The Quantum Geometry® Grand Challenge: A Multiyear Prize and Benchmark Program for Quantum Computing in Geometry, CAD, GIS, and Meshing

W. Randolph Franklin
contest@quantum-geometry.com
Quantum Geometry® LLC
Henderson, Nevada, USA

Abstract

This paper proposes the Quantum Geometry® Grand Challenge, a five-year prize and benchmark program for quantum and hybrid quantum–classical methods in computational geometry, computer-aided design (CAD), geographic information science (GIS), and meshing. The central question is where, if anywhere, quantum computation improves a complete geometric workflow. Comparisons would include data preparation, encoding, execution, and output validation. The program would begin by defining benchmarks, without requiring an initial demonstration of quantum advantage. A proposed 2027 pilot would select a small number of spatial-optimization and mesh-partitioning problems and publish classical reference solvers and independent validators. Separate award categories would recognize reproducible positive and negative findings. Two illustrative problems show how the evaluation would work. The paper sets out submission requirements, comparison metrics, judging safeguards, and a staged expansion to CAD, geometric algorithms, and data representations. Quantum Geometry® LLC would initially administer the program and offer conditional founding matching support of up to $100,000 over five years. Expansion would depend on secured resources and independently reproducible pilot results. The contribution is a research-program design that can be tested and evaluation resources that others can reuse; it is not a claim of demonstrated quantum advantage.

CCS Concepts

• **Theory of computation → Computational geometry**; • **General and reference → Design**; • **Computing methodologies → Parallel algorithms**; • **Hardware → Quantum technologies**.

Keywords

quantum computing, computational geometry, computer-aided design, geographic information science, mesh generation, spatial optimization, benchmarks, prize competitions

ACM Reference Format:
W. Randolph Franklin. 2026. The Quantum Geometry® Grand Challenge: A Multiyear Prize and Benchmark Program for Quantum Computing in Geometry, CAD, GIS, and Meshing. In *The 2nd ACM SIGSPATIAL International Workshop on Quantum Computing and Quantum-Inspired Technologies for Spatial Data Systems and Applications (Q-Spatial '26), November 03–06, 2026, Riverside, CA, USA.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3849728.3849841



This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License.
Q-Spatial '26, Riverside, CA, USA
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-3054-2/2026/11
https://doi.org/10.1145/3849728.3849841

Disclosure

The author is President of Quantum Geometry® LLC and owns the Quantum Geometry® trademark. The proposed program would initially be administered by that company. The views expressed are the author's own and do not imply endorsement, sponsorship, or an official position of Rensselaer Polytechnic Institute (RPI), ACM SIGSPATIAL, SIAM, IBM, XPRIZE, or any other organization.

1 Introduction

Geometry-centered computation supports engineering design, spatial decision making, manufacturing, robotics, and scientific simulation. Classical algorithms and software are mature, but quantum computation may still improve some search, optimization, sampling, or linear-algebra tasks. Testing that possibility requires more than translating a familiar problem into a quantum circuit. The comparison must preserve the original task and its accuracy requirements, and include the costs of the complete workflow.

The Quantum Geometry® Grand Challenge would organize this investigation around shared problem statements, public test instances, strong classical reference methods, independent validation, and prizes for reusable research contributions. Its first goal would be to identify suitable problems and reliable ways to evaluate them. A careful finding that an approach is uncompetitive can be as valuable as a promising algorithm.

This paper makes three contributions. First, it defines a program connecting quantum-computing researchers with the computational geometry, CAD, GIS, and meshing communities. Second, it proposes a benchmark and evaluation procedure, illustrated by two pilot problems. Third, it sets out governance, conditional funding, and milestones for deciding whether to expand the pilot. These arrangements are proposals. Partnerships, awards, and technical outcomes are not presented as established facts.

2 History of Competitions

A competition can accomplish several things that conventional research solicitations usually do not. It creates common deadlines and shared deliverables, attracts students and interdisciplinary teams, and exposes weak comparison methods and poorly specified problems. It can also produce reusable datasets, software, and reports.

A competition makes an emerging field more visible to sponsors, institutions, and potential participants. By rewarding careful negative results as well as positive claims, the proposed program could help build a research community.

Competitions have repeatedly focused attention and accelerated technical progress. Alcock and Brown's 1919 transatlantic flight won the *Daily Mail* prize [1]. DARPA's Grand Challenge helped stimulate progress in autonomous vehicles [2]. IARPA has sponsored automatic speech-recognition challenges [3]. Kaggle provides a widely used platform for data-science competitions built around shared benchmarks [4]. Machine learning conferences routinely use shared tasks and competition tracks to define measurable progress [5]. A comparable effort could help the computational geometry, GIS, CAD, and meshing communities understand and evaluate quantum methods.

3 Proposer Background

Franklin has worked in parallel computational geometry since the 1980s [6]. He supervised doctoral dissertations on parallel geometric computation in the early 1990s [7, 8], and his more recent work includes parallel mesh intersection and exact geometric predicates on GPUs [9, 10]. While serving as Director of the Numeric, Symbolic, and Geometric Computation Program in NSF/CISE, Franklin helped lead two joint NSF/DARPA solicitations on *Computational and Algorithmic Representations of Geometric Objects (CARGO)* [11]. CARGO targeted three areas: computational cartography, computational topology, and computer-aided design. In 2021 he developed and taught the first Quantum Computer Programming course in RPI's School of Engineering [12]. The learning outcomes of that course were to (i) demonstrate proficiency with the mathematics behind quantum computing, including qubits and quantum gates, (ii) understand important quantum computing algorithms such as Grover's and Shor's, (iii) understand the three platforms covered in the course: transmon qubits (IBM Q), trapped ions (IonQ), and quantum annealing (D-Wave), and (iv) apply that to write and run programs on those platforms.

He is now an Emeritus Professor at RPI and President of Quantum Geometry® LLC [13]. His consulting covers computer engineering; parallel and CUDA programming; efficient GPU data structures and algorithms; geographic information science data structures; applied computational geometry; CAD algorithms, including overlaying triangulations in 3D; and quantum computer programming.

The proposed challenge also draws on the author's experience supporting students and technical competitions. Since 2022, he has donated $14,500 to SIAM for student travel to the International Meshing Roundtable, $25,000 to ACM SIGSPATIAL for GIS Cup prizes and travel support, and $5,000 to RPI's student Quantum Computing Club. These activities span the three communities the challenge seeks to connect: geometry and meshing, GIS and spatial computing, and quantum computing. A focused multiyear program could give these separate contributions a lasting framework.

4 Positioning and Technical Scope

4.1 Relationship to existing work

Quantum algorithms for geometric problems are not new. Sadakane, Sugawara, and Tokuyama studied intersection and proximity problems in 2001 [14]. Aaronson et al. later studied the quantum complexity of closest pair and related problems [15]. These results motivate further investigation. They also show why each study must state its computational model and data-access assumptions.

Benchmark resources are also developing outside geometry. The Quantum Optimization Benchmarking Library (QOBLIB) provides ten optimization problem classes and a framework for comparing classical and quantum methods [16]. XPRIZE Quantum Applications runs a broader applications competition [17]. The proposed challenge should build on these efforts where appropriate. It would add geometric task definitions, correctness checks tailored to each application, comparisons of complete workflows, and sustained participation from spatial and engineering communities.

Speedup claims also need care. Rønnow et al. distinguish comparison scenarios and identify pitfalls in detecting quantum speedup [18]. A favorable result against specified classical methods should therefore be reported with its scope and assumptions. It should not be treated as universal evidence of quantum advantage.

4.2 What belongs in the challenge?

Here, *quantum geometry* refers to the program's focus on quantum methods for geometry-centered computation. It does not refer to a new mathematical theory of geometry or quantum gravity. Quantum Geometry® is the program's brand; technical descriptions would use familiar disciplinary terms.

Eligible quantum approaches would include gate-based algorithms, quantum annealing, and hybrid workflows. Quantum-inspired methods run entirely on classical hardware would be labeled as such. They could serve as classical comparison methods or enter a separate contribution category. A simulator demonstration would show how an implementation behaves at the simulated scale, not how it performs on quantum hardware.

Geometry must be essential to a benchmark. A routing, clustering, or graph-partitioning task should preserve the coordinates, distances, topology, constraints, or engineering meaning relevant to the application. It should not become an arbitrary optimization problem with a geometric label. The presence of a difficult optimization subproblem also does not, by itself, show that a quantum method will help.

4.3 A focused pilot within a broader scope

The program covers five application families (Table 1). Every family should report hardware comparisons and data-encoding costs; these should not be separate application leaderboards. The pilot should start with at most two families so that organizers can maintain reference implementations and validate submissions reliably.

5 Two Illustrative Pilot Benchmarks

The following examples are proposed for advisory-board review. They are not completed experiments or adopted challenge rules.

Table 1: Candidate application families. These are research directions, not claims of quantum advantage.

Family	Example tasks and required outputs
GIS and spatial optimization	Facility placement, routing, regionalization; feasible placements, routes, or assignments.
Meshing and simulation preprocessing	Mesh partitioning and quality improvement; valid partitions or meshes preserving specified constraints.
CAD and geometric constraints	Assembly configuration, tolerance analysis, surface fitting; configurations meeting stated tolerances.
Computational geometry	Proximity, intersection, visibility, range searching; answers meeting exact or declared approximate specifications.
Geometric and topological representations	Point-cloud, mesh, and complex encodings; useful task outputs with preparation and readout costs.

Table 2: Two station-placement plans for the four-town example. Every town is assigned to its nearest selected station.

Selected sites (km)	Nonzero contributions	Total
0, 2	$1 \times 6 + 2 \times 8$	22
2, 10	$1 \times 2 + 1 \times 2$	4

They show how precisely a task must be defined before an algorithm prize can be evaluated.

5.1 Spatial facility location

Consider choosing locations for emergency-response stations. A municipality has a budget for a fixed number of stations and a list of eligible sites. Neighborhoods generate different levels of service demand. Where the stations are placed determines how far responders must travel. A simple objective is to minimize distance weighted by expected demand, with each neighborhood assigned to a selected station.

For a small example, consider four towns along a straight road at 0, 2, 8, and 10 km. Their relative demand weights are 1, 2, 1, and 2. Each town is an eligible site, and two stations must be selected. Stations at 0 and 2 serve the western towns locally but leave the eastern towns far away. The eastern towns contribute 1×6 and 2×8 to weighted distance, for a total of 22. Stations at 2 and 10 serve both clusters. Only the towns at 0 and 8 then have nonzero travel distances, giving $1 \times 2 + 1 \times 2 = 4$ (Table 2). Both plans use two stations, but their service-distance costs differ. A GIS application could use road-network distances or travel times instead of this one-dimensional model.

Let I be demand locations and J a finite set of candidate facility sites. Each demand has weight $w_i \geq 0$, and $d_{ij} \geq 0$ is its specified distance to candidate j. For a prescribed number of facilities p, binary variables y_j indicate selected sites and z_{ij} indicate assignments. A pilot could use the discrete weighted p-median formulation

$$\begin{aligned} \min_{y,z} \quad & \sum_{i \in I} \sum_{j \in J} w_i d_{ij} z_{ij}, \\ \text{subject to} \quad & \sum_{j \in J} y_j = p, \\ & \sum_{j \in J} z_{ij} = 1 \quad (i \in I), \\ & z_{ij} \leq y_j, \quad y_j, z_{ij} \in \{0, 1\}. \end{aligned} \tag{1}$$

The required output is a set of p sites and a valid assignment for every demand. The validator would recompute the objective from the original distance data. If distances derive from a road network rather than Euclidean coordinates, their construction cost and provenance would be reported. Capacity, equity, and temporal constraints would be reserved for subsequent versions with separately specified validators.

Test instances should vary the numbers of demand locations and candidate sites, the degree of clustering, and the distance structure. Small cases would have certified optima. Larger cases would include solver bounds or clearly labeled best-known feasible values. Classical reference methods should include a suitable mixed-integer optimization solver and a documented facility-location heuristic. A quantum or annealing implementation could use a binary optimization encoding. Its comparison would include the costs of auxiliary variables, penalties, embedding, and any repairs needed to make a solution feasible.

The research question is specific: for a stated family of test instances and a fixed resource budget, does the complete method improve the validated objective value or reach a fixed quality target faster than the classical methods? A small encoded problem that is easy to solve classically can still be useful for education or for developing a formulation. It would not, by itself, support a performance claim.

5.2 Balanced partitioning of a mesh dual graph

A motivating application is distributing a simulation mesh among processors. The partition should balance computational work while reducing communication between neighboring cells assigned to different processors. This is an established use of mesh partitioning [19].

For example, consider heat flow through a plate divided into a 2×4 grid of square cells. Each cell requires the same computational work. Two processors, A and B, receive four cells each. In the dual graph, each cell is a vertex, and cells sharing an edge are connected. Figure 1 shows two balanced assignments. The left/right partition cuts two neighbor connections; the top/bottom partition cuts four. If each crossing connection has the same data-exchange cost, the left/right assignment has half as many communication interfaces, with the same workload balance. Finding a good assignment on a larger, irregular mesh is a useful optimization task. If work differs between cells, the balance requirement should use cell weights.

For a mesh with an even number n of cells, let $G = (V, E)$ be its dual graph: vertices represent cells, and edges connect cells sharing a face. Assign a nonnegative edge weight a_{uv}, such as a declared

Left/right

A	A	B	B
A	A	B	B

Two crossing connections

Top/bottom

A	A	A	A
B	B	B	B

Four crossing connections

Figure 1: Two balanced partitions of an eight-cell mesh. Letters identify processors; double rules mark their shared boundary. Both processors receive four cells, but the left/right partition has fewer communication interfaces.

communication weight. A simple pilot task is balanced bisection:

$$\min_{x\in\{0,1\}^n} \quad \sum_{(u,v)\in E} a_{uv}(x_u - x_v)^2, \tag{2}$$
$$\text{subject to} \quad \sum_{v\in V} x_v = n/2.$$

Here x_v identifies the processor assigned to cell v. The term $(x_u - x_v)^2$ is zero for neighbors on the same processor and one for neighbors on different processors. With unit edge weights, the two assignments in Figure 1 therefore have objective values two and four, respectively. The binary quadratic objective provides a concrete test case for comparing classical and quantum optimization methods under the same balance constraint.

The output specifies a partition for every cell. The validator would check the number of cells in each partition and recompute the cut from the original graph. Allowing a balance tolerance or more than two partitions would require a new version of the specification. Connectivity would be required only when explicitly included as an application constraint.

Test instances should include regular and irregular meshes, with documented generation procedures or reusable source data. Classical reference methods would include a multilevel graph-partitioning method subject to the same balance requirement and an exact solver for small cases. A hybrid method should also be tested with its quantum component replaced by a classical solver, keeping the rest of the workflow equivalent.

This benchmark concerns simulation preprocessing, not mesh generation or smoothing. A lower cut weight improves one partitioning objective. It does not show that a finite-element simulation runs faster unless that downstream effect is measured. A useful result could identify a competitive subproblem solver or show that overhead prevents practical improvement.

6 Benchmark and Evaluation Protocol

6.1 Minimum benchmark package

Before opening an algorithm competition, each selected task should publish the following:

(1) **Task specification:** input and output formats; objective; constraints; coordinate units; numeric precision; degeneracies; and exactness or approximation requirements.
(2) **Instance suite:** provenance and reuse permissions; generators where available; size and difficulty parameters; and separate development and evaluation instances.
(3) **Independent validator:** checks on the original task, including feasibility, residuals, tolerances, and objective values. Encoded energies alone would not establish task correctness.
(4) **Classical references:** runnable implementations, versions, configurations, hardware descriptions, tuning budgets, and certified optima or bounds where available.
(5) **Reporting template:** execution environment, costs, repeated-run results, uncertainty, and the precise claim supported by the evidence.

Evaluation instances should be withheld until teams freeze their submission configurations, then published after evaluation. Alternatively, organizers could announce a generator in advance and select its seeds independently. Either approach should prevent undisclosed tuning on the evaluation cases while producing an archived benchmark that others can inspect.

6.2 Accounting for the complete workflow

All methods should be evaluated from a common input representation to a validated output. A useful accounting identity is

$$T_{\text{total}} = T_{\text{prepare}} + T_{\text{encode}} + T_{\text{solve}} + T_{\text{decode}} + T_{\text{validate}}. \tag{3}$$

For hybrid methods, T_{solve} includes classical optimization, communication, repeated quantum execution, and any error-mitigation procedures. Compilation and annealer embedding belong in the relevant preparation or encoding stage. Reported costs must also include measurements, unsuccessful attempts, and solution repairs. Shared setup costs may be spread across repeated tasks only when the reuse scenario and accounting rule are explicit and equally available to classical methods.

Reports should separate cloud queue delay from active execution time and also give the elapsed time experienced by the user. Where available, they should include memory use, hardware access costs, and quantum resources alongside runtime. A circuit-level improvement can be useful, but it does not replace a comparison of complete workflows.

Theoretical submissions should state the computational model, oracle or memory assumptions, dependence on precision, output requirements, and algorithmic resource bounds. Fault-tolerant estimates should distinguish logical from physical qubits and give the error-correction and hardware assumptions. Asymptotic results and projected hardware runtimes are different forms of evidence and should be labeled accordingly.

6.3 Quality, cost, and statistical evidence

Each benchmark should announce its primary metrics in advance. For optimization, these would include feasibility rate, objective quality at a fixed budget, and time to a fixed quality target. For geometric decision or construction tasks, a result must be correct within the stated tolerance before its speed is compared.

An objective gap should be measured against a certified optimum or valid bound, with its normalization stated. A best-known feasible value is a useful reference, but it is not proof of optimality. Reports should cover the stated family of instances, including failures and timeouts, rather than only successful cases. Randomized methods should report repeated trials, seeds where meaningful, and the

Table 3: Proposed rubric for pilot benchmark proposals after minimum requirements are met. Later categories would have separately published rubrics.

Criterion	Weight
Geometric relevance and precise task definition	25%
Instance design, validation, and reproducibility	25%
Classical baselines and fair comparison plan	20%
Insight into quantum suitability or limitations	20%
Clarity, documentation, and educational value	10%

Table 4: Illustrative pilot award pool, contingent on secured funds and published rules.

Award	Amount
Best overall benchmark proposal	$5,000
Best student submission	$2,500
Best GIS or spatial-computing proposal	$2,500
Best CAD or meshing proposal	$2,500
Best rigorous negative or skeptical analysis	$2,500
Total	$15,000

spread of results or confidence intervals. Equal budgets need not mean equal hardware, but the budget and performance question must be clear.

In a hybrid workflow, replacing the quantum component with a suitable classical method would help show where any gains come from. They may come from that component, or from the surrounding preprocessing and problem decomposition. Organizers should ask domain specialists to review the classical reference methods and update benchmarks when stronger methods become available.

Negative results need the same care as positive ones. One implementation losing to a classical method on one family of instances does not rule out all quantum methods. A useful negative result identifies a reproducible limitation, failed assumption, resource requirement, or bottleneck that can guide later research.

7 Competition Structure and Judging

7.1 Separate contribution categories

The program should have separate categories for benchmark proposals, theoretical and resource-estimation studies, empirical algorithm studies, and educational contributions. Hardware experiments and simulator studies would be clearly labeled and ranked separately. Entirely classical and quantum-inspired entries could improve the reference methods and receive appropriate recognition.

Before scoring an empirical entry, judges should require correct task outputs, complete resource accounting, an adequate classical comparison, and enough code and data to inspect the main result. A benchmark proposal would instead need a complete problem statement, a validator, and a defensible plan for classical reference methods. Each public award should name the category and contribution being recognized.

Table 3 does not require a favorable conclusion about quantum computing. Rewarding only positive findings would conflict with the program's support for negative results. New formulations and algorithms are welcome, but novelty cannot make up for invalid comparisons or incomplete task definitions.

7.2 Proposed 2027 pilot awards

The first competition could be titled *Quantum Geometry® Grand Challenge 2027: Benchmark Problems for Quantum and Hybrid Methods in Geometry*. The illustrative awards in Table 4 retain the proposed amounts and total $15,000. This is a proposed award pool, not a secured operating budget or an announced obligation.

Before launch, the rules should define eligibility, student status, minimum award standards, ties, and appeals. They should also say whether one entry can receive multiple awards. Domain awards would be offered only if the pilot solicits proposals in those areas. Judges should be able to withhold an award when no entry meets the published standard. A skeptical-analysis entry should offer substantive evidence, not just an unsupported objection to quantum computing.

8 Governance and Community Participation

8.1 Administration and scientific independence

Quantum Geometry® LLC would initially administer the website, schedule, sponsor agreements, communications, and payments. An external scientific advisory board should oversee task selection, evaluation rules, and review standards. It should include expertise in quantum algorithms, geometric and spatial computation, meshing, optimization, and reproducible science.

The administrator owns the program's trademark and may receive sponsorship, so scientific decisions need explicit safeguards. The rules should require disclosure of financial and institutional relationships, recusal of judges with conflicts, and a judging panel with an independent majority. Rubrics should be published before submission, and appeals should follow a documented process. Sponsors could contribute resources and expertise. Sponsorship should give them no entitlement to favorable results or influence over particular awards. Naming recognition would be separate from scientific authority.

An openly licensed repository should preserve versions of the benchmark specifications, validators, and reference implementations. Hardware submissions should archive code, configurations, available calibration information, and run records. Later reviewers may lack access to the original device. Reports should therefore distinguish rerunning the analysis from independently repeating an experiment on hardware.

8.2 Participants and possible partners

Conferences and professional societies could help with workshops, special sessions, student travel, publicity, and independent reviewers. RPI's IBM Quantum System One installation provides relevant institutional context [20]. However, neither access to that system nor an RPI partnership is assumed. Participation by ACM SIGSPATIAL, SIAM's International Meshing Roundtable, and other quantum, CAD, GIS, and meshing organizations is also a possibility, not an established arrangement.

Student clubs could organize reading groups and local projects. Faculty could build the benchmarks into seminars, capstones, or

independent studies. Starter instances, documented classical reference methods, and simulator-compatible exercises would let students participate without privileged hardware access. Clubs should help bring in participants, while a stable organization maintains administration and archives.

9 Funding and a Five-Year Roadmap

9.1 Conditional founding support

The proposed founding commitment is *up to $100,000 over five years from Quantum Geometry® LLC as matching support, provided suitable complementary sponsorship is secured.* This cap covers all five years; it is not an annual contribution. Before announcing awards, organizers must settle the matching terms, eligible expenditures, payment schedule, and treatment of in-kind support. No match ratio is assumed here.

Companies, universities, foundations, and other sponsors could provide cash support. Eligible university collaborators could also seek research grants. Cloud credits, software, datasets, and staff time would be valuable, but should be reported separately from cash available for prizes. The operating budget must cover benchmark curation, independent verification, administration, hosting, and student travel as well as awards. Expansion should depend on secured resources, not anticipated donations.

Sponsor packages could be developed once the pilot's costs are known. Fixed annual tiers and promised judging roles would be premature while matching terms, participation, and verification workload remain uncertain. Sponsor recognition should preserve the scientific independence described above.

9.2 Milestones and decisions

The five-year horizon would begin with the proposed 2027 pilot; subsequent phases are relative milestones, not commitments to particular hardware capabilities.

Year 1: formulate and validate. Form the advisory board, settle funding and rules, solicit benchmark proposals, select at most two pilot tasks, and publish validators and classical references. Complete an independent trial of the submission and evaluation process. The principal product would be usable benchmark packages and a public pilot report.

Year 2: evaluate algorithms. Run a limited competition on the selected tasks. Require full workflow accounting and classical comparison from the first algorithm round. Archive submissions and publish both positive and negative findings. Before expansion, an independent reviewer should be able to rerun the principal analyses and verify reported outputs.

Year 3: expand selectively. Add CAD constraints, proximity problems, point-cloud registration, or representation studies only when a task has a domain reviewer, maintained baselines, a validator, and funding for its evaluation. Preserve separate categories for theory, empirical performance, and education.

Year 4: test practical workflows. Seek industrially motivated, shareable instances and compare hardware and hybrid implementations under common output requirements. Resource estimates and hardware experiments should remain separately labeled. Downstream engineering or spatial benefits would require direct measurements.

Year 5: synthesize and preserve. Publish an archival benchmark release and a synthesis of methods, costs, limitations, and unresolved questions. Award capstone prizes only within secured funds, and decide whether continuing administration or institutional adoption is justified.

10 Risk Analysis

10.1 The Topic Is Too Unfamiliar

Some communities may see quantum computing as remote from their current concerns. Familiar problem descriptions and easy entry points would help. For SIGSPATIAL, the program should emphasize spatial optimization, routing, clustering, map matching, indexing, and facility location. For SIAM IMR, it should emphasize mesh quality, graph partitioning, refinement, and simulation preprocessing. For CAD communities, it should emphasize constraint solving, design-space exploration, and shape optimization.

10.2 Hype Damages Credibility

The competition should explicitly reward careful negative results and require strong classical baselines. It should not suggest that useful quantum advantage is imminent in all proposed areas.

10.3 Student Groups Lack Continuity

Student clubs are valuable participation channels but weak administrative homes. Administration should remain with Quantum Geometry® LLC or another stable entity, with students engaged as participants, ambassadors, and local organizers.

10.4 Professional Societies Decline Ownership

Professional societies should initially be asked for recognition, sessions, student awards, or publicity, not full administration. If the pilot succeeds, deeper institutional adoption may become possible later.

10.5 Vendor Lock-in

It is not yet clear which quantum technologies will ultimately prevail. The project should avoid dependence on a single vendor or hardware platform. Comparing architectures may itself provide useful insights. Middleware, libraries, and algorithmic abstractions that work across platforms would help. For example, a geometric problem expressed naturally through a Fourier transform might benefit from existing quantum Fourier-transform algorithms. Such an approach would still need a complete-workflow analysis and comparison with classical methods.

10.6 Classical Methods Remain Superior

The principal scientific risk is that classical methods outperform quantum methods on every pilot task. That would limit performance prizes, but could still produce useful benchmarks and explain why particular quantum approaches fail. A failed pilot should not be interpreted as proof that all quantum methods must fail.

10.7 Benchmark Design and Hardware Access

Poor benchmark design is a separate risk. An encoding may discard geometric structure, penalty choices may favor one method, or

small test instances may be too easy to support an application claim. Domain review, validation on the original task, and exact solutions for small cases would help address these problems. When stronger classical methods emerge, organizers should update the comparisons.

Unequal hardware access could distort participation. Separate theoretical, simulator, hardware, and educational categories would allow useful contributions under different resource constraints.

10.8 Funding and Evaluation Workload

Organizational risks include uncertain matching funds, excessive evaluation work, conflicts of interest, and loss of continuity. A limited pilot, independent judging, documented administration, and an archival repository would reduce these risks. If funding or participation falls short, the program should preserve its benchmark resources and scale down rather than announce commitments it cannot support.

10.9 Success Criteria

Success should be measured by independently checked benchmark releases, reproducible studies, maintained classical comparisons, student participation, and reuse by other researchers. Annual reports should record these outcomes, verification costs, and unresolved failures. A quantum performance improvement would be valuable, but the program could produce scientific value without one.

11 Conclusion

The Quantum Geometry® Grand Challenge would provide a structured way to test quantum methods in geometry-centered computation. Its first contribution would be shared problems and credible evaluation. Quantum advantage would remain a hypothesis to investigate. The proposed 2027 pilot offers a concrete first step: define a small number of benchmark tasks, publish validators and classical reference methods, secure the award and operating budgets, and invite independent review.

Conditional founding matching support of up to $100,000 over five years could help establish these resources. Broader sponsorship and expansion should depend on the pilot's findings. The lasting outcome should be evidence about useful methods and their limitations, along with research resources that remain valuable when classical methods prevail.

Acknowledgments

The author thanks the RPI quantum-computing student community, ACM SIGSPATIAL GIS Cup organizers, and SIAM International Meshing Roundtable community for providing examples of student-centered technical engagement. Any errors or omissions in this concept paper are the author's responsibility.

References

[1] Science Museum. 2017. Science Museum Anniversary. June 26, 2017. https://blog.sciencemuseum.org.uk/science-museum-anniversary/. Accessed October 4, 2026.

[2] DARPA. 2026. *Innovation Timeline*. https://www.darpa.mil/about/innovation-timeline. Accessed October 4, 2026.

[3] Kay Peterson, Audrey N. Tong, and Jennifer Yu. 2021. OpenASR20: An Open Challenge for Automatic Speech Recognition of Conversational Telephone Speech in Low-Resource Languages. In *Proceedings of Interspeech 2021*. https://doi.org/10.21437/Interspeech.2021-1930. Accessed October 4, 2026.

[4] Kaggle. 2026. *Kaggle Competitions*. https://www.kaggle.com/competitions. Accessed October 4, 2026.

[5] MLSys 2026. 2026. *Ninth Annual Conference on Machine Learning and Systems*. https://mlsys.org/Conferences/2026. Accessed October 4, 2026.

[6] W. Randolph Franklin, Narayanaswami Chandrasekhar, Mohan Kankanhalli, Manoj Seshan, and Varol Akman. 1988. Efficiency of uniform grids for intersection detection on serial and parallel machines. In *New Trends in Computer Graphics (Proceedings of Computer Graphics International '88)*, Nadia Magnenat-Thalmann and D. Thalmann (Eds.). Springer-Verlag, 288–297.

[7] Mohan Kankanhalli. 1990. *Techniques for Parallel Geometric Computations*. Ph.D. Dissertation. Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute.

[8] Chandrasekhar Narayanaswami. 1991. *Parallel Processing for Geometric Applications*. Ph.D. Dissertation. Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute. UMI no. 92-02201.

[9] Salles V. G. de Magalhães, W. Randolph Franklin, and Marcus V. A. Andrade. 2017. Fast exact parallel 3D mesh intersection algorithm using only orientation predicates. In *25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM. https://sigspatial2017.sigspatial.org/accepted-papers/. Accessed October 4, 2026.

[10] Marcelo de Matos Menezes, Salles Viana Gomes Magalhães, Matheus Aguilar de Oliveira, W. Randolph Franklin, and Rodrigo Eduardo de Oliveira Bauer Chichorro. 2022. Fast parallel evaluation of exact geometric predicates on GPUs. *Computer-Aided Design*, Article 103285. https://doi.org/10.1016/j.cad.2022.103285. Accessed October 4, 2026.

[11] U.S. National Science Foundation. 2001. *Computational and Algorithmic Representations of Geometric Objects (CARGO)*. NSF 01-111. https://www.nsf.gov/funding/opportunities/cargo-computational-algorithmic-representations-geometric-objects/5244/nsf01-111. Accessed October 4, 2026.

[12] W. Randolph Franklin. 2021. *Quantum ECSE-4964/6964 Quantum Computer Programming, Fall 2021, Rensselaer Polytechnic Institute*. Course website. https://wrfranklin.org/Teaching/quantum-f2021/. Accessed October 4, 2026.

[13] W. Randolph Franklin. 2026. *Quantum Geometry®*. https://quantum-geometry.com/. Accessed October 4, 2026.

[14] Kunihiko Sadakane, Norito Sugawara, and Takeshi Tokuyama. 2001. Quantum Algorithms for Intersection and Proximity Problems. In *Algorithms and Computation (ISAAC 2001)*, 148–159. Springer. https://link.springer.com/book/10.1007/3-540-45678-3. Accessed October 4, 2026.

[15] Scott Aaronson, Nai-Hui Chia, Han-Hsuan Lin, Chunhao Wang, and Ruizhe Zhang. 2020. On the Quantum Complexity of Closest Pair and Related Problems. In *35th Computational Complexity Conference (CCC 2020)*, Article 16, 16:1–16:43. https://doi.org/10.4230/LIPIcs.CCC.2020.16. Accessed October 4, 2026.

[16] Thorsten Koch, David E. Bernal Neira, Ying Chen, et al. 2026. The Quantum Optimization Benchmarking Library. *Nature Computational Science* 6, 653–671. https://doi.org/10.1038/s43588-026-00991-1. Accessed October 4, 2026.

[17] XPRIZE Foundation. 2026. *XPRIZE Quantum Applications*. Competition description. https://www.xprize.org/competitions/qc-apps. Accessed October 4, 2026.

[18] Troels F. Rønnow, Zhihui Wang, Joshua Job, Sergio Boixo, Sergei V. Isakov, David Wecker, John M. Martinis, Daniel A. Lidar, and Matthias

Troyer. 2014. Defining and detecting quantum speedup. *Science* 345, 420–424. https://doi.org/10.1126/science.1252319. Accessed October 4, 2026.

[19] George Karypis. 2013. *METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices*. Version 5.1.0. University of Minnesota, March 30, 2013. Sections 1 and 3.4. https://karypis.github.io/glaros/files/sw/metis/manual.pdf. Accessed October 4, 2026.

[20] Mike Murphy. 2024. The first IBM Quantum computer comes to a university campus. IBM Research, April 5, 2024. https://research.ibm.com/blog/rpi-ibm-quantum-system-one-installed. Accessed October 4, 2026.