UBA
Universidad de Buenos Aires
DEPARTAMENTO DE COMPUTACION
Facultad de Ciencias Exactas y Naturales - UBA

SEDLab
Discrete Event Simulation Laboratory

ICC Instituto de Ciencias de la Computación
CONICET

Status, optimisations and new tools for quantized steppers in Geant4 (QSStepper)

Rodrigo Castro, Matías Grynberg Portnoy,
Alejandro Mignanelli, Gerónimo Romczyk and Lucio Santi

University of Buenos Aires and
ICC-CONICET, Argentina.
rcastro@dc.uba.ar

CERN

XXIX Geant4 Collaboration Meeting
Catania, Italy
October 7-11, 2024

GEANT4
A SIMULATION TOOLKIT

Outline

1. Main Motivation for quantization-based stepping in Geant4

2. Quick refresher on QSS (more in the backup slides)

3. Milestone: QSS in the Geant4 11.2 release (Dec. 2023)

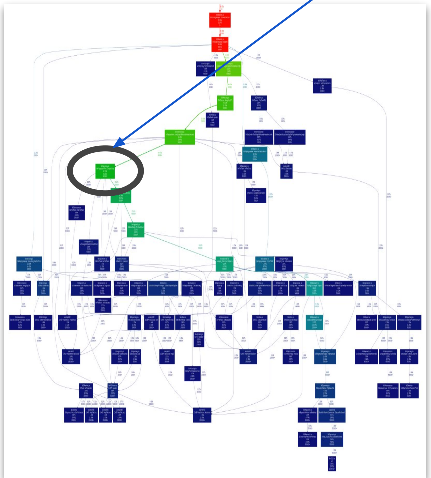
4. Progresses with new experimental versions of QSS
    - Automated benchmarking toolkit
    - Benchmarks with the ATLAS detector (FullSimLigth toolik)
    - New experimental flavors of the QSS family *[Matías Portnoy]*

5. Conclusions

Motivation

- **Simulation domain:** Tracking of subatomic particles
  - Undergoing physics processes within complex detector geometries
- **Key issue:** Handling of boundary crossings across discrete volumes
  - Can require **CPU-intensive ad-hoc iterative algorithms**
  - **Can we do better?**
- **Approach:** Family of hybrid (continuous/discrete-event) integrators
  - **Quantized State System (QSS) numerical methods**
  - Attractive performance features for HEP applications

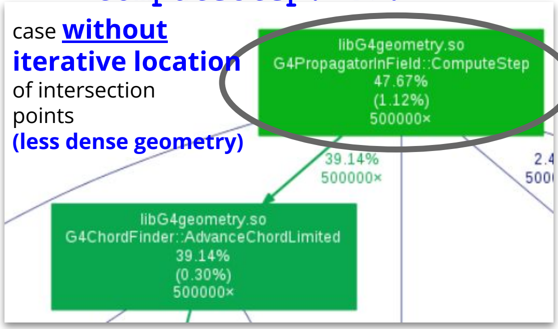
`G4PropagatorInField::ComputeStep`



callgrind performance analysis

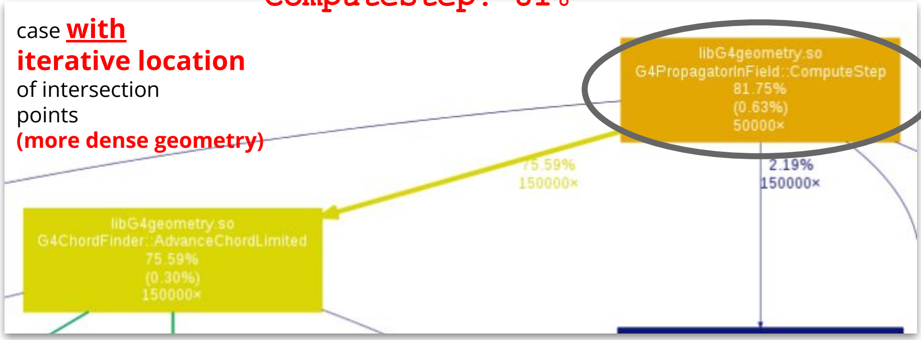
`ComputeStep:~47%`

case **without iterative location** of intersection points **(less dense geometry)**



`ComputeStep:~81%`

case **with iterative location** of intersection points **(more dense geometry)**

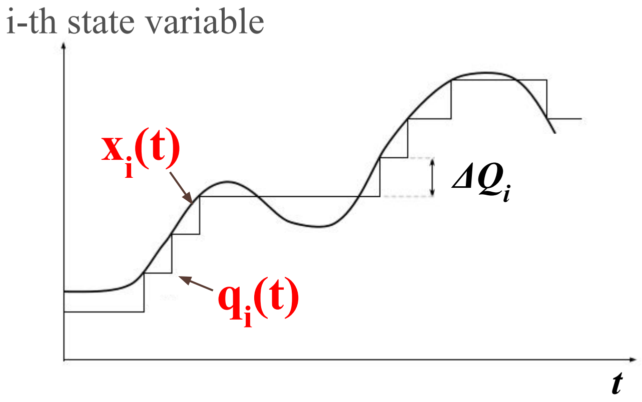


Synthetic benchmark
Perfect 2D particle oscillator
radius: 45 mm
Geometry: Parallel planes
G4 params:
epsilon = 1E− 7
deltaChord = 0.25 mm
stepMax = 20 mm
trackLength = 1000 m

Quantized State System (QSS) numerical methods

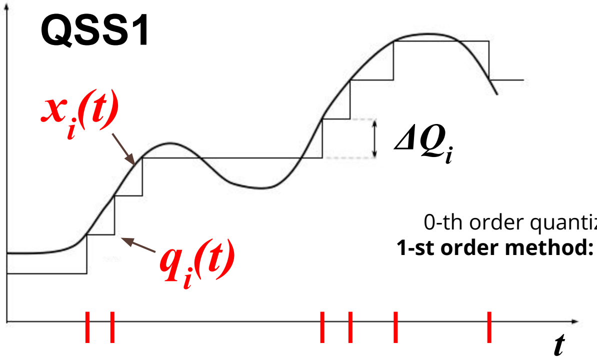
- Based on **state variables quantization**
- QSS methods discretize the system **state variables** as opposed to classical solvers which discretize the **time** (e.g. family of Runge-Kutta methods)
- ***Continuous* state variables** are approximated by ***Quantized* state variables**
  - A **quantization function** is in charge of controlling **error and accuracy** throughout the simulation

$$\underbrace{\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(\mathbf{t}))}_{\text{ODE system}} \quad \Rightarrow \quad \underbrace{\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{q}(\mathbf{t}))}_{\text{ODE quantized system}}$$



Higher order QSS

$$q_i(t) = \begin{cases} x_i(t) & \text{if } |q_i(t^-) - x_i(t)| \geq \Delta Q_i \\ q_i(t^-) & \text{otherwise} \end{cases}$$

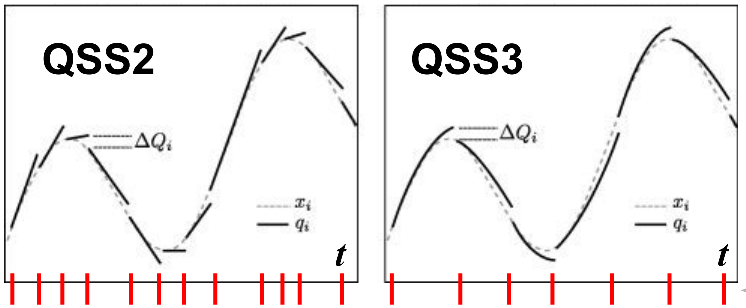


0-th order quantizer
1-st order method: QSS1

- ΔQ_i is the **quantum**
  - **Maximum deviation allowed** between x_i and q_i (**error control**)
  - Derived from the **accuracy** demanded by the user
- **Higher order QSS methods** (**QSS*n***) follow a similar principle
  - In a **QSS1** method, $q(t)$ follows piecewise constant trajectories
  - In a **QSS*n*** method, $q(t)$ follows piecewise (*n-1*)-*th* order polynomial trajectories

1-st order quantizer
2-nd order method: QSS2

2-nd order quantizer
3-rd order method: QSS3

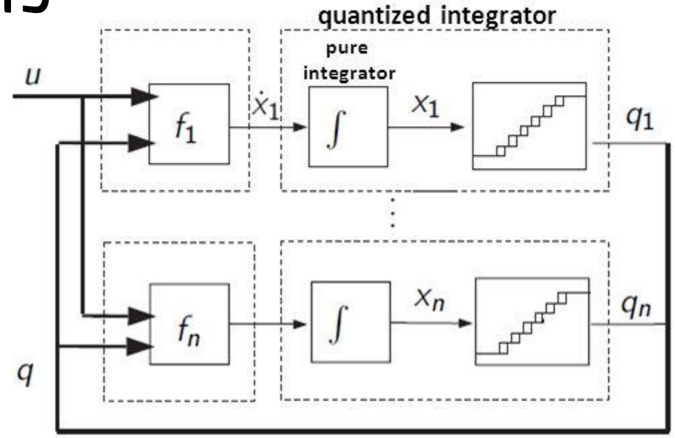


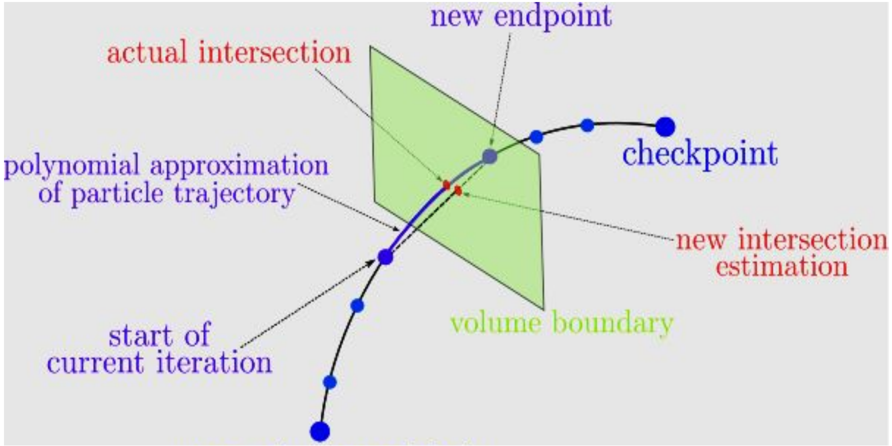
Asynchronous **discrete events**
No "regular" time steps

Goal: Fewer "steps" for the same global accuracy

Main QSS features for HEP problems

- **Inherent asynchronicity**
  - Decoupled, independent computation of changes in each state variable (no "global clock")
- **Dense trajectory output**
  - Supported by piecewise polynomial approximations of trajectories
- **Lightweight discontinuity handling**
  - Boundary crossings detected by **lightweight detection of** simple polynomial roots



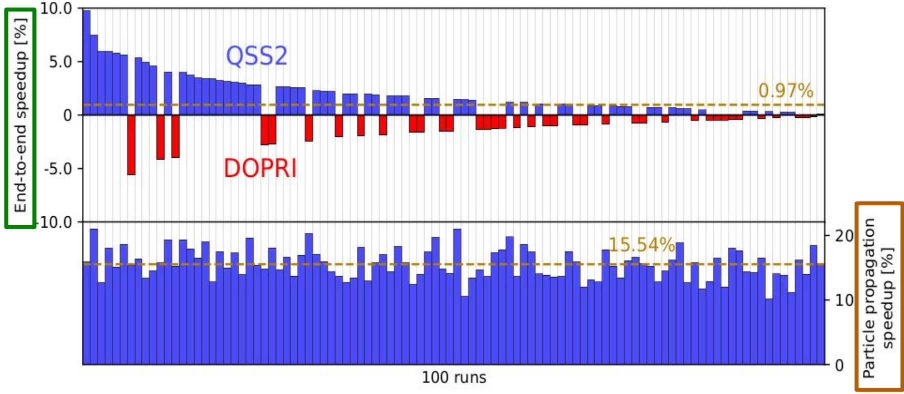


particle trajectory

CMS Benchmark results

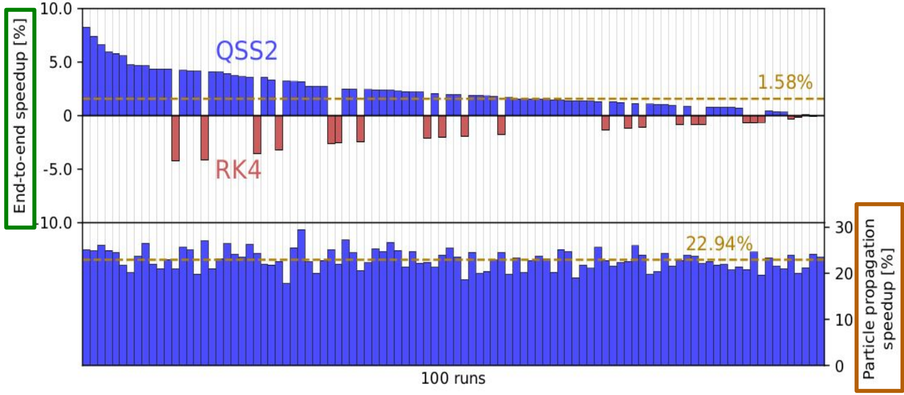
- Experimental results
  - CMS full Run 1 geometry

    Single π⁻ particles, Physics list FTFP_BERT
  - 100 independent runs, 2000 particle gun events
- QSS2 vs. DOPRI
  - 62 runs favorable for QSS; 38 for Geant4
  - Avg. End to End speedup: **~1%** (max. ~10%)
  - Avg. Stepping speedup: **~15%** (max. ~20%)
- QSS2 vs. RK4
  - 77 runs favorable for QSS; 23 for Geant4
  - Avg. End to End speedup: **~1.5%** (max. ~8%)
  - Avg. Stepping speedup: **~23%** (max. ~30%)



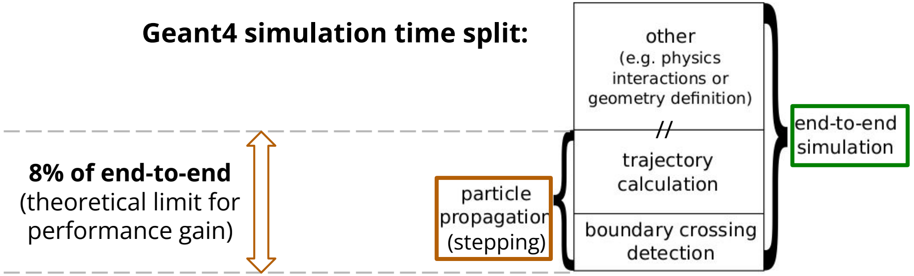
(a) QSS2 vs. DOPRI

(comparison with "DOPRI with Interpolation" is work in progress)



(b) QSS2 vs. RK4

Geant4 simulation time split:



Timeline (simplified)

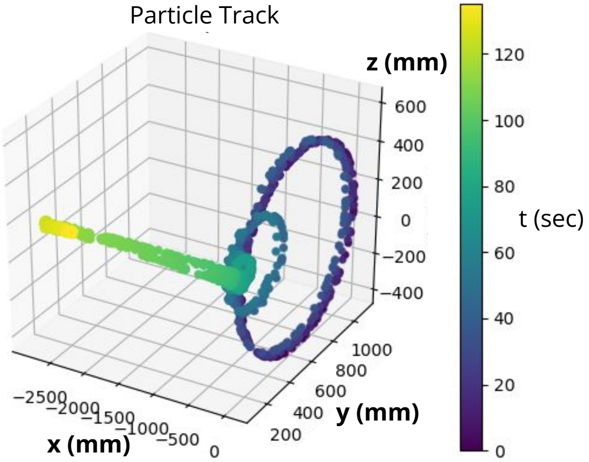
- 2019-2020 - Initial ideas, exploration of viability (10.5)
    - Toy examples
- 2021-2022 - First Implementations and version upgrades (10.7, 11.0, ...)
    - Geant4 official suite of test examples

- **2023 - QSS Stepper first incorporated into a Geant4 public release (11.2.0)**
    - Submitted to the Geant4 Testing and Quality Assurance process
    - v11.2.0, December 8th, 2023, https://geant4.web.cern.ch/download/release-notes/notes-v11.2.0.html

- 2024 - Current work: optimisations, housekeeping, research
    - Code cleaning, better documentation, more examples covered
    - Debugging of known issues (QSS3 debugging still pending)
    - Tooling: Automated benchmarking framework for QSS steppers
    - Optimised default steppers: codenames **new**QSS2, **new**QSS3
    - New experimental flavors: codenames **HelixMixed**QSS2, **Rotation**QSS2
    - ATLAS first tested (FullSimLight, FSL toolkit)

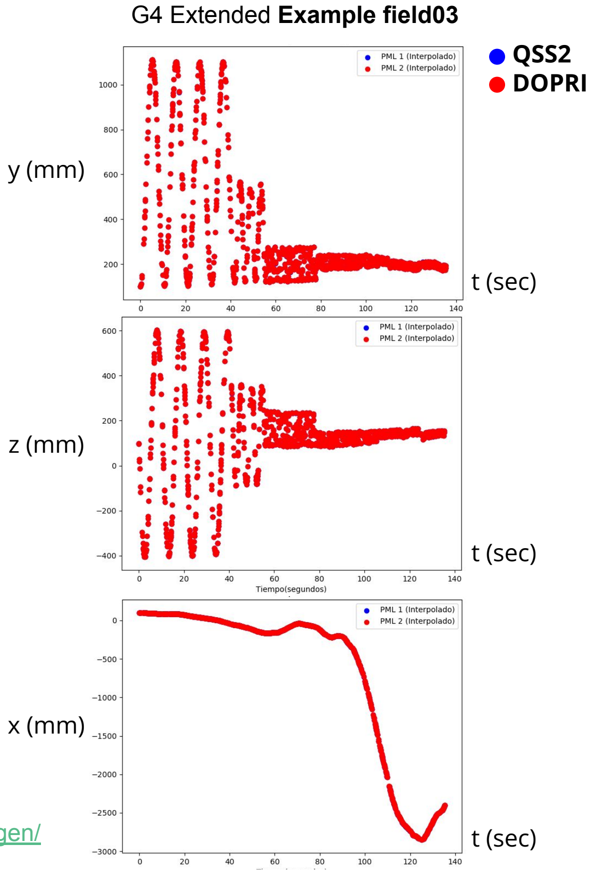
New: Logging for error assessment (A. Mignanelli)

- Calculation of the Mean Square Error (MSE) for x(t), y(t), z(t) and the Track Length L(t)
- Thorough systematic comparison of deviation between methods for **different accuracies**
- Interpolation of asynchronous time series
- E.g.: QSS2 vs DOPRI



dQRel=1e-5, dQMin=1e-6

```
X_MSE = 1.64
Y_MSE = 0.00072
Z_MSE = 0.0014
L_MSE = 0.0
```



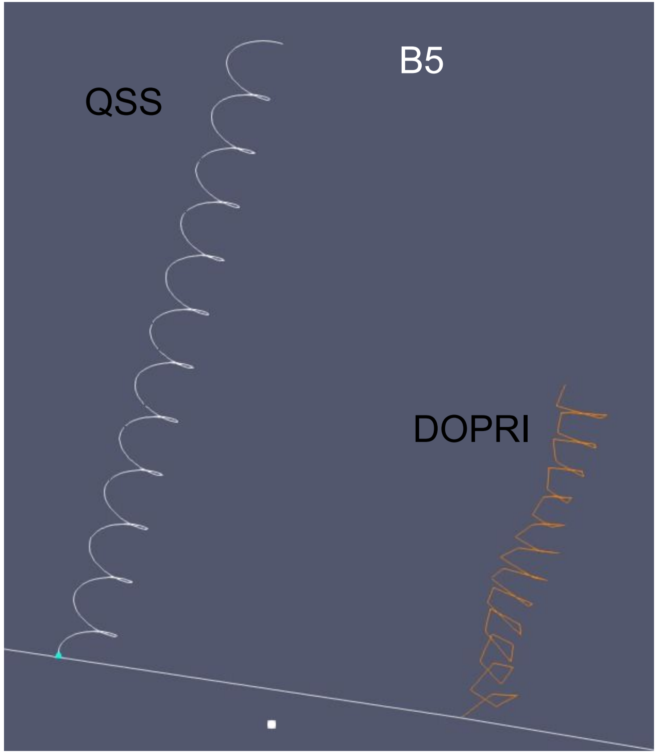
https://geant4-userdoc.web.cern.ch/Doxygen/examples_doc/html/Examplefield03.html

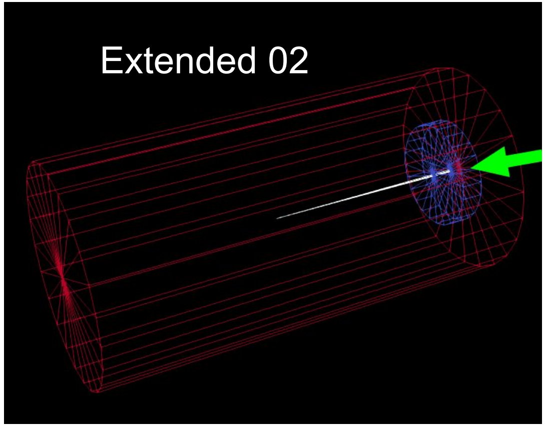
Examples tested

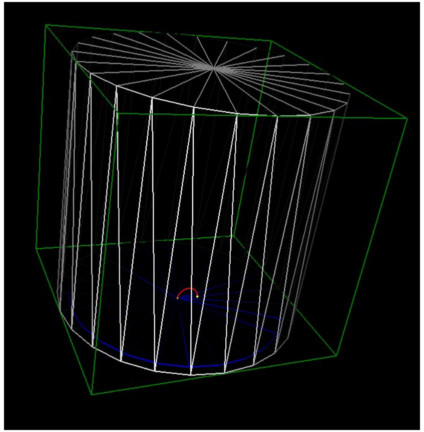
- Examples taken from the Geant4 Examples testing validation suite
  https://geant4-userdoc.web.cern.ch/Doxygen/examples_doc/html/index.html
- Examples considered:
  - Basic: **B2a, B2b, B4c, B4d, B5**
  - Extended: **field01, field03**
  - Advanced: **ams_Ecal**
- Examples **not** considered
  - Basic: B1, B3, B4a, B4b
  - Extended: F02, F04, F05, F06

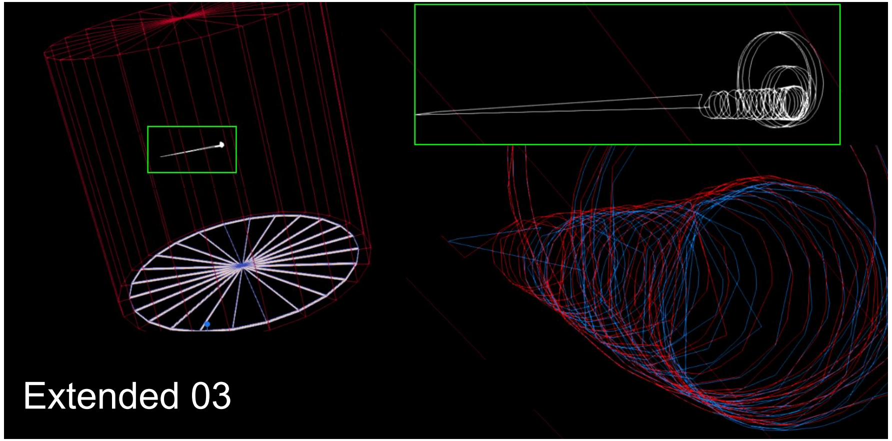
- Tests with models of Full Detectors:
  - **CMS**
    - Extensively tested
  - **ATLAS**
    - Recent efforts, using the FullSimLight (FSL) simulation package

Example visualizations









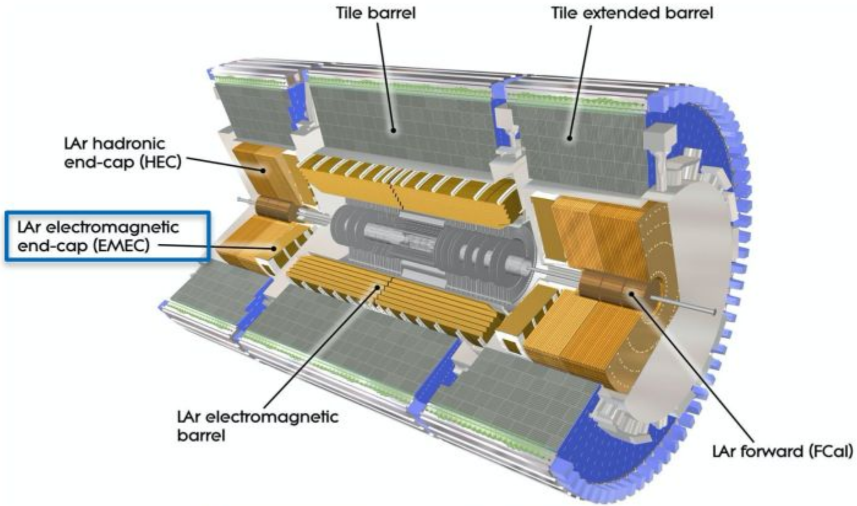
ATLAS as a new reference model

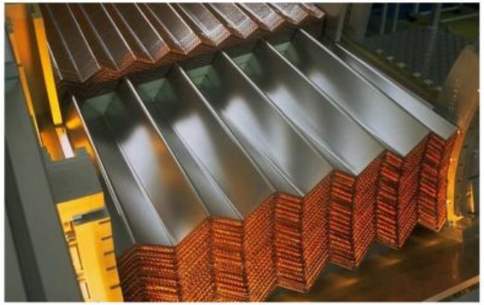
- **ATLAS experiment** as next reference model for QSS methods

- Explore simulation scenarios that use QSS methods integrated into the recently developed **FullSimLight** simulation prototype [1][2], a lightweight standalone Geant4 simulation tool that supports the full ATLAS geometry and the ATLAS magnetic field map

- Of particular interest is the simulation of the **EMEC** detector

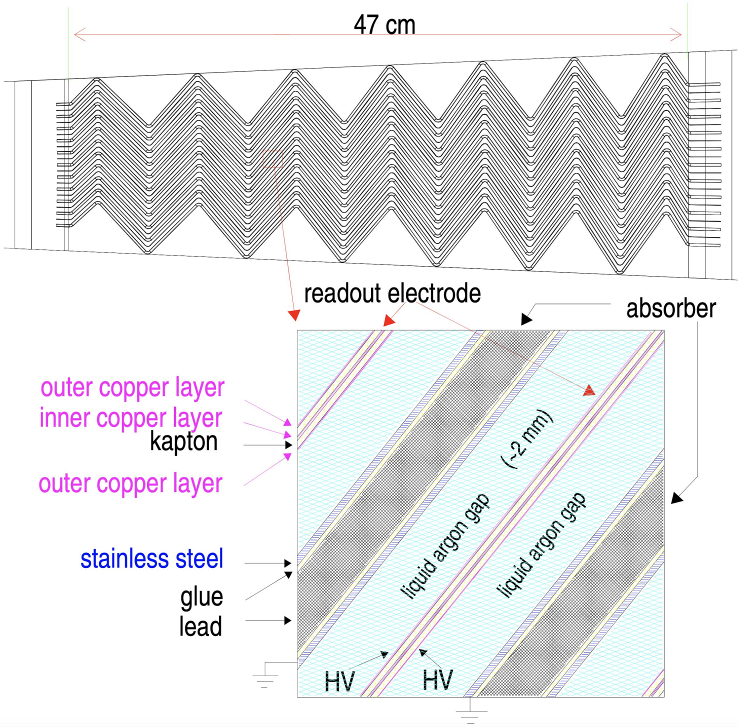
- The **EMEC (ElectroMagnetic End-Cap) [3]** is a lead-liquid argon sampling calorimeter with interleaved accordion-shaped absorbers and electrodes.
  - The accordion fold amplitude varies with the radius from the wheel center leading to a varying gap size
  - It has been implemented as a **custom solid** ***(GeoSpecialShape)*** in GeoModel and in Geant4





Picture of an electromagnetic end-cap module during stacking, showing the accordion structure of the ATLAS EM calorimeters.

ATLAS as a new reference model



- The EM calorimeters comprise accordion-shaped copper-kapton electrodes positioned between lead absorber plates and kept in position by honeycomb spacers while the system is immersed in LAr [1]

- The **EMEC** special shape is a well-known hotspot in the ATLAS simulation:
  - takes a significant amount of the total full Geant4 simulation CPU time: **~11.5%**

- The research hypothesis is that the **EMEC's densely layered geometry** is a very **suitable** application case where the efficient discontinuity handling property of QSS can be effectively leveraged.

Module / Class / Source Function / Call Stack	CPU Time ▼	Instructions Retired	Microarchitecture Usage	
			Microarchitecture Usage	CPI Rate
▸ libG4geometry.so	27.0%	26.9%	40.2%	0.665
▸ libG4processes.so	24.2%	21.4%	32.2%	0.693
▾ libGeoSpecialShapes.so	11.5%	14.2%	53.9%	0.571
▸ LArWheelCalculator_Impl::DistanceCalculatorSaggingOff	6.7%	8.9%	50.1%	0.563
▸ LArWheelCalculator	2.4%	3.3%	66.7%	0.570
▸ LArWheelCalculator_Impl::WheelFanCalculator<LArWheelCalculator_Impl::SaggingOff_t>	2.1%	1.8%	45.5%	0.610

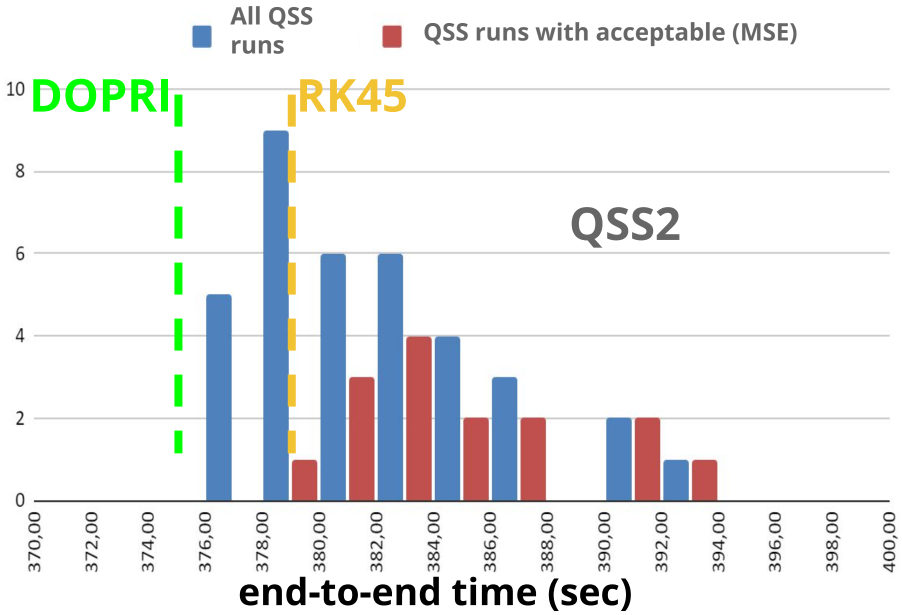
Contributed by Marilena Bandieramonte, U. of Pittsburgh

QSS Stepper in ATLAS - FullSimLight (G. Romczyk)

- First thorough validation of QSS2 in FSL

- Accuracy parameter sweeping:
  - dQRel=1e-4, dQMin=1e-7
  - dQRel=1e-5, dQMin=1e-8
  - dQRel=1e-6, dQMin=1e-9

- Preliminary conclusions:
  - QSS2 can achieve performance similar to the RK45 stepper in FSL (with acceptable accuracy)
  - More investigation is needed to see if extra performance gains can be achieved by **focusing QSS2 in the EMEC hotspot region**



- Experiment configuration in FSL:
  - **ATLAS Extension:** https://geomodel.web.cern.ch/home/fullsimlight/atlas-extensions/
  - **Geo File:** https://geomodel.web.cern.ch/atlas-geometry-data/geometry-ATLAS-R3S-2021-03-02-00.db
  - **Magnetic Field:** https://geomodel.web.cern.ch/atlas-magnetic-field/bmagatlas_09_fullAsym20400.data

New variants of QSS - testing phase (M. Portnoy)

- **newQSS**
  - **Reimplementation of the current QSS version** available in Geant4 release
  - More performant in some examples
  - Easier to understand and to extend

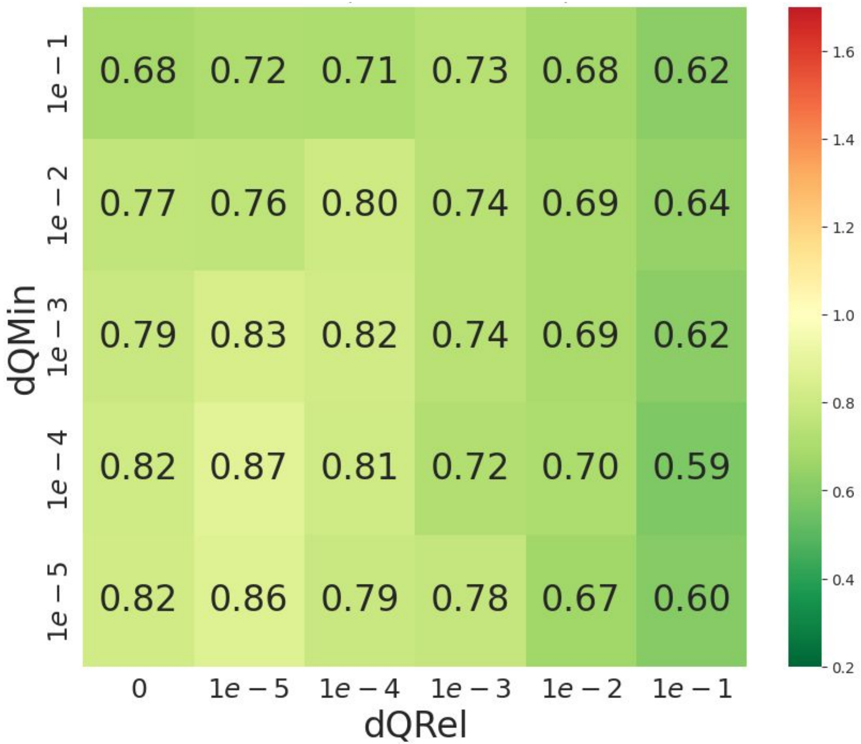
- **HelixMixedQSS** (experimental)
  - Combines with helix advances by *measuring the field variability*
  - *Good for slowly varying B fields*. Accuracy degrades with rapidly changing B fields.
  - Work in progress: Still need to fine tune some parameters of our heuristics

- **RotationQSS** (experimental)
  - A new ***coordinate system rotation-invariant*** version of QSS (orders 2 and 3)
  - Achieves better accuracy for a same set of QSS error control parameters (dQRel, dQMin)
  - Reduces some operations, but imposes an overhead that cancels out the performance boost.

- **Work in progress**
  - They all need further explorations to assess in what situations they yield a better performance

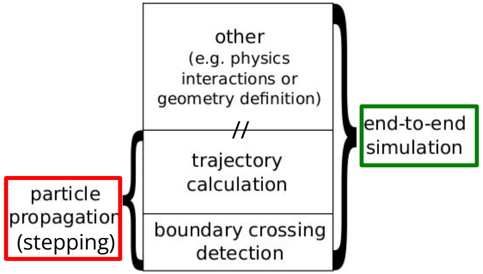
New Implementations: newQSS - Example: field01

QSS **stepping time**
newQSS2/**old**QSS2 **ratio**
Example: field01



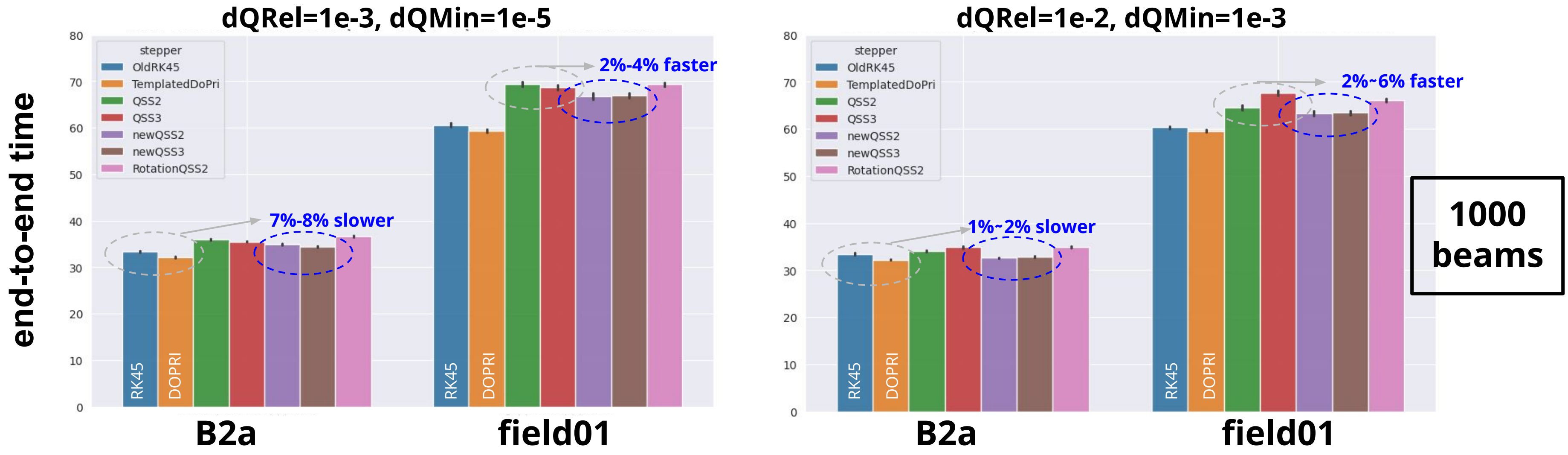
- Optimized re-implementation of the QSS family of Steppers in Geant4
- We can observe cases where the new versions present better performance

- The figure shows reduction of the simulation **QSS <u>stepping</u> time** *(not end-to-end time)*
- Improvements are *highly dependent on the type of example tested and its configuration*



Getting comparable to DOPRI in “unfavorable cases”

- The new methods show *comparable performance to DOPRI in some cases*
  - As expected, not particularly efficient for simulations with low number of intersections/step
- We reduced the QSS processing time on test examples
  - The figures shows end-to-end wall clock time for exampleB2a and field01 for 1000 beams

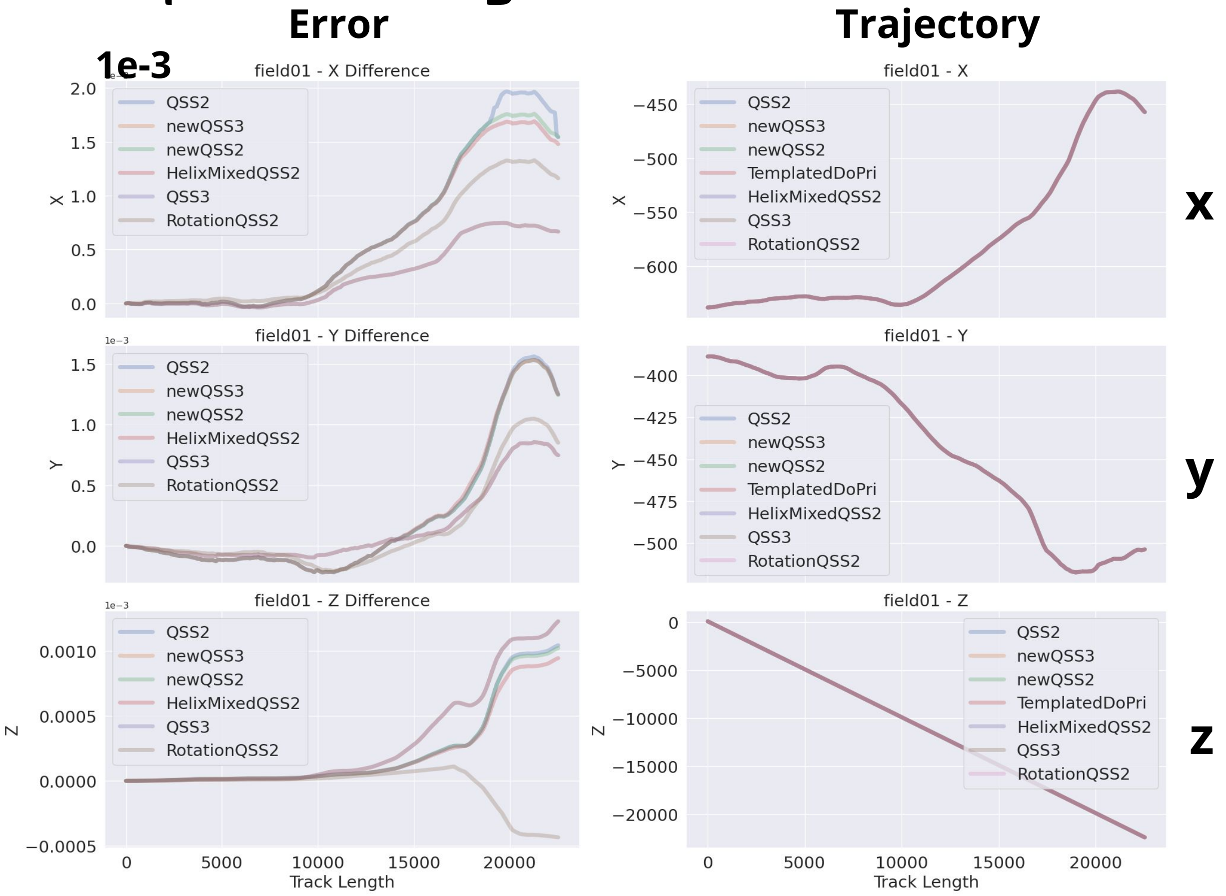


Accuracy analysis - Comparison against DOPRI

- **field01 example**

- We show the difference (error) on each of the 3 spatial coordinates, and the respective trajectories
    - *relative* error order of magnitude: <1e-6

- Trajectories are indistinguishable to the naked eye



Conclusions

- Performance gains in Geant4 achievable by QSS methods are a fact
  - Largely application-dependent
- We are progressing into a phase of more comprehensive benchmark-based performance characterizations
  - ATLAS recently added to the list, bringing in new particular challenges
    - Multi-Stepper approach?
  - CMS continues to serve as a reference model (add more test cases)
  - The impacts of the new QSS flavors (new-, Rotate-, HelixMixed-) on CMS and ATLAS need to be studied soon
- We entered into a more stable and productive stage
  - Solid automated benchmarking tools + new QSS methods to propose and test

Thank you

Backup Slides

QSS solvers for HEP applications

- Started as a collaboration with the Detector Simulation Group in Fermilab
  - w/Daniel Elvira & Team, Software for Physics Applications Dept., Scientific Computing Div.
- Since 2015 - Research on **efficient simulation** of particle systems (HEP and other apps.)
- 1 completed PhD Thesis (Santi)
- 3 completed Master's Thesis (Ponieman ,Rossi, Mignanelli*)
- 2 ongoing Master's Thesis (Grynberg Portnoy, Romczyk)
- 5 peer-reviewed publications
- Successful case of a HEP/Computer Science interdisciplinary collaboration
  - Results relevant and innovative both for the Physics and the Computer communities

Efficient discrete-event based particle tracking simulation for high energy physics
L. Santi, L. Rossi, and R. Castro
https://doi.org/10.1016/j.cpc.2020.107619

SED Discrete Event Simulation Lab

GEANT4 A SIMULATION TOOLKIT

Fermilab

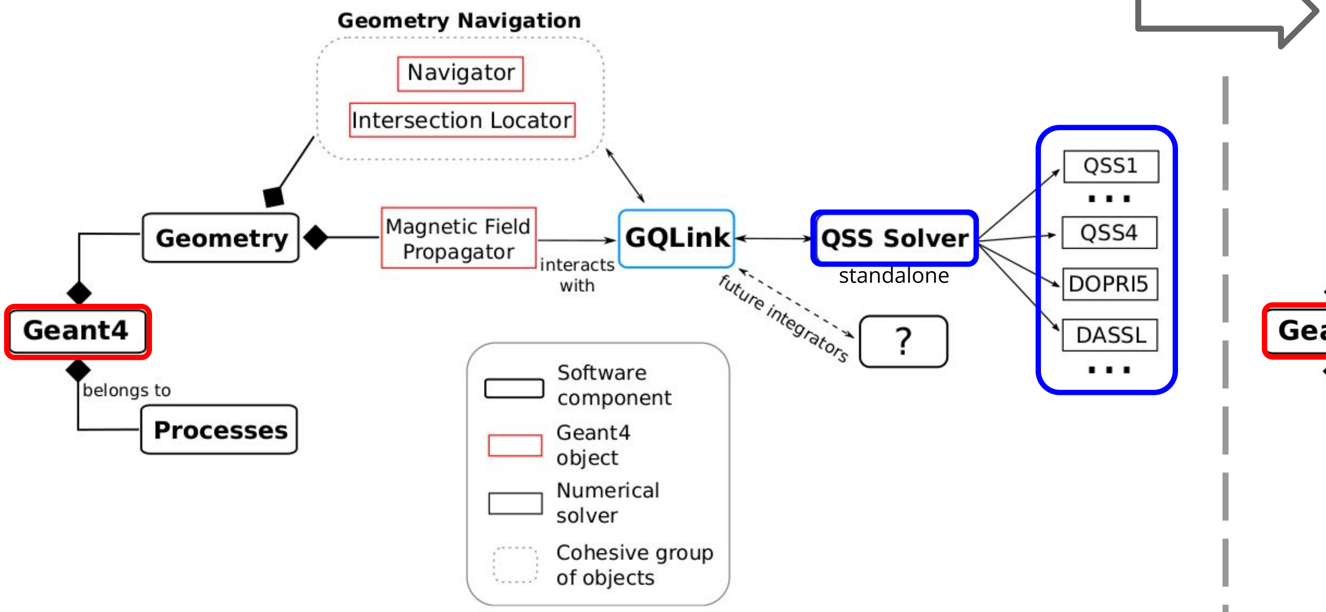
Accuracy analysis - Comparison against DOPRI

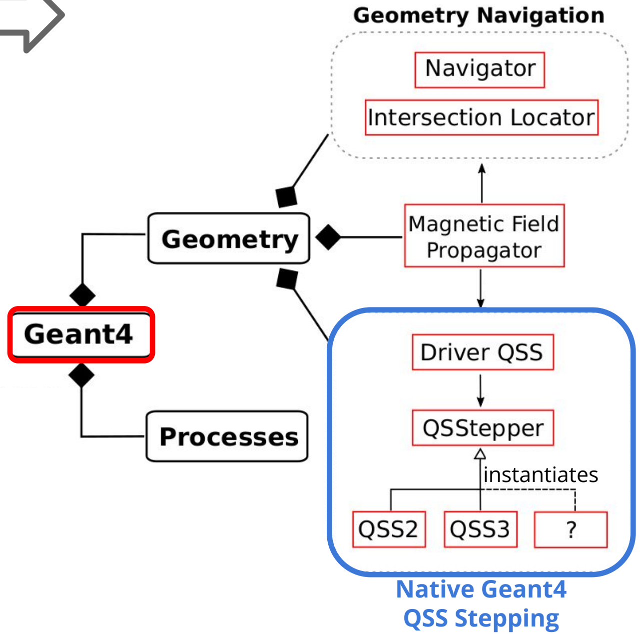
experiment	stepper	dQRel	dQMin	Real time	Substeps per Step	% intersection Per step	Speedup relative dopri
exampleB2a Multi-beam (1000)	newQSS2	0.001	1E-05	34.87	17.31	5.98	-8.40
	newQSS3	0.001	1E-05	34.47	7.85	5.81	-7.15
	OldRK45	N/A	N/A	33.37	0.00	N/A	-3.62
	RotationQSS2	0.001	1E-05	36.58	14.69	5.90	-13.71
	QSS2	0.001	1E-05	36.00	18.29	6.39	-11.91
	QSS3	0.001	1E-05	35.53	7.90	6.17	-10.45
	TemplatedDoPri	N/A	N/A	32.16	0.00	N/A	0.0
field01 Multi-beam (1000)	newQSS2	0.001	1E-05	66.74	8.10	0.11	-12.50
	newQSS3	0.001	1E-05	66.94	4.78	0.11	-12.85
	OldRK45	N/A	N/A	60.64	0.00	N/A	-2.22
	RotationQSS2	0.001	1E-05	69.33	7.14	0.11	-16.87
	QSS2	0.001	1E-05	69.43	8.12	0.11	-17.04
	QSS3	0.001	1E-05	68.67	4.77	0.11	-15.77
	TemplatedDoPri	N/A	N/A	59.32	0.00	N/A	0.0

Accuracy analysis - Comparison against DOPRI

experiment	stepper	dQRel	dQMin	Real time	Substeps per Step	% intersection Per step	Speedup relative dopri
exampleB2a Multi-beam (1000)	newQSS2	0.01	1E-03	32.59	5.08	5.88	-1.20
	newQSS3	0.01	1E-03	32.88	3.47	5.27	-2.08
	OldRK45	N/A	N/A	33.38	0.00	N/A	-3.64
	RotationQSS2	0.01	1E-03	34.94	4.54	6.01	-8.48
	QSS2	0.01	1E-03	34.10	6.74	5.62	-5.86
	QSS3	0.01	1E-03	34.89	4.43	5.86	-8.31
	TemplatedDoPri	N/A	N/A	32.21	0.00	N/A	0.0
field01 Multi-beam (1000)	newQSS2	0.01	1E-03	63.40	3.51	0.10	-6.49
	newQSS3	0.01	1E-03	63.47	2.83	0.10	-6.60
	OldRK45	N/A	N/A	60.31	0.00	N/A	-1.30
	RotationQSS2	0.01	1E-03	66.14	3.29	0.11	-11.08
	QSS2	0.001	1E-03	64.52	3.50	0.11	-8.36
	QSS3	0.001	1E-03	67.60	2.85	0.10	-13.54
	TemplatedDoPri	N/A	N/A	59.54	0.00	N/A	0.0

Integration with G4: High Level architectures



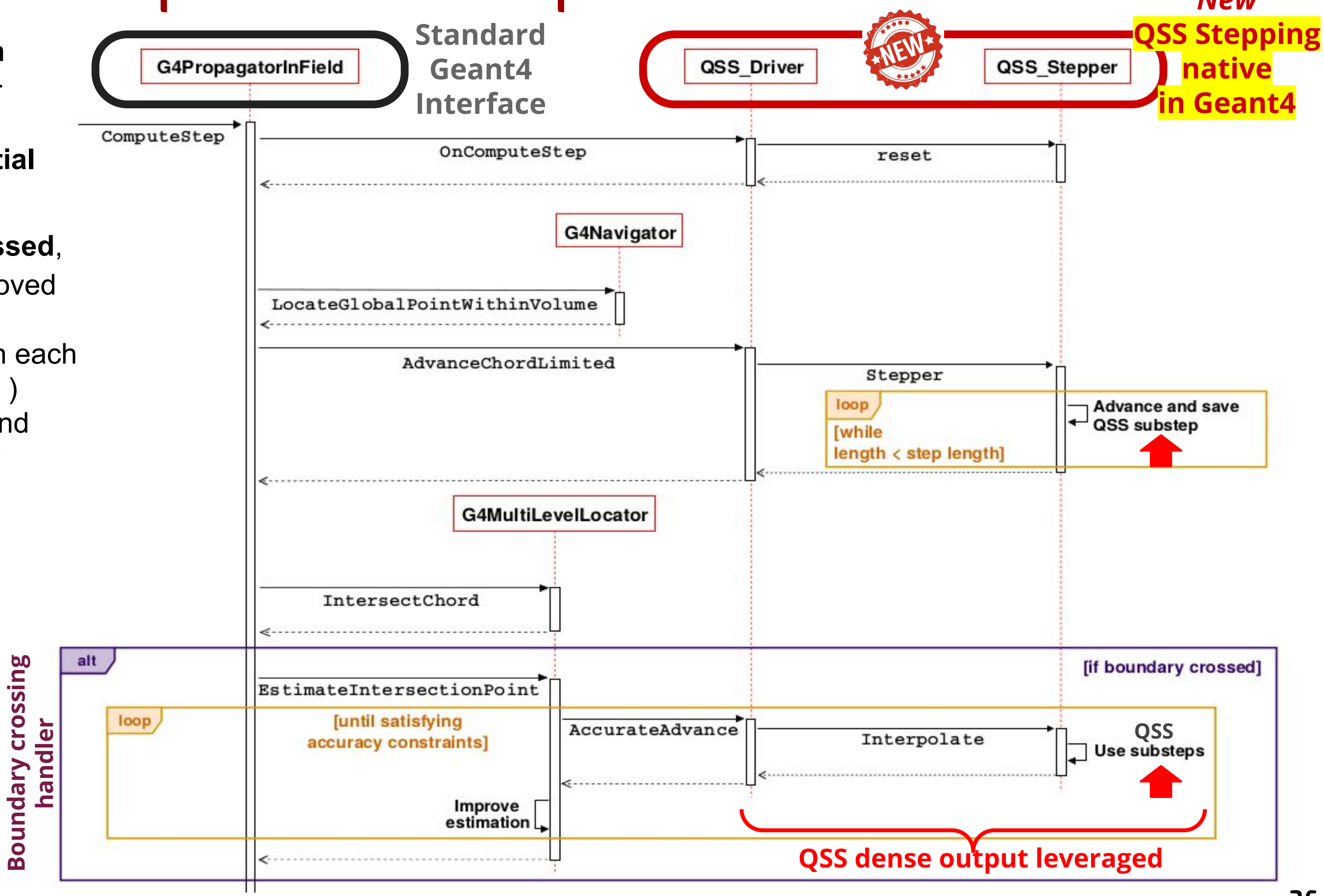


Strategy 2: "Embedded QSS"

- **QSStepper** for Geant4
- New native G4 Steppers

QSS-based step computation sequence in Geant4

- The **intersection-finding algorithm** starts with a quick test using a linear segment joining the step endpoints (`IntersectChord`) yielding an **initial estimation** of the intersection point
- In case a **volume boundary is crossed**, this estimation is progressively improved (`EstimateIntersectionPoint`, that queries the Integration Driver on each of its iterations (`AccurateAdvance`) in order to advance a given length and then test which side of the boundary the particle lies in
- The **QSS Driver**, by means of the Interpolation Driver's custom behavior, issues an `Interpolate` call to the **QSS Stepper**
- `Interpolate` **is handled very efficiently** leveraging the **polynomial QSS Substeps** previously computed and saved



Summary of results: QSS vs. DOPRI

Example	Method	QSS accuracy parameters dQrel	dQmin	% of Intersections per G4 Step	QSS Substeps per G4 Step	User Time (seg)	System Time (seg)	Real Time (seg)	Average Time per G4 Step (seg)	Speedup (QSS vs. DOPRI) Real Time
B2a	DOPRI	N/A	N/A	3.79%	N/A	2.052	0.175	2.614	1.3E-04	N/A
B2a	QSS	1.0E-02	1.0E-03	3.75%	10.191	2.067	0.176	2.654	1.3E-04	-1.53%
B2b	DOPRI	N/A	N/A	3.73%	N/A	2.081	0.178	2.651	1.3E-04	N/A
B2b	QSS	1.0E-02	1.0E-03	3.77%	10.209	2.107	0.178	2.680	1.3E-04	-1.09%
B4c (*)	DOPRI	N/A	N/A	4.31%	N/A	1.623	0.180	2.202	1.1E-03	N/A
B4c (*)	QSS	1.0E-02	1.0E-03	4.02%	2.517	1.603	0.182	2.170	2.1E-03	1.43%
B4d (*)	DOPRI	N/A	N/A	4.31%	N/A	1.637	0.183	2.217	1.1E-03	N/A
B4d (*)	QSS	1.0E-03	1.0E-04	4.19%	5.026	1.605	0.178	2.164	1.1E-03	2.39%
B5 SingleBeam (*)	DOPRI	N/A	N/A	2.78%	N/A	3.442	0.257	4.004	1.1E-01	N/A
B5 SingleBeam (*)	QSS	1.0E-03	1.0E-04	2.78%	1,494.940	3.259	0.245	3.841	1.1E-01	4.06%
Extended Field 01 (*)	DOPRI	N/A	N/A	6.51%	N/A	1.020	0.096	1.347	7.4E-04	N/A
Extended Field 01 (*)	QSS	1.0E-02	1.0E-03	5.99%	37.787	1.014	0.096	1.333	6.7E-04	1.03%
Extended Field 02 (*)	DOPRI	N/A	N/A	19.17%	N/A	1.270	0.124	1.612	9.7E-04	N/A
Extended Field 02 (*)	QSS	1.0E-02	1.0E-03	19.17%	3.056	1.265	0.128	1.610	9.7E-04	0.07%
Extended Field 03	DOPRI	N/A	N/A	14.76%	N/A	1.375	0.186	1.783	1.9E-04	N/A
Extended Field 03	QSS	1.0E-02	1.0E-03	9.99%	62.279	2.608	0.451	3.281	8.2E-05	-83.95%
Extended Field 06	DOPRI	N/A	N/A	0.08%	N/A	0.030	0.010	0.037	3.1E-05	N/A
Extended Field 06	QSS	1.0E-02	1.0E-03	0.08%	1.190	0.032	0.012	0.040	3.3E-05	-7.27%

(*) In all cases where **QSS is able to outperform DOPRI**, only the best combination of QSS accuracy parameters is shown (**rel**ative and **min**imum **Q**uantum **d**elta sizes, **dQrel** and **dQmin**). Other combinations may exist that could even perform worse than DOPRI.

Results highlights

- 11 examples tested and verified successfully:
  - **Basic** (B2a, B2b, B4c, B4d, B5), **Extended** (with magnetic field: 01, 02, 03, 06), **Advanced** (ams_ECAL)
  - **FullSimLight**, a lightweight standalone Geant4 simulation tool that supports the full ATLAS geometry and the ATLAS magnetic field map
- Benchmarks made against G4 (ver. 11.0.0-ref-02) with default stepper (DOPRI with Interpolation Driver)
- In 5 cases *there exist **QSS accuracy parameters*** that **can outperform DOPRI**
  - **However**, the ratio of geometry intersections per G4 step remains below 19% in all tested examples (typically around 5%) => these are **not** “QSS-friendly” scenarios (not “too many” intersections per step)
- Particle trajectories were compared visually using Paraview and VTK output files
- Benchmarking software: we continue developing a **toolset for repeatable benchmarking** that can be parameterized to produce systematic performance comparisons across G4 Steppers

Benchmark computing platform

- All experimentations carried out in CERN’s OpenLab (controlled environment)
- Hardware specs: Intel(R) Xeon(R) CPU E5-2683 v4 @ 2.10GHz (64 CPUs) 64 GB RAM

2023 Plans for QSS integration into the G4 release

- March/April
  - Goal: Integrate the already developed QSS capabilities (last integration: v10.5)
  - Incorporate members of the UBA Team (Simulation Lab, CS Dept.) to the Geometry and Transport WG
  - Initial tests, code housekeeping, documentation for final users.
- June/July/August
  - Goal: Include QSStepper into the Geant4 Quality Assurance regular procedures (collab. with Soon Yung Jun, Fermilab)
  - Reproduce benchmarks already run by the UBA Team in Argentina
  - Start adding more applications (based on the success of previous benchmarks)
- September/October
  - Goal: QSStepper in the next *development version*
  - Assess performance, identify bottlenecks and opportunities for improvements
  - Design/start new projects for extensions/refinements/enhancements
    - Typically advanced undergrad students, Master's Thesis, 6mo-1yr. Potentially a new PhD student
- November/December
  - Goal: QSStepper in the next *release version*
  - Design/start maintenance procedures/plan
  - More goals TBD according to the progresses made so far

Summary 2023

- Performance gains in Geant4 achievable by QSS methods are a fact
  - But also largely application-dependent
- We are entering a new phase of more comprehensive benchmark-based performance characterizations
  - CMS continues to serve as a reference model (add more test cases)
  - ATLAS to be soon added to the list, bringing in new particular challenges
- HEP as a provider of challenging applications for continued Simulation-specific R&D