

Representations and the Geometry of Discovery

How does changing the description of a problem change the search?

Complexity, Information, and Cross-Domain Transfer

Why suspect the coordinates rather than the objective?

In a narrow quadratic valley, a step safe across the steep direction barely moves along the flat one. Raising one shared step size threatens instability.

The minimum is unchanged by renaming coordinates. Could unequal progress therefore be a property of the units in which we search, rather than missing information about the answer?

Constructing alternatives from the observed mismatch

Candidate	Consequence for this quadratic
Larger shared step	Faster flat motion; steep direction limits stability
Per-direction step sizes	Can compensate for known unequal curvature
Rescale the second coordinate	Can express the same correction as round level sets

The diagnosis suggests changing step size, per-direction movement, or coordinates. These are related designs, with different information and cost requirements.

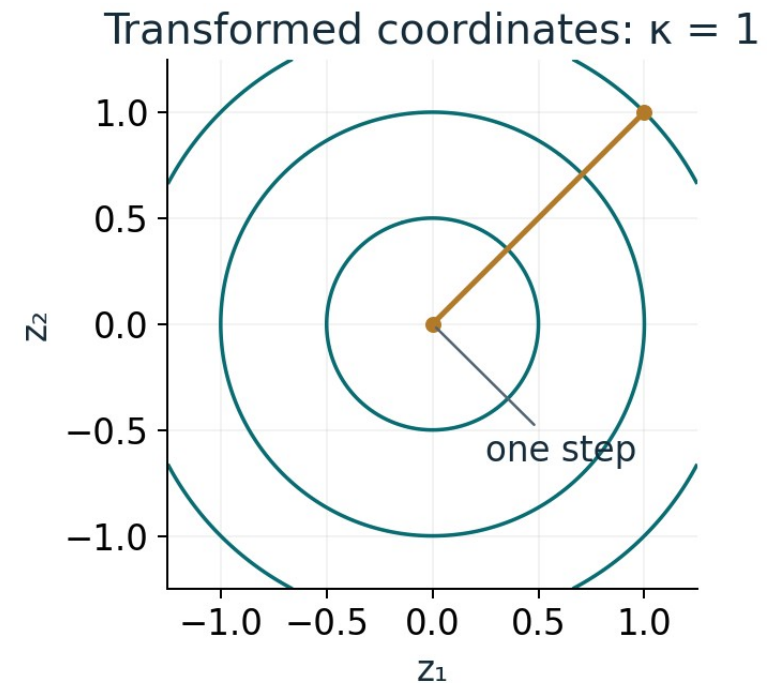
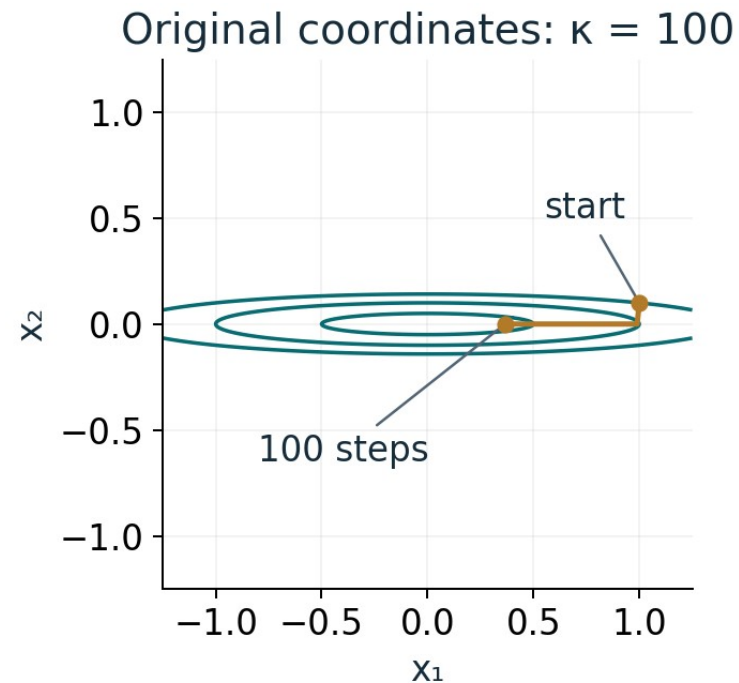
Choosing the rescaling from the curvature

$$\kappa(\nabla^2 f): 100 \rightarrow 1; \quad z = (x_1, 10x_2)$$

Start with $f(x) = \frac{1}{2}(x_1^2 + 100x_2^2)$. To absorb the coefficient 100 into a square, choose $z_2 = 10x_2$ because $10^2 = 100$; keep $z_1 = x_1$.

Then $f = \frac{1}{2}\|z\|^2$: step 1 reaches zero. In x , step 0.01 leaves x_1 multiplied by 0.99 per iteration. The gain assumes the curvature is known.

The coordinate change makes level sets circular



The same objective has condition number 100 in x and 1 in z . The coordinate scales are equal within each panel. The gain assumes the useful scaling is already known. Estimating and applying a preconditioner consumes resources.

Objects and their encodings

$$z \in Z \rightarrow d(z) \in X \rightarrow \text{evaluation}$$

Let X be scientific objects and Z their encodings, with a decoder $d: Z \rightarrow X$. Search operates on Z and evaluates decoded candidates.

A representation can omit valid objects, give several encodings to one object, or make nearby codes decode to very different objects.

A useful metric must preserve relevant distinctions

For molecular graphs, edit distance measures graph changes, not necessarily changes in binding or synthesis cost. For programs, a one-character edit can radically change behavior.

A smooth surrogate is valuable only where representation distance predicts a useful relationship in objective or feasibility.

Constraint-preserving parameterization

n^2 entries $\rightarrow n(n+1)/2$ free symmetric entries

Searching directly over symmetric matrices requires equality constraints $A_{ij}=A_{ji}$. Storing only the upper triangle makes symmetry automatic.

The number of free entries falls from n^2 to $n(n+1)/2$, but this does not resolve every remaining optimization difficulty.

Symmetry creates redundant candidates

$$f(g \cdot x) = f(x) \text{ for all } g \in G$$

A group G acts on X . If feasibility and objective are invariant under this action, an entire orbit has the same scientific value.

Searching one representative per orbit may save work, provided finding that representative is itself feasible.

The orbit count is not always $|X|/|G|$

8 strings $\rightarrow$ 4 singleton orbits + 2 pairs = 6 orbits

If no nonidentity symmetry fixes an object, all orbits have size $|G|$. Otherwise orbit sizes vary.

For binary strings of length three under reversal, 000, 010, 101 and 111 are fixed, while $001 \leftrightarrow 100$ and $011 \leftrightarrow 110$ form pairs: six orbits.

Cap sets under affine transformations

$$(Ax+b)+(Ay+b)+(Az+b)=A(x+y+z)$$

Invertible affine maps over F_3 preserve the absence of nontrivial triples summing to zero. Searching affine-equivalent copies wastes effort when the objective is set size.

For $x \mapsto Ax+b$, the transformed triple sum is $A(x+y+z)+3b=A(x+y+z)$.

Compositional representations create reusable moves

A grammar builds candidates from primitives and composition rules. Search can add a meaningful component instead of changing individual symbols.

A small grammar may make useful programs short while excluding other functions or hiding expensive primitives.

When a library pays for itself

$$12T - (20 + 3T) = 9T - 20$$

A repeated program fragment costs 12 units to write inline. Defining a reusable primitive costs 20; each call costs 3.

With T uses, inline cost is $12T$ and library cost is $20 + 3T$. The library is cheaper from $T=3$ onward.

DreamCoder makes library learning concrete

DreamCoder alternates program search with learning reusable abstractions and a recognition model. A changing library alters which solutions are short and easy to propose.

Its results support learned reuse within studied domains, not unrestricted transfer between arbitrary scientific tasks.

Description length and computation are different

$$K_u(x) = \min \{|p| : U(p)=x\}$$

Kolmogorov complexity $K_u(x)$ is the length of the shortest program that outputs x on a fixed universal machine U .

A short program may run for an enormous time. Exact K_u is uncomputable in general, so practical compressors provide limited proxies.

Compression is evidence of regularity, not truth

$$L(\text{model}) + L(\text{data} \mid \text{model})$$

A concise formula can summarize observations while failing outside the observed regime. Description length penalizes complexity only relative to a code and model class.

A meaningful comparison charges for both the model and the data left unexplained by it.

A two-part code must pay for the model

$$L(\text{model}, \text{data}) = L(\text{model}) - \log_2 P(\text{data} \mid \text{model})$$

Encode 00001111 with two predeclared models. A fair-coin model uses 1 model bit plus 8 data bits: 9 total.

A split-at-four model assigns each observed bit probability 0.9. With a 4-bit model code, its total is $4 - 8 \log_2(0.9) \approx 5.216$ bits.

A short law can hide a difficult inverse problem

The equation $y=f_{\theta}(x)$ may be easy to evaluate for known θ , while inferring θ from noisy y is ill-conditioned or nonidentifiable.

Compact scientific laws do not automatically make discovery of their parameters or mechanisms easy.

Ill-conditioning amplifies observational error

$$\theta_2 = (y_2 - y_1) / \eta$$

Suppose $y_1 = \theta_1 + \theta_2$ and $y_2 = \theta_1 + (1 + \eta)\theta_2$. Subtraction gives $\theta_2 = (y_2 - y_1) / \eta$.

For $\eta = 0.01$, an error of 0.001 in the difference creates an error of 0.1 in θ_2 . More precise computation cannot remove measurement uncertainty.

Effective dimension depends on the signal

A high-dimensional representation may concentrate useful variation in a smaller subspace. This can reduce estimation or optimization cost if the target respects that structure.

Low reconstruction error does not guarantee preservation of a rare causal signal or a decisive design constraint.

Why should a rotated molecule need new training?

An isolated molecule has the same energy after a rigid rotation. A force vector, however, must rotate with the atoms. Treating every coordinate as an unrelated feature hides that structure.

This suggests constraining how predictions transform, before choosing a particular network. The relevant symmetry depends on whether the environment rotates too.

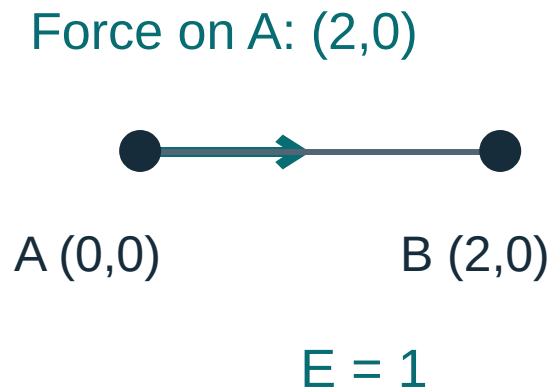
Three designs follow from the same constraint

Candidate	Design move	Trade-off
Rotation augmentation	Train on transformed inputs	More examples; approximate consistency
Invariant features	Use distances for energy	Vector output needs direction
Equivariant model	Constrain the transformation	Architecture must match the physics

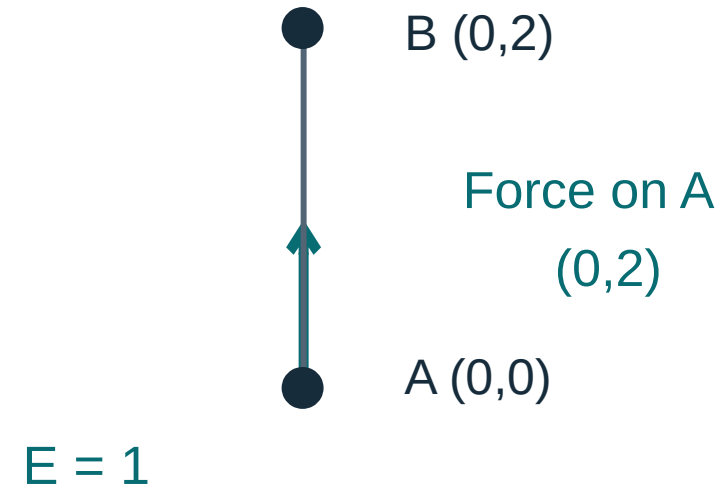
For a scalar target, invariant features may suffice. For a vector target, retain directions and specify how they transform. The required output helps choose the design.

Energy stays fixed while forces rotate

Original coordinates



After a 90° rotation



Toy spring: $E = \frac{1}{2}k(r-r_0)^2$, with $k=2$, $r_0=1$ and $r=2$. Both orientations have $E=1$. The force on atom A rotates from (2,0) to (0,2).

An invariant energy yields equivariant forces

$$E(RX)=E(X) \Rightarrow F(RX)=RF(X)$$

Let X collect all atomic positions and R rotate every position. Differentiate $E(RX)=E(X)$: the chain rule gives $R^T \nabla E(RX) = \nabla E(X)$.

Since $R^T R = I$ and $F = -\nabla E$, multiplying by R gives the force rule above. Symmetry guarantees consistency under rotation; it cannot guarantee correct energy or chemistry.

A new variable changes what the model can explain

$$R = dM, M \sim \text{Poisson}(\mu/d) \Rightarrow E[R]=\mu, \text{Var}(R)=d\mu$$

Let M count independent founding events and d be descendants per event. The resistant count $R=dM$ has mean μ and variance $d\mu$; an independent-count Poisson model has both mean and variance μ .

With $\mu=4$ and $d=8$, the means are both 4 but the variances are 4 and 32. Measuring only the mean hides the distinction.

An invertible encoding preserves information

$$H \rightarrow X \rightarrow Z \implies I(H;Z) \leq I(H;X)$$

If Z is a deterministic invertible recoding of X , it preserves mutual information with a target H . A lossy map can only discard information about H .

Yet even an information-preserving encoding can radically change the computational ease of using that information.

Sufficient for a target need not mean invertible

$$P_{\theta}(X=x) = \theta^{S(x)}(1-\theta)^{n-S(x)}$$

For iid Bernoulli observations $X_1, \dots, X_n$ with unknown θ , the count $S = \sum X_i$ preserves the likelihood for θ ; the order of outcomes can be discarded.

S is sufficient for θ in this model. It loses information needed to study serial dependence or changing success rates.

A representation has an acquisition cost

$$c_0 + T c_1 < T c_2 \iff T > c_0/(c_2 - c_1)$$

Let c_0 be the cost of building a representation, c_1 the new per-task cost and c_2 the old per-task cost. Assume $c_2 > c_1$; reuse pays after enough comparable tasks.

The threshold ignores maintenance and distribution shift; these can reverse the apparent saving.

The four examples after changing representation

Problem	Useful change	Required check
Cap set	Affine normalization	No desired orbit removed
Program	Learned library	Correct semantics and coverage
Molecule	Geometry-aware encoding	Function still validates
Mechanism	Causal variables	Interventions retain meaning

Each gain must be paired with a coverage or validation question.

What a convincing representation result measures

Compare success at fixed total budget and fixed validation standard. Report the cost of acquiring the representation and the objects it cannot express.

Ablations can test whether improvement comes from constraints, learned priors or reusable components.

Open question: a useful geometry of discovery

Can measurable properties of a representation predict search performance across genuinely new problem families?

Candidate descriptors include local improvement probability, feasible-region connectivity, reusable substructure and evaluator smoothness. Their joint predictive value is not established.

Conclusion: structure must match the target

Representations can eliminate redundancy, enforce constraints and expose reusable structure. They can also hide the scientific distinction that matters.

Next: how evaluation and experimental feedback turn a representation into an adaptive discovery process.

References

Sumit Gulwani, Oleksandr Polozov and Rishabh Singh (2017). Program Synthesis. Foundations and Trends in Programming Languages 4, 1–119; survey monograph

Kevin Ellis et al. (2021). DreamCoder: Bootstrapping Inductive Program Synthesis with Wake-Sleep Library Learning. PLDI 2021; conference paper; preprint has earlier title

Joseph L. Watson et al. (2023). De novo design of protein structure and function with RFdiffusion. Nature 620, 1089–1100; journal article

Bernardino Romera-Paredes et al. (2024). Mathematical discoveries from program search with large language models. Nature 625, 468–475; online 14 December 2023

References · continued

Jorma Rissanen (1978). Modeling by shortest data description. Automatica 14, 465–471; journal article

Andrey N. Kolmogorov (1965). Three approaches to the quantitative definition of information. Problems of Information Transmission 1, 1–7; journal article

Quentin Berthet and Philippe Rigollet (2013). Computational Lower Bounds for Sparse PCA. arXiv:1304.0828v2, 26 April 2013; primary preprint as inspected

Víctor García Satorras, Emiel Hoogeboom and Max Welling (2021). E(n) Equivariant Graph Neural Networks. ICML 2021, PMLR 139, 9323–9332; conference paper

References · continued

Benjamin Fry, Kaia Slaw and Nicholas F. Polizzi (2026). Zero-shot design of drug-binding proteins via neural iterative selection–expansion. Nature 656, 237–249; online 24 June 2026

S. E. Luria and M. Delbrück (1943). Mutations of Bacteria from Virus Sensitivity to Virus Resistance. Genetics 28(6), 491–511; published primary research

Claude E. Shannon (1948). A Mathematical Theory of Communication. Bell System Technical Journal 27, 379–423, 623–656; journal article