

Finanziato dall'Unione europea NextGenerationEU



Ministero dell'Università e della Ricerca

The OpenGADGET3 code for cosmological simulations

- An update in preparation of the Key Science Projects -



Spoke 3 Technical Workshop, Bologna Dec 17-19, 2024

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





Milena Valentini

Dipartimento di **Fisica** Dipartimento d'Eccellenza 2023-2027







The Open GADGET3 code: a state-of-the-art code for HPC



ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing















Technical Objectives, Methodologies and Solutions

- TreePM+SPH code
- Highly optimised code: MPI parallelised + OpenMP
- Two hydro solvers: improved SPH formalism or MFM
- Two sub-grid models (Muppi, and one based on Springel&Hernquist 2003)
- Several modules for sub-resolution physics: star formation, stellar feedback, BH accretion and feedback, chemical enrichment, dust evolution, magnetic fields, cosmic rays
- Runs on CPUs and GPUs

MUPPI sub-resolution model

- description of a multi-phase ISM with H₂-based star formation
- thermal, kinetic, and low-metallicity stellar feedback
- improved cooling table interpolation
- stellar evolution and chemical enrichment
- angular-momentum-dependent gas accretion, dynamical friction, spin evolution
- isotropic, thermal AGN feedback + mechanical AGN feedback

formation and evolution of dust, and dust-assisted cooling

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





USM

The OpenGadget3 code











Technical Objectives, Methodologies and Solutions

- TreePM+SPH code
- **Highly optimised code:** MPI parallelised + OpenMP
- Two hydro solvers: improved SPH formalism or MFM
- **Two sub-grid models** (Muppi, and one based on Springel&Hernquist 2003)
- Several modules for sub-resolution physics: star formation, stellar feedback, BH accretion and feedback, chemical enrichment, dust evolution, magnetic fields, cosmic rays
- **Runs on CPUs and GPUs**

Core team in Trieste: S. Borgani, L. Tornatore, G. Murante, M. Valentini, T. Castro, P. Monaco, G. Taffoni, A. Damiano, G. Granato, D. Goz, P. Barai, M. Gitton-R., A. Saro, M. Viel



ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





USM

The OpenGadget3 code



Main tasks within the WP 2 of Spoke 3 –

Develop Open-GADGET further:

- including additional physics modules
- enhancing code modularity and readability
- improving code performance

Core teams in Trieste and Munich











- working strategy
- → Quite large (> 30 people from different institutes) user community



ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





penGadget3 - Development	
enGadget3 - Develop	ment ⋳ Star 2 % Fo
mits 😵 5 Branches 🔗 0 Tags vn	🗔 163.1 MiB Project Storage
e branch 'FG_MFM_Indent' into 'main_dev Karademir authored 2 hours ago	elopment' •••
elopment ~ OpenGadget3 / + ~	History Find file Edit
e CI/CD configuration	i 🗄 Add README 🗄 Add CHANGELOG 🗄 Add CONTRIBUTING 🔯 Configure Int
	Last commit
	Update CIPipeline.yml -> adding hydro tests to m
S	Update Verbose levels
	Update Makefile.Dorc as done by Klaus
/	Fixed isses with the natural constants defined
	remove un-initialized pmpotential (non)periodic f
r	Fix inconsistencies in comoving time integration f







- Section of the the files setting the many parameters.
- Construction of a reference structure for the file configure several reference production runs and files of parameters for the OpenGADGET3 c

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





The OpenGADGET3 project aims at making the use of the many complex physics modules more user friendly.

Substantial effort in cleaning and making more transparent the definition of the code configurations and of

	<u>Name</u>	Last modified	Size Description
es which	Parent Directory		_
•	Config_MV_Muppi.sh	2024-03-28 17:27	9.9K
code.	Config_MV_SH03.sh	2024-03-28 17:28	7.5K
	<u>Files_aux/</u>	2024-02-13 13:08	-
	DenGadget3.tar	2024-02-02 15:24	14M
	DpenGadget3/	2024-02-02 15:23	-
	<u>dfrogin 25x Muppi/</u>	2024-02-12 12:08	-
	<u>dfrogin 25x SH03/</u>	2024-02-12 12:08	-
	<u>dfrogin 250x Muppi/</u>	2024-02-12 12:08	-
	dfrogin 250x SH03/	2024-02-12 12:08	-
	<u>dianoga g15_25x_Muppi/</u>	2024-02-12 12:09	-
	<u>dianoga_g15_25x_SH03/</u>	2024-02-12 12:10	-
	magneticum box4 25x Muppi/	2024-02-12 11:57	-
	magneticum box4 25x SH03/	2024-02-12 12:09	-





- The OpenGADGET3 project aims at making the use of the many complex physics modules more user friendly.
- Substantial effort in cleaning and making more transparent the definition of the code configurations and of the files setting the many parameters.
- Construction of a reference structure for the files which configure several reference production runs and files of parameters for the OpenGADGET code.
- **Bug fixing** and tackling subtleties of the sub-grid modelling.



ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





- **Re-structuring** of the code (modularity)
- **Cleaning** the code **and documenting** its status







ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing









Adopting different numerical prescriptions for BH re-positioning has an impact on **BH dynamics**, AGN feedback, BH-BH mergers



DYNMASS

DYNFRIC

Event 1

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





Damiano, Valentini, Borgani, Tornatore+ 2024









1. GPU scalability

2. Performance issues

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





- OpenGadget has most of the modules running on GPUs (thanks to A. Ragagnin).
- We are assessing in detail the scalability of this implementation in order to highlight the blocking factors, mitigate their impact or turn to new strategies with greater parallelism

Detailed profiling with the assistance of POP and SPACE Centers of Excellence

Coordinator of the work: L. Tornatore





1) GPU scalability: Speed-Up

2×1024³, 120 Mpc, up to **512 GPUs**

1024³ -- from 008 to 128 Nodes



ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





2×2048³, 240 Mpc, up to **1024 GPUs**

2048³ -- from 064 to 256 Nodes







1) GPU scalability: more in detail

2×1024³, 120 Mpc, up to **512 GPUs**

1024³ TOTAL run time scaling-- from 8 to 128 nodes



ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing











ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing









1) GPU scalability: more in detail

The gravity tree has some noteworthy performance issues, mostly in

- Tree Walk \rightarrow Barnes&Hut is not GPU-friendly
- Communication

Communication & Nodes update have scalability issues in the SPH part, too.

- In the longer term (Dec 2025), we aim for a different implementation: We have extracted a kernel of the code which reproduces the conditions under which gravity is computed in OG3 and which will feature the new, restructured implementation of the tree, where
- the walk is done for a bunch of particles all together instead of for every 1. single particle, by grouping particles per tree node (they belong to);
- the Barnes and Hut scheme is not adopted anymore: rather, we opt for a 2. direct computation of the force within a given radius, to avoid to check whether nodes have to be opened and the tree walked further.

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing









Finanziato dall'Unione europea extGenerationEU



Ongoing: Assessing scalability, targeting performance issues

Report	Pecult		Size	Time	Cycles	GPU	SM Frequency	Process	Attributes			
Current profile	0.1.0 37 - main 49	5 7 7	(1488, 1, 1)x(32, 4, 1	1) 687.01 ms	853,127,893	0 - NVIDIA A100-SXM-6	54GB 1.24 Ghz	[2334582] toy apuized activelis	· 简			
Bacolino 1 profilo y	0.0 o		(1400 1 1) (22 4 1	1) 5450	6 70 / 072 002		4CP 1 24 Chz	[2224116] toy, anuized activalia	- 			
Baseline i prome.	0.0.0 37 - main_42	9	(1488, 1, 1)x(32, 4,	1) 5.45 5	0,764,973,863	0 - NVIDIA ATUU-SAN-0	0466 1.24 Ghz	[2334116] toy_gpuized_activelis	(- <u></u>			
Summary Details	Source Con	text Comme	ents Raw	Session	-					🗘 Compare 🚽 💥 Tools 🛫	◎ View 🚽 🕒	Export 🔵 🗏
	d 61											
Ine report contains importe	d source files.											
 GPU Speed Of Light Through 	put									GPU Thro	ughput Chart	Q v
High-level overview of the throug identify the highest contributor. I	hput for compute and ligh-level overview of	memory resource the utilization for c	s of the GPU. For eacl compute and memory	h unit, the throug resources of the	ghput reports th e GPU presente	he achieved percentage o ed as a roofline chart.	of utilization with res	pect to the theoretical maximum. B	reakdowns show the t	hroughput for each individual sub-metric of Com	pute and Memor	y to clearly
Compute (SM) Throughput [%]						63.01 (+694.32%	6) Duration [ms]				687.0	1 (-87.40%)
Memory Throughput [%]						19.84 (-48.51%	6) Elapsed Cycles [cycle]			853,127,89	3 (-87.43%)
L1/TEX Cache Throughput [%]						20.09 (+33.299	6) SM Active Cycle	s [cycle]			842,570,760.8	4 (-87.22%)
L2 Cache Throughput [%]						18.41 (-64.749	6) SM Frequency [0	ihz]			1.2	4 (-0.07%)
DRAM Throughput [%]						8.74 (-71.08%	6) DRAM Frequenc	y [Ghz]			1.5	9 (-0.21%)
너희 High Compute Throughp	ut Compute is more look-up tables. The following tal	e heavily utilized th ole lists the metric:	an Memory: Look at ti s that are key perform	he <u>> Compute W</u> ance indicators:		sis section to see what th	e compute pipelines	are spending their time doing. Also	o, consider whether an	y computation is redundant and could be reduced	d or moved to	•
	Metric Name		V	alue Guidano	ce							
	smthroughput	.avg.pct_of_peak_	sustained_elapsed 1	9.8406 63.010	- 19.841 >= 10.	.000						
FP64/32 Utilization Est. Speedup: 31.90%	The ratio of peak floa consider using 32-bit	t (fp32) to double (precision floating	(fp64) performance or point operations to im	n this device is 2 prove its perform	2:1. The kernel a mance. See the	achieved 0% of this devic e <u> Kernel Profiling Guide</u>	e's fp32 peak perfori for more details on	nance and 25% of its fp64 peak pe roofline analysis.	rformance. If <u>▶ Compu</u>	<u>ite Workload Analysis</u> determines that this kernel	is fp64 bound,	♥
	The following table lis	ts the metrics that	t are key performance	indicators:								
	Metric Name				Value Gui	iidance						
	smspsass_thread_	inst_executed_op_	.dfma_pred_on.sum.p	er_cycle_elapse	d 731.451 De	ecrease fp64 FMA instruc	tions					
	smspsass_thread_	inst_executed_op_	_dadd_pred_on.sum.pe	er_cycle_elapsed	d 304.771 De	crease fp64 ADD instruc	tions					
	smspsass_thread_	inst_executed_op_	dmul_pred_on.sum.pe	er_cycle_elapsed	d 243.817 De	crease fp64 MUL instruc	tions					
						GPU	Throughput					
Compute (SM) [%]												
Memory [%]												
0.0												
	10.0		20.0			40.0	50.0	60.0	70.0	80.0 90.0)	100.0

Comparison of two kernels through NVIDIA's NCU profiler

Branch Instructions [inst]	31,905,570,725 (-98.05%)	Branch Efficiency [%]
Branch Instructions Ratio [%]	0.15 (-1.46%)	Avg. Divergent Branches

Increased branch efficiency due to a much smaller thread divergence

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





1) GPU scalability: as for now...

Two strategies have been tested:

kernel 1: reproduces the standard OG3 tree walk strategy -> specific tree walk for each particle (each particle has a specific seed number)

kernel 2: reproduces a modified tree walk strategy, where the geometric centre of the node is considered instead of different tree leaves -> common tree walk for a bunch of particles

> Lower threads divergence, higher branch efficiency and better parallelism







	20	048(512x4)[1]	4096(1024x4)[2]	8192(2048x4)[3]		100				
Global efficiency	-	78.23	73.34	49.75		100	2) Performance is	sues: ve	ectorization	on
Parallel efficiency	-	78.23	73.74	55.54	- ;	80				
Load balance	-	85.89	83.68	71.74		(%)				
Communication efficiency	-	91.09	88.11	77.41	- (60 ⁶)	With the assistance of the	POP CoE,	and within	the SI
Computation scalability	-	100.00	99.47	89.59		ente ente	CoE, we are profiling in d	etails the co	ode's behav	iour.
IPC scalability	-	100.00	100.87	102.63		Pero				
Instruction scalability	-	100.00	98.69	86.81	- :	20	The results are summariz	ed in tables	s, as sketche	ed in t
Frequency scalability	-	100.00	99.91	100.56		_	figure on the left	ha aravity t		a diffe
		2048(512x4)[1] 4096(1024x4)[2] 8192(2048x4)[3]		0	metrics, columns refer to	the total nu	mber of three	e une eads)
Hybrid Parallel efficiency		- 78.23	73.74	55.54		100	from which come key indi	antoro ann l	aa aallaatad	I
MPI Parallel efficiency		80.33	76.67	68.64		80	nom which some key hu	calors carri	Je collected	
MPI Load balance		- 88.14	86.91	88.59				0040	4000	0100
MPI Communication efficiency		- 91.14	88.22	77.48	-	60 ^(%) a	Number of processes	2048	4096	8192
Serialization efficiency		-				Itag	Elapsed time (sec)	47.714394	25.446344	18.75
Transfer efficiency		-			-	40 J	Efficiency	1.0	0.937549	0.635
OpenMP Parallel efficiency		- 97.39	96.17	80.91		B	Speedup	1.0	1.875098	2.543
OpenMP Load Balance		97.45	06.20	80.02	-	20	Average IPC	0.961925	0.970340	0.987
		57.45	90.29	00.90			Average frequency (GHz)	3.190112	3.187294	3.207
OpenMP Communication efficie	ency	- 99.94	99.88	99.91		0		1	1	1

2010(0122/1/[1] 1000(1021/1/[1] 0102(2010/1/[0]	2048(512x4)[1]	4096(1024x4)[2]	8192(2048x4)[3]
---	----------------	-----------------	-----------------

	2048(512X4)[]	.] 4096(1024x4)[2]	8192(2048x4)[3]	- 100			1. A.	
fficiency -	78.23	73.34	49.75	- 100	2) Performance is	sues: ve	ectorization	on
el efficiency -	78.23	73.74	55.54	- 80				
balance -	85.89	83.68	71.74	(%				
munication efficiency -	91.09	88.11	77.41	- 60)) gge(With the assistance of the	POP CoE,	and within	the SI
utation scalability -	100.00	99.47	89.59	- 40	CoE, we are profiling in d	etails the co	ode's behav	iour.
calability -	100.00	100.87	102.63	Pero o.				
uction scalability -	100.00	98.69	86.81	- 20	The results are summariz	ed in tables	, as sketche	ed in t
uency scalability -	100.00	99.91	100.56		tigure on the left	ho arovity t	ant rouve or	o diffe
	2048(512x	4)[1] 4096(1024x4)[2	2] 8192(2048x4)[3]		metrics, columns refer to	the total nu	mber of three	e unie eads)
Parallel efficiency	- 78.23	73.74	55.54	- 100	from which come kay indi	actora con l	aa aallaatad	
arallel efficiency	- 80.33	76.67	68.64	- 80	nom which some key hu	Calors Carri	Je collected	
Load balance	- 88.14	86.91	88.59			0040	4000	0100
Communication efficiency	- 91.14	88.22	77.48	- 60 %	Number of processes	2048	4096	8192
Serialization efficiency	-			tage	Elapsed time (sec)	47.714394	25.446344	18.75
ransfer efficiency	-			- 40	Efficiency	1.0	0.937549	0.635
MP Parallel officiency	07.20	06.17	90.01	Per la	Speedup	1.0	1.875098	2.543
	57.59	50.17	00.91	- 20	Average IPC	0.961925	0.970340	0.987
enmp Load Balance	- 97.45	96.29	80.98		Average frequency (GHz)	3.190112	3.187294	3.207
enMP Communication efficien	icy- 99.94	99.88	99.91	0				

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing











-- Hybrid



Ongoing: Assessing scalability, targeting performance issues

		2048(512x4)[1]	4096(1024x4)[2]	8192(2048x4)[3]
Global efficiency	-	78.23	73.34	49.75
Parallel efficiency	-	78.23	73.74	55.54
Load balance	-	85.89	83.68	71.74
Communication efficiency	-	91.09	88.11	77.41
Computation scalability	-	100.00	99.47	89.59
IPC scalability	-	100.00	100.87	102.63
Instruction scalability	-	100.00	98.69	86.81
Frequency scalability	-	100.00	99.91	100.56

2010(0122/1/[1] 1000(1021/1/[1] 0102(2010/1/[0]	2048(512x4)[1]	4096(1024x4)[2]	8192(2048x4)[3]
---	----------------	-----------------	-----------------

		1	1	
Hybrid Parallel efficiency	-	78.23	73.74	55.54
MPI Parallel efficiency	-	80.33	76.67	68.64
MPI Load balance	-	88.14	86.91	88.59
MPI Communication efficiency	-	91.14	88.22	77.48
Serialization efficiency	-			
Transfer efficiency	-			
OpenMP Parallel efficiency	-	97.39	96.17	80.91
OpenMP Load Balance	-	97.45	96.29	80.98
OpenMP Communication efficiency	/-	99.94	99.88	99.91

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





2) Performance issues: vectorization 100 - 80 Percentage(%) The low IPC (Instructions Per Cycle), although constant with - 60 decreasing workload, indicates that the computational efficiency is not high. 40 Further inspection returned that in particular the - 20 vectorization ratio is very small (~10%) and limited to 128bits registers 0 the main target is to re-formulate the data structures - 100 that now consists in Arrays of (large)Structures - 80 Number of processes 2048 4096 8192 Percentage(%) - 60 Elapsed time (sec)47.714394 25.44634418.755917 Efficiency 0.6359911.00.937549- 40 1.875098 1.02.543965Speedup

0.961925

3.190112

- 20

Average IPC

Average frequency (GHz)

Missione 4 • Istruzione e Ricerca

0.970340

3.187294











Vectorization ratio achieved on average (= fraction of vector floating point (FP) instructions issued to the total number of FP instructions) under different assumptions.

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing





2) Performance issues: vectorization

We have tested the effect of different data layout on the achievable vectorization in a loop that reproduces the N-Body pattern, assuming that:

- A fraction of particle is active
- Every active particle interacts with its neighbours
- Neighbours are not close in memory

We experimented AoS, AosS and SoA with some carefully crafted loops to

- enhance auto-vectorization by the compiler (AoS, SoA)
- test compilers vector extensions (AosSv)
- explicitly use vector intrinsics (AosSi, SoAi)

Also, we have tested the effect of enhancing the memory contiguity (v1 VS v2) on different compilers (gnu VS intel)

Cons of vector instructions: every instruction requires more CPU cycles, the CPU frequency is generally decreased for an intense vector burst









2) Performance issues: vectorization

Credits: L. Tornatore



Results from LEONARDO DCGP, obtained by measuring performance counters via PAPI

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing

Timing run-time AOS AOS AOS I GOA GOAI POB ADS ADS ST SOR SOR A0505540551 50A 50A AOS AOS AOSSI GOA GOA VI 1CX VI GCC VZ ICX

- 1. A large vectorization fraction with the wrong data layout is not an advantage (e.g. AosSv) because a larger # of instructions is issued and the cpu frequency is decreased
- 2. Smaller structures offer ~10% of gain in terms of run-time (e.g. AosSv)
- **3.** Memory contiguity seems to be the most promising trick (go from v1 to v2), especially if the compiler is good in spotting opportunities (see icx vs gcc in v2.AoS)

Comparison of the required time per time step at different numbers of particles in each time bin.

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing

3) CPU optimization

Loop restructuring leads to a 2x performance in timesteps with a small # of particles (blue VS black curves)

Updates on the gradient computation and more precise memory allocation further increase the performance (red VS blue)

In total, these improvements speed up the calculation of the smallest time bins by up a factor of ~5 (red VS black).

Framework (developed within SPACE) to explore the topology of a given infrastructure and build a hierarchy of MPI communicators

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing

4) **Topology awareness** (= capability of the code to explore the NUMA topology of a machine)

A hierarchy of communicators groups the MPI tasks based on their NUMA affinity.

Every MPI task can understand on what node it is running and which are the other MPI tasks that run on the same node. The tasks running on the same node are grouped in a dedicated communicator and share the node memory via the MPI's shared-memory windows.

Every node has a designated master task that is in charge of MPI communications with other nodes, and the master tasks of all nodes participate in a dedicated MPI communicator.

Final goal: avoid too many communications and develop algorithms that are increasingly communication-free.

Next Steps and Expected Results

So far, results in line with timescale, milestones and KPIs identified.

EAGER: Evolution of gAlaxies and Galaxy clustErs in high-Resolution cosmological simulations

Stefano Borgani, Milena Valentini, Luca Tornatore, Alice Damiano, Alex Saro, Giuliano Taffoni, Tiago Castro

SLOTH: Shedding Light On dark matter wiTH cosmological simulations

Milena Valentini, Stefano Borgani, Tiago Castro, Luca Tornatore, Matteo Viel, Alice Damiano, Pierluigi Monaco, Giuliano Taffoni

ICSC Italian Research Center on High-Performance Computing, Big Data and Quantum Computing

Key Science Projects