical	8	0.284	0.001	0.284	0.017	1.000	0.782	1.000	0.050	1.000	0.124
empirical	9	0.223	0.001	0.223	0.008	1.000	0.758	1.000	0.029	1.000	0.123
empirical	10	0.180	0.001	0.180	0.004	1.000	0.745	1.000	0.019	1.000	0.117
empirical	11	0.146	0.000	0.146	0.002	1.000	0.729	1.000	0.012	1.000	0.110
empirical	12	0.121	0.000	0.121	0.001	1.000	0.711	1.000	0.007	1.000	0.107

B.7 The LINKS dataset in detail

The published LINKS dataset reports one hundred million planar linkage mechanisms, with a reduced release of ten million mechanisms (Heyrani Nobari et al., 2022). These are upstream dataset sizes, not a claim that this work transformed every record. Each mechanism is a triple (A, x_0, n):

- $A \in \{0,1\}^{N\times N}$ is the symmetric adjacency matrix of the joint graph.
- $x_0 \in \mathbb{R}^{N\times 2}$ holds the initial planar positions of all N joints.
- $n \in \{0,1\}^{N}$ flags each joint as moving (0) or grounded (1).

The motor is fixed by convention: joints 0 and 1 are connected, joint 0 is always grounded, and the link $0{-}1$ acts as the driver crank.

B.8 The Purwar Lab dataset in detail

The published Purwar Lab dataset reports approximately three million single-DOF planar four-, six-, and eight-bar linkage mechanisms with open and closed coupler curves (Nurizada et al., 2024). The build streams 2,966,754 of them, divided into 602,079 four-bar, 397,525 six-bar, and 1,967,150 eight-bar mechanisms (Appendix E, which also gives the release link). Each mechanism is given as:

- $B \in \{0,1\}^{L\times N}$ is the link-incidence matrix, one row per rigid link over N joints, with row 0 the ground link, so a ternary or quaternary body is a single row and never a ring of bars.
- $\tau : J \to \{\mathrm{R}, \mathrm{P}\}$ marks each joint revolute or prismatic, read off the named topology (`RRRR`, `RRRP`, `RRPR`, `PRPR`), which is what makes the slider families reachable at all.
- one coupler curve per record, travelling inline with the record and traced by a point carried on a link.

The motor is fixed by the source topology table, which supplies the grounded joints on incidence row 0 together with the driver pair. No joint-coordinate array is released, so the pose x_0 is decoded from the source file name and the resulting design frame is accepted only once the traced point maps through it onto the stored curve (§3.4).

C Full Benchmark Tables

These tables reproduce the completed experiment summaries used in §5.

C.1 Compositional validity

Section 5.2 reports the decline in the empirical branch and leaves its mechanism here. Extra stationary pivots can make the distance between two boundary joints vary, over a revolution, beyond what a randomly sized dyad can close. The attachment stays structurally admissible throughout, because admissibility reads only the groups and their dependencies, and it is the sampled lengths that fail. This is why the decline is a property of how geometry is drawn, which leaves the construction rule intact, and why the oracle records its geometric condition separately from its three structural ones.

Table 3: Experiment 2 hash and fingerprint summary. Adjacency hashes use the bare joint graph; solve-graph fingerprints use group-order counts and the order-labelled condensation DAG.

Quantity	Value
Mechanisms hashed	10,020,000
Distinct adjacency hashes D_A	1,173,078
Distinct Assur fingerprints D_F	908,186
Coarsening factor D_A/D_F	1.29
Mean mechanisms per Assur fingerprint	11.03
Largest Assur-fingerprint bucket	330,809
Max adjacency hashes in one Assur fingerprint	300

Table 4: Experiment 3 learning curve for the 128-point output-joint coupler curve. Scores are mean symmetric Chamfer distances $\pm$ standard deviations over 10 stratified fold units. The p column is the one-sided Nadeau–Bengio-corrected paired-test probability for "Assur is better."

Train n	Adj. Chamfer	Assur Chamfer	Adj. $-$ Assur	NB p
50	0.1822 ± 0.0290	0.1624 ± 0.0063	0.0197	0.4802
100	0.1602 ± 0.0038	0.1505 ± 0.0045	0.0097	0.4251
200	0.1473 ± 0.0022	0.1396 ± 0.0027	0.0077	0.3622
500	0.1359 ± 0.0022	0.1276 ± 0.0014	0.0083	0.2607
1,000	0.1274 ± 0.0014	0.1203 ± 0.0014	0.0070	0.1587
2,000	0.1196 ± 0.0008	0.1140 ± 0.0012	0.0056	0.0078
4,000	0.1131 ± 0.0008	0.1085 ± 0.0007	0.0047	0.0022
8,000	0.1069 ± 0.0008	0.1031 ± 0.0007	0.0038	7.4×10^{-5}
10,000	0.1051 ± 0.0008	0.1019 ± 0.0007	0.0033	0.0002
20,000	0.1002 ± 0.0008	0.0988 ± 0.0007	0.0014	0.0035
40,000	0.0964 ± 0.0005	0.0967 ± 0.0006	-0.0004	0.9321
72,013	0.0942 ± 0.0005	0.0957 ± 0.0004	-0.0014	1.0000

C.2 Structural coarsening

C.3 Curve-regression learning curve

The p column tests "Assur is better," so it reads near 1 on the last two rows and separates them only under the reverse test. At 40,000 examples the -0.0004 difference sits inside fold noise, with a corrected paired interval of $[-0.0009, +0.0001]$ and a one-sided probability of 0.068 for adjacency-better. At the full pool the -0.0014 difference is resolved, at 6.6×10^{-6} with all ten folds favoring adjacency.

C.4 The default feature vector

Section 5.1 and the ablation below both speak of a *default* vector, which is the eight blocks named in Table 5, 86 numbers in all, each of them independent of pose, of absolute scale, and of the joint numbering. Group-order counts (`order_count`, 3) record how many groups of each order the decomposition returned. Condensation-DAG shape (`dag_shape`, 6) records its depth, its group and fixed-joint counts, and the three leading singular values of the group-by-group adjacency matrix. Link lengths (`lengths`, 37) are the complete set in solve order, divided by the crank length, and assembly signs (`branches`, 20) carry one branch per dyad in that same order, which together pin the configuration that lengths alone leave ambiguous. The tracer dyad contributes its own triangle (`coupler_tri`, 5) and the tracer's angle, area and perpendicular offset on the coupler body (`coupler_shape`, 3). Path geometry (`path_geom`, 9) holds the per-stage lengths along the chain from driver to coupler with that chain's total reach, and shape ratios (`shape_ratios`, 3) hold dimensionless link proportions. The structural sub-block of §5.4 is the part that reads off the decomposition with no geometry at all, the order counts and the DAG-shape block together with a gated DAG-branching block, 13 numbers.

Comparison	Block	Gain	Std.
Default leave-one-out	`order_count`	0.00032	0.00074
Default leave-one-out	`dag_shape`	0.00219	0.00057
Default leave-one-out	`lengths`	0.00022	0.00049
Default leave-one-out	`branches`	-0.00209	0.00061
Default leave-one-out	`coupler_tri`	-0.00016	0.00074
Default leave-one-out	`coupler_shape`	0.00017	0.00063
Default leave-one-out	`path_geom`	-0.00039	0.00048
Default leave-one-out	`shape_ratios`	-0.00048	0.00067
Candidate add	`grashof`	0.00397	0.00065
Candidate add	`coupler_param`	-0.00036	0.00049
Candidate add	`transmission`	0.00295	0.00089
Candidate add	`reach_extent`	-0.00006	0.00056
Candidate add	`loop_spectral`	0.00023	0.00069
Candidate add	`interactions`	0.00106	0.00066

Table 5: Experiment 3 feature-ablation summary at train $n = 5{,}000$ over 10 fold units. Default rows report leave-one-out importance; candidate rows report add-one-block gain. Positive values mean lower Chamfer relative to the corresponding baseline.

N	Trials	Fingerprint	Assur	Assur struct.	Adjacency
4	760	1.000	1.000	1.000	0.050
5	580	1.000	1.000	1.000	0.005
6	620	1.000	1.000	1.000	0.000
7	620	1.000	1.000	1.000	0.000
8	640	1.000	1.000	1.000	0.000
9	620	1.000	1.000	1.000	0.000
10	680	1.000	1.000	1.000	0.000
11	800	1.000	1.000	1.000	0.000
12	680	1.000	1.000	1.000	0.000
All	6,000	1.000	1.000	1.000	0.007

Table 6: Experiment 4 relabelling invariance: the fraction of trials left unchanged by a random joint permutation. 300 generated mechanisms, 20 permutations each. The adjacency column is the Experiment 3 flattened adjacency-plus-position baseline, which is keyed by joint index and is expected to change.

C.5 Assur-feature ablation

Section 5.1 states the outcome and leaves the magnitudes here. The condensation-DAG block is the only clear contributor at $+0.00219 \pm 0.00057$ Chamfer, and the branch block is the only one whose removal helps, lowering error by 0.00209 ± 0.00061. The remaining six default blocks sit inside fold noise, and the structural sub-block on its own scores 0.10375 against 0.10576 for the full default vector, so the vector as shipped is carried by its condensation-DAG component. Among blocks outside the default, a Grashof indicator gains 0.00397 and a transmission-angle summary 0.00295, both larger than anything currently included. Each describes motion feasibility, which the solve graph leaves for simulation to settle, so neither enters the default vector on this evidence alone.

C.6 Relabelling invariance

Mechanisms are generated by the Experiment 1 compositional sampler, decomposed once, then re-decomposed under a random permutation of all joint indices with the motor pair and coupler joint remapped through the same permutation. Rates are the fraction of trials whose descriptor is bitwise unchanged.

D Conditions and Boundaries for Predictive Composition

The central question of Section 4 is what makes a familiar constraint module reusable in a new mechanism. A dyad illustrates the idea: its internal joint is found by intersecting two circles. The same conditional rule applies in many assemblies, while the boundary positions, bar lengths, and assembly branch specify each instance. Composition reuses that rule and passes its solution to later modules.

We first ask when accurate local predictions remain accurate after this composition. The transfer bound separates familiarity of local inputs from the amplification of local errors. Two physical constructions then explain why amplification and loss of smoothness can occur even in chains of simple dyads. Next, we ask what information must reach each module: a port-assignment example and a dimension bound show how compression can merge different behaviours. Finally, we distinguish the information retained by a lossless decomposition from the computational structure it exposes, and show why undirected graph width does not capture that structure.

The proofs use local implicit solves and the established Assur decomposition (Servatius et al., 2010; Shai et al., 2013); the transfer argument builds on the broader study of compositional prediction (**???**).

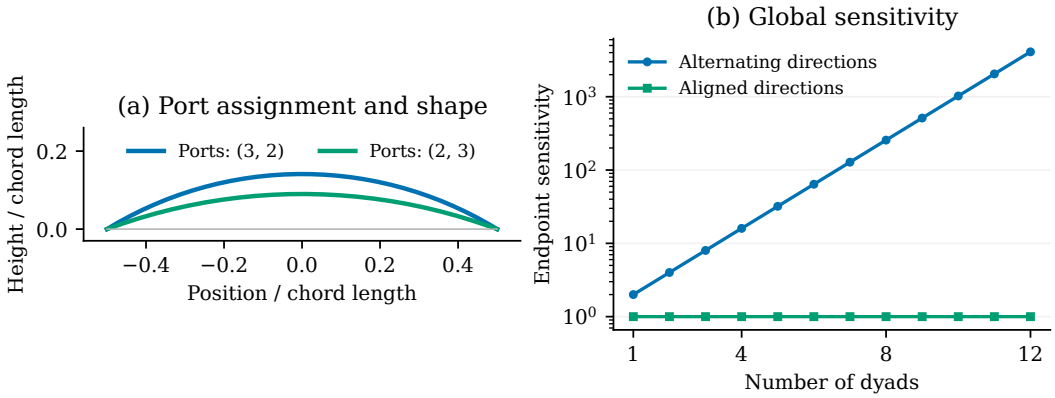


Figure 10: **What local structure alone leaves unresolved.** (a) Exchanging lengths between the boundary ports of a four-bar dyad changes its output arc, shown after normalizing both chords to unit length (Proposition 9). (b) Dyad chains with identical local Jacobian singular values have endpoint configuration sensitivities 1 and 2^m at their reference poses (Theorem 7). Panel (a) shows why a module needs the correspondence between constraints and boundary ports; panel (b) shows why the directions in which modules transmit motion matter. Both are analytic constructions. The included script reproduces the figure and checks the constructions against circle-intersection solves.

D.1 Generalization Through Module Interfaces

An assembly may be new even though each conditional problem it contains is familiar. For example, a dyad predictor can be reused wherever its boundary motion, geometry, and assembly branch resemble those seen during training. The difficulty is that an upstream prediction also supplies the boundary conditions for later modules. We therefore separate two sources of error: the error a module makes with the correct inputs, and the additional error caused by imperfect inputs arriving from its parents. The first is governed by local coverage; the second by propagation through the solve graph.

D.1.1 Spaces, Domains, and Error Propagation

The setup below formalizes this separation. The input and output spaces allow a module to predict an entire path, so that the same argument applies to mechanism motion as well as to a single configuration.

Fix a finite solve DAG with nodes $V = \{1, \dots, m\}$ in topological order. Write $\mathrm{pa}(v)$ for the parents of v, and $t(v)$ for its module type. For each type t, let X_t be a separable Banach space of module outputs and (Z_t, d_t) a Polish space of module inputs. For trajectory prediction one may take

$$\mathsf{X}_t = C([0,1]; \mathbb{R}^{d_t^{\mathrm{out}}}), \qquad \|x\|_{\mathsf{X}_t} = \sup_{\theta \in [0,1]} \|x(\theta)\|_2.$$

Here θ is a common driving parameter, not an independently resampled arc-length coordinate. The input space includes the full boundary paths, geometric parameters, and branch information. The input metric is fixed before stating a transport bound; any rescaling of its coordinates changes the associated Lipschitz constants.

For each type, a measurable exact conditional solver $S_t : D_t \to \mathsf{X}_t$ is defined on an admissible Borel set $D_t \subseteq \mathsf{Z}_t$. Its measurable learned counterpart $\widehat{S}_t : U_t \to \mathsf{X}_t$ is defined on a Borel set $U_t \supseteq D_t$. Branch-conditioned solver charts may be included in the type index. This avoids assuming a single-valued exact solver across incompatible assembly branches.

Let ω collect the random mechanism geometry, source motion, and other exogenous data, on a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. The measurable routing map Q_v supplies the inputs to occurrence v:

$$z_v = Q_v\big(\eta_v, (x_u)_{u \in \mathrm{pa}(v)}\big), \qquad x_v = S_{t(v)}(z_v), \tag{7}$$

$$\widehat{z}_v = Q_v\big(\eta_v, (\widehat{x}_u)_{u \in \mathrm{pa}(v)}\big), \qquad \widehat{x}_v = \widehat{S}_{t(v)}(\widehat{z}_v). \tag{8}$$

All quantities depend on ω. The same observed exogenous input η_v is used in the two recursions; uncertainty in that input is not included in the bounds below.

We require $z_v \in D_{t(v)}$ almost surely. To compare the two recursions, also consider every hybrid input obtained by independently choosing either x_u or $\widehat{x}_u$ for each parent. Every such input must lie in $U_{t(v)}$. The exact solver need not be defined at these hybrid inputs. For two hybrid tuples differing only in parent u, assume

$$\Big\| \widehat{S}_{t(v)}\big(Q_v(\eta_v, \dots, a_u, \dots)\big) - \widehat{S}_{t(v)}\big(Q_v(\eta_v, \dots, b_u, \dots)\big) \Big\|_{\mathsf{X}_{t(v)}} \le A_{vu} \|a_u - b_u\|_{\mathsf{X}_{t(u)}}. \tag{9}$$

The constants are deterministic and uniform over the stated evaluation population. Set $A_{vu} = 0$ when $u \notin \mathrm{pa}(v)$. Thus A is a nonnegative strictly lower-triangular matrix. Bounds on rectangular domains of admissible parent states imply Eq. (9), but are stronger than needed.

The hybrid inputs let us replace predicted parent outputs one at a time. This attributes each incoming error to a specific dependency, with A_{vu} bounding its effect. They are comparison points for the learned predictor; requiring them to be physically feasible exact assemblies is unnecessary.

Write $x = (x_1, \dots, x_m)$ and define $\widehat{x}$ analogously. Let a measurable readout produce $y = R(\eta, x)$ and $\widehat{y} = R(\eta, \widehat{x})$ in a separable Banach space Y. Assume

$$\|\widehat{y} - y\|_{\mathsf{Y}} \le \sum_{v=1}^{m} c_v \|\widehat{x}_v - x_v\|_{\mathsf{X}_{t(v)}}, \qquad c_v \ge 0. \tag{10}$$

Selecting a tracer coordinate is one such readout. Define

$$w^\top = c^\top (I - A)^{-1} = c^\top \sum_{j=0}^{m-1} A^j. \tag{11}$$

Each w_v sums products of edge gains over directed paths from v to the readout, including the direct readout contribution c_v.

For intuition, an error passing through two successive dependencies with gains a and b contributes at most ab times its original magnitude. When several paths reach the output, their contributions add in the bound. The weight w_v collects this downstream influence for occurrence v.

Proposition 1 (Propagation from errors at true module inputs)**.** Under the domain and gain assumptions above, let

$$\delta_t(z) = \|\widehat{S}_t(z) - S_t(z)\|_{\mathsf{X}_t}, \qquad z \in D_t.$$

Then, almost surely,

$$\|\widehat{y} - y\|_{\mathsf{Y}} \le \sum_{v=1}^{m} w_v \delta_{t(v)}(z_v). \tag{12}$$

Consequently, if ν_v is the law of the true input z_v,

$$\left(\mathbb{E}\|\widehat{y} - y\|_{\mathsf{Y}}^2\right)^{1/2} \le \sum_{v=1}^{m} w_v \|\delta_{t(v)}\|_{L^2(\nu_v)}. \tag{13}$$

Proof. Put $e_v = \|\widehat{x}_v - x_v\|_{\mathsf{X}_{t(v)}}$ and $b_v = \delta_{t(v)}(z_v)$. Insert $\widehat{S}_{t(v)}(z_v)$ between the true and predicted outputs. Changing one parent at a time from its true to its predicted output uses only the stipulated hybrid inputs, so the triangle inequality and Eq. (9) give

$$e_v \le b_v + \sum_{u \in \mathrm{pa}(v)} A_{vu} e_u.$$

In vector form $e \le b + Ae$, componentwise. Iteration terminates because $A^m = 0$, giving $e \le \sum_{j=0}^{m-1} A^j b$. Multiplication by the nonnegative readout vector c proves Eq. (12). Taking scalar $L^2(\mathbb{P})$ norms and using Minkowski's inequality proves Eq. (13). No independence between errors at different occurrences is required. □

D.1.2 Transport from training interfaces to new assemblies

The propagation result assumes we know a module's error on the inputs of the new assembly. Training instead characterizes its error under a reference distribution. The next step connects the two: if nearby local inputs have similar prediction errors, shifting the input distribution incurs a controlled additional cost. The Wasserstein distance W_2 expresses the smallest RMS displacement needed to match inputs from the two distributions.

Let μ_t be a reference training distribution of complete local inputs for type t, supported on D_t. The quantity

$$\epsilon_t = \|\delta_t\|_{L^2(\mu_t)} \tag{14}$$

is a population error of the learned module, not its empirical training error. A separate statistical bound may control it from a finite sample. The following statements can also be read conditionally on fitted module predictors and any event on which their local error bounds hold.

Theorem 2 (Generalization controlled by local interface transport)**.** Assume the conditions of Proposition 1. Suppose μ_t and ν_v have finite second moments in their respective input metrics and, on a set supporting both measures, the scalar residual δ_t is K_t-Lipschitz. Then

$$\left(\mathbb{E}\|\widehat{y} - y\|_{\mathsf{Y}}^2\right)^{1/2} \le \sum_{v=1}^{m} w_v \left[\epsilon_{t(v)} + K_{t(v)} W_2(\nu_v, \mu_{t(v)})\right], \tag{15}$$

where

$$W_2(\nu, \mu) = \inf_{\pi \in \Pi(\nu,\mu)} \left(\int d_t(z, z')^2 \, \mathrm{d}\pi(z, z') \right)^{1/2}.$$

It suffices for the exact and learned module maps to be Lipschitz on that set with constants L_t and $\widehat{L}_t$, respectively; one may then take $K_t = L_t + \widehat{L}_t$.

Proof. Fix an occurrence v and write $t = t(v)$. For any coupling $(Z, Z') \sim \pi \in \Pi(\nu_v, \mu_t)$, the Lipschitz condition gives $\delta_t(Z) \le \delta_t(Z') + K_t d_t(Z, Z')$ almost surely. Minkowski's inequality therefore yields

$$\|\delta_t\|_{L^2(\nu_v)} \le \epsilon_t + K_t \left(\int d_t(z, z')^2 \, \mathrm{d}\pi(z, z') \right)^{1/2}.$$

Taking the infimum over couplings proves the local transport bound; existence of an optimal coupling is not needed. Substitute these bounds in Eq. (13). Finally, the reverse triangle inequality gives

$$|\delta_t(z) - \delta_t(z')| \le \|\widehat{S}_t(z) - \widehat{S}_t(z')\|_{\mathsf{X}_t} + \|S_t(z) - S_t(z')\|_{\mathsf{X}_t},$$

which proves the stated sufficient condition for K_t. □

An alternative is useful when the new assembly encounters the same local regimes with different frequencies. A bounded density ratio limits how much more often any regime can occur, giving the following transfer bound.

Corollary 3 (Density-ratio transfer)**.** Under Proposition 1, suppose instead that $\nu_v \ll \mu_{t(v)}$ and $\mathrm{d}\nu_v/\mathrm{d}\mu_{t(v)} \le C_v < \infty$ almost everywhere. Then

$$\left(\mathbb{E}\|\widehat{y} - y\|_{\mathsf{Y}}^2\right)^{1/2} \le \sum_{v=1}^{m} w_v \sqrt{C_v}\, \epsilon_{t(v)}. \tag{16}$$

No residual Lipschitz assumption is required for this corollary.

Proof. For $t = t(v)$,

$$\int \delta_t^2 \,\mathrm{d}\nu_v = \int \delta_t^2 \frac{\mathrm{d}\nu_v}{\mathrm{d}\mu_t}\,\mathrm{d}\mu_t \le C_v \epsilon_t^2.$$

Take square roots and apply Eq. (13). □

These bounds permit new dependency graphs. More precisely, if the same module bank is used on a family of graphs and the right-hand side of Eq. (15) is at most B for every graph, the RMS bound B holds uniformly on that family. If a random graph has conditional bound B_G, its unconditional RMS error is at most $(\mathbb{E}_G B_G^2)^{1/2}$, by the law of total expectation. The number of possible graphs does not appear separately; their effects enter through local input laws and path weights.

Physical units and normalized module charts. Reuse can also exploit a shared coordinate system. Translating, rotating, and scaling a dyad into a common frame lets one predictor handle many placements. To propagate its error through the physical mechanism, we must then account for the conversion back to world coordinates. For a dyad, let $d(\theta) = \|p_2(\theta) - p_1(\theta)\| > 0$, and let $R(\theta)$ be the rotation whose first column is $(p_2(\theta) - p_1(\theta))/d(\theta)$. A normalized solver s_t receives the ratio paths $\rho_i = \ell_i/d$ and the branch. The physical solver is

$$S_t(z)(\theta) = p_1(\theta) + d(\theta) R(\theta) s_t(\chi(z))(\theta).$$

Its learned counterpart uses the same reconstruction formula with $\widehat{s}_t$. At a true input, if $\sup_\theta d(\theta) \le d_{\max}$, then

$$\delta_t(z) \le d_{\max} \|\widehat{s}_t(\chi(z)) - s_t(\chi(z))\|_\infty. \tag{17}$$

This follows pointwise from orthogonality of R and then by taking a supremum. Thus the local term in Eq. (13) can alternatively be bounded by

$$d_{\max}\left[\overline{\epsilon}_t + \overline{K}_t W_2(\chi_\# \nu_v, \overline{\mu}_t)\right],$$

where $\overline{\mu}_t$ is the training law of normalized inputs, $\overline{\epsilon}_t$ is the normalized local RMS error, and $\overline{K}_t$ bounds the normalized scalar residual's Lipschitz constant. The proof is Eq. (17) followed by the coupling argument in Theorem 2. Crucially, the edge gains A_{vu} must still bound the complete physical predictor: the chart, reference point, scale, and rotation also change when parent predictions change. For a C^1 path norm the reconstruction derivatives must additionally be included; $d_{\max}$ alone is then insufficient.

D.1.3 Uniform trajectory error and normalized point clouds

The learning experiment observes normalized curve samples, whereas the composition bound first controls a trajectory at common driving phases. The following results connect these descriptions in two steps: control the effect of centering and scaling, then the effect of sampling each curve by its own arc length. A lower bound on curve scale prevents normalization from magnifying arbitrarily small positional errors.

Corollary 4 (Uniform trajectory error)**.** Under Proposition 1, suppose $\delta_t(z) \le e_t$ for every admissible true input in the evaluation population. Then

$$\|\widehat{y} - y\|_{\mathsf{Y}} \le B := \sum_v w_v e_{t(v)} \tag{18}$$

almost surely. In particular, for $\mathsf{Y} = C([0,1];\mathbb{R}^2)$ with the supremum norm this bounds the error at every common driving phase.

Proof. Substitute the uniform local bounds in Eq. (12). □

For an ordered cloud $P = (p_1, \dots, p_N)$ in $\mathbb{R}^2$, write

$$\|P\|_N^2 = \frac{1}{N}\sum_{i=1}^{N} \|p_i\|_2^2, \qquad CP = (p_i - \overline{p})_{i=1}^N, \qquad \sigma(P) = \|CP\|_N,$$

and define the centered, RMS-normalized cloud $\mathcal{N}(P) = CP/\sigma(P)$ when $\sigma(P) > 0$. We use the following explicit symmetric squared Chamfer convention:

$$\mathrm{Ch}_2(P, Q) = \frac{1}{2N}\left(\sum_{i=1}^{N} \min_j \|p_i - q_j\|_2^2 + \sum_{j=1}^{N} \min_i \|q_j - p_i\|_2^2\right). \tag{19}$$

For the convention that sums the two directional means, the following Chamfer upper bounds are multiplied by two.

Proposition 5 (Stable normalization and Chamfer comparison)**.** If $\sigma(P), \sigma(Q) \geq s_0 > 0$, then

$$\mathrm{Ch}_2\big(\mathcal{N}(P), \mathcal{N}(Q)\big) \leq \|\mathcal{N}(P) - \mathcal{N}(Q)\|_N^2 \leq \frac{\|P - Q\|_N^2}{s_0^2}. \tag{20}$$

Consequently, sampling true and predicted trajectories at the same N phases, under Corollary 4 and these scale floors, gives a normalized Chamfer bound B^2/s_0^2. If only $\sigma(P) \geq s_0$ is known and $\|P - Q\|_N \leq B < s_0$, then Q is also nonconstant and the bound $4B^2/s_0^2$ holds instead.

Proof. The centering map C is an orthogonal projection for the inner product $\langle P, Q\rangle_N = N^{-1}\sum_i p_i^\top q_i$, and hence $\|C(P - Q)\|_N \leq \|P - Q\|_N$. For nonzero vectors a, b in any real Hilbert space, put $r = \|a\|$ and $s = \|b\|$. Direct expansion gives

$$\|a - b\|^2 = (r - s)^2 + rs\left\|\frac{a}{r} - \frac{b}{s}\right\|^2.$$

Apply this identity with $a = CP$, $b = CQ$ and $r, s \geq s_0$ to obtain the second inequality in Eq. (20). In each directed Chamfer term the point with the same index is an available match, proving the first inequality. Shared-phase sampling satisfies $\|P - Q\|_N \leq \|y - \widehat{y}\|_\infty \leq B$.

For the final assertion, the reverse triangle inequality gives $\sigma(Q) \geq \sigma(P) - \|C(P - Q)\|_N \geq s_0 - B > 0$. For arbitrary nonzero a, b,

$$\left\|\frac{a}{\|a\|} - \frac{b}{\|b\|}\right\| \leq \frac{\|a - b\| + |\|a\| - \|b\||}{\|a\|} \leq \frac{2\|a - b\|}{\|a\|}.$$

Use $a = CP$, the true scale floor, and the preceding Chamfer comparison. □

The same proof gives an expected normalized Chamfer bound B_{rms}^2/s_0^2 from any supremum-path RMS bound $\big(\mathbb{E}\|y - \widehat{y}\|_\infty^2\big)^{1/2} \leq B_{\mathrm{rms}}$, provided both cloud scale floors hold almost surely. This comparison uses points sampled at the same driving phases. With independent arc-length sampling, corresponding samples can occur at different phases. Controlling the derivative error also controls that displacement in phase; a positive speed floor makes the arc-length maps stably invertible. These are the additional conditions in the next result.

Proposition 6 (An explicit arc-length resampling condition)**.** Let $y, \widehat{y} \in C^1([0, 1]; \mathbb{R}^2)$ satisfy

$$0 < a \leq \|y'(\theta)\|_2, \|\widehat{y}'(\theta)\|_2 \leq b < \infty, \qquad \|y - \widehat{y}\|_\infty \leq E_0, \quad \|y' - \widehat{y}'\|_\infty \leq E_1.$$

Let $s_y(\theta)$ and $s_{\widehat{y}}(\theta)$ be cumulative arc length divided by the respective total lengths, each starting at phase zero. For any common fractions $u_1, \dots, u_N \in [0, 1]$, set

$$p_i = y(s_y^{-1}(u_i)), \qquad q_i = \widehat{y}(s_{\widehat{y}}^{-1}(u_i)).$$

Then

$$\|P - Q\|_N \leq E_0 + 2(b/a)^2 E_1 =: E_{\mathrm{arc}}. \tag{21}$$

If both sampled clouds have RMS scale at least s_0, their normalized squared Chamfer distance is at most E_{arc}^2/s_0^2.

Proof. Write $\ell_y(\theta) = \int_0^\theta \|y'(r)\|_2 \, \mathrm{d}r$, $L_y = \ell_y(1)$, and define their predicted counterparts analogously. The reverse triangle inequality for speeds implies

$$\|\ell_y - \ell_{\widehat{y}}\|_\infty \le E_1, \qquad |L_y - L_{\widehat{y}}| \le E_1.$$

Since $L_y, L_{\widehat{y}} \in [a, b]$ and $\ell_{\widehat{y}}(\theta) \le L_{\widehat{y}}$,

$$\|s_y - s_{\widehat{y}}\|_\infty \le \frac{E_1}{L_y} + \frac{|L_y - L_{\widehat{y}}|}{L_y} \le \frac{2E_1}{a}.$$

Both normalized arc-length maps have derivative at least a/b and are strictly increasing bijections of $[0, 1]$. Their inverses are therefore b/a-Lipschitz. Comparing their values at any common fraction u gives

$$|s_y^{-1}(u) - s_{\widehat{y}}^{-1}(u)| \le \frac{b}{a}\|s_y - s_{\widehat{y}}\|_\infty \le \frac{2bE_1}{a^2}.$$

Use the position error E_0 at the true inverse phase, and then the b-Lipschitz continuity of $\widehat{y}$, to obtain $\|p_i - q_i\|_2 \le E_0 + 2b^2E_1/a^2$ for every i. Take the cloud RMS norm and apply Proposition 5. □

For example, the abstract propagation result may be instantiated in C^1 path spaces with norm $\|f\|_{C^1} = \max\{\|f\|_\infty, \|f'\|_\infty\}$, provided the local domain, error, and gain assumptions hold in that norm. A uniform output bound B then supplies $E_0, E_1 \le B$ in Proposition 6. Its nonzero-speed condition states explicitly the additional regularity used by arc-length inversion.

What the bound makes visible. The local error, distribution shift, and path weight answer three practical questions: how accurately is a conditional rule learned, how familiar are its inputs in a new assembly, and how much does its error matter to the output? They identify where additional training coverage could be useful: unfamiliar interface conditions with large downstream influence. The next two constructions examine that influence directly. They show why small, familiar module types can still form a sensitive mechanism.

D.2 Local regularity does not bound global configuration sensitivity

The transfer bound weights local errors by their downstream influence. Can that influence be inferred from how well-conditioned each module is in isolation? A dyad gives a concrete reason to look further. Its output joint moves tangentially to the circle around its fixed anchor. The next dyad receives that motion in a particular direction, which determines how much of its potential amplification is realized.

The construction below arranges these directions in two ways. One chain passes the crank's instantaneous displacement through unchanged; the other doubles it at every stage. Both chains have the same graph, unit bar lengths, and local Jacobian singular values. Even their local solution derivatives have the same operator norm. The difference lies in how the responses line up across interfaces.

Let $e_x = (1, 0)^\top$ and $e_y = (0, 1)^\top$. A dyad with internal joint q, moving boundary joint p, and fixed boundary joint a has constraints

$$f(q, p; a) = \frac{1}{2}\begin{pmatrix} \|q - p\|^2 - 1 \\ \|q - a\|^2 - 1 \end{pmatrix} = 0. \tag{22}$$

At a feasible configuration write $u = q - p$ and $v = q - a$, so that u and v are unit vectors. The Jacobian with respect to the unknown joint is

$$J = \partial_q f = \begin{pmatrix} u^\top \\ v^\top \end{pmatrix}. \tag{23}$$

The factor $1/2$ in (22) fixes the constraint scaling: both rows of J have unit Euclidean norm. Throughout this subsection, a local Jacobian means this normalized Jacobian.

Theorem 7 (Equal local regularity, exponentially different global sensitivity)**.** For every integer $m \ge 1$, there are two grounded planar bar–joint mechanisms, denoted $\mathcal{M}_m^{\text{constant}}$ and $\mathcal{M}_m^{\text{amplifying}}$, with the following properties.

1. Both mechanisms have the same incidence graph: a unit crank drives q_0, and, for $i = 1, \dots, m$, a two-bar dyad determines q_i from q_{i-1} and a fixed anchor a_i. Every moving bar has unit length. The anchors lie on one rigid ground body. The two mechanisms have the same Assur solve graph and a chain of m successive dyads.

2. Both admit a regular analytic assembly branch through their reference configurations for $|\theta| < \delta_m$, for some $\delta_m > 0$, under the same unit-speed-at-reference crank input

$$q_0(\theta) = (-1 + \cos\theta, \sin\theta)^\top. \tag{24}$$

3. At $\theta = 0$, every dyad in both mechanisms has the same two local Jacobian singular values,

$$\sigma_- = \sqrt{1 - \frac{2}{\sqrt{5}}}, \qquad \sigma_+ = \sqrt{1 + \frac{2}{\sqrt{5}}}. \tag{25}$$

   The operator norm of each local solution derivative with respect to its moving boundary is also the same, namely $\sqrt{5}$.

4. Their endpoint configuration sensitivities at that input differ exponentially:

$$\left\| \frac{dq_m^{\text{constant}}}{d\theta}(0) \right\| = 1, \qquad \left\| \frac{dq_m^{\text{amplifying}}}{d\theta}(0) \right\| = 2^m. \tag{26}$$

Proof. We first derive the local transfer formula, then specify the two grounded mechanisms explicitly.

The transfer through one dyad. Let $R(x, y)^\top = (-y, x)^\top$ denote counterclockwise rotation through $\pi/2$, and put $t = Rv$. Because a is fixed, differentiating (22) gives

$$u^\top(dq - dp) = 0, \qquad v^\top dq = 0. \tag{27}$$

Thus dq is a multiple of t. Whenever $u^\top t \neq 0$, solving these two equations yields

$$dq = A\, dp, \qquad A = \frac{tu^\top}{u^\top t}. \tag{28}$$

In particular, if the preceding joint moves with differential $dp = t_{\text{in}}\, ds$, the scalar gain into the outgoing unit tangent t is

$$dq = t\, \frac{u^\top t_{\text{in}}}{u^\top t}\, ds. \tag{29}$$

A chain with constant endpoint sensitivity. At the reference configuration set $q_0^* = 0$ and, for every $i = 1, \dots, m$, define

$$u_i = \frac{2e_x + e_y}{\sqrt{5}}, \qquad v_i = e_x, \qquad t_i = e_y, \qquad q_i^* = q_{i-1}^* + u_i, \qquad a_i = q_i^* - v_i. \tag{30}$$

These equations specify both the joint coordinates and the fixed anchors. They give $\|q_i^* - q_{i-1}^*\| = \|q_i^* - a_i\| = 1$, as required. The crank is grounded at $a_0 = -e_x$, so (24) passes through $q_0^* = 0$ with derivative $q_0'(0) = e_y$. In fact this chain has the explicit assembly

$$q_i(\theta) = q_i^* + q_0(\theta). \tag{31}$$

The interjoint bars retain vector u_i, while $q_i(\theta) - a_i = (\cos\theta, \sin\theta)^\top$. Equation (31) therefore satisfies every bar constraint and gives $q_i'(0) = e_y$ for all i.

A chain with amplifying endpoint sensitivity. Again set $q_0^* = 0$. For odd i, set

$$t_i = e_x, \qquad v_i = -e_y, \qquad u_i = \frac{e_x + 2e_y}{\sqrt{5}}; \tag{32}$$

for even i, set

$$t_i = e_y, \qquad v_i = e_x, \qquad u_i = \frac{2e_x + e_y}{\sqrt{5}}. \tag{33}$$

Define the coordinates and anchors recursively by

$$q_i^* = q_{i-1}^* + u_i, \qquad a_i = q_i^* - v_i. \tag{34}$$

Again all moving bars have unit length. The internal joints advance strictly in both coordinate directions. Their explicit coordinates are

$$q_{2r}^* = \frac{3r}{\sqrt{5}}(e_x + e_y), \qquad q_{2r+1}^* = \frac{(3r+1)e_x + (3r+2)e_y}{\sqrt{5}}. \tag{35}$$

Put $t_0 = e_y$, the crank tangent at $\theta = 0$. At every step,

$$u_i^\top t_i = \frac{1}{\sqrt{5}}, \qquad u_i^\top t_{i-1} = \frac{2}{\sqrt{5}}. \tag{36}$$

The transfer formula (29) therefore gives, inductively,

$$q_i'(0) = 2^i t_i. \tag{37}$$

Grounding and the common constraint structure. In each mechanism all a_i, including a_0, are fixed points carried by a single rigid ground body. Ground-anchor positions may differ between the two mechanisms; their ground-link incidence is identical. All internal joints are distinct. There are also no accidental identifications between an anchor and an internal joint: internal-joint coordinates lie in $(1/\sqrt{5})\mathbb{Z}^2$, whereas the difference $q_i^* - a_i$ is a nonzero coordinate unit vector. Irrationality of $\sqrt{5}$ excludes equality with any other internal joint. The same argument, together with distinct internal coordinates, distinguishes the anchors from each other. The two constraints of each dyad are independent at the reference configuration, as verified below. With its two boundary joints pinned, a dyad is therefore an irreducible isostatic module. Each module uses only the preceding moving joint and its ground anchor. If the source group includes all ground pins, the full solve graph also has source-to-module edges; these are the same in both constructions.

Equal local singular values and transfer norms. For any of the constructed dyads,

$$J_i J_i^\top = \begin{pmatrix} 1 & u_i^\top v_i \\ u_i^\top v_i & 1 \end{pmatrix}, \qquad \left|u_i^\top v_i\right| = \frac{2}{\sqrt{5}}. \tag{38}$$

Its eigenvalues are $1 \pm 2/\sqrt{5}$, proving (25). In particular, $\det J_i \neq 0$ and both reference assemblies are regular. Since u_i and t_i are unit vectors and $|u_i^\top t_i| = 1/\sqrt{5}$, the rank-one formula (28) also gives $\|A_i\|_2 = \sqrt{5}$ for every module in both mechanisms.

Existence of the assembly branches. Substitute the analytic crank (24) into all dyad constraints, and treat $(q_1, \ldots, q_m)$ as the unknown vector. The Jacobian of the resulting system at the reference configuration is block lower triangular, with invertible diagonal blocks $J_1, \ldots, J_m$. The analytic implicit function theorem gives a unique analytic assembly branch through that configuration on an open interval about $\theta = 0$. Continuity allows the interval to be reduced so that every dyad remains regular and all distinct joints remain distinct. Taking the smaller interval for the two mechanisms gives the stated $\delta_m > 0$. This is an existence statement for each finite m; it does not require an interval of uniform width as m grows. The derivatives computed above prove (26) and complete the proof. □

The local operator norm records the largest possible response over input directions. Composition selects a particular direction at each stage. Theorem 7 shows that these choices can make the endpoint sensitivity constant or exponential even when all local norms agree. Geometric interface information therefore explains differences that the solve graph and local conditioning alone leave open. Here sensitivity refers to the configuration map $\theta \mapsto q_m(\theta)$; its relation to a curve observation also depends on the readout and sampling, as formalized in Appendix D.1.

D.3 Repeated tangencies reduce configuration regularity

The preceding chains remain regular near their reference poses: their responses have finite derivatives, although those derivatives can become large with depth. At a circle tangency, the form of the response

itself changes. Moving two tangent circles slightly toward one another opens an intersection height proportional to the square root of the displacement. If that height supplies the displacement for the next dyad, two stages produce a fourth-root response, three an eighth-root response, and so on.

The theorem realizes this repeated square-root effect with unit bars. The motion remains continuous at the limiting pose, but becomes progressively less smooth: a small change in the driving angle can produce a much larger change in the final joint position. The exponent 2^{-m} states precisely how this regularity deteriorates with the number of successive tangencies.

Theorem 8 (Depth-dependent loss of regularity at a singular assembly)**.** For every integer $m \geq 1$, there is a grounded planar mechanism formed by a unit crank followed by m unit-bar dyads, and a continuous assembly branch $q_1(\theta), \ldots, q_m(\theta)$ for $0 \leq \theta < \eta_m$, with $\eta_m > 0$, such that every dyad is regular for $0 < \theta < \eta_m$ and

$$\lim_{\theta \downarrow 0} \frac{\|q_m(\theta) - q_m(0)\|}{\theta^{2^{-m}}} = 1. \tag{39}$$

Consequently, the endpoint configuration map is not Hölder continuous with exponent $\beta > 2^{-m}$ at $\theta = 0$. In particular, over mechanisms with arbitrarily many dyads, no positive depth-independent Hölder exponent holds at all singular assemblies.

Proof. Let $n_k = e_x$ for odd k and $n_k = e_y$ for even k. In particular, n_k and n_{k+1} are perpendicular unit vectors. Put $q_0^* = 0$ and specify the reference positions and ground anchors by

$$q_k^* = q_{k-1}^* + n_k, \qquad a_k = q_{k-1}^* + 2n_k \qquad (1 \leq k \leq m). \tag{40}$$

Each reference dyad has two unit bars meeting at an external tangency: q_k^* is the midpoint of q_{k-1}^* and a_k. The crank, grounded at $a_0 = e_y$, is

$$q_0(\theta) = (\sin\theta, 1 - \cos\theta)^\top = q_0^* + \theta n_1 + o(\theta). \tag{41}$$

All a_k belong to one fixed ground body. The reference joints satisfy $q_{2r}^* = (r, r)^\top$ and $q_{2r+1}^* = (r+1, r)^\top$. The odd anchors have first coordinate minus second coordinate equal to 2, and the even anchors, including a_0, have this difference equal to -1. Thus the internal joints and anchors are all distinct.

We construct the branch inductively and prove the more precise asymptotic statement

$$q_k(\theta) - q_k^* = \theta^{2^{-k}} n_{k+1} + o\big(\theta^{2^{-k}}\big), \qquad 0 \leq k \leq m. \tag{42}$$

The case $k = 0$ is (41). Suppose the statement holds for $k-1$, and write

$$z = q_{k-1}(\theta) - q_{k-1}^*, \qquad \alpha = 2^{-(k-1)}. \tag{43}$$

Then $z = \theta^\alpha n_k + o(\theta^\alpha)$. The vector between the two boundary joints is

$$r = a_k - q_{k-1}(\theta) = 2n_k - z, \qquad d = \|r\|. \tag{44}$$

The squared height of a unit-circle intersection above the midpoint is

$$h^2 = 1 - \frac{d^2}{4} = n_k^\top z - \frac{\|z\|^2}{4} = \theta^\alpha\big(1 + o(1)\big). \tag{45}$$

For every sufficiently small positive θ, this gives $h > 0$ and $0 < d < 2$, so the two circles intersect transversely. Choose the unit vector $t(r)$ perpendicular to r whose limit as $\theta \downarrow 0$ is n_{k+1}. A continuous local choice is obtained by taking one of the two fixed signs of a 90-degree rotation of r/d. Define the chosen assembly by the circle-intersection formula

$$q_k(\theta) = \frac{q_{k-1}(\theta) + a_k}{2} + h\, t(r). \tag{46}$$

This point is at unit distance from both boundaries. Moreover,

$$\begin{aligned} q_k(\theta) - q_k^* &= \frac{z}{2} + h\, t(r) \\ &= \theta^{\alpha/2} n_{k+1} + o\big(\theta^{\alpha/2}\big), \end{aligned} \tag{47}$$

because $z = O(\theta^\alpha)$, $h/\theta^{\alpha/2} \to 1$, and $t(r) \to n_{k+1}$. This proves the induction. Each stage permits a sufficiently small positive interval; their intersection is nonempty for any fixed finite m. On a possibly smaller such interval all distinct joints remain distinct. The two constraint-gradient rows of a dyad have determinant of magnitude $dh > 0$, so every module is regular there. The formula is continuous at $\theta = 0$ with limit q_k^*.

Taking norms in (42) proves (39). If a Hölder bound of exponent $\beta > 2^{-m}$ held at zero, there would be a finite constant C with $\|q_m(\theta) - q_m(0)\| \le C\theta^\beta$ for all sufficiently small positive θ. Equation (39) contradicts this, since $\theta^{2^{-m}-\beta} \to \infty$. Finally, for any fixed $\beta > 0$ one can choose m with $2^{-m} < \beta$, proving the last assertion. □

Together, the two chain constructions distinguish amplification on regular branches from loss of smoothness at singular limits. Both arise with the simplest Assur modules. Thus identifying reusable module types is only one part of characterizing a prediction problem; the geometric regimes in which those modules operate determine the regularity available to a learner. We next turn from sensitivity to information: what happens when the representation fails to distinguish two such conditional problems at all?

D.4 Why boundary-port assignments matter

Sensitivity concerns how prediction errors propagate when the conditional problem is specified. A more basic obstruction occurs when two different problems receive the same representation. Consider a dyad attached to a moving crank joint and a fixed ground joint, with bar lengths 2 and 3. An unordered list of lengths records the same module in both cases, but does not say which length connects to the moving boundary. Swapping that assignment changes the resulting motion.

The construction below makes the difference visible even after removing position, orientation, scale, and traversal order. Both output traces are circular arcs, but their central angles differ: after equalizing their chords, one arc bends more than the other. If the representation merges these mechanisms, every predictor must give them the same answer. The separation between the true arcs therefore forces a nonzero error on at least one of them, regardless of the predictor's capacity or training data.

Proposition 9 (A port-assignment obstruction)**.** Let $O = (0, 0)$ and $B = (3, 0)$ be fixed joints and let $A(\theta) = (\cos\theta, \sin\theta)$, $0 \le \theta \le 2\pi$, be the end of a unit input crank about O. Attach a dyad with internal joint C by

$$\|C - A(\theta)\| = a, \qquad \|C - B\| = b, \tag{48}$$

and select the branch satisfying $\det(B - A(\theta), C - A(\theta)) > 0$. Consider the two assignments $(a, b) = (3, 2)$ and $(a, b) = (2, 3)$. They have identical module types, solve dependencies, boundary motions, branch signs, and unordered dyad-length multisets. Both have a regular solution throughout a complete crank rotation. Their output curves are circular arcs with central angles

$$\begin{aligned} \psi_1 &= \arccos(-3/4) - \arccos(1/4), \\ \psi_2 &= \arccos(-17/18) - \arccos(-1/2), \end{aligned} \tag{49}$$

where $0 < \psi_2 < \psi_1 < \pi$. Consequently, their output curves remain distinct after quotienting by Euclidean similarities, reflection, and traversal order.

Let $\overline{\mathcal{C}}_i$ denote these arcs after their chord endpoints are mapped to $(0, 0)$ and $(1, 0)$, with the arc above the chord, and put

$$s_i = \frac{1}{2}\tan(\psi_i/4), \qquad \Delta = s_1 - s_2 > 0. \tag{50}$$

If a representation ϕ assigns the same features to the two mechanisms, every deterministic predictor g of nonempty compact output curves satisfies

$$\max_{i\in\{1,2\}} d_H\big(g(\phi(M_i)), \overline{\mathcal{C}}_i\big) \;\ge\; \frac{\Delta}{2} \;\approx\; 0.0255918, \tag{51}$$

where d_H is Hausdorff distance. The corresponding worst-case absolute error for predicting the similarity-invariant sagitta-to-chord ratio s_i has the same lower bound.

Proof. Write $D(\theta) = \|B - A(\theta)\|$. Since $D(\theta)^2 = 10 - 6\cos\theta$, we have $2 \le D(\theta) \le 4$. For both assignments,

$$|a - b| = 1 < D(\theta) < 5 = a + b. \tag{52}$$

The two defining circles therefore intersect transversely at every input angle. More explicitly, set

$$e = \frac{B - A}{D}, \qquad t = \frac{a^2 + D^2 - b^2}{2D}, \qquad h = \sqrt{a^2 - t^2} > 0,$$

and let $e^\perp$ be the counterclockwise quarter-turn of e. The selected solution is $C = A + te + he^\perp$, a smooth 2π-periodic function of θ. For the two squared-distance constraints in Eq. (48), the Jacobian with respect to C has determinant $4\det(C - A, C - B) = 4Dh > 0$. Thus the local position solve is nonsingular throughout the complete rotation.

The selected branch lies strictly above the horizontal axis. To see this, a solution with $C_y = 0$ would have $C_x = 3 - b$ or $3 + b$. For $(a, b) = (3, 2)$ these are 1 and 5, whose distances to points of the unit circle lie in $[0, 2]$ and $[4, 6]$, respectively, and hence never equal $a = 3$. For $(a, b) = (2, 3)$ they are 0 and 6, with distance ranges $\{1\}$ and $[5, 7]$, neither containing $a = 2$. Neither branch can cross the horizontal axis. At $\theta = 0$ the selected branch has positive y coordinate, and the other branch has negative y coordinate; continuity establishes these signs for the whole cycle.

Write a point on the upper half of the output circle as

$$C = B + b(\cos\varphi, \sin\varphi), \qquad 0 < \varphi < \pi.$$

Such a point is reached for some input angle if and only if the circle centred at C with radius a intersects the unit circle centred at O. Because $a > 1$, this is precisely

$$a - 1 \le \|C\| \le a + 1.$$

Every such intersection belongs to the selected branch, since its C_y coordinate is positive. This proves that the complete traced set, including its endpoints, is exactly the arc specified by

$$\frac{(a-1)^2 - 9 - b^2}{6b} \;\le\; \cos\varphi \;\le\; \frac{(a+1)^2 - 9 - b^2}{6b}. \tag{53}$$

Substituting the two assignments gives Eq. (49). Both angles lie strictly between zero and π, and

$$\cos\psi_1 = \frac{-3 + \sqrt{105}}{16}, \qquad \cos\psi_2 = \frac{17 + \sqrt{105}}{36}.$$

Their difference satisfies $\cos\psi_2 - \cos\psi_1 = (95 - 5\sqrt{105})/144 > 0$, so $\psi_1 > \psi_2$. A Euclidean similarity, including a reflection, preserves the central angle of a proper circular arc. Traversal changes also leave the traced set unchanged. The two curves are therefore different under all of the stated identifications.

An arc with radius R and central angle $\psi < \pi$ has chord length $2R\sin(\psi/2)$ and sagitta $R(1 - \cos(\psi/2))$. Their ratio is $\frac{1}{2}\tan(\psi/4)$, which establishes Eq. (50). Numerically,

$$\psi_1 \approx 1.1007423341, \quad \psi_2 \approx 0.7123013928, \qquad s_1 \approx 0.1411744917, \quad s_2 \approx 0.0899909200.$$

For nonempty compact sets $K, L \subset \mathbb{R}^2$, the height functional satisfies $|\max_{x\in K} x_y - \max_{x\in L} x_y| \le d_H(K, L)$. Hence $d_H(\overline{\mathcal{C}}_1, \overline{\mathcal{C}}_2) \ge \Delta$. If their features coincide, both mechanisms receive the same prediction $K = g(\phi(M_1)) = g(\phi(M_2))$. The Hausdorff triangle inequality then gives

$$\Delta \le d_H(\overline{\mathcal{C}}_1, K) + d_H(K, \overline{\mathcal{C}}_2),$$

which proves Eq. (51). The scalar statement follows from the same triangle inequality on $\mathbb{R}$. □

The numerical floor is expressed for the specified chord normalization and Hausdorff loss. The preceding non-equivalence of the two shapes holds under any faithful normalization by similarities and traversal order. The proposition concerns representations that discard port assignments; a record retaining complete link incidence distinguishes the two mechanisms.

The observation in the experiment is a finite point cloud. To connect the construction to that target, the next proposition shows that the difference in arc angle remains visible after sampling. The obstruction does not depend on resolving arbitrarily fine details of a continuous curve.

Proposition 10 (The distinction survives finite curve sampling)**.** For either mechanism in Proposition 9, sample one complete crank cycle at m equally spaced periodic arc-length positions, with arbitrary starting phase. The sampled points occupy a circular arc whose minimal covering central angle $\widehat{\psi}_i$ satisfies

$$\psi_i(1 - 2/m) \le \widehat{\psi}_i \le \psi_i. \tag{54}$$

For every $m \ge 6$, the two sampled point sets are not related by a Euclidean similarity, including reflection. This includes 128 periodic samples and 128 samples with the final point repeating the initial point.

Proof. Each point in the interior of the arc in Eq. (53) admits exactly two input-crank positions: the circle centred at C with radius a intersects the unit circle in two distinct points. Each endpoint admits exactly one, where those circles are tangent. All these configurations are on the selected dyad branch, as established in Proposition 9. The crank therefore traverses the output arc once in each direction. Equivalently, the derivative of the output angle can vanish only when $(C - A)$ is parallel to A, which is precisely one of those two tangencies; the denominator of implicit differentiation is nonzero by dyad regularity. The total output-path length is consequently $2b\psi_i$.

Every arc-length position on this closed traversal is within cyclic arc-length distance $b\psi_i/m$ of a sample. In particular, each turning point has a sampled point within angular distance ψ_i/m. All samples lie between the two turning points, which proves Eq. (54). For $m \ge 6$, there are at least three distinct sampled points, since each interior output point has at most two preimages on the traversal. They determine the output circle, and the shortest containing arc has angle $\widehat{\psi}_i < \pi$. This angle is invariant under similarities and reflection. Finally,

$$\widehat{\psi}_1 \ge \tfrac{2}{3}\psi_1 \approx 0.7338282 > 0.7123014 \approx \psi_2 \ge \widehat{\psi}_2.$$

Thus the sampled shapes are distinct under the stated identifications. With 128 samples including a duplicate closing point, the periodic set has $m = 127$ samples, so the same conclusion applies. □

D.5 The dimension required to retain observable behaviour

The port example gives one explicit collision between different behaviours. The next result asks when such collisions become unavoidable for an entire family. The relevant size is the number of independently observable variations: normalization may remove some parameter changes, while others remain visible in the target. A continuous representation with too few coordinates must merge some of these visible variations.

The proof uses a geometric fact: on the boundary of a q-dimensional parameter ball, a continuous map into fewer than q coordinates identifies some opposite points. If their true outputs remain separated, a shared prediction must incur an error. The theorem turns this idea into a quantitative lower bound. Let $B_r^q = \{\theta \in \mathbb{R}^q : \|\theta\| \le r\}$ and let $(\mathcal{Y}, d_{\mathcal{Y}})$ be a metric space of outputs after the chosen normalization. The dimension argument uses the Borsuk–Ulam theorem; related latent-dimension obstructions for invariant representations are developed by **?**.

Theorem 11 (An observable-dimension obstruction)**.** Let $T : B_r^q \to \mathcal{Y}$ describe a mechanism family with $r > 0$. Suppose there is a constant $c > 0$ such that

$$d_{\mathcal{Y}}(T(\theta), T(\theta')) \ge c\|\theta - \theta'\| \qquad \text{for every } \theta, \theta' \in B_r^q. \tag{55}$$

If $d < q$, every continuous representation $\phi : B_r^q \to \mathbb{R}^d$ and every decoder $g : \mathbb{R}^d \to \mathcal{Y}$ satisfy

$$\sup_{\theta \in B_r^q} d_{\mathcal{Y}}(g(\phi(\theta)), T(\theta)) \ge cr. \tag{56}$$

In particular, exact prediction of this family by continuous features requires at least q feature coordinates.

Proof. Restrict ϕ to the boundary sphere rS^{q-1}. For $q \ge 2$, pad its values with zeros to obtain a continuous map to $\mathbb{R}^{q-1}$. The Borsuk–Ulam theorem gives a point $\theta \in rS^{q-1}$ with $\phi(\theta) = \phi(-\theta)$. For $q = 1$ the only admissible d is zero and the same equality holds directly. Both parameters receive the same prediction y. By Eq. (55) and the metric triangle inequality,

$$2cr \le d_{\mathcal{Y}}(T(\theta), T(-\theta)) \le d_{\mathcal{Y}}(T(\theta), y) + d_{\mathcal{Y}}(y, T(-\theta)).$$

At least one of the two errors is at least cr. □

How can the separation assumption be checked for a concrete mechanism family? Measure several quantities from its output and differentiate them with respect to the family parameters. Full column rank means that every local parameter direction changes those measurements to first order. The next proposition turns this observable rank into the separation required by the theorem on a sufficiently small neighbourhood.

Proposition 12 (An observable-rank criterion)**.** Let $U \subset \mathbb{R}^q$ be open, $\theta_0 \in U$, and $T : U \to \mathcal{Y}$. Suppose $a : \mathcal{Y} \to \mathbb{R}^p$ is L-Lipschitz with $L > 0$, $F = a \circ T$ is continuously differentiable near θ_0, and $DF(\theta_0)$ has column rank q. Write $\sigma > 0$ for its smallest singular value. There exists $r > 0$ such that, on $\theta_0 + B_r^q$, condition (55) holds with $c = \sigma/(2L)$. Consequently, every continuous representation of dimension $d < q$ has worst-case prediction error at least $\sigma r/(2L)$ on this neighbourhood.

Proof. Set $J = DF(\theta_0)$. By continuity choose $r > 0$ with $\theta_0 + B_r^q \subset U$ and

$$\sup_{\theta \in \theta_0 + B_r^q} \|DF(\theta) - J\|_{\mathrm{op}} \le \sigma/2.$$

For θ, θ' in this convex ball, the fundamental theorem of calculus along the segment between them gives

$$F(\theta) - F(\theta') = J(\theta - \theta') + \int_0^1 \big(DF(\theta' + t(\theta - \theta')) - J\big)(\theta - \theta')\, dt.$$

Therefore $\|F(\theta) - F(\theta')\| \ge (\sigma/2)\|\theta - \theta'\|$. The L-Lipschitz property of a gives

$$d_{\mathcal{Y}}(T(\theta), T(\theta')) \ge \frac{\|F(\theta) - F(\theta')\|}{L} \ge \frac{\sigma}{2L}\|\theta - \theta'\|.$$

Apply Theorem 11 after translating the parameter ball by θ_0. □

Finite observations. For the paper's target of 128 sampled two-dimensional curve points, a regular local coordinate description has at most 256 continuous coordinates before normalization; any independently removed continuous symmetries further reduce its dimension. Thus the observable rank q in Proposition 12 is bounded by the dimension of that chosen observation. A dimension bound increasing without limit with mechanism size would require increasingly detailed observations as well.

D.6 Lossless structure exposes computation without adding information

The preceding obstructions concern representations that merge outputs the task must distinguish. A lossless structural decomposition raises a different question: if the complete mechanism can be recovered, what has changed for learning? The proposition below isolates the answer. An unrestricted learner can translate between the raw and decomposed records, so both permit the same predictions. The benefit of an explicit decomposition must therefore be understood through the computation and inductive bias available to the chosen learner.

Proposition 13 (Statistical equivalence of lossless representations)**.** Let $E : \mathcal{R} \to \mathcal{A}$ be a measurable encoding of raw mechanism records, and let $P : E(\mathcal{R}) \to \mathcal{R}$ be a measurable recovery map satisfying $P(E(R)) = R$. For every sample size, every loss, and every family of joint laws of (R, Y), the sets of achievable prediction risks from R and from $E(R)$ coincide when learners have unrestricted computation and may use E and P. Their optimal and minimax risks therefore coincide.

Proof. A learner receiving raw training records applies E to every training input and to the test input, then runs any learner for the encoded representation. Conversely, a learner receiving encoded records applies P to the training and test inputs and runs any raw-record learner. These simulations give exactly the same predictions for every data set and test input; randomized learners can use the same random seed. Consequently, both simulations preserve risk under every joint law. Taking infima over learners, and then suprema over any specified family of laws, proves the assertions. □

This is the reason to compare representations with the same learner in the paper's prediction experiment: it asks whether making constraint organization explicit helps that learner use the available data. The lossless Assur record and the fixed feature vector extracted from it are distinct objects. The former exposes a recoverable computational structure; the latter retains the interface and geometric

details chosen by feature extraction. The port and dimension results explain why those choices matter.

D.7 Constraint modules and graph width measure different structure

The preceding results make the conditional solve structure central to reuse. Could an ordinary measure of undirected graph complexity already capture that structure? The size of an Assur component measures how many joint positions must be determined together in the Assur decomposition. Treewidth measures the size of overlapping vertex sets needed in an undirected tree decomposition. The following two constructions show why they differ.

In the first, a grid is built one dyad at a time: each new joint attaches to two joints whose positions are already known. The undirected grid becomes wide, while each conditional solve remains small. In the second, joints form a cycle and each also attaches to one of two fixed pins. The graph has bounded treewidth, but no proper set of internal joints is determined from the pins alone; the whole mechanism forms one Assur component. Both families use only two boundary joints per component.

Throughout this subsection, $G = (I \dot\cup P, E)$ is a simple pinned bar–joint graph in the plane: I contains the internal joints, P contains the fixed boundary joints, and every edge has an internal endpoint. We take treewidth on this graph, including the pins and without adding edges between them. "In the plane" refers to the coordinates of the joints; crossings between bars are permitted and are not additional joints. For a generically pinned-isostatic graph, let $\kappa(G)$ be the largest number of internal joints in an Assur component and let $p(G)$ be the largest number of distinct boundary joints of a component. Component boundaries may include joints determined by earlier components.

Theorem 14 (Separation of Assur component size and treewidth)**.** There are two families of generically pinned-isostatic planar bar–joint graphs, each with exactly two pins, with the following properties:

1. For every integer $r \geq 2$, a graph G_r with $(r+1)^2$ internal joints satisfies
$$\kappa(G_r) = 1, \qquad p(G_r) = 2, \qquad \mathrm{tw}(G_r) \geq r.$$
2. For every integer $n \geq 3$, a graph H_n with n internal joints satisfies
$$\kappa(H_n) = n, \qquad p(H_n) = 2, \qquad \mathrm{tw}(H_n) \leq 4.$$

Consequently, bounded Assur component size does not imply bounded treewidth; bounded treewidth and bounded interface arity together do not imply bounded Assur component size.

Proof. **Dyadic graphs containing large grids.** Set $s = r + 1$, take two distinct pins a, b, and introduce joints q_{ij} for $1 \leq i, j \leq s$ in row-major order. Attach each new joint to two previously available joints as follows:

new joint	attachment joints
q_{11}	a, b
$q_{1j},\ j > 1$	$q_{1,j-1}, a$
$q_{i1},\ i > 1$	$q_{i-1,1}, b$
$q_{ij},\ i, j > 1$	$q_{i-1,j}, q_{i,j-1}$.

There are no other edges. The two attachment joints are always distinct, and an edge is introduced only with its later endpoint, so the graph is simple. Every addition is a dyad with one internal joint and two boundary joints. At a generic placement, its two incident bar directions are linearly independent. Ordering the constraint rows and internal coordinate columns by insertion gives a block lower-triangular pinned rigidity matrix with invertible 2×2 diagonal blocks. Thus the graph is generically pinned-isostatic. Its successive dyads form an Assur decomposition, so $\kappa(G_r) = 1$ and $p(G_r) = 2$; equivalently, the Assur decomposition theorem identifies these singleton components (Servatius et al., 2010; Shai et al., 2013).

The graph induced by the q_{ij} contains the $s \times s$ grid, including every horizontal and vertical grid edge. We give a direct lower bound on its treewidth. For each row i and column j, let C_{ij} be the union of that row and that column. Every C_{ij} is connected and any two such crosses intersect. In any tree decomposition, the bags meeting a connected vertex set form a subtree. The subtrees corresponding

to the crosses therefore intersect pairwise. The Helly property of subtrees of a tree gives a bag meeting every cross. Any set of fewer than s grid vertices misses an entire row and an entire column, and hence misses their cross. That bag must consequently contain at least s grid vertices. Its width is at least $s - 1 = r$. Restricting a tree decomposition to a subgraph cannot increase its width, so $\mathrm{tw}(G_r) \geq r$.

Large irreducible components of bounded treewidth. Take internal vertices $v_1, \ldots, v_n$ forming a simple cycle and two pins a, b. Give each internal vertex exactly one additional edge, to a when its index is odd and to b when its index is even. Both pins are used for every $n \geq 3$. There are n cycle edges and n spokes, hence $|E(H_n)| = 2n$.

We verify the pinned-Laman counts, which characterize generic pinned isostaticity in the plane (Servatius et al., 2010). For a nonempty internal vertex set S and a selected pin set P', the relevant induced edge bounds are $2|S|$ when $|P'| \geq 2$, $2|S| - 1$ when $|P'| = 1$, and $2|S| - 3$ when $P' = \emptyset$ and $|S| \geq 2$. A single unpinned vertex has no edges and is treated separately.

If $\emptyset \neq S \subsetneq I$, the induced cycle edges form a forest and number at most $|S| - 1$. Including any selected pins adds at most $|S|$ spokes. Thus a subset with either one or two pins has at most $2|S| - 1$ edges, satisfying the corresponding bound. With no pins and $|S| \geq 2$, there are at most $|S| - 1 \leq 2|S| - 3$ edges. If $S = I$, the no-pin subgraph has $n \leq 2n - 3$ edges, including equality at $n = 3$. With exactly one pin it has at most $n + (n - 1) = 2n - 1$ edges because both pins are used. With both pins it has exactly $2n$ edges. Induced edge counts also bound all non-induced subgraphs, completing the count verification.

It follows that H_n is generically pinned-isostatic. Moreover, every proper nonempty internal subset has strictly fewer than $2|S|$ constraints even with both pins available. Such a subset cannot be pinned-rigid: its pinned rigidity matrix has $2|S|$ columns and fewer rows. There is therefore no proper pinned-isostatic subassembly. The whole graph is an Assur component, so $\kappa(H_n) = n$ and $p(H_n) = 2$.

Finally, the bags

$$B_j = \{a, b, v_1, v_j, v_{j+1}\}, \qquad j = 2, \ldots, n - 1,$$

ordered along a path form a tree decomposition. They cover the first and last cycle edges through v_1, every other cycle edge through its corresponding bag, and every spoke through its internal endpoint. Each vertex occurs in a connected interval of bags. All bags have five vertices, so the width is at most four. For $n = 3$, the same construction consists of a single five-vertex bag and remains valid. □

Corollary 15 (Linear conditional evaluation despite unbounded treewidth)**.** For the family G_r in Theorem 14, suppose the pin positions, all bar lengths, and a feasible assembly branch are given, and every dyad on that branch has two distinct attachment points and transverse circle intersections. All $(r + 1)^2$ internal positions can then be evaluated in $O(r^2)$ scalar operations in a model assigning unit cost to arithmetic operations and square roots.

Proof. Let a dyad's attachment positions be u, v, its bar lengths be ℓ_u, ℓ_v, and its specified branch be $\sigma \in \{-1, 1\}$. Writing $d = \|v - u\|$, $e = (v - u)/d$, and $e^\perp$ for a fixed quarter-turn of e, its internal position is

$$q = u + \alpha e + \sigma h e^\perp, \qquad \alpha = \frac{\ell_u^2 - \ell_v^2 + d^2}{2d}, \qquad h = \sqrt{\ell_u^2 - \alpha^2}.$$

The hypotheses give $d > 0$ and $h > 0$, so this uses a constant number of operations. Row-major traversal evaluates each of the $(r+1)^2$ dyads once, with its attachment positions already available. □

The grid gives a concrete computational consequence: a known feasible branch is evaluated by one circle intersection per joint, even as its treewidth grows. The cycle shows the converse distinction: small interfaces and small treewidth can coexist with a single large Assur component. Thus undirected connectivity and conditional solve structure describe different aspects of a mechanism's complexity. The corollary counts scalar operations for evaluating a supplied feasible branch at one boundary configuration; numerical bit complexity and finding that branch are separate computational questions.

Table 7: **The two source corpora store what a mechanism is, and neither stores how it is built.** Rows are LINKS (Heyrani Nobari et al., 2022), the Purwar Lab four-, six-, and eight-bar corpus (Nurizada et al., 2024), and the layer derived from them, with source counts as those publications report. The geometry discussion below qualifies the two marked cells.

		Mechanism				Construction	
Resource	Records	Typed incidence	Pose x_0	Coupler curve	Sliders, n-ary	Solve order	Group status
LINKS	10M	×	✓	✓†	×	×	×
Purwar Lab	3M	✓	✓‡	✓	✓	×	×
Assur-KB (this work)	12.99M	✓	✓	✓	✓	✓	✓

Consequences for representation and evaluation. The results give a connected account of predictive reuse. Interfaces must retain distinctions visible in the target; training must cover the local conditions encountered at those interfaces; and downstream geometry determines how much local error matters. A solve decomposition makes these quantities explicit, even when the undirected graph is large or has high treewidth. They suggest concrete axes for future compositional evaluations: vary interface coverage and propagation sensitivity separately, while holding the module vocabulary fixed. The present common-MLP experiment addresses the preceding representational question—whether exposing constraint organization helps a fixed learner predict behaviour from limited data—as discussed in Appendix **??**.

E Census of the Built Knowledge Base

Here we report a complete census of the Assur-KB build present in this checkout. The census describes the dataset. The topology name and the source directory are summarised at a top-50 cutoff with the tail collapsed into a single catch-all bucket.

Throughout, *module* means a candidate Assur group and *order* is the stored interface arity of §3.4, the number of distinct boundary joints. Group counts labelled *total* include the ground group of each mechanism; *Assur groups* exclude it.

E.1 Corpus totals

Table 8: Build totals per source. Extraction time is the decomposition cost recorded on each record, summed over the source.

Quantity	LINKS	Purwar Lab	Total
Shards	101	304	405
Records	10,020,000	2,966,754	12,986,754
Unparsable lines	0	0	0
JSONL size (GB)	131.1	26.7	157.8
Mean extraction time (ms/record)	1.64	4.17	—
Total extraction time (CPU h)	4.57	3.43	8.00

The LINKS side is the `LInK-10M` release, split 10,000,000 train and 20,000 test, all revolute, all decomposed by the dyadic route. The Purwar Lab side is streamed from the upstream `purwarlab` release and decomposed by the link-incidence route; its records divide into 602,079 four-bar, 397,525 six-bar, and 1,967,150 eight-bar mechanisms. Extraction per record: eight CPU hours cover the whole build, and the tail is short (the slowest LINKS record takes 8.2 s and the slowest Purwar Lab record 4.8 s).

E.2 Mechanism-level structure

The two sources cover complementary regimes. LINKS supplies the size range, spanning 6 to 20 joints roughly uniformly and reaching solve graphs of depth 14, but every one of its mechanisms

Table 9: Per-mechanism distributions. Groups include the ground group; DAG depth is the longest path in the condensation DAG.

	LINKS				Purwar Lab			
Quantity	min	med	mean	max	min	med	mean	max
Joints	6	13	12.91	20	5	11	9.49	11
Links	7	19	19.47	35	2	8	6.82	8
Ground joints	2	3	2.68	7	2	3	2.79	4
Groups (total)	4	10	10.23	18	1	4	3.57	5
Assur groups	3	9	9.23	17	0	3	2.57	4
Condensation edges	4	17	16.79	33	0	4	3.70	7
DAG depth	2	6	6.08	14	0	2	2.16	4

decomposes into dyads alone. Purwar Lab is small and shallow by comparison, capped at 11 joints and depth 4, and supplies the structural variety: higher interface arity, n-ary bodies, and prismatic pairs. The 153,124 Purwar Lab records with joint code `PRPR` account for every zero in the Assur-group column: they carry a ground group and no module, and the build marks their topology as incomplete. They are the reason the Purwar Lab minimum for groups, edges, and depth is not the four-bar value.

E.3 Modules

Table 10: Module inventory by source. Percentages are shares of that source's Assur groups (top block) or of its mechanisms (bottom block).

	LINKS		Purwar Lab	
Quantity	count	%	count	%
Groups (total)	102,534,069	—	10,599,777	—
Ground groups	10,020,000	—	2,966,754	—
Assur groups	92,514,069	100.0	7,633,023	100.0
order 2 (dyads)	92,514,069	100.0	5,761,578	75.5
order 3	0	0.0	1,703,554	22.3
order 4	0	0.0	167,891	2.2
Distinct shape keys	1	—	33	—
Mechanisms, dyads only	10,020,000	100.0	1,081,258	36.4
Mechanisms with order > 2	0	0.0	1,732,372	58.4
Mechanisms with no module	0	0.0	153,124	5.2

Table 11: Per-module distributions. Carried points are tracer points attached to a module's links and are counted over all groups, including ground.

	LINKS				Purwar Lab			
Quantity	min	med	mean	max	min	med	mean	max
Internal joints	1	1	1.00	1	1	1	1.84	7
Boundary joints	2	2	2.00	2	2	2	2.27	4
Bodies	2	2	2.00	2	1	2	1.99	6
Carried points	0	0	0.00	0	0	0	0.28	2

The LINKS module population is a single point: 92,514,069 groups, all with the shape key `c2/i1/e2/L2-2`, one internal joint, two boundary joints, and two binary bars. This is a property of the source, whose mechanisms are generated dyadically, and it is why the audited LINKS-derived records support no claim about higher-order modules. The Purwar Lab population is where the schema's generality is exercised: a quarter of its modules have interface arity above two, modules reach seven internal joints and six bodies, and 33 distinct shape keys appear. A shape key records

order, internal-joint count, boundary-joint count, and the sorted body arities, so `c2/i3/e2/L3-3` denotes an order-2 module with three internal joints closed by two ternary links.

Table 12: The ten most frequent Purwar Lab shape keys, covering 90.1% of its Assur groups.

Shape key	Groups	% of Assur groups
`c2/i1/e2/L2-3`	1,975,188	25.9
`c2/i2/e2/L2-3`	1,577,230	20.7
`c3/i1/e3/L3`	969,112	12.7
`c2/i1/e2/L2-2`	787,543	10.3
`c2/i3/e2/L3-3`	538,901	7.1
`c2/i3/e2/L2-4`	338,516	4.4
`c3/i2/e3/L4`	231,528	3.0
`c2/i4/e2/L3-4`	170,360	2.2
`c4/i1/e4/L3`	146,165	1.9
`c3/i3/e3/L2-2-3-3`	146,078	1.9

The classical dyad, two binary bars on two boundary joints, is only the fourth most common Purwar Lab module at 10.3%. The head of the distribution is a binary-plus-ternary pair, and the third entry is a single ternary link on three boundary joints, the order-3 module that a binary joint graph cannot express.

E.4 Bodies and joints

Table 13: Typed incidence in the build. Body and joint counts are instances summed over mechanisms, not distinct types.

Quantity	LINKS	Purwar Lab
Bodies	195,048,138	20,698,934
binary	195,048,138	10,321,008
ternary	0	8,938,010
quaternary	0	1,439,916
n-ary bodies ($n > 2$)	0	10,377,926
Mechanisms with an n-ary link	0	2,966,754
Nested bodies	0	614,818
Revolute joints	129,374,453	27,423,968
Prismatic joints	0	733,017
Mechanisms with a prismatic joint	0	439,591

Half of all Purwar Lab bodies (50.1%) are ternary or quaternary, and every Purwar Lab mechanism contains at least one such link, so the whole of that source depends on incidence being preserved: flattened to pairwise adjacency, each of these bodies would be indistinguishable from a ring of bars. Prismatic pairs are rarer, 2.6% of joint instances, but they appear in 14.8% of Purwar Lab mechanisms, which is what makes the three slider families (`RRRP`, `RRPR`, `PRPR`) reachable at all.