ISSN: 2560-5429 Journal of Applied Technical and Educational Sciences http://jates.org/
Vol. 16, No. 1: 430 https://doi.org/10.24368/jates430 1-35

From Drawing to Construction: Automated Drag-Test Diagnostics for ICT-Supported Geometry Teaching

Fidan Sadigli [1*]

[1*] Baku State University, Faculty of Mechanics and Mathematics, Department of Higher Mathematics and Methods of Teaching Mathematics, Z. Khalilov str. 23, AZ1148, Baku, Azerbaijan, fidan.sarmasli@bsu.edu.az

ORCID: 0009-0009-7838-7376

Abstract

Dynamic geometry environments are standard equipment in school geometry, yet the property that gives them their didactic value — that a genuine construction survives dragging while a figure drawn by eye does not — is still checked by hand, one file at a time, by a teacher who moves a point and looks. This study asks whether that judgement can be automated in a form that is diagnostic rather than merely correct-or-wrong. A learner construction is modelled as a directed acyclic graph over free parameters and the task goal as a conjunction of atomic predicates, identities with dimensionless residuals and strict order conditions with signed margins; the given points of the task and the free parameters on which the goal depends are randomly re-sampled under a local and a global regime, and the resulting violation profile is mapped to a not-satisfied outcome and four diagnostic categories. The central design move is to separate identity goals, which a relation that has not been established violates almost everywhere, from order and side goals, which define open regions and are only locally stable. On a purpose-built benchmark of twelve construction tasks and thirty-eight generated solution variants, the engine reproduced every design label at its default operating point and, across twenty random seeds, from twelve drag samples per regime onwards. It also reproduced the analytically derived outcomes of a suite of thirty-six negative, near-miss, boundary and degenerate submissions, and a report took 0.5 ms on average on one processor core. A single randomly directed global drag failed to expose 9.6% of the non-robust variants; this describes one random drag, not the performance of a teacher. Because the benchmark and the suite were developed alongside the classification rule, these results establish internal consistency rather than diagnostic accuracy on authentic learner work, and the sample budget is an empirical property of this benchmark, not a guarantee for unseen constructions. The contribution is a computational diagnostic prototype with potential formative-assessment applications, together with an explicit account of what its sample budget, tolerance and precision requirements do and do not guarantee.

Keywords: dynamic geometry environments; drag test; robust construction; automated formative assessment; geometry education; ICT in mathematics teaching

1. Introduction

Dynamic geometry environments (DGE) have been part of ordinary school practice for three decades, and the quantitative case for them is settled well enough. A meta-analysis of quasi-experimental studies reported a standardised mean difference of 1.02 (95% CI 0.56–1.48) for dynamic-geometry instruction over conventional teaching across nine studies and 587

License: CC BY 4.0

participants (Chan & Leung, 2014), and a recent three-level meta-analysis of dynamic mathematics software confirmed a positive but strongly heterogeneous effect (Ji, Guo, & Song, 2024). What these aggregate figures do not settle is the question that decides whether a particular lesson has worked: what exactly has a given learner established when the figure on the screen looks right?

The distinction at stake is old and precise. A drawing satisfies the target property in the position in which it was made; a construction satisfies it because of the relations that generated it, and therefore keeps satisfying it when the free points are moved. Dragging is what makes the difference visible, which is why it has been the central object of study in the field since Hölzl (1996), and why Laborde (2005) named the two poles robust and soft constructions. Learners readily produce figures that look like the answer and are not (Healy & Hoyles, 2001), and the classification of dragging modalities that followed — wandering, maintaining, dragging test (Arzarello, Olivero, Paola, & Robutti, 2002; Baccaglini-Frank & Mariotti, 2010) — is in effect a taxonomy of how expert users interrogate a figure that a novice merely admires.

The practical consequence is a bottleneck. To find out whether a class of thirty has constructed or drawn, a teacher must open thirty files and drag in each of them, knowing in advance which point to drag and how far. That is neither scalable nor systematic, and an unguided check is less complete than it appears: as the results below show, a single drag in a random direction leaves a measurable fraction of defective constructions undetected. Moreover, the information a teacher actually needs — which part of the target property the learner secured and which part is still decoration — is not produced at all by a pass-or-fail glance.

Automated reasoning offers one route out. GeoGebra has shipped symbolic automated reasoning tools for a decade (Botana et al., 2015; Kovács, Recio, & Vélez, 2018), and neuro-symbolic provers now solve olympiad geometry at medallist level (Trinh, Wu, Le, He, & Luong, 2024). These systems answer the question “is this statement true of this configuration?” with certainty. They are, however, oriented towards proof rather than towards diagnosis: elimination is computationally heavy, the target statement must be expressible in the prover’s language, and a negative verdict tells a teacher that something is wrong without saying what the learner nevertheless achieved. Formative-feedback research is unambiguous that the useful information lies precisely there, at the level of the process and of the specific gap (Shute, 2008), and that tutoring systems derive their effect from the granularity of the step they can address (VanLehn, 2011).

This paper addresses that gap. It reports the design and computational evaluation of DRACE (Drag-Robustness Analysis and Classification Engine), a lightweight numerical engine that

License: CC BY 4.0

replaces the single drag by systematic random re-sampling of the free parameters — many drags rather than one, but a finite number of them, so that the test is probabilistic and never exhaustive — and returns a structured diagnosis rather than a verdict. The study is the first cycle of a design-based research programme: the artefact and its computational characterisation are the contribution; claims about learners, teachers and learning outcomes are deliberately not made, and the classroom study that must follow is specified rather than claimed.

2. Background and literature review

2.1. Drawing, figure and the epistemic role of dragging

Hölzl (1996) showed that the drag mode is not a neutral viewing device but a constituent of the geometry the learner meets on the screen, and that its introduction changes what counts as a correct answer. Laborde (2001) traced how task design had to be rebuilt around that fact, and later (Laborde, 2005) formulated the robust/soft opposition that organises the present work: a robust construction realises the target property as a consequence of the construction relations, a soft one realises it through the placement of a point that the user adjusts. Mariotti (2000) had already argued that the environment mediates the meaning of proof itself, since the invariance a learner observes under dragging is the empirical shadow of the theoretical necessity a proof would establish.

The literature then turned to how invariance is perceived. Arzarello et al. (2002) distinguished dragging modalities by the cognitive work they perform; Baccaglini-Frank and Mariotti (2010) modelled maintaining dragging as an instrument of conjecture generation; Leung (2008) read dragging through variation theory; and Leung, Baccaglini-Frank and Mariotti (2013) analysed the discernment of invariants as a layered process in which some invariants are directly induced by the construction and others are consequences of it. The survey of Sinclair et al. (2016) confirms that the relation between dragging and proving remains one of the field's principal threads.

Two implications matter here. First, robustness is a property of the construction and not of the picture, and it is therefore in principle checkable by machine. Second, robustness is not binary in the way a grade is: a learner who produces a rhombus when asked for a square has secured the metric relations and missed the angular ones, which is a different didactic situation from a learner who has secured nothing. The classification developed below is designed to preserve that difference.

License: CC BY 4.0

2.2. Automated reasoning in dynamic geometry environments

The symbolic line of work embeds algebraic-geometry provers in the learning environment. Botana et al. (2015) describe the integration of several provers into GeoGebra; Kovács, Recio and Vélez (2018) show how the resulting commands can be used in the teaching and learning of proving, and also document how easily an intuitive use of them produces unexpected output. The approach is mathematically decisive — a Gröbner-basis certificate settles the statement — and the classroom scenarios it opens are genuinely new. Its cost profile and output format, however, are those of a prover: elimination is expensive, degenerate branches must be excluded by explicit non-degeneracy conditions, and the answer is a truth value about a stated proposition. Trinh et al. (2024) push the proving frontier further with a neuro-symbolic system, but the target is still the proof of a given statement rather than the diagnosis of a learner artefact.

Numerical checking sits on the other side. It is already used inside GeoGebra as a pre-filter before symbolic elimination is attempted, and its epistemic status is well understood: it cannot certify truth, but for relations expressed as identities it can refute with probability one, because an identity that fails somewhere on a connected domain fails almost everywhere on it (made precise for the constructions used here in Section 4.5). The present work takes that observation as a design principle rather than as an optimisation, and asks what a purely numerical instrument can deliver when the aim is not a proof but an interpretable account of what a learner has established. The natural deployment target is the browser-resident dynamic-geometry kernel opened up by libraries such as JSXGraph (Gerhäuser, Valentin, & Wassermann, 2010).

2.3. Feedback, van Hiele levels and the teacher

Hattie and Timperley (2007) locate effective feedback at the level of the task process and of self-regulation rather than of the person, and Shute (2008) shows that elaborated feedback specifying what to change outperforms verification alone. VanLehn (2011) attributes the effectiveness of intelligent tutoring systems to the granularity of the step at which they can intervene. A diagnosis that names the failing sub-property of a construction is exactly a process-level, step-granular, elaborated feedback signal, which is why the mapping from diagnostic category to feedback prompt is treated below as part of the design rather than as an afterthought.

The van Hiele levels (van Hiele, 1986) supply the interpretive frame, in the form in which Battista (2007) restates them for research on geometric and spatial thinking. A learner working at the visual level judges a figure by its appearance; at the analysis level, properties are attended to individually but not yet organised; at the informal-deduction level, properties are ordered and one is seen to follow from another. The diagnostic categories introduced below were designed

License: CC BY 4.0

with that progression in mind, and Table 2 proposes a hypothesised alignment between them and the levels, so that a single report could serve both as feedback to the learner and as one piece of evidence for the teacher's own assessment — a technological-pedagogical-content-knowledge concern in the sense of Mishra and Koehler (2006). The alignment is a theoretically motivated interpretation of one artefact, not a classification of the learner: a single construction cannot establish the level at which a learner reasons, and the alignment has yet to be validated against learner data (Section 6.5).

3. Research goal and research questions

The goal of the study is to design an instrument that performs the drag test automatically, with many randomly chosen drags in place of one, and that reports not only whether a learner construction is correct but which parts of the task goal it establishes. Four questions guide the computational evaluation:

RQ1. Can a purely numerical procedure reproduce the mathematical status of learner-typical construction variants whose status is known exactly, and does it distinguish partially correct work from work that is merely drawn?

RQ2. How many drag samples are required, and does the required budget differ between goal properties expressed as algebraic identities and those expressed as order or side conditions?

RQ3. How sensitive is the diagnosis to the parameters of the procedure — the residual tolerance, the local drag radius and the stability threshold — and to the finite precision of stored coordinates?

RQ4. Is the computational cost compatible with real-time use during a lesson, and how often does a single drag in a random direction fail to expose a defective construction?

4. Materials and methods

4.1. Design approach

The study follows the first cycle of a design-based research programme (Wang & Hannafin, 2005): an artefact is specified, implemented and evaluated against criteria that can be tested without human participants, so that the classroom cycle begins from an instrument whose internal behaviour is already characterised. All evaluation reported here is computational. No learner data were collected and none are simulated as though they had been: the benchmark consists of solution variants generated programmatically, whose mathematical status follows from the algebra of the variant itself and is therefore known exactly rather than assigned by a rater. The benchmark and the validation suite were, however, written by the author alongside

License: CC BY 4.0

the classification rule; what this implies for the interpretation of the results is discussed in Section 6.4.

4.2. Representing a learner construction

A construction is a directed acyclic graph. Its sources are free parameters: free points $f = (f_1, \ldots, f_k)$, which a user may drag, and free scalars $t = (t_1, \ldots, t_l)$, which parameterise semi-free points constrained to a line or a circle. Some free points are given by the task — the given points, shared by every submission — and the others are introduced by the learner, as points placed by eye or as auxiliary points. The remaining nodes are produced by straightedge-and-compass primitives: line and circle through given points, midpoint, perpendicular and parallel line, perpendicular bisector, angle bisector, foot of a perpendicular, quarter turn, translation, and the three intersection operators (line–line, line–circle, circle–circle). Intersection operators that admit two solutions take an explicit branch rule, because a learner who builds on an index-selected branch has built something whose behaviour under dragging differs from the same figure built on a determined branch — a difference the engine is designed to expose.

Every primitive raises a degeneracy exception when its output is undefined at the current sample: a line through coincident points, a circle of vanishing radius, the intersection of parallel lines, a line that misses or touches a circle. Two cases are distinguished, because they mean different things. A task-domain degeneracy occurs when an object of the task data — the givens shared by every solution, together with the non-degeneracy conditions under which the task is posed, such as a segment of positive length, a triangle whose vertices are not collinear or a line that cuts a circle in two points — is undefined; the sample then lies outside the domain on which the task is posed, no construction can be judged there, and the sample is rejected and counted. If the submitted figure itself lies outside that domain, the engine returns TASK UNDEFINED and makes no diagnosis. A construction failure occurs when the task data are defined but an object of the learner construction is not; the task is still meaningful there, so the failure is a genuine lack of robustness. It violates an implicit existence condition of the construction — a non-vanishing denominator, a positive discriminant — which is a strict inequality and is therefore counted with the order conditions of Section 4.3; the conjuncts that refer to the undefined object cannot be evaluated at that sample and are not. Treating the first case as a violation would penalise a learner for configurations in which the task has no answer; treating the second as a rejection would hide a real defect, and scoring it as a failure of every identity on the undefined object would report a construction that is correct wherever it exists as a figure drawn by eye.

License: CC BY 4.0

4.3. Goal predicates, residuals and margins

A task goal is a conjunction of atomic predicates $g_1, \ldots, g_m$. Each identity predicate carries a residual $r_j \geq 0$ that vanishes exactly when it holds, and each order predicate a signed margin w_j that is positive exactly when its strict relation holds and zero on its boundary. Quantities that would otherwise carry the dimension of a length are divided by a configuration scale (1),

$$s(f) = \max_{i,j \in G} |f_i - f_j| , \tag{1}$$

where G is the set of all given points of the task. G is fixed by the task statement, so s is a property of the task figure and not of the submission: it depends neither on the free points that the learner introduces nor on the dependency graph of the construction, that is, on the given points from which an object named in the goal happens to be constructed. The scale is evaluated at every sample, so that every residual and margin is dimensionless and a single tolerance ε governs all tasks at all scales. This is not a cosmetic choice: without it the same tolerance would be strict for a figure drawn small and permissive for the same figure drawn large, so that one tolerance could not serve every task and verdicts would not be comparable across learners. Because s is fixed by the task, all submissions of a task are measured with the same yardstick: two submissions that produce the same figure receive the same handed-in values and the same handed-in outcome, however they were constructed, and a point added far away cannot shrink the residual of a near miss below the tolerance (Section 5.8). A goal that concerns a small part of a large task figure is accordingly judged at the precision of the whole figure, which is the precision at which a teacher who drags that figure perceives it. The predicates implemented are listed in Table 1, where u and v denote unit direction vectors of the lines involved, × the scalar cross product and · the scalar product. Lengths are divided by s, and the residuals of perp and par and the margin of orient are formed from unit vectors, so that every residual and margin is a dimensionless number.

Numerically, an identity conjunct is satisfied when $r_j \leq \varepsilon$, and an order conjunct only when $w_j > \varepsilon$. The tolerance thus acts as one resolution for both families: a residual that cannot be told from zero at resolution ε counts as zero, and a strict relation counts as established only when its margin can be told from zero. A zero margin — collinear points for orient, a point on the line for sameside, a point at an endpoint for within — is therefore always a violation, and so is any margin within ε of zero; unit tests in the supplementary archive check this for every predicate on interior, boundary, near-boundary and degenerate configurations. between(P, A, B), “P lies on the segment AB, strictly between A and B”, is not atomic: it is the conjunction of the identity onl(P, AB) and the order condition within(P, A, B), which concerns only the position of the

License: CC BY 4.0

projection of P along AB. The engine expands it into these two conjuncts, so that collinearity is judged as an identity and the position along the segment as an order condition; an order test on the projection alone would accept a point that is not on the segment at all.

Table 1. Atomic goal predicates, their residuals (identities) or margins (strict order conditions) and their type; between is the conjunction of two of them.

Predicate	Meaning	Residual (identity) or margin (order)	Type
onl(P, ℓ)	P lies on the line ℓ = AB	$\lvert u \times (P - A)\rvert / s$	identity
eqlen(A,B,C,D)	|AB| = |CD|	$\lvert\,\lvert AB\rvert - \lvert CD\rvert\,\rvert / s$	identity
perp(ℓ_1, ℓ_2)	the two lines are perpendicular	$\lvert u_1 \cdot u_2 \rvert$	identity
par(ℓ_1, ℓ_2)	the two lines are parallel	$\lvert u_1 \times u_2 \rvert$	identity
oncircle(P, k)	P lies on the circle k(O, r)	$\lvert\,\lvert P - O\rvert - r\,\rvert / s$	identity
eqdistl(P, ℓ_1, ℓ_2)	P is equidistant from two lines	$\lvert\, d(P,\ell_1) - d(P,\ell_2)\,\rvert / s$	identity
midp(M,A,B)	M is the midpoint of AB	$\lvert M - (A+B)/2 \rvert / s$	identity
within(P,A,B)	the projection of P onto AB lies strictly between A and B	$\min(\lambda, 1-\lambda)$, with $\lambda = (P-A)\cdot(B-A) / \lvert B-A\rvert^2$	order
orient(A,B,X)	A, B, X are strictly positively oriented	$u \times v$	order
sameside(P, ℓ, Q)	P and Q lie strictly on the same side of ℓ	$\mathrm{sgn}(\sigma(P)\sigma(Q)) \cdot \min(\lvert\sigma(P)\rvert, \lvert\sigma(Q)\rvert)$, with $\sigma(\cdot)$ the signed distance to ℓ divided by s	order
between(P,A,B)	P lies on the segment AB, strictly between A and B	compound: onl(P, AB) and within(P, A, B)	identity + order

4.4. Two re-sampling regimes

The engine first checks that the task data are defined at the configuration in which the construction was handed in — if they are not, the outcome is TASK UNDEFINED and no diagnosis is made — and that the goal holds there; if it does not, the outcome is NOT SATISFIED (Section 4.6). Otherwise the given points of the task and the free parameters of the submission on which the goal depends — its support — are re-sampled under each of two regimes until N samples have been accepted in each; the free points re-sampled form the set S. The given points are always re-sampled, because they are what a teacher drags and because the scale (1) is computed from them; free parameters of the submission outside the support are held fixed, since they affect neither an object named in the goal nor the scale. The local regime (2) perturbs each free point of S by a small displacement proportional to the configuration scale, while the global regime (3) discards the handed-in position altogether and samples uniformly from a square window centred on the centroid c of the given points G:

(2) $$f_i^{(loc,n)} = f_i + \delta\, s(f)\, u_i^{(n)}\,, \quad u_i^{(n)} \sim N(0, I_2)\,, \quad i \in S\,,$$

(3) $$f_i^{(glob,n)} \sim U(\Omega)\,, \quad \Omega = c + s(f)\cdot[-1, 1]^2\,, \quad i \in S\,.$$

Free scalars are perturbed correspondingly: in the local regime by δ times their admissible range, in the global regime uniformly over that range. For each regime ρ, the violation rate of each atomic conjunct g_j and that of the existence condition are given by (4),

License: CC BY 4.0

(4) $$v_j^{(\rho)} = |\, V_j^{(\rho)} \,| \,/\, |\, D_j^{(\rho)} \,|\,, \quad e^{(\rho)} = |\, U^{(\rho)} \,| \,/\, N_\rho\,,$$

where $D_j^{(\rho)}$ is the set of accepted samples of regime ρ at which the objects named in g_j are defined, $V_j^{(\rho)} \subseteq D_j^{(\rho)}$ the subset at which g_j is violated (an identity residual above ε, an order margin not above ε), $U^{(\rho)}$ the set of accepted samples at which an object named in the goal is undefined, and N_ρ the number of accepted samples; $e^{(\rho)}$ is the violation rate of the existence condition. A sample is rejected only in the case of a task-domain degeneracy (Section 4.2); construction failures are accepted. Sampling continues until N samples have been accepted, up to a cap of 20N draws; the cap was never reached in any run reported here, so $N_\rho = N$ throughout, and rejections cost draws without changing the denominator; no construction failure occurred on the benchmark, so there $|\, D_j^{(\rho)} \,| = N$ as well. A conjunct is called empirically stable in a regime when its violation rate does not exceed a threshold $\tau = 0.05$, and the construction stably defined when $e^{(\rho)}$ does not. The term is chosen deliberately: a stable conjunct has not been shown to be invariant in the mathematical sense, only not to fail in more than a fraction τ of the sampled configurations. For identity conjuncts the choice of τ is immaterial in practice, because by Proposition 1 a conjunct that has not been established is violated almost everywhere; τ matters for order conjuncts and for the existence condition, and its role and the sensitivity of the results to it are examined in Sections 4.5 and 5.3.

4.5. Separating identities from order conditions

Pooling the two families of predicates is the natural first design, and it is wrong. The reason is measure-theoretic, and it also explains the sample budget reported in Section 5.2.

Proposition 1. Let D be a connected open subset of the space of free parameters (f, t) on which every primitive of the construction is defined and every index-selected intersection keeps a strict order between its two points, and let g_j be an identity conjunct whose residual vanishes on D exactly where a real-analytic function q_j vanishes. If q_j does not vanish identically on D, its zero set has Lebesgue measure zero in D; consequently, for samples drawn from any distribution on D that is absolutely continuous with respect to Lebesgue measure, g_j is violated with probability one.

The measure-theoretic fact is that the zero set of a real-analytic function that does not vanish identically on a connected open set is null (Mityagin, 2020). The proposition is formulated for analytic rather than rational maps because the construction operators of DRACE are not all rational. On D, the line through two points, the line–line intersection, midpoint, perpendicular, parallel, perpendicular bisector, foot of a perpendicular, quarter turn, translation and a semi-free point on a line are rational in the free parameters; the line–circle and circle–circle

License: CC BY 4.0

intersections add square roots of discriminants that are strictly positive on D (a vanishing discriminant is a tangency and lies on the boundary of D); the angle bisector and the circle through a point add Euclidean norms of non-zero vectors; and a semi-free point on a circle adds the cosine and sine of its parameter. All of these are real-analytic where they are defined, and so is their composition. The identity residuals of Table 1 are absolute values of analytic functions on D (cross and dot products of unit directions, differences of distances); taking the absolute value does not change the zero set, and for eqdistl, $|d_1| - |d_2|$ vanishes exactly where the analytic product $(d_1 - d_2)(d_1 + d_2)$ does. Every primitive and residual in Table 1 is therefore covered. Because identity rates are taken only over samples at which the learner construction is defined (Section 4.4), they are estimated on exactly the set on which the proposition is stated.

Two qualifications matter. First, the configuration space on which a construction is evaluated is in general the union of several components of this kind, separated by the null set on which an index-selected branch changes order and by the regions in which a primitive is undefined. An identity can hold identically on one component and fail on another — for example when an index-selected branch swaps and the construction switches to a different intersection point — and its violation probability under global sampling is then the probability mass of the components on which it fails, not one. Second, in floating-point arithmetic with tolerance ε the false-negative probability is bounded by the measure of the ε-tube around the zero set, which is the quantity probed empirically in Sections 5.3 and 5.5. The didactic reading is simpler than the mathematics: an identity that a learner has not actually established fails at almost every position of the free points, so a single global drag exposes it almost surely.

Order and side predicates behave in the opposite way, and Proposition 1 does not apply to them. They define open semi-algebraic sets, and a sample taken in the interior of such a set remains inside it under any sufficiently small perturbation. A point placed by eye on the segment AB therefore satisfies within(P, A, B) under every local drag, however arbitrary its placement was, although the same drags move it off the line and violate onl(P, AB). If the two families are pooled, that accidental local stability is counted as an established relation and a figure drawn entirely by eye is reported as partially correct — an error that is not merely cosmetic, since it would tell the learner that something has been secured when nothing has. The engine therefore evaluates the two families separately: identities decide what the learner established, order conditions decide whether what was established is bound to the configuration. The existence of a learner construction is a condition of the same kind — a strict inequality on a denominator or a discriminant — and is evaluated with the order family.

License: CC BY 4.0

The two families also need different sample budgets. For an identity, one global sample suffices almost surely. For an order conjunct whose violating region has probability p under the global regime, the number K of violating samples among the N accepted ones is binomial, and the conjunct is reported as not stable when $K > \tau N$, which happens with probability $D(N, p) = P(\mathrm{Bin}(N, p) > \tau N)$. Three properties of D matter, and they hold only in the sense stated. First, $D(N, p) > 0$ for every finite N and every $p > 0$: no violating fraction, however small, is excluded from detection. While $N < 1/\tau$, that is $N < 20$ at $\tau = 0.05$, the threshold τN is below one, a single violating sample suffices and $D(N, p) = 1 - (1 - p)^N$; with $p = 0.01$ and $N = 12$, for instance, $D = 11.4\%$. Second, τ is a threshold only asymptotically: as N grows, $D(N, p)$ tends to 0 for $p < \tau$, to 1/2 for $p = \tau$ and to 1 for $p > \tau$. At $N = 200$, D is 0.3% for $p = 0.02$, 18.0% for $p = 0.04$, 41.7% for $p = 0.05$ and 99.2% for $p = 0.1$. τ is therefore the violating fraction below which detection becomes improbable as the budget grows, not a fraction that is never reported. Third, because τN is compared with an integer count, D is not monotone in N: the smallest budgets that reach a detection probability of 0.99 are $N = 7$ at $p = 0.5$, $N = 31$ at $p = 0.2$ and $N = 156$ at $p = 0.1$, but at $p = 0.1$ the probability falls below 0.99 again for $N = 160$–169 and 180–182 and stays above it from $N = 183$ onwards. The same model governs the existence condition of a construction that fails on part of the configuration space. The empirical budget reported in Section 5.2 is therefore a property of the violating fractions that occur in the present benchmark, not a sample requirement for unseen constructions.

4.6. Diagnostic classification

Write I for the index set of identity goals and O for that of order goals, and let I* and O* be the subsets that are empirically stable under the global regime, and $e = e^{(\mathrm{glob})}$ the global rate of construction failure. The verdict follows (5) and (6):

(5) *ROBUST if $I^* = I$, $O^* = O$ and $e \leq \tau$; FRAGILE if $I^* = I$ and ($O^* \neq O$ or $e > \tau$) ;*

(6) *PARTIAL if I^* is a non-empty proper subset of I ; VISUAL if I^* is empty .*

Before this rule is applied, the handed-in check of Section 4.4 can return a fifth outcome, NOT SATISFIED, when at least one conjunct already fails in the submitted figure or refers to an object that is undefined there. It is part of the diagnostic framework rather than an error state: it is likely to be the most frequent outcome in authentic classroom data, and the report then names the failing conjuncts, which is itself process-level feedback. If the given points as submitted violate the non-degeneracy conditions of the task, the check returns TASK UNDEFINED instead, an input condition rather than a diagnosis. Two submissions that produce the same figure receive the same outcome from both checks, whatever their

License: CC BY 4.0

constructions, because the residuals are measured against a scale fixed by the task (Section 4.3). The local regime is not used by the rule but is retained in the report, where it distinguishes an order condition that fails only far from the handed-in position — the configuration-bound case — from one that fails immediately, which indicates a figure drawn on the boundary of the intended region. Every task in the benchmark has at least one identity conjunct; for a task posed only through order conditions, $I = I^* = \emptyset$ and the verdict is decided by (5) alone. Table 2 gives the reading of each outcome, a hypothesised alignment with the van Hiele levels and the feedback move attached to it, and Fig. 1 shows the resulting pipeline. The alignment column states what kind of reasoning a submission with that outcome is compatible with; it does not assign a level to the learner, and it is a hypothesis for the classroom cycle rather than a result of this study.

Table 2. Diagnostic outcomes, what each establishes, their hypothesised alignment with the van Hiele levels and the associated feedback move. The alignment is theoretically motivated and has not been validated with learners.

Outcome	**What the submission establishes**	**Hypothesised van Hiele alignment (not validated)**	**Feedback move**
ROBUST	every goal conjunct is empirically stable under global dragging	compatible with informal deduction (level 3): properties are ordered and derived	ask for the reason: which step forces the property, and would another order of steps work?
FRAGILE	all identity conjuncts are stable, but an order or side conjunct, or the existence of the construction, fails in part of the configuration space	compatible with a transition from level 2 to level 3: the figure is right, its configuration dependence is not yet seen	name the configuration: drag until the property fails, then ask why the construction is still correct
PARTIAL	a proper non-empty subset of the identity conjuncts is stable	compatible with analysis (level 2): single properties attended to but not coordinated	report the stable sub-property, then target only the failing one
VISUAL	no identity conjunct survives dragging	compatible with visualisation (level 1): the figure is judged by its appearance	show the drag, then rebuild the same figure from relations rather than positions
NOT SATISFIED	at least one conjunct already fails in the submitted figure; the failing conjuncts are named	none proposed: the submission does not yet realise the target figure	point to the failing conjunct in the submitted figure before any dragging, and ask the learner to repair it

License: CC BY 4.0

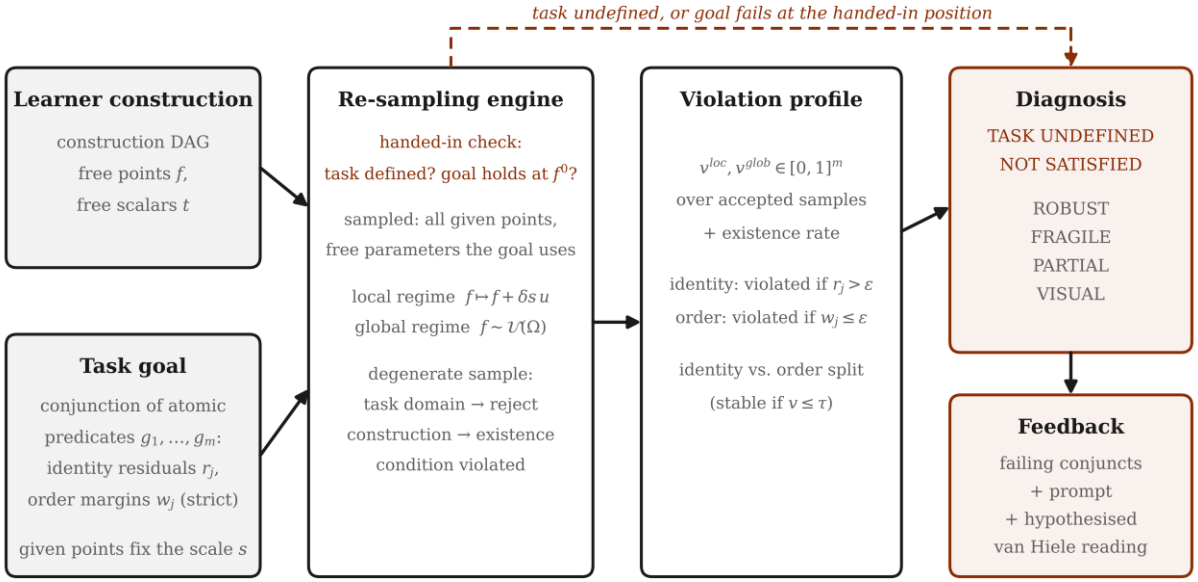

Fig. 1. The DRACE pipeline. The learner construction and the task goal enter separately; a handed-in check returns TASK UNDEFINED when the given objects are degenerate and NOT SATISFIED when the goal already fails; otherwise the given points and the free parameters on which the goal depends are re-sampled, a violation profile is formed over the accepted samples, and the identity/order split turns that profile into a diagnosis and a feedback prompt.

4.7. A worked example

The square on a given side illustrates every element of the design. The goal has six conjuncts: three equal-length relations and three right angles. Fig. 2 shows three submissions that are visually indistinguishable as handed in. After a single global drag, the correct construction is unchanged, the rhombus keeps its three equal-length relations and loses all three right angles, and the figure placed by eye loses everything. The engine reports ROBUST, PARTIAL with the failing conjuncts named, and VISUAL respectively — three different didactic situations that no inspection of the first row can separate. For the rhombus, the report at N = 12 reads: verdict PARTIAL; conjuncts not stable under global dragging: perp(A,B,B,C), perp(B,C,C,D), perp(C,D,D,A); global violation rates 0, 0, 0, 1, 1, 1 in the order of the goal; no rejected samples and no construction failures; feedback: “Report the stable sub-property, then target only the failing one.” This structured record, rather than a grade, is what the engine hands to the learner-facing layer and to the teacher.

License: CC BY 4.0

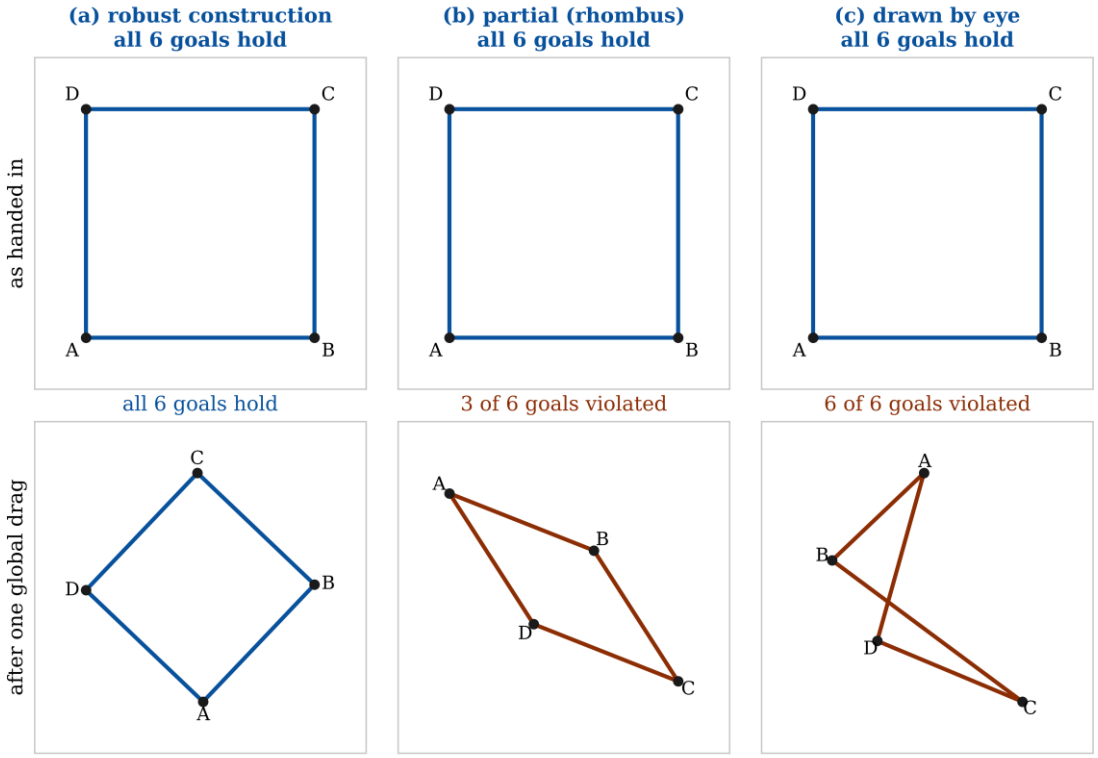

Fig. 2. Three submissions for the task "construct a square on the segment AB". Top row: as handed in, all three satisfy every goal. Bottom row: after one sample from the global regime. Blue indicates that all six goals hold, dark red that at least one is violated.

4.8. The benchmark

Twelve construction tasks from the lower- and upper-secondary curriculum were implemented, each with three or four solution variants, giving thirty-eight instances. The twelve goals have 37 atomic conjuncts between them; since each variant is judged on every conjunct of the goal of its task, the thirty-eight variants carry 122 conjuncts in total, the sum of m over the variants (Table 3). Every variant satisfies its goal exactly at the handed-in configuration — the largest initial residual over the whole benchmark is 2.2×10^{-16}, that is floating-point zero, and the smallest initial order margin is 0.17 — so no instance can be separated by inspecting the submitted figure alone. That is the point of the benchmark: it isolates exactly the discrimination the drag test is supposed to perform. The benchmark was written for this study and is designed to contain every diagnostic category and each documented error type; it is not a sample of learner work, and its role is to test whether the rule does what it was designed to do.

The variants instantiate error types documented in the dynamic-geometry literature. Robust variants are canonical straightedge-and-compass solutions; one task also contains a non-canonical robust variant, so that the engine is required not to penalise a correct solution differing from the reference one. Visual variants place the answer point by eye at its correct location. Partial variants enforce a proper subset of the goal relations — the rhombus and the rectangle

License: CC BY 4.0

for the square, one bisector cut with a median for the incentre, a trapezium for the parallelogram. Fragile variants are correct constructions whose order property is configuration bound (the foot of an altitude leaves the base when the triangle becomes obtuse; the circumcentre leaves the triangle in the same circumstance) or constructions built on an index-selected branch that silently swaps under dragging, so that an internal angle bisector is replaced by the external one. In T10 the goal is that the foot D of the bisector at A lies on the segment BC; it is stated as between(D, B, C) and therefore contains the identity onl(D, BC) as well as the order condition within(D, B, C). Table 3 lists the tasks.

Table 3. Benchmark tasks. R = robust, F = fragile, P = partial, V = visual; m is the number of atomic conjuncts in the goal of the task. The total of m is taken over the thirty-eight variants, each of which is judged on all m conjuncts of its task; the twelve goals themselves have 37 conjuncts.

ID	Task	m	Variants	Design label
T1	Midpoint of a segment	2	3	R, P, V
T2	Perpendicular from a point to a line	2	3	R, P, V
T3	Square on a given side	6	4	R, P, P, V
T4	Circumcentre of a triangle	2	3	R, P, V
T5	Incentre of a triangle	5	4	R, F, P, V
T6	Fourth vertex of a parallelogram	2	3	R, P, V
T7	Foot of the altitude inside the base	3	3	F, P, V
T8	Equilateral triangle above a side	3	3	F, P, V
T9	Right angle in a semicircle	2	3	R, R, V
T10	Bisector foot on the opposite side	3	3	R, F, V
T11	Circumcentre inside the triangle	5	3	F, P, V
T12	Chord midpoint under the centre	2	3	R, P, V
	Total, m summed over the 38 variants	122	38	10 R, 5 F, 11 P, 12 V

Because every benchmark variant satisfies its goal as handed in, the benchmark cannot exercise the NOT SATISFIED outcome, and it contains no configuration on the boundary of an order condition and no degenerate input. A validation suite of thirty-six further submissions to the same twelve tasks was therefore written for these cases (Table 5). It has five groups. Thirteen negative submissions, at least one per task, fail the goal in the submitted figure: a rhombus drawn at 80° for the square, a centroid offered as circumcentre or incentre, the parallelogram ABDC for ABCD, the correct foot of an altitude in an obtuse triangle. Five near misses place a point 10^{-6} s off the required relation; three of them add an unused free point at 10^{2} s, 10^{4} s or 10^{6} s, and one uses an auxiliary point placed 10^{6} s away. Five boundary submissions put the foot of an altitude at either vertex of a right triangle or 10^{-10} and 10^{-6} of the base from a vertex, and the circumcentre of a right triangle on its hypotenuse. Eight degenerate submissions contain coincident or collinear givens, a line that misses or touches the circle of T12, a learner construction that is undefined in the submitted figure, a point at an end of the Thales diameter,

License: CC BY 4.0

where the right angle is undefined, and two constructions with a fixed compass opening that are correct wherever they exist but undefined on an open part of the configuration space. Five submissions for T10, finally, place D on the bisector but off BC (the incentre among them), on BC only through an auxiliary point placed there by eye, on the bisector only because the submitted triangle is isosceles, or on BC outside the segment.

The outcome of every case, and for NOT SATISFIED the conjuncts that must be named, were derived from the geometry of the case and the rules of Sections 4.2–4.6 and recorded before the engine was run on the suite; the two expected rates of construction failure were computed by a Monte Carlo simulation that does not use the engine. The suite therefore tests whether the rules do what they are designed to do in situations the benchmark does not reach, and it contains the configurations on which an earlier version of the engine failed (Section 5.8). Like the benchmark, it was written by the author together with the rules it tests: agreement on it is evidence of internal consistency, not of diagnostic accuracy on learner work (Section 6.4).

A regression set R, added in response to the third review, tests the scale (1), which the benchmark and the suite cannot do: in all of their submissions the goal depends on every given point of its task. R consists of one task whose goal does not name every given point — the midpoint M of BC in a given triangle ABC whose vertex A lies far from BC ($|BC| = 1$, $s = 2.60$) — and four pairs of submissions that produce the same figure, bit for bit, at the handed-in configuration. The first submission of a pair constructs M through A: as the intersection of BC with a line from A through a point placed by eye, or with the diagonal AD of the parallelogram ABDC; the second places M by eye at the coordinates to which the first evaluates. In two pairs M is the midpoint, in one it lies 2×10^{-8} |BC| from it, within the tolerance relative to the task figure, and in one 5×10^{-8} |BC| from it, beyond the tolerance. The two submissions of a pair must receive the same handed-in values and the same handed-in outcome; their diagnoses after dragging may differ, since a constructed point and a point placed by eye behave differently under dragging. In addition, every submission of the benchmark, the suite and R is paired with an exact twin in which each learner point named in the goal is routed through a translation by the zero vector from a given point to itself: the twin produces the same figure at every configuration, but its dependency graph contains one more given point, so the two must receive identical reports. The outcomes of R, too, were derived before the engine was run on it.

4.9. Experimental design, implementation and computational environment

The following experiments were run. (E1) Agreement between the verdict and the design label at the default operating point ($N = 200$, $\varepsilon = 10^{-8}$, $\delta = 0.02$, $\tau = 0.05$), repeated over twenty independent random seeds. (E2) A sample-budget sweep over N from 1 to 400, twenty seeds

License: CC BY 4.0

each, reported overall and by category. (E3) A tolerance sweep over ε from 10^{-14} to 1, a sweep of the local drag radius δ from 0.002 to 0.2 (ten seeds each), and a sweep of the stability threshold over every attainable value τ = k/200 from 0 to 1 (twenty seeds). (E4) Wall-clock cost per report. (E5) The probability that a single drag drawn at random from the global regime fails to expose a non-robust construction, over the twenty-eight non-robust instances with two hundred seeds each, at five tolerances. (E6) The tolerance sweep repeated after every free parameter has been perturbed to a relative precision η, from 10^{-6} to 10^{-2}, under two perturbation models — random jitter and deterministic grid snapping (Section 5.5) — over five seeds at N = 200. (E7) Accounting of rejected samples and construction failures, with an ablation in which rejected samples are scored as violations. (E8) The magnitude of genuine identity and order violations, and the margins of order relations that hold by construction. (E9) An ablation in which identity and order conjuncts are pooled: a conjunct counts as established when it is stable under the local regime, and a submission is reported ROBUST when every conjunct is stable globally, PARTIAL when at least one is established, and VISUAL otherwise. (E10) The violating fraction of each fragile instance, the budget it implies and the fragile agreement curve it predicts. (E11) The size of the global window, with half-width κs for κ = 0.5, 1, 2 and 4 in place of s in (3), twenty seeds. (E12) The validation suite at N = 12 and N = 200 over twenty seeds, scored against the outcome derived for each case and, for NOT SATISFIED, against the conjuncts that must be named. (E13) Invariance: every benchmark and suite submission re-judged, five seeds each, after adding an unused free point at 10^3 s or 10^6 s, an unused point feeding a construction step that the goal does not use, or an unused free scalar, and after scaling the whole figure by 10^{-3} or 10^3 or translating it by 10^3 s; the diagnosis (verdict, failing conjuncts, violation and existence rates, draws, rejections and construction failures) and every other field of the two reports were compared, and the largest change of a handed-in residual and of a handed-in margin was recorded. (E14) Scale independence (Section 4.8): the four pairs of the regression set R, the exact twins of all eighty-two benchmark, suite and regression submissions over five seeds, and, at the handed-in configuration and at fifty global samples of each of the eighty-one submissions whose scale is defined (the given points of suite case D1 coincide), a check that s equals the diameter of all given points of the task, each judged with the present engine and with version 1.2.0, in which s was the diameter of the given points on which the goal depends.

The engine is implemented in Python with NumPy as its only run-time dependency (Matplotlib is used for the figures only), in about 610 lines for the geometry, the predicates and the classification, plus about 240 lines of benchmark definitions, 240 lines for the validation suite

License: CC BY 4.0

and 100 lines for the regression set (version 1.3.0). The results reported here were produced single-threaded on a Linux virtual machine with an Intel Xeon processor (2.30 GHz, two virtual cores) and 8 GB of RAM, with Python 3.13.18 and NumPy 2.5.3. Every number except the timings is platform independent: the results shipped with version 1.1.0 of the archive, produced on a Windows 11 laptop (Intel Core i3-1315U, Python 3.12.10, NumPy 2.5.2), were reproduced bit for bit on a Linux virtual machine with that version of the code, and version 1.3.0 reproduces every number of the run shipped with version 1.2.0 except the timings (Section 5.8). Absolute timings depend on the machine and its load; the figures reported are those of the run whose results are shipped in the archive. Wall-clock time was measured with time.perf_counter around one complete call of the report function (handed-in check, both regimes, all conjuncts); for each budget N and each of the thirty-eight instances, three untimed warm-up reports were followed by ten timed reports with fresh seeds, so each timing figure summarises 380 reports. All random streams are seeded, so every number in Section 5 except the timings is reproduced exactly by the supplementary scripts. The source code, the benchmark and suite definitions, the experiment scripts and the raw and summary result files are provided as a supplementary archive (DRACE_supplementary.zip); its file summary.json contains every number quoted in Section 5, the raw sample bank contains the residual and margin matrices from which most of them are computed, and a verification script recomputes the main results from that bank.

5. Results

5.1. Agreement with the design labels

At the default operating point the engine reproduced all thirty-eight design labels. The confusion matrix (Table 4) is diagonal, and the result was identical for each of the twenty seeds, so the standard deviation of the agreement across seeds is zero. No instance was reported as NOT SATISFIED, confirming that every variant does satisfy its goal in the position in which it was handed in; the outcomes and inputs that the benchmark cannot produce are evaluated on the validation suite (Section 5.8). For the present benchmark this answers RQ1: a purely numerical procedure recovers the mathematical status of every variant, including the separation of partially correct work from work that is merely drawn. Because the benchmark and the rule were developed together, the result shows that the rule is internally consistent — it does what it was designed to do on inputs whose status is known — and not that it is accurate on authentic learner work (Section 6.4). The benchmark is also small: even if its instances were regarded as a sample from a population of submissions, which they are not, 38 correct verdicts out of 38 would only bound the agreement rate below by 90.7% (one-sided 97.5% Clopper–Pearson bound), and 5 out of 5 fragile verdicts by 47.8%.

License: CC BY 4.0

Table 4. Confusion matrix at the default operating point (N = 200, $\varepsilon = 10^{-8}$, $\delta = 0.02$, $\tau = 0.05$); rows are the design labels, columns the verdict.

Design label	ROBUST	FRAGILE	PARTIAL	VISUAL	n
ROBUST	10	0	0	0	10
FRAGILE	0	5	0	0	5
PARTIAL	0	0	11	0	11
VISUAL	0	0	0	12	12

5.2. The required sample budget

The sample-budget sweep connects most directly to practice, because N is the number of drags the engine makes in place of the teacher. Agreement rises from 93.9% (SD 3.0) at N = 1 to 97.6% at N = 2, 98.6% at N = 3, 99.6% at N = 5 and 99.9% at N = 8, and reaches 100% at N = 12, where it remains for every larger budget tested (Fig. 3a). This is an empirical result for these thirty-eight instances and these twenty seeds; it does not establish that N = 12 suffices for unseen constructions or authentic submissions.

The per-category breakdown in Fig. 3b explains the shape of the curve and is consistent with Proposition 1. The robust, partial and visual categories are already classified perfectly at N = 1: they are decided by identity conjuncts, and one global sample suffices almost surely. The entire deficit at small N belongs to the fragile category, which is decided by order conjuncts: agreement there is 54% at N = 1, 82% at N = 2, 89% at N = 3, 97% at N = 5, 99% at N = 8 and 100% from N = 12 onwards. The budget is governed by the violating fractions of the five fragile instances (E10). Because a sample exposes a fragile construction when it violates any of its order conjuncts, the relevant quantity at small N is the fraction of global samples that violate at least one of them; it ranged from 0.33 to 0.73. Applying the rule to N samples resampled from the 8000 pooled samples of each instance predicts a fragile agreement of 48%, 71%, 83%, 94%, 98.3% and 99.7% at N = 1, 2, 3, 5, 8 and 12, and a 99% detection budget between 4 and 12 samples per instance. The observed curve lies somewhat above this prediction at small N (54% against 48% at N = 1, 82% against 71% at N = 2): the twenty seeds supply only one hundred fragile trials, and because the curves for successive N are nested prefixes of the same sample streams, an early excess carries over to larger N. Both curves pass 98% at N = 8 and reach full or nearly full detection at N = 12. A fragile construction with a smaller violating region needs more samples, and one whose violating fraction is below τ is reported with a probability that is positive at every budget but falls towards zero as the budget grows (Section 4.5). This answers RQ2 for the present benchmark.

License: CC BY 4.0

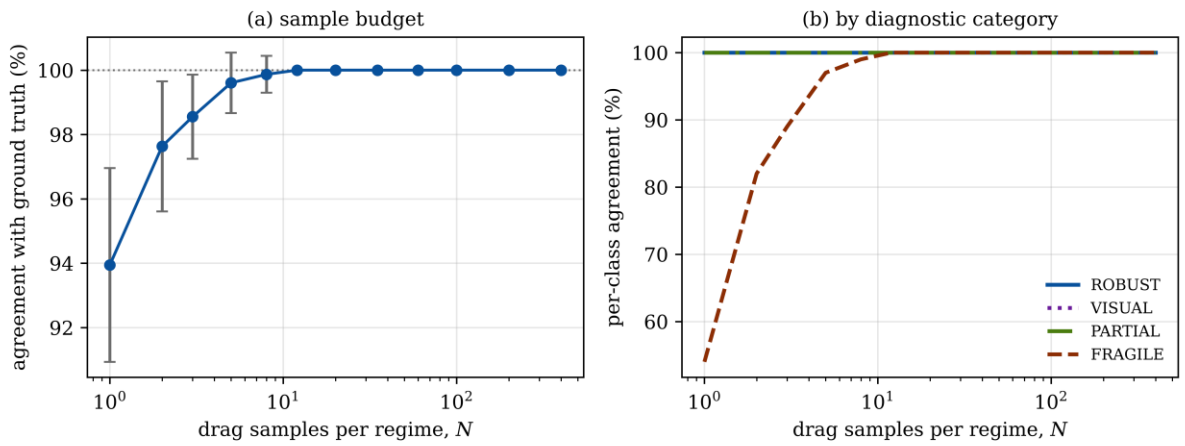

Fig. 3. Effect of the sample budget. (a) Overall agreement with the design labels, mean ± SD over twenty seeds. (b) Agreement by diagnostic category; only the fragile category, which is decided by order conjuncts, requires more than a single sample. The budget at which agreement reaches 100% is specific to this benchmark (Section 4.5).

5.3. Sensitivity to the parameters of the procedure

Agreement was 100% at every tolerance from $\varepsilon = 10^{-14}$ to $\varepsilon = 10^{-3}$ (Fig. 4a). Above $\varepsilon = 10^{-3}$ it begins to fall (99.7% at $\varepsilon = 10^{-2}$, 97.4% at $\varepsilon = 0.1$, 85.0% at $\varepsilon = 0.2$), because a tolerance of that size starts to reject correct order relations whose margin is small and, further up, to absorb genuine identity violations (Section 5.5). The stability threshold behaves similarly (Fig. 4c): agreement was 100% for every attainable τ from 0 to 0.24 at $N = 200$ and fell to 96.8% at $\tau = 0.3$ and 88.6% at $\tau = 0.5$. Conjuncts that hold by construction were never violated in any sample (largest observed rate 0), so on this benchmark $\tau = 0$ would have produced the same verdicts. The default $\tau = 0.05$ is a convention: it leaves a margin against isolated numerical failures near degenerate configurations, which did not occur here, and it sets the violating fraction below which a configuration dependence becomes improbable to detect as N grows (Section 4.5). A user who wants every observed violation to count can set $\tau = 0$; below $N = 20$ the default already behaves in this way, because a single violating sample then exceeds τN. The width of the tolerance plateau should nevertheless be read for what it is: the benchmark constructions are exact, so a satisfied residual is of order 10^{-16}, and most violated ones are of order 10^{-2} to 1. Input taken from a real dynamic-geometry file, where coordinates are rounded and points may be snapped to a grid, will narrow that gap, and ε will then require calibration to the precision of the source. Section 5.5 measures how far.

The local drag radius δ does not enter the verdict at all, because rule (5)–(6) uses only the global regime; the constant 100% agreement over $\delta = 0.002$ to 0.2 in Fig. 4b is therefore a check of the implementation rather than evidence of robustness. What δ does change is the local profile shown to the teacher: the mean local violation rate of the fragile instances' order conjuncts was

License: CC BY 4.0

0% for $\delta \le 0.02$, 1.9% at $\delta = 0.1$ and 10.8% at $\delta = 0.2$, so the default $\delta = 0.02$ keeps the local regime inside the region in which a configuration-bound property still holds, which is what lets the report separate a configuration-bound failure from a figure drawn on the boundary of the intended region.

The size of the global window, by contrast, cannot matter. Every predicate of Table 1 is invariant under translations and uniform scalings, free scalars are scale-free path parameters, and the global regime discards the handed-in position; enlarging the window therefore rescales every sampled configuration without changing the distribution of its shape. E11 confirms this: with the half-width in (3) set to κs for $\kappa = 0.5$, 1, 2 and 4, agreement at $N = 200$ was 100%, 100%, 100%, 100%, agreement at $N = 12$ was 100.0%, 100.0%, 100.0%, 99.9%, and the mean fraction of global samples violating at least one order conjunct of a fragile instance was 0.47, 0.48, 0.47, 0.47, differences that are within sampling noise. The invariance tests of Section 5.8 make the same point for whole figures: scaling a submission by 10^{-3} or 10^{3}, or translating it by 10^{3} s, left every diagnosis unchanged. What does matter is the shape of the sampling distribution — here independent uniform points in a square — because it determines the violating fractions of order conjuncts and hence the budget; that choice is not varied here (Section 6.4). In answer to RQ3, none of the parameters needed tuning per task on this benchmark.

5.4. Computational cost

On the test machine described in Section 4.9, a complete report took on average 0.55 ms at $N = 12$ (median 0.54 ms, maximum 0.97 ms over 380 timed reports), 0.21 ms at $N = 1$, 1.71 ms at $N = 50$, 6.62 ms at $N = 200$ and 32.0 ms at $N = 1000$, scaling linearly in N as expected (Fig. 4d). At the budget at which full agreement was first reached on this benchmark, a class set of thirty submissions is therefore processed in well under a second on a single processor core, which makes in-editor feedback technically feasible; whether feedback at that speed is pedagogically useful is a question for the classroom cycle.

License: CC BY 4.0

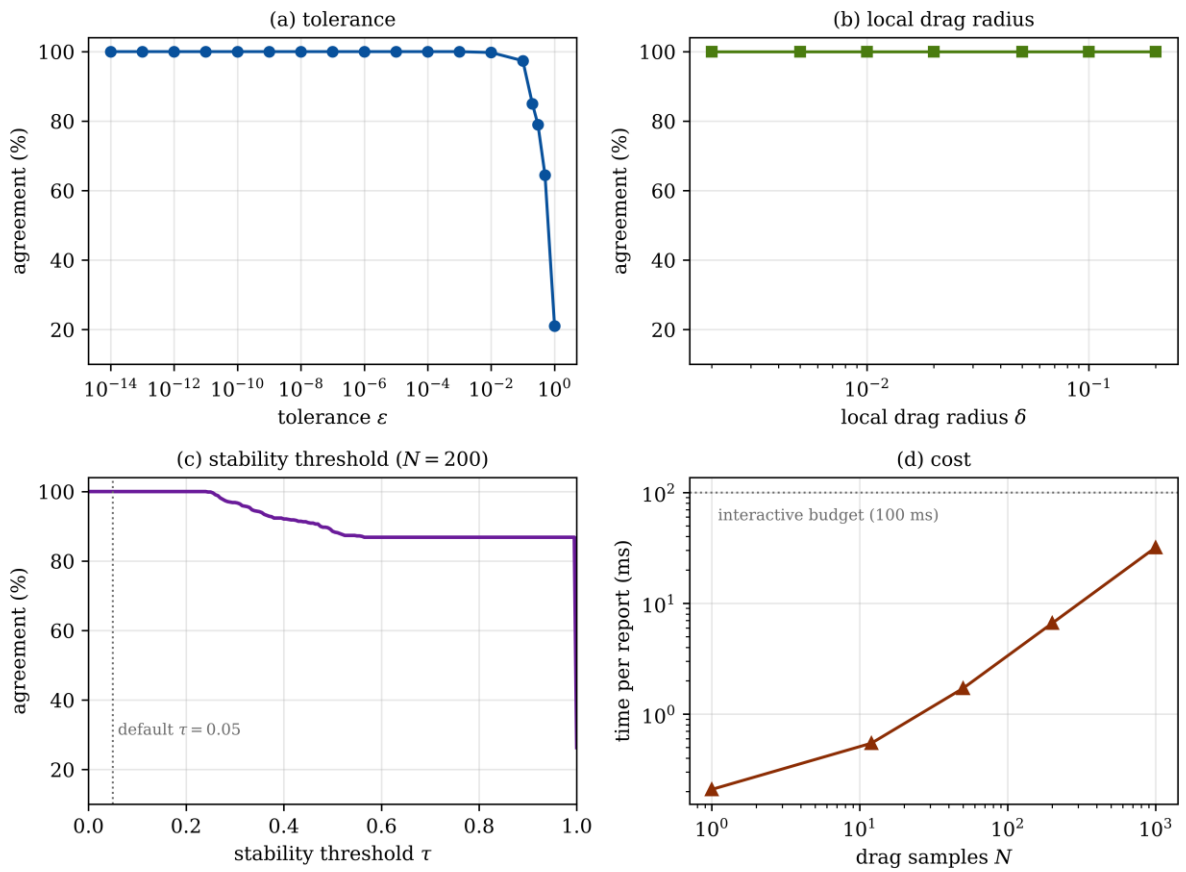

Fig. 4. Parameter sensitivity and cost. (a) Tolerance ε on exact input. (b) Local drag radius δ, an implementation check, since the verdict does not depend on δ by construction. (c) Stability threshold τ at N = 200. (d) Mean wall-clock time per report against the sample budget on the test machine of Section 4.9; the dotted line marks a 100 ms interactive budget.

5.5. Input precision and the operating window

The tolerance plateau of the previous section is an artefact of exact input. To estimate what remains of it when coordinates carry finite precision, every free point and free scalar of every instance was perturbed before the analysis under two models, for η from 10^{-6} to 10^{-2}, over five seeds at N = 200: random jitter, in which each coordinate is displaced uniformly by at most η s, and grid snapping, in which each coordinate is rounded to a randomly offset grid of spacing η s (each free scalar to a grid of spacing η times its range). Grid snapping models a file that stores positions to a fixed number of decimals, or an environment with snap-to-grid enabled. Neither model is an analysis of the files of a particular dynamic geometry environment, which may also store display-rounded values, construction histories or measurement-based steps; how closely the models represent such files will be checked once an importer exists (Section 6.4).

The outcome (Fig. 5a) is sharp. Below ε ≈ η the agreement collapses to 39.5%, and that number is not arbitrary: it is exactly 15 of 38, the robust and fragile instances, whose goals hold for any placement of the free parameters. The remaining twenty-three instances rely on a point or a

License: CC BY 4.0

slider that the learner tuned, so once the stored coordinates are perturbed their goal no longer holds exactly and the engine correctly reports NOT SATISFIED. Under grid snapping the floor is somewhat higher (53% to 56%), because snapping all points to the same lattice leaves some figures placed by eye exactly in their intended relation. Raising ε restores the diagnosis in both models: agreement was 100% from $\varepsilon = 10\eta$ under both models, for every precision tested up to $\eta = 10^{-3}$ (Table 6).

The upper end of the window is set by two effects that a large tolerance has on the two families: it absorbs genuine identity violations, and it rejects correct strict order relations whose margin falls below ε. On this benchmark the second effect binds first. Genuine identity violations under global dragging had median residual 0.47, fifth percentile 0.031 and first percentile 0.0053, so absorbing them requires tolerances well above 10^{-3}; the smallest observed, 1.1×10^{-6}, came from samples close to the boundary of a violating region, which is why the bound is governed by violation rates rather than by the single smallest residual. The margins of order relations that hold by construction, by contrast, had first percentile 0.0024 and were at most 10^{-2} in 3.8% of global samples; for the incentre of T5, whose margin is its inradius relative to the triangle, the share was 4.2%, close enough to τ that one of ten seeds reported the robust construction FRAGILE at $\varepsilon = 10^{-2}$ (agreement 99.7%). On exact input, agreement therefore stayed at 100% up to $\varepsilon = 10^{-3}$. Combining the two bounds gives, for this benchmark and these perturbation models, the operating window $10\eta \le \varepsilon \le 10^{-3}$ shown in Fig. 5b, which is non-empty when stored coordinates are accurate to about one part in 10,000 of the figure diameter. Both bounds are empirical properties of the present thirty-eight instances and of the two perturbation models, not general properties of dynamic geometry files or of the method: a task whose genuine violations are smaller, or whose correct order relations come closer to their boundary, would lower the upper bound, and the lower bound depends on how a given environment rounds or snaps. At one part in 1,000, full agreement was reached only at $\varepsilon = 10^{-2}$ under both models, a value at the edge of the exact-input plateau that should not be relied on; at one part in 100, no tolerance recovered the full diagnosis under either model. This completes the answer to RQ3: on this benchmark the procedure had no parameter that needed tuning per task, but the tolerance must be set once, from the precision of the source, and the window should be re-established for any new task set.

License: CC BY 4.0

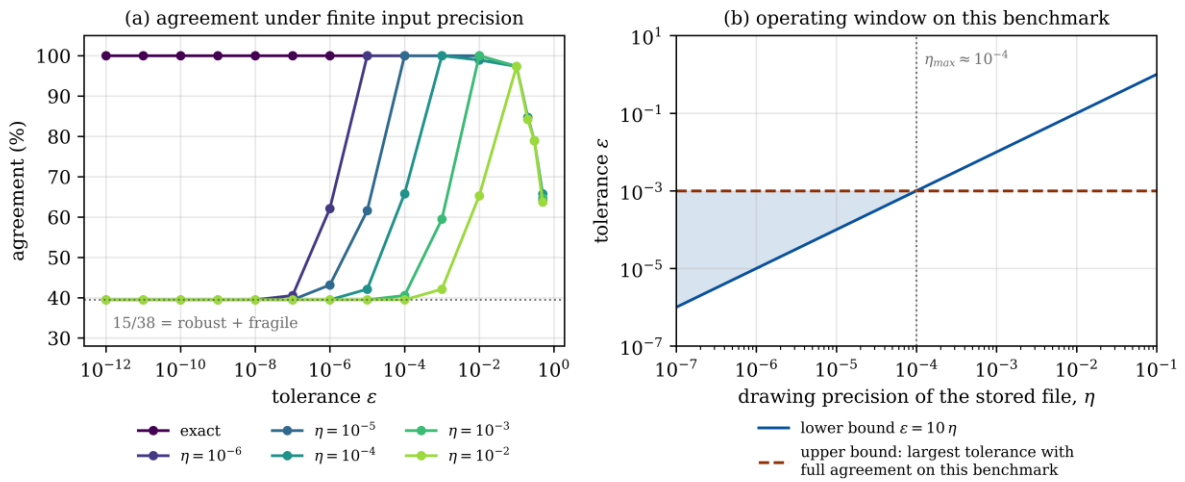

Fig. 5. Behaviour under finite input precision. (a) Agreement against the tolerance for several drawing precisions η under random jitter; the dotted line marks the 15 of 38 instances whose goals hold for any placement. (b) The resulting operating window on this benchmark, bounded below by a multiple of η and above by the largest tolerance with full agreement on exact input; both bounds are benchmark-specific.

5.6. Degeneracy and construction failure

In the global regime 2.4% of draws were rejected as task-domain degeneracies; no local draw was rejected (1.2% over both regimes). The figure is dominated by T12, where the chord is part of the task data: in 23.5% of global draws the line misses the circle, so the task itself is undefined and no construction can be judged. Because sampling continues until N samples have been accepted, these rejections cost draws but do not change the denominator of (4). Scoring them as violations instead would have turned all three T12 variants, including the robust one, into VISUAL. Construction failures — samples at which the task is defined but an object of the learner construction is not — are scored as violations of the existence condition (Section 4.2). None occurred on the present benchmark, in either regime, because its variants use only primitives that remain defined wherever the task data are, apart from sets of measure zero. This is a property of the benchmark, not of the method: authentic submissions are expected to contain such failures, and the validation suite contains two constructions that fail on an open part of the configuration space (Section 5.8). The distinction between an undefined task and a failing construction is the same non-degeneracy issue that symbolic provers must handle explicitly.

5.7. The miss rate of a single random drag

Over the twenty-eight non-robust instances, one drag drawn at random from the global regime returned ROBUST in 9.6% of trials at every $\varepsilon \leq 10^{-4}$ and in 10.1% at $\varepsilon = 10^{-2}$. At the stricter tolerances all misses fall on the five fragile instances, of which 53.8% were missed, close to the 52% expected from the fractions of samples that violate at least one of their order conjuncts

License: CC BY 4.0

(Section 5.2); partial and visual instances were never missed. This figure describes one randomly directed global drag, which is the engine itself at N = 1: it moves all free points at once and anywhere in the window, unlike a manual drag of one point along a path, and it must not be read as an estimate of teacher error. Experienced users drag purposefully: the literature on dragging modalities (Arzarello et al., 2002; Baccaglini-Frank & Mariotti, 2010) describes precisely the strategies by which a skilled user chooses where to drag, and such a user would do better than this figure. What the number shows is narrower: how much a single unguided drag leaves undetected, and that the cases it misses are exactly the configuration-bound ones. Together with the timing figures, this answers RQ4.

5.8. Negative, boundary and degenerate cases

The validation suite (Section 4.8) was judged at N = 200 and at N = 12, over the same twenty seeds as the benchmark (E12). Every one of its thirty-six outcomes was reproduced at N = 200 in all twenty seeds (Table 5). The twenty-seven NOT SATISFIED cases cover all twelve tasks, and in every one of them the report named exactly the conjuncts that fail in the submitted figure — the three right angles of a rhombus drawn at 80°, only the condition at A for a centroid offered as the incentre of a triangle that is isosceles at B, only the incidence with AB for a foot raised off the line along its normal — so the pre-check returns process-level feedback rather than a bare refusal. The near misses, whose point lies 10^{-6} s off the required relation, stayed NOT SATISFIED when an unused point was added at 10^{2} s, 10^{4} s or 10^{6} s and when an auxiliary point that the construction uses was placed 10^{6} s away. The boundary submissions — a foot of an altitude at a vertex of a right triangle or 10^{-10} of the base away from it, a circumcentre on the hypotenuse — were reported NOT SATISFIED with the strict order conjunct named, whereas a foot 10^{-6} of the base inside, a margin above the resolution, was accepted and diagnosed FRAGILE like the corresponding benchmark variant. Coincident or collinear givens and a line that misses or touches the circle of T12 returned TASK UNDEFINED; a construction that is undefined in the submitted figure and a point at an end of the Thales diameter returned NOT SATISFIED; and the two constructions with a fixed compass opening, correct wherever they exist but undefined on 19% and 25% of the global window, were reported FRAGILE, with the existence condition as the failing condition (observed failure rates 0.19 and 0.24 at N = 200). The five T10 cases separate what the corrected goal distinguishes: points on the bisector but off BC, the incentre among them, are NOT SATISFIED in the submitted figure; a point that lies on BC only because an auxiliary point was placed there by eye, and a median foot submitted in an isosceles triangle, are PARTIAL, with the stable sub-property named; the foot of the external bisector is NOT SATISFIED because it lies outside the segment.

License: CC BY 4.0

At N = 12 the agreement was 717 of 720 reports (99.6%). The three exceptions are the two fixed-compass cases, reported ROBUST in two and one of the twenty seeds because none of their twelve global samples fell into the region in which the construction is undefined; the binomial model of Section 4.5 predicts this for $(1 - p)^{12} = 8\%$ and 3% of runs at $p = 0.19$ and 0.25. A construction that fails on part of the configuration space behaves in this respect like any order condition, and the budget it needs is set by the size of that part.

The invariance tests (E13) re-judged all seventy-four benchmark and suite submissions under seven transformations: an unused free point at 10^3 s or 10^6 s, an unused point feeding a construction step that the goal does not use, an unused free scalar, uniform scaling by 10^{-3} and 10^3, and translation by 10^3 s. In all 2590 pairs of reports the diagnosis was unchanged: verdict, failing conjuncts, violation and existence rates, draws, rejections and construction failures were identical. This is invariance of the diagnosis, not numerical equality of every field. The four transformations that add unused objects left every field of all 1480 reports identical. Scaling and translation change the floating-point values of residuals and margins by rounding: 210 of the 1110 reports differed from the original in a margin at the handed-in configuration, and the largest changes were 1.6×10^{-10} for a residual and 2.0×10^{-12} for a margin. No value crossed the tolerance, which is why every diagnosis was unchanged. Judged by the earlier version of the engine (1.1.0, kept in the archive), eighteen of the thirty-six suite cases received another outcome than the one derived for them: the four near misses with a distant point, the four boundary cases at or within 10^{-10} of a boundary, the four degenerate task configurations, both fixed-compass constructions and four of the five T10 cases. The suite thus separates a rule that has the properties claimed in Section 4 from one that does not; like the benchmark, it was written together with the rule it tests, and its agreement is a statement about internal consistency, not about authentic learner work.

The regression test of the scale (E14) documents the correction made in response to the third review. Judged by version 1.2.0, in which s was the diameter of the given points on which the goal depends, three of the four pairs of R received different handed-in values, and in one of them the same figure was VISUAL when M was constructed through A but NOT SATISFIED when M was placed by eye — the inconsistency the review described: the copy placed by eye depends on B and C only, so it was measured against $|BC| = 1$ instead of $s = 2.60$, and its residual of $2 \times 10^{-8}\ |BC|$ exceeded the tolerance. Judged by the present engine, every pair received identical handed-in values and the outcome derived for it: VISUAL for both submissions of the two pairs built from a point placed by eye, ROBUST for the construction through the parallelogram and VISUAL for its copy, and NOT SATISFIED for both

License: CC BY 4.0

submissions of the pair whose point lies beyond the tolerance. All 410 reports of the exact twins (eighty-two submissions, five seeds each) were identical to those of the submissions in every field, against 405 of 410 with version 1.2.0, and s equalled the diameter of all given points at all 4131 configurations examined (eighty-one submissions, fifty-one configurations each). Because the goal of every benchmark and suite submission depends on all given points of its task, the correction changes no other result of this section: the run of version 1.3.0 reproduces every number of the run of version 1.2.0 except the timings. Table 6 collects the operational figures of the whole evaluation.

Table 5. Validation suite: outcomes derived for each case and agreement over twenty seeds.

Group	n	Derived outcomes	What the cases test	Agreement, N = 200 / N = 12
Negative	13	13 NOT SATISFIED	a wrong figure for every task; the failing conjuncts must be named	100.0% / 100.0%
Near miss	5	5 NOT SATISFIED	a point 10^{-6} s off its relation; unused points up to 10^{6} s away, an auxiliary point 10^{6} s away	100.0% / 100.0%
Boundary	5	4 NOT SATISFIED, 1 FRAGILE	feet and a circumcentre on an order boundary or 10^{-10} from it; one foot 10^{-6} inside	100.0% / 100.0%
Degenerate	8	4 TASK UNDEFINED, 2 NOT SATISFIED, 2 FRAGILE	coincident or collinear givens, a line missing or touching the circle, constructions undefined as submitted or on an open set	100.0% / 98.1%
T10 off segment	5	3 NOT SATISFIED, 2 PARTIAL	on the bisector but off BC, on BC or on the bisector only by accident, outside the segment	100.0% / 100.0%
Total	36	27 NOT SATISFIED, 4 TASK UNDEFINED, 3 FRAGILE, 2 PARTIAL		100.0% / 99.6%

Table 6. Operational summary of the computational evaluation.

Quantity	Value	Condition
Agreement with design labels	38 / 38 (100%)	N = 200, twenty seeds, SD = 0
Smallest budget with 100% agreement	N = 12 per regime	this benchmark, twenty seeds (empirical)
Agreement at N = 1	93.9% (SD 3.0)	deficit entirely in the fragile class
Validation suite outcomes reproduced	36 / 36 (100%)	N = 200, twenty seeds; 99.6% at N = 12 (Table 5)
Diagnosis unchanged under invariance tests	2590 / 2590	unused points and scalars, dead steps: every field identical; scaling, translation: values change by rounding only (at most 1.6×10^{-10})
Same figure, different construction (regression set R)	4 / 4 pairs; 410 / 410 twin reports	same handed-in values and outcome; exact twins identical in every field (version 1.2.0: one pair NOT SATISFIED against VISUAL; 405 / 410)
Tolerance plateau	ε from 10^{-14} to 10^{-3}	exact benchmark input

License: CC BY 4.0

Quantity	Value	Condition
Local drag radius	δ from 0.002 to 0.2	verdict independent of δ by design; $\delta = 0.02$ keeps the local profile clean
Global window half-width	κ from 0.5 to 4	verdicts unchanged (scale invariance); agreement at N = 200: 100% for every κ
Stability threshold plateau	τ from 0 to 0.24	N = 200
Time per report	0.55 ms	N = 12, single thread, Intel Xeon processor at 2.30 GHz
Time per report	6.62 ms	N = 200, single thread, same machine
Task-domain samples rejected	2.4% overall, 23.5% in T12	global regime; construction failures: none occurred
Single random global drag misses a defect	9.6%	non-robust instances, $\varepsilon \le 10^{-4}$; not a teacher error rate
Usable tolerance window (this benchmark)	$10\eta \le \varepsilon \le 10^{-3}$	η = relative precision of the stored coordinates
Precision required of the input (this benchmark)	$\eta \le 10^{-4}$	about four significant digits

6. Discussion

6.1. What the instrument adds

The drag test has been the conceptual centre of dynamic-geometry didactics since Hölzl (1996), but as a classroom procedure it has remained manual and unsystematic. The contribution here is not the idea of dragging but its conversion into a procedure with characterised behaviour: an empirically determined sample budget for this benchmark together with a model of how that budget depends on the construction, a measured single-drag miss rate, a measured cost on stated hardware, and an output that is a description rather than a grade. The engine reports which conjunct of the goal survived and which did not, and that is the level at which feedback research locates the effect (Shute, 2008; VanLehn, 2011).

The identity/order separation is the part of the design that is least obvious and most consequential. It is a mathematical fact about two kinds of condition, but it carries a direct didactic meaning: securing a metric relation and securing a configuration are different achievements, and an instrument that conflates them will report a figure drawn by eye as partially correct. On the present benchmark the pooled rule made exactly that error on five of the twelve visual instances.

6.2. Naming the fragile case

The fragile category deserves separate comment because it corresponds to a situation teachers meet constantly and have no word for. A learner constructs the foot of an altitude correctly, sees it inside the base, and takes “inside the base” to be part of what was proved. Dragging the

License: CC BY 4.0

triangle into an obtuse position destroys the appearance while leaving the construction intact. Under a pass-or-fail check this is simply correct and the misconception passes unremarked; under the diagnosis proposed here it is a distinct verdict with a distinct feedback move — drag until the property fails, then ask why the construction is nevertheless still right. This is the discernment of a derived rather than an induced invariant in the sense of Leung et al. (2013), and on the hypothesised alignment of Table 2 it corresponds to the transition from the second to the third van Hiele level (Battista, 2007) — a reading that remains to be tested with learners.

6.3. Relation to symbolic automated reasoning

Nothing here competes with a Gröbner-basis certificate. A numerical drag test cannot prove that a construction is correct; it can only fail to refute it. What Proposition 1 supplies is a refutation guarantee for identity conjuncts at a cost of well under a millisecond per submission on a single processor core. No like-for-like comparison with symbolic elimination was carried out, so no claim is made about relative speed; the case for combining the two rests on their different outputs and guarantees. A natural deployment is to run the numerical diagnosis on every submission continuously, and to invoke a symbolic prover (Botana et al., 2015; Kovács et al., 2018) only where a certificate is actually wanted — typically on the robust verdicts, which are the ones a numerical method cannot establish. The two are complementary layers of a single tool rather than alternatives. Table 7 sets out the division of labour.

Table 7. Division of labour between symbolic automated reasoning as deployed in dynamic geometry and the diagnostic engine proposed here.

Aspect	**Symbolic reasoning tools**	**Drag-robustness diagnosis**
Question answered	is a stated proposition true of this configuration?	which parts of the task goal did this learner establish?
Guarantee	proof, subject to explicit non-degeneracy conditions	for identity conjuncts, refutation with probability one (Proposition 1); for order conjuncts, detection with a probability that tends to one as N grows when the violating fraction exceeds τ and to zero when it is below τ; no certificate of correctness
Output	a truth value, optionally with the conditions of validity	a per-conjunct violation profile and a diagnostic category
Partial achievement	not represented: the proposition holds or it does not	represented directly, as the empirically stable subset of the goal
Configuration dependence	handled as a non-degeneracy condition attached to the statement	reported as a distinct verdict with its own feedback move
Cost per submission	depends on the prover and the construction; not measured in this study	about 0.5 ms at $N = 12$ on one processor core, linear in the number of drag samples
Natural role	certify the constructions the numerical layer calls robust	run continuously on every submission, as the first layer

License: CC BY 4.0

Read this way the numerical layer is not a weaker prover but a different instrument. It is what makes continuous, per-submission diagnosis affordable, and it narrows the set of cases on which the expensive certificate is worth computing.

6.4. Limitations

Seven limitations bound the claims. First, and decisively, there are no classroom data. No classroom study, usability study, learning-outcome assessment or teacher validation has been carried out; the evaluation establishes internal validity — that the rule does what it was designed to do on inputs whose status is known — and DRACE is therefore a computational diagnostic prototype with potential formative-assessment applications, not a validated formative-assessment instrument. Second, the benchmark and the validation suite were designed alongside the classification rule: the identity/order separation was introduced in response to failures observed on the benchmark, and the treatment of between, of the scale, of boundary configurations and of construction failures in response to counterexamples; perfect agreement on cases that informed the design is evidence of coherence, not of generalisation, and an independent corpus of authentic learner constructions is required before the figures reported here may be read as accuracy in any external sense. Third, the sample budget is empirical: full agreement from N = 12 holds for these instances and seeds, a fragile construction with a small violating region needs more samples, and one whose violating fraction is below τ is detected with a probability that, although positive at every budget, falls towards zero as the budget grows (Section 4.5).

Fourth, the precision study rests on two perturbation models, random jitter and grid snapping, rather than on an analysis of the files of actual environments; learner files may also carry display-rounded values and construction histories with measurement-based steps, and the operating window is specific to this benchmark. Fifth, the evaluation procedure is generic once a task has been encoded, but each task must still be written as an explicit conjunction of the predicates of Table 1: DRACE is task-generic within that vocabulary, not task-independent. The vocabulary covers the plane geometry addressed here but not conics, transformations beyond isometries or three-dimensional constructions, and no importer for GeoGebra or JSXGraph files has yet been written, so the present system consumes constructions in its own format. Sixth, the van Hiele alignment of Table 2 is a hypothesis: a single construction cannot establish the level at which a learner reasons, and the alignment must be validated against an independent assessment of learners' levels.

Seventh, the global regime is a sampling model, not a model of continuous dragging: it relocates all free points at once, whereas a learner drags one point along a path. Dynamic geometry

License: CC BY 4.0

environments differ in how they follow the two points of an intersection along such a path, and an environment that tracks them by continuity may keep a branch that DRACE, which orders the points by coordinate, reports as swapping; the index-selected variants of T5, T8 and T10 represent the coordinate-ordered behaviour only, and an importer must reproduce the branch rule of the source environment. The violating fractions of order conjuncts, and with them the sample budget, also depend on the sampling distribution, which here places the free points independently and uniformly in a square; a distribution closer to how learners actually drag would give different fractions. Finally, encoding a task costs effort: the goal must be written once per task, not per submission, and the twelve tasks here needed between two and six conjuncts each, but the authoring burden for teachers has not been studied.

One further property is a design decision rather than a limitation, but it should be stated plainly: the engine certifies the goal and not the method. A robust construction that differs from the reference solution is accepted, as the second robust variant of task T9 demonstrates. That is appropriate for formative use and inappropriate for a task in which a particular construction sequence is itself what is being assessed.

6.5. The next design cycle

The cycle that follows is a classroom study with three questions and a pre-specified analysis. The first is diagnostic validity: whether the verdicts agree with the judgements of experienced teachers on authentic submissions, measured by weighted kappa against two independent raters on a corpus of learner files collected across several classes and tasks, with the sample budget and the tolerance re-established on that corpus. The second is the hypothesised van Hiele alignment: whether the outcomes of Table 2 relate to learners' levels as measured by an independent van Hiele test. The third is instructional effect: whether the category-specific feedback of Table 2 produces greater gains than verification-only feedback, in a design that separates the effect of the diagnosis from the effect of receiving any feedback at all, accompanied by a usability study with teachers. Two extensions are needed first: an importer for the file formats of the major environments, which will also allow the precision of stored files to be measured directly, and a natural-language layer that renders the diagnosis in the learner's own language. The second is where large language models belong in this architecture — not as the judge of the construction, which they are not reliable enough to be, but as the phrasing of a verdict already established by the numerical procedure, with the failing conjunct supplied to the model as a constraint it is not permitted to contradict. How far learners and teachers are equipped to work with such a layer is itself an open question, and instruments for

License: CC BY 4.0

measuring generative-AI literacy in technical education are only now being validated (Peculea, 2026).

7. Conclusion

Dragging is what separates a construction from a drawing, and that separation is the whole didactic content of dynamic geometry. This paper turned the drag test into a computational procedure. Representing a learner construction as a graph over free parameters and a task goal as a conjunction of identities with dimensionless residuals and strict order conditions with signed margins makes the test task-generic within the supported predicate vocabulary; randomised re-sampling under two regimes replaces one drag by many; separating identities from order conditions makes its output a diagnosis rather than a verdict. On a purpose-built benchmark of twelve tasks and thirty-eight variants, whose labels follow from the algebra of each variant, the engine reproduced every label at its default operating point, reached full agreement from 12 drag samples per regime across twenty seeds, was insensitive to its drag radius and stability threshold over wide ranges and to its tolerance up to 10^{-3}, reproduced the derived outcomes of thirty-six negative, near-miss, boundary and degenerate submissions, gave the same diagnosis when points the goal does not depend on were added or the figure was rescaled, measured every submission of a task against the same scale, so that two constructions of the same figure received the same handed-in outcome, and produced a report in under a millisecond on a single processor core. Because the benchmark, the validation suite and the rule were developed together, these results show internal consistency, not accuracy on authentic learner work, and the sample budget is a property of this benchmark rather than a guarantee for unseen constructions.

Two findings carry beyond the artefact. The first is structural: the properties a geometry task asks for fall into two families that behave in opposite ways under dragging, and the distinction is not a technicality but the difference between securing a relation and securing a configuration. The second is operational: multi-sample checking is cheap, whereas one randomly directed global drag failed to expose 9.6% of the non-robust benchmark variants and 54% of the configuration-bound ones. That figure describes a single random drag, not the performance of a teacher, who drags purposefully; what it shows is that the cases an unguided check misses are exactly the configuration-bound ones. The precision study fixes, for this benchmark, the single parameter that must be calibrated and the condition under which it can be set. What remains is empirical validation: the categories, the hypothesised van Hiele alignment and the feedback moves of Table 2 have not been tested with teachers or learners, and until they are, DRACE is

License: CC BY 4.0

a computational diagnostic prototype with potential formative-assessment applications rather than a validated assessment instrument.

Ethics statement

This article does not involve any original experimental studies with human or animal subjects. Therefore, no ethical approval was required for this work.

Funding statement

The author received no financial support for the research, authorship, and/or publication of this article.

Declaration of competing interest

The author declares no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability statement

The engine (DRACE version 1.3.0), the benchmark definitions, the validation suite, the experiment scripts, the raw sample bank and the summary result files are provided as a supplementary archive (DRACE_supplementary.zip). With them every number in Section 5 can be reproduced exactly, except the wall-clock timings, which depend on the machine; a verification script recomputes the main results from the raw sample bank, and the version history of the archive states which version produced the results of each version of the manuscript.

References

Arzarello, F., Olivero, F., Paola, D., & Robutti, O. (2002). A cognitive analysis of dragging practises in Cabri environments. ZDM – Mathematics Education, 34(3), 66-72. https://doi.org/10.1007/BF02655708

Baccaglini-Frank, A., & Mariotti, M. A. (2010). Generating conjectures in dynamic geometry: The maintaining dragging model. International Journal of Computers for Mathematical Learning, 15(3), 225-253. https://doi.org/10.1007/s10758-010-9169-3

Battista, M. T. (2007). The development of geometric and spatial thinking. In F. K. Lester (Ed.), Second handbook of research on mathematics teaching and learning (pp. 843-908). Charlotte, NC: Information Age Publishing.

Botana, F., Hohenwarter, M., Janičić, P., Kovács, Z., Petrović, I., Recio, T., & Weitzhofer, S. (2015). Automated theorem proving in GeoGebra: Current achievements. Journal of Automated Reasoning, 55(1), 39-59. https://doi.org/10.1007/s10817-015-9326-4

Chan, K. K., & Leung, S. W. (2014). Dynamic geometry software improves mathematical achievement: Systematic review and meta-analysis. Journal of Educational Computing Research, 51(3), 311-325. https://doi.org/10.2190/EC.51.3.c

License: CC BY 4.0

Gerhäuser, M., Valentin, B., & Wassermann, A. (2010). JSXGraph – Dynamic mathematics with JavaScript. International Journal for Technology in Mathematics Education, 17(4), 211-215. https://eric.ed.gov/?id=EJ913356

Hattie, J., & Timperley, H. (2007). The power of feedback. Review of Educational Research, 77(1), 81-112. https://doi.org/10.3102/003465430298487

Healy, L., & Hoyles, C. (2001). Software tools for geometrical problem solving: Potentials and pitfalls. International Journal of Computers for Mathematical Learning, 6(3), 235-256. https://doi.org/10.1023/A:1013305627916

Hölzl, R. (1996). How does 'dragging' affect the learning of geometry. International Journal of Computers for Mathematical Learning, 1(2), 169-187. https://doi.org/10.1007/BF00571077

Ji, Z., Guo, K., & Song, S. (2024). Effects of dynamic mathematical software on students' performance: A three-level meta-analysis. Journal of Educational Computing Research, 62(4), 1035-1060. https://doi.org/10.1177/07356331241226594

Kovács, Z., Recio, T., & Vélez, M. P. (2018). Using automated reasoning tools in GeoGebra in the teaching and learning of proving in geometry. International Journal for Technology in Mathematics Education, 25(2), 33-50. https://doi.org/10.1564/tme_v25.2.03

Laborde, C. (2001). Integration of technology in the design of geometry tasks with Cabri-Geometry. International Journal of Computers for Mathematical Learning, 6(3), 283-317. https://doi.org/10.1023/A:1013309728825

Laborde, C. (2005). Robust and soft constructions: Two sides of the use of dynamic geometry environments. In S. C. Chu, H. C. Lew, & W. C. Yang (Eds.), Proceedings of the 10th Asian Technology Conference in Mathematics (pp. 22-36). Cheong-Ju: Korea National University of Education.

Leung, A. (2008). Dragging in a dynamic geometry environment through the lens of variation. International Journal of Computers for Mathematical Learning, 13(2), 135-157. https://doi.org/10.1007/s10758-008-9130-x

Leung, A., Baccaglini-Frank, A., & Mariotti, M. A. (2013). Discernment of invariants in dynamic geometry environments. Educational Studies in Mathematics, 84(3), 439-460. https://doi.org/10.1007/s10649-013-9492-4

Mariotti, M. A. (2000). Introduction to proof: The mediation of a dynamic software environment. Educational Studies in Mathematics, 44(1-2), 25-53. https://doi.org/10.1023/A:1012733122556

Mishra, P., & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowledge. Teachers College Record, 108(6), 1017-1054. https://doi.org/10.1111/j.1467-9620.2006.00684.x

Mityagin, B. S. (2020). The zero set of a real analytic function. Mathematical Notes, 107(3), 529-530. https://doi.org/10.1134/S0001434620030189

Peculea, L. (2026). Pilot adaptation and preliminary validation of the Generative AI Literacy for Learning Scale (GenAI-LLs) in engineering education. Journal of Applied Technical and Educational Sciences, 16(1), ArtNo: 425. https://doi.org/10.24368/jates425

Shute, V. J. (2008). Focus on formative feedback. Review of Educational Research, 78(1), 153-189. https://doi.org/10.3102/0034654307313795

Sinclair, N., Bartolini Bussi, M. G., de Villiers, M., Jones, K., Kortenkamp, U., Leung, A., & Owens, K. (2016). Recent research on geometry education: An ICME-13 survey team report. ZDM – Mathematics Education, 48(5), 691-719. https://doi.org/10.1007/s11858-016-0796-6

License: CC BY 4.0

Trinh, T. H., Wu, Y., Le, Q. V., He, H., & Luong, T. (2024). Solving olympiad geometry without human demonstrations. Nature, 625(7995), 476-482. https://doi.org/10.1038/s41586-023-06747-5

van Hiele, P. M. (1986). Structure and insight: A theory of mathematics education. Orlando, FL: Academic Press.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. Educational Psychologist, 46(4), 197-221. https://doi.org/10.1080/00461520.2011.611369

Wang, F., & Hannafin, M. J. (2005). Design-based research and technology-enhanced learning environments. Educational Technology Research and Development, 53(4), 5-23. https://doi.org/10.1007/BF02504682

License: CC BY 4.0