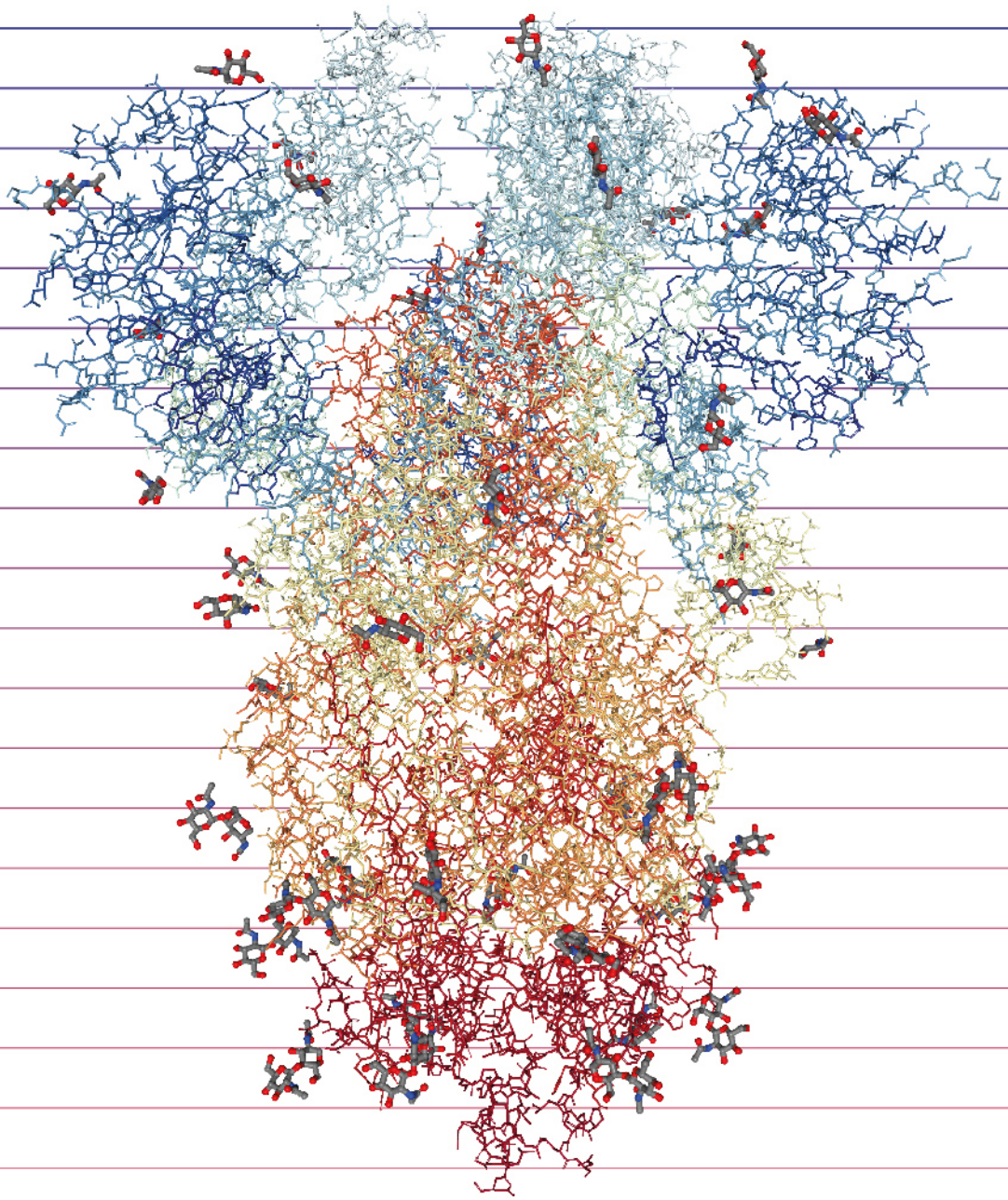
Italian National Agency for New Technologies,
Energy and Sustainable Economic Development

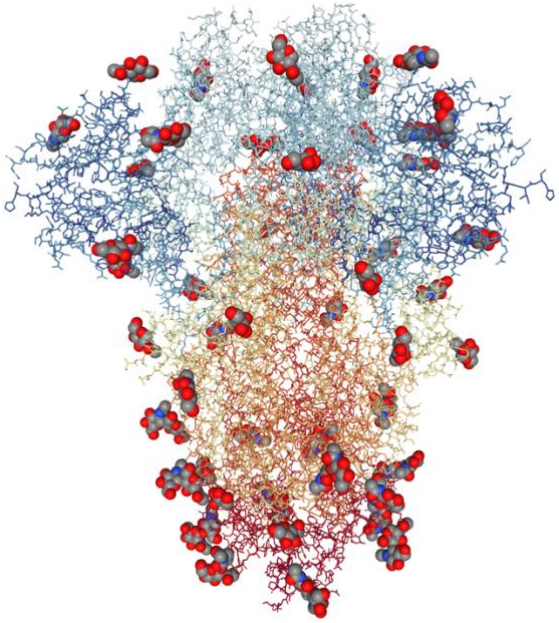
# ENEA CRESCO IN THE FIGHT AGAINST COVID-19



June 2021

Italian National Agency for New Technologies,
Energy and Sustainable Economic Development

# ENEA CRESCO IN THE FIGHT AGAINST COVID-19



SARS-CoV-2 spike glycoprotein

June 2021

Contributions provided by the Workshop:
ENEA CRESCO in the fight against COVID-19 – Jan.26th – Feb. 23rd 2021
Scientific Editor: Francesco Iannone, ENEA, TERIN-ICT-HPC, CR Frascati
Acknowledgements: We wish to thank
Cover: Flavio Miglietta, ENEA, REL-PROM, CR Frascati
ISBN: 978-88-8286-415-6

# Index

# FOREWORD

A two-day workshop dedicated to the exploitation of ENEA HPC CRESCO in the fight against COVID-19 was held on Jan.26th and Feb.23rd, 2021. The workshop topic was on the responses provided by computing scientist community to the COVID-19 pandemic emergency. The workshop talks covered a wide range of applications needing of HPC resources for numerical simulations manly in the research areas of Molecular Dynamics, Computational Fluid-Dynamic and Artificial Intelligence. Mostly of those talks are the result of collaborations between ENEA TERIN-ICT, universities and public research institutes for exploiting the computing resources of ENEA CRESCO: Computational RESsearch Centre on COmplex systems, able to provide HPC multicores-multiprocessors clusters interconnected by high bandwidth and low latency networks including high performance storage areas.
In the first day of the workshop, Giorgio Graditi, head of TERIN: Energy Technologies and Renewable Sources Department of ENEA, gave his welcome message to the participants emphasizing that ICT has a key enabling technology role to achieve the goals of the Green Deal for climate and environment an High Performance Computing, Artificial Intelligence and High Performance Data Analysis of BigData can help accelerate its realization. After that the workshop program for the first day was as follow:

| timetable | Program Jan.26th – Chair M.Celino |
|---|---|
| 9:30 | – *Welcome Message* – Giorgio Graditi head of TERIN: Energy Technologies and Renewable Sources Department of ENEA<br>– *ENEA HPC CRESCO in the time of COVID-19.* – F.Iannone - ENEA |
| 9.50 | – *Long timescale molecular dynamics and one trillion virtual screening on HPC5* – F.Frigerio - ENI |
| 10:20 | – *Advanced simulations on HPC5* – N.Bienati - ENI |
| 10:50 | – *Computational methods applied to the discovery of Sars-Cov-2 inibithors targeting the spike glycoprotein* – A.Romeo - University of Rome “Tor Vergata” |
| 11:20 | – *Bringing AI pipelines onto cloud-HPC: the classification of interstitial pneumonia example* – M.Aldinucci - University of Turin |

In the second day of the workshop, Silvio Migliori, head of TERIN-ICT: Information and Communication Technologies Division of ENEA, gave his welcome message to the participants emphasizing that ENEA ICT ENEA has always been an important player in scientific advanced computing and has achieved a relevant role onto the national HPC ecosystem already since the end of

2000 with the CRESCO project. The participation of the ENEA ICT division in national and international projects has allowed to repeatedly update the computing resources located in the ENEA Portici site, enhancing the overall investment and consolidating an activity based on the availability of advanced computing systems and enabling the growth of an important competence group. After that the workshop program for the first day was as follow:

| timetable | Program Feb.23rd – Chair M.Chinnici |
|---|---|
| 9:30 | – *Welcome Message* – Silvio Migliori head of TERIN-ICT: Information and Communication Technologies Division of ENEA<br>– *ENEA HPC CRESCO to support research on COVID-19.* – M.Celino - ENEA |
| 9.50 | – *Enhancing CFD simulations of COVID-19 diffusion by coughing and sneezing using data assimilation* – R.Arcucci - Imperial College London |
| 10:20 | – *Multiscale modelling of the Sars-Cov2 spike protein in interaction with the human ACE2 receptor* – S.Succi - IIT, M.Lauricella - CNR-IAC, L.Chiodo - Campus Biomedico University |
| 11:50 | – *Thermo-fluid dynamics of saliva droplets airborne diffusion in relation to Sars-Cov-2 transmission* – V.D'alessandro - Università Politecnica delle Marche |
| 11:20 | – *HPC-driven hit-to-lead process for Sars-Cov2 main protease inhibition* – M.Macchiagodena, P.Procacci - Università degli Studi di Firenze |

M.Celino, M.Chinnici,F.Iannone,S.Migliori
Dipartimento Tecnologie Energetiche e Fonti Rinnovabili
Divisione per lo Sviluppo Sistemi per l'Informatica e l'ICT

# ENEA HPC CRESCO IN THE TIME OF COVID-19 AND NEW SUPERCOMPUTING FRONTIERS

F.Iannone*, and *CRESCO team*:
D.Alderuccio, F.Ambrosino, G.Baldassarre, T.Bastianelli, R.Bertini, G.Bracco, L.Bucci, F.Buonocore, M.Caiazzo, B.Calosso, G.Cannataro, M.Caporicci, G.Carretto, M.Celino, M.Chinnici, R.Clemente, M.De Rosa, D.Di Mattia, G.Formisano, S.Ferriani, G.Ferro, A.Funel, D.Giammattei, S.Giusepponi, G.Guarnieri, M.Gusso, W.Lusani, M.Marano, A.Mariano, S.Migliori, M.Mongelli, P.Ornelli, S.Pagnutti, P.Palazzari, F.Palombi, S.Pecoraro, A.Perozziello, G.Ponti, S.Pierattini, M.Puccini, G.Santomauro, A.Scalise, F.Simoni, M.Steffè, D.Visparelli,

*Energy Technologies & Renewable Sources Department - Information Communication Technologies, Lungotevere Thaon di Revel, 76, 00196 Rome Italy*

* Corresponding author. E-mail: *francesco.iannone@enea.it*

**ABSTRACT.** High Performance Computing and Data Analysis (HPC/HPDA) infrastructures are important to face up the outbreak of SARS-COV2 in several application areas, such as: the design of new drugs, new physical barriers to reduce contagion as well as to plan strategies of containment, mitigation and suppression using predictive models for infection diseases. ENEA has made available CRESCO, its own HPC main facility with 1.4 Pflops peak computing power, for projects, requiring massive computing resources, related to mitigate the impact of COVID-19 pandemic. Within this framework, ENEA CRESCO has been carried out a huge amount of numerical simulations since spring 2020 providing to the scientific community a powerful tool for modelling the molecular dynamics (MD) of drug-receptor able to calculate actual binding free energy with accuracy as well as to carry out high-fidelity computational fluid dynamics (CFD) simulations of the transport mechanisms and related fluid dynamics of saliva droplets causing virus contagion. This paper provides the operations data of ENEA CRESCO gathered on the last year emphasising the considerable usage of the computing power in the middle of the Italian lockdown. Furthermore the paper describes the main numerical simulations carried out on ENEA CRESCO for applications relevant to counteract the pandemic as well as ENEA initiatives on the new supercomputing frontiers for the current exascale challenge and beyond for the quantum computing era.

## 1 Introduction

The HPC landscape is in the limelight for about two decades as hot topic both for computer scientists and end users. The level of expectations is increasing, motivated by the noticeable technical advances and what is announced at the horizon. The usage of a high fraction of the available processing power to solve real life problems is a central goal to achieve, as the gap between the theoretical performance and the sustained efficiency is more and more perceptible on modern supercomputers. From the scientific viewpoint, there are number of challenging achievements that are expected in order to come up with efficient and scalable computing solutions. Each involved topic is subject to intensive researches, with significant discoveries that are already effective. Solving large-scale problems in a short period of time using heterogeneous supercomputers is the main concern of the *exascale* challenge in the next few

years, whilst the sustainability in terms of energy efficiency pave the way for new generations of computing paradigms based on the quantum mechanics. The HPC is doing round breaking work that might not be possible without them, and this has changed the rules of science and industry. With computing possibilities running up against the far edge of current technology, researchers are looking for new ways to shrink processors, combine their power, and gather enough energy to make them all work efficiently. Computational capabilities are nowadays an essential part in cutting-edge scientific and engineering experiments. The capability to analyse and predict from huge amount data has incredibly improved with the use of the HPC and HPDA. Modern computational chemistry drives innovation in areas ranging from drug discovery to material design; neuroscientists can evaluate a large number of parameters in parallel to find good models for brain activity; automobile manufacturers can perform more realistic crash simulation to improve safety; astronomers can analyse different regions of the sky in parallel to search for supernovae; nuclear and particle physics are moving beyond common belief with large-scale simulations; search engines can launch parallel search across large-scale clusters of machines and instantly aggregates the results, thus reducing the latency of each request while improving relevance and accuracy; cryptography and computer systems security will benefit from the computation of gigantic prime numbers; researchers in artificial intelligence are trying to use large supercomputers to replicate (or surpass) a high-functioning human's ability to answer questions; social networking services are increasing their pervasiveness through large-scale Deep/Machine Learning processing. While keep striving to provide breath taking faster computers, designers need to contend with power and energy constraints. For decades, computers got faster by increasing their (aggregated) central processor unit. However, high processor frequency means lot of heat. This question of energy is more crucial as computing are being reported to the "Cloud", which is another innovative and affordable way to fulfil the need of high-range computing facilities. Indeed, Cloud computing offers a great alternative on mass storage, software and computing devices. Federating available computing resources, assuming a fast network, is certainly a valuable way to offer a more powerful computing system to the community. Energy, both dissipated and consumed, is also a critical concern, which is subject to active investigation from both the hardware and software standpoints.

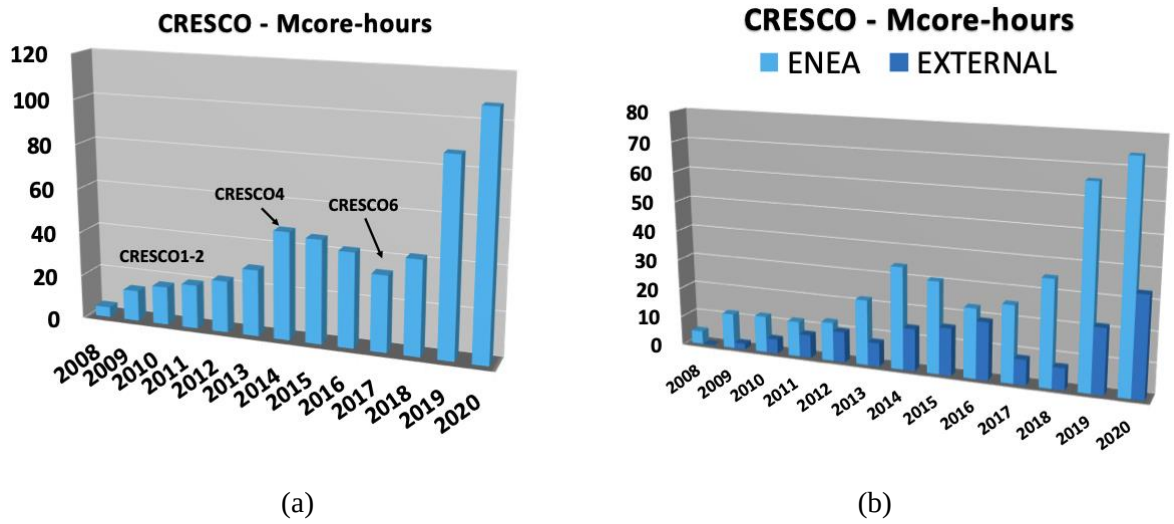
From the programming point of view, harvesting hardware advances to rich the level of cutting-edge research expectations is more challenging. Indeed, beside the ambient enthusiasm around the evolution of supercomputers, the way to peak performances is far from straightforward. In addition to algorithmic efforts to express and quantify all levels of parallelism, specific hardware and system considerations have to be taken into account when trying to provide an efficient, robust, and scalable implementation on (heterogeneous) multi-core processors. This has brought an unprecedented level of complexity in program design. Adapting a code for a given architecture or optimize it accordingly requires a complex set of program transformations, each of them addressing one or more aspects (e.g. registers, cache, instruction pipeline, data exchanges) of the target architecture. When the program is complex enough, or when the target architecture is a combination of different processing units (hybrid or accelerated computing), devising highly efficient programs becomes seriously hard. This is the price anyone should be aware of, when it comes to current and future states of HPC.

In the first part of this paper the statistic of CRESCO usage, during the 2020 pandemic year, will shown and the second part, it faces the main topics of the new challenges of the HPC landscape and beyond.

## 2 CRESCO usage

The CRESCO project, started in 2008, opened the ENEA HPC era of the HPC systems based on architecture of multicore/multiprocessor nodes interconnected by a low latency and high bandwidth networks. This architecture has been held in the various stages of the HPC systems scaling up driven

by the development of the CPU technology. Thanks to this development strategy, a community of users, internal ENEA and external, has grown more and more causing an exponential demand of computing time (figure 1).




(a) (b)

**Fig.1:** (a) CRESCO usage since 2008; (b) CRESCO usage since 2008 broken down in ENEA internal and external users

The figure 1(a) shows the growth in terms of millions of core-hours over about a decade, starting in 2008 with the first HPC systems, CRESCO1-2 with a peak power of 25 Tflops, moving in 2015 to CRESCO4 with a peak power of 100Tflops, up to CRESCO6 installed at its peak power of 1.4 Pflops in 2018. The figure 1(b) shows the same growing trend of the CRESCO usage broken down for ENEA internal and external CRESCO users. The usage of CRESCO HPC systems in 2020 is ~110 Mcore-hours corresponding to ~1.4 M€ with Amazon Web Service rate of a node similar to the CRESCO ones.

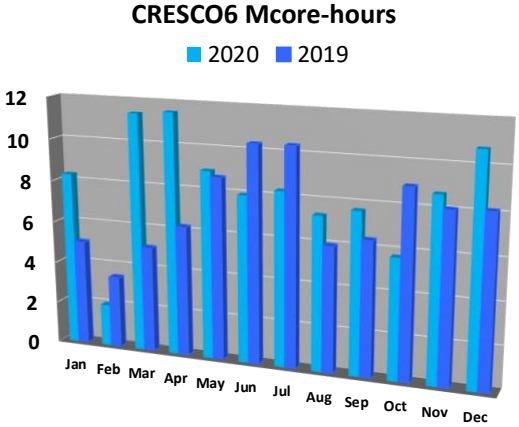
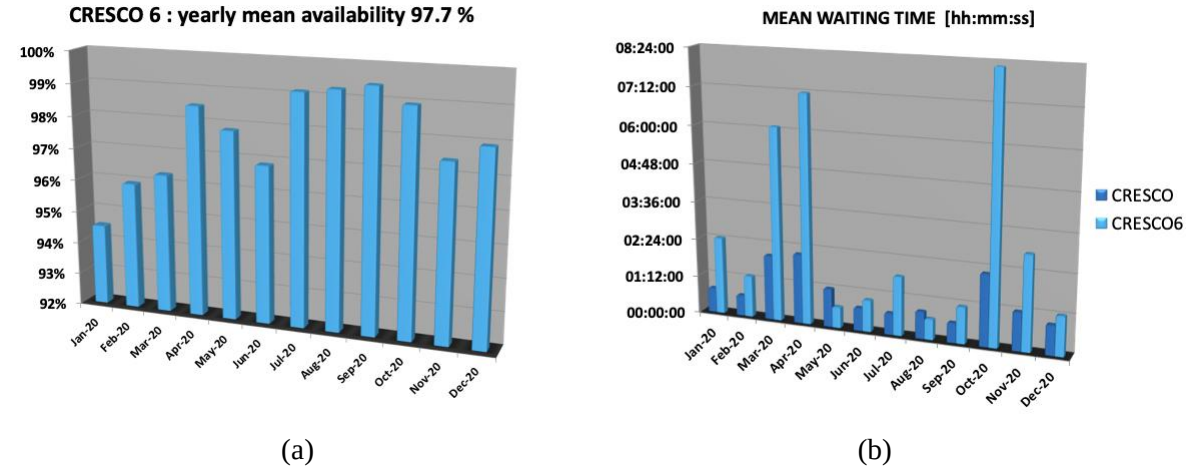



**Fig.2:** CRESCO6 usage in 2020;

What happened during the pandemic year is shown in the figure 2 CRESCO6 usage peak in 2020 was in the months of March and April, during the Italian lockdown whilst the usage peak in 2019 was as usual in the months of June and July.

To complete the statistics of the CRESCO6 operations in 2020, the figure 3(a) shows the availability that is 97.7 % as annual mean, whilst the figure 3(b) shows the wait main time of the jobs in the CRESCO and CRESCO6 queues.

The main numerical simulations, carried out during the year of pandemic, have used the Molecular Dynamic (MD) codes. Several of these codes belong to the open source material science community and they are undergoing the process of code-refactoring in order to run efficiently on exascale heterogeneous systems.




(a) (b)

**Fig.3:** (a) CRESCO6 availability in 2020; (b) Total CRESCO clusters and CRESCO6 mean waiting time in the queues in 2020;

Among the main MD codes undergoing the code-refactoring process include the following:
*Gromacs*: it is a versatile package to perform molecular dynamics, i.e. simulate the Newtonian equations of motion for systems with hundreds to millions of particles. It is primarily designed for biochemical molecules like proteins and lipids that have a lot of complicated bonded interactions, but since GROMACS is extremely fast at calculating the nonbonded interactions (that usually dominate simulations) many groups are also using it for research on non-biological systems, e.g. polymers [1].
*LAMMPS*: it is a classical molecular dynamics code, and an acronym for Large-scale Atomic/Molecular Massively Parallel Simulator. LAMMPS has potentials for solid-state materials (metals, semiconductors) and soft matter (biomolecules, polymers) and coarse-grained or mesoscopic systems. It can be used to model atoms or, more generically, as a parallel particle simulator at the atomic, meso, or continuum scale. LAMMPS runs on single processors or in parallel using message-passing techniques and a spatial-decomposition of the simulation domain. The code is designed to be easy to modify or extend with new functionality [2].
*CP2K*: it is a quantum chemistry and solid state physics software package that can perform atomistic simulations of solid state, liquid, molecular, periodic, material, crystal, and biological systems. CP2K provides a general framework for different modelling methods such as DFT using the mixed Gaussian and plane waves approaches GPW and GAPW. Supported theory levels include DFTB, LDA, GGA, MP2, RPA, semi-empirical methods (AM1, PM3, PM6, RM1, MNDO, …), and classical force fields (AMBER, CHARMM, …). CP2K can do simulations of molecular dynamics, metadynamics, Monte Carlo, Ehrenfest dynamics, vibrational analysis, core level spectroscopy, energy minimization, and transition state optimization using NEB or dimer method. CP2K is written in Fortran 2008 and can be run efficiently in parallel using a combination of multi-threading, MPI, and CUDA. It is freely available under the GPL license. It is therefore easy to give the code a try, and to make modifications as needed [3].
Some of these code have been used in customized implementation provided by users themselves as well as from ENEA CRESCO support. Some numerical simulations used molecular dynamics-based technique for the calculation of the absolute binding free energies in drug-receptor systems of the of the

main protease ($3CL^{pro}$) of the SARS-CoV2 [4]; another one used a customized version of LAMMPS to obtain a multiscale model of the SARS-CoV2 spike protein interacting with the human ACE2 receptor. Numerical simulations were carried out on HPC CRESCO based on a thermo-fluid dynamic models of saliva droplet diffusion using a customized version of Openfoam [5].

# 3 The exascale challenge

MD simulations are usually used successfully on HPC systems playing a significant role in drug design. The MD model is a *N*-body classical physical problem consisting to obtain the dynamic of *N* mass particle interacting according to a given force law. The problem arises in several contexts of the classical physic from molecular scale in structural biology systems to stellar scale research in astrophysics.

A typical simulation consists of a force evaluation step, where the force law and the current configuration of the system are used to the compute the forces on each particle, and an update step, where the dynamical equations (usually Newton's laws) are numerically stepped forward in time using the computed forces. The updated configuration is then reused to calculate forces for the next time step and the cycle is repeated as many times as desired. The simplest force models are pairwise additive, that is the force of interaction between two particles is independent of all the other particles, and the individual forces on a particle add linearly. The force calculation for such models is of complexity $O(N^2)$ as shown in the algorithm 1 of figure 4 with details in pseudo-code.

Since typical studies involve a large number of particles ($10^3$ to $10^6$) and the desired number of integration steps is usually very large ($10^6$ to $10^{15}$), the computational requirements often limit both the problem size as well as the simulation time and consequently, the useful information that may be obtained from such simulations. Numerous methods have been developed to deal with these issues. For molecular simulations, it is common to reduce the number of particles by treating the solvent molecules as a continuum.

**Algorithm 1**: n-body

```
set initial positions and velocities
for each time-step Δt do
    for each particle i-th do
        for each particle j-th do
            evaluate the force on j-th particle
        end for j-th particle
        integrate to calculate Δx for i-th particle
    end for i-th particle
end for time-step
```

**Fig**.4: n-body algorithm.

The current petascale HPC systems are based on architectures with conventional multi-cores processors, composing a SMP compute node, sharing main memory and interconnected to other nodes by means a low latency network. Thanks to a multi-thread handling in a SMP node, hybrid parallel applications, using MPI and OpenMP (or others threading API) are still suitable on petascale HPC systems with the growing the number of cores. So far the MD applications, based on *N*-body algorithms such as Gromacs, Lammps, have still a good speed up for simulations with 100M particles on large HPC systems running over 150k cores reaching performance of 24 ns/day. As *N*-body algorithm needs to update the whole configuration domains by means MPI all-to-all communications among nodes interconnected by low latency network, it is a biggest computational challenge for MD simulations as well as parallel applications based on numerical models strongly coupled, such us CFD, to scale-up on exascale HPC systems with tens of million cores in tens of thousands nodes consuming a huge quantity of electric energy. Hence the main challenge in the exascale transition is the containment of the power

consumption to a certain value, e.g. 20 mWatt per GFlops, while improving compute performance by, say, doubling at least the current top HPC systems. The issue of the efficiency loss, major and disruptive changes in hardware architectures are taking into account. There is increasing consensus that the constraints set by power consumption can only be met by heterogeneous architectures, with specialized core processors that maximize efficiency for a specific set of instructions. This will make the overall computations more time- and energy-efficient by mapping different compute-intensive tasks to different specialized processing units, allowing performance to grow while keeping the power budget under control. The ensuing architectural complexity will set nontrivial requirements in terms of data movement, heterogeneous memory management, and fault tolerance, which will most probably require a major, possibly joint, re-design of circuits and algorithms and the adoption of different programming paradigms. In the last decade, mainly because of the continuously increasing graphics processing demands of the video game industry, Graphics Processing Units (GPUs) have evolved into massively parallel computing engines. On the other hand, FPGA solutions can also offer high throughput to numerous data-intensive applications with critical time constraints in a reconfigurable environment.
In particular the FPGA accelerator boards got 2/4 QSFP network interfaces that allow to design HPC systems based on *heterogenous* architectures with network topologies: 1D torus or 2D mesh/torus with 2/4 node degrees and diameters as $n/2$ or $n^{1/2}$.
The heterogenous architectures based on FPGA got a level of flexibility able to set dynamically mixed precision methods suitable to save energy when solving $Ax=b$ using Iterative Refinement (IR). The idea of the mixed precision method is to utilize a low precision for $O(N^3)$ LU solver, while attaining a solution accuracy through $O(N^2)$ refinement, where $N$ is a matrix size. Further the energy efficiency can be improved shifting thermal evaluation from the chip design phase to the run-time thermal management. In this context, accurate, fast and flexible thermal simulators help understand the power dissipation requirements, tailoring the cooling to the chip requirements to best utilize HPC infrastructures while keeping cooling costs at a minimum and enabling run-time thermal management. It can be implemented with more flexibility on programmable accelerator boards FPGA based.
Starting from European Processor Initiative activities on the stencil/tensor accelerator, boosting it using mixed-precision/trans-precision arithmetic and/or Posits and also developing an accelerator with an hardware posit processing unit for HPC computation. Alternative to floats Posits are promising to increase bandwidth and memory efficiency and hence boosting performance and saving power either for AI services (Posit8 instead of float16/32), or for scientific computation (posit16/32 instead of float64/128). In order to design HPC heterogenous system running on FPGA accelerators, the availability of a High-Level Synthesis tool and the possibility to use in the flow pre-designed computation/interfacing Intellectual Property (IP) is mandatory to lower the access barrier currently limiting the widespread adoption of FPGA devices. In this context several toolchains provide new programming models for developing on GPU/FPGA accelerators.
One of the basic guidelines in energy efficient computing is the optimization and the acceleration of algorithms and software libraries that provide a reduction of the elapsed time of HPC applications and, as consequence turn, a significant cut in energy consumption.
The new power-to-solution metrics requires a rethinking of many computational kernels of HPC applications looking for a trade-off between the reduction of the total energy and the minimization of the time-to-solution, promoting scalability also for solving ever larger problems as required by high-resolution simulations and big data applications.
Within this context, new open-source high-performance algorithms and software libraries for some, among the most widely used, kernels in numerical linear algebra and graph computation, shall be deployed. New algorithms shall be designed and optimized having as target platforms clusters of hybrid nodes, with thousands of simple computing units and a memory hierarchy that is much more exposed to the developer's control with respect to the traditional multi-level cache based systems. It is not

unusual for algorithms, inefficient on traditional computing platforms, to become very much competitive on accelerators like GPU because additional computations are well tolerated and convenient when using complex memory access patterns. On the new architectures also data structures may need an in-depth revision. As for GPU, for instance, thread-locality rather than data-locality must be privileged. Single-precision floating-point arithmetic offers many advantages in terms of memory footprint and computational efficiency on GPUs (latest generation GPUs also offer a half-precision, 16 bit-based floating-point arithmetic). Although in general algorithms should be oblivious to the precision of the floating-point arithmetic, at the level of software implementation, any algorithm must be double checked under different conditions (size of the problem, range of the values that the variables can assume, presence of reduction operators), in order to guarantee that single precision and potentially half precision can be safely used, within several iterative refinement techniques for obtaining user's required accuracy.

Heterogenous architecture based on GPU/FPGA is already used to reduce the huge processing time in MD simulations. Its programming model provides a top level abstraction for low level hardware routines as well as consistent memory and execution models for dealing with massively-parallel code execution. A standard programming model is OpenCL, set from Khronos, for heterogenous parallel computing on cross-vendor and cross-platform hardware. Figure 5 shows an overview of the OpenCL architecture. One CPU-based "Host" controls multiple "Compute Devices" (for instance CPUs, GPUs & FPGAs are different compute devices). Each of these coarse grained compute devices consists of multiple "Compute Units" (akin to execution units & arithmetic processing unit groups on multi-core CPUs - think "cores") and within these are multiple "Processing Elements". At the lowest level, these processing elements all execute OpenCL "Kernel Functions".

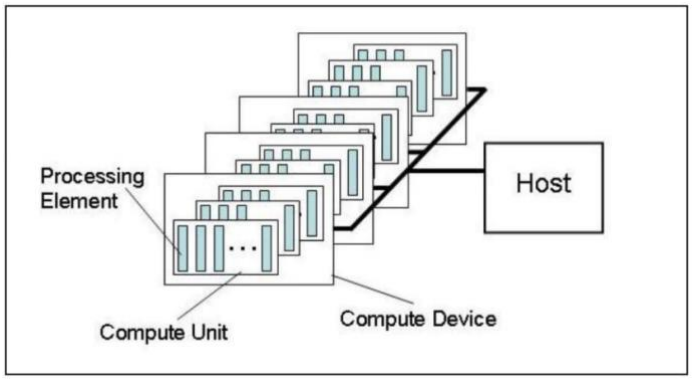



**Fig.5**: OpenCL programming model set by Khronos.

The "Kernel Functions" can be thought of as functions that transform each element of an input stream into a corresponding Processing Element of an output stream. When expressed this way, the "Kernel Function" can be applied to multiple "Processing Elements" of the input stream in parallel. Instead of blocking data to fit caches, the data is streamed into the compute units. Since streaming fetches are predetermined, data can be fetched in parallel with computation.

Since all coordinates are fixed during the force calculation, the force computation can be parallelized for the different values of *i*. In terms of streams and kernels, this can be expressed as the kernel function shown in pseudo-code of figure 6.

The kernel is applied to each element of the stream coordinates to produce an element of the forces stream. Note that the kernel can perform an indexed fetch from the coordinates stream inside the *j-loop*. An out-of-order indexed fetch can be slow, since in general, there is no way to prefetch the data. However in this case the indexed accesses are sequential. Moreover, the *j-loop* is executed

simultaneously for many *i-particles*; even with minimal caching, the position of the *j-th* particle can be reused for many *N i-particles* without fetching from memory thus the performance of this algorithm would be expected to be high.

**Algorithm 2**: n-body kernel function

```
__kernel void kforce(global double i-th partcle)
        for each particle j-th do
            evaluate the force on j-th particle by i-th particle
        end for j-th particle
end __kernel kforce
```

**Fig.6**: N-body kernel function algorithm.

There are dozens of MD packages in production use, many of which have been successfully accelerated with GPUs. Scaling, however, remains problematic for the small simulations (20K - 50K particles) commonly used in critical applications, e.g., drug design, where long timescales are also extremely beneficial. FPGAs have been explored as possible MD accelerators for many years [6–12]. The first generation of complete FPGA/MD systems accelerated only the range limited (RL) force and used CPUs for the rest of the computation. While performance was sometimes competitive, high cost and lack of availability of FPGA systems meant that they were never in production use. In the last few years, however, it has been shown that FPGA clusters can have excellent for the Long Range force computation (LR) [17–20], the part of MD that is most difficult to scale.

It remains to be demonstrated, however, whether a single FPGA MD engine can be sufficiently competitive to make it worth developing such a cluster. And if so, how should it be implemented? One thing that is certain is that previous CPU-centric approaches are not viable: long timescales require ultra-short iteration times which make the cost of CPU-device data transfers prohibitive. This leads to another question: is it possible to build such an FPGA MD engine where there is little interaction with other devices? One advantage with current FPGAs is that it is now possible–for simulations of great interest (up to roughly 40K particles)–for all data to reside entirely on-chip for the entire computation. Although this does not necessarily impact performance (double-buffering off-chip transfers still works), it simplifies the implementation and illuminates a fundamental research question: what is the best mapping among particles, cells, and force computation pipelines? Whereas the previous generation of FPGA/MD systems only dealt with a few cells and pipelines at a time, the concern now is with hundreds of each. Not only does this lead to a new version of the problem of computing pairwise forces with cutoff (see [21, 22]), it also requires orchestrating LR with the other force computations, and then all of those with motion update and particle movement.

| *Kernel* | *Formula* | *Flop per Interaction* |
|---|---|---|
| *Coulomb* | $\frac{q_j}{r_{ij}^3}\bar{\boldsymbol{r}}_{ij}$ | *19* |
| *LJC (constant)* | $\frac{q_i q_j}{r_{ij}^3}\bar{\boldsymbol{r}}_{ij} + \epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^6 - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12}\right]$ | *30* |
| *LJC (linear)* | $\frac{q_i q_j}{r_{ij}^4}\bar{\boldsymbol{r}}_{ij} + \epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^6 - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12}\right]$ | *30* |
| *LJC (sigmoidal)* | $\frac{q_i q_j}{\zeta(r_{ij}) r_{ij}^3}\bar{\boldsymbol{r}}_{ij} + \epsilon_{ij}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^6 - \left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12}\right]$ <br> $\zeta(r) = e^{(\alpha r^3 + \beta r^2 + \gamma + \delta)}$ | *43* |

**Tab.1**: Forces formulas and number of float pointing operations per instructions in MD algorithms.

Basically the main MD algorithms are force calculation and time integration based on the *n-body* algorithm. The forces computed depend on the system being simulated and may include bonded terms, pairwise bond, angle, and dihedral; and non-bonded terms, van der Waals and Coulomb [13]. It allows to avoid $O(N^2)$ calculations treating the solvent as a continuum model. In such models, the quantum interaction of non-bonded atoms is given by a Lennard-Jones function and the electrostatic interaction is given by Coulomb's Law suitably modified to account for the solvent. The LJC(constant) kernel calculates the Coulomb force with a constant dielectric, while the LJC(linear) and LJC(sigmoidal) kernels use distance dependent dielectrics. The equations used for each kernel as well as the arithmetic complexity of the calculation are shown in Table 1.

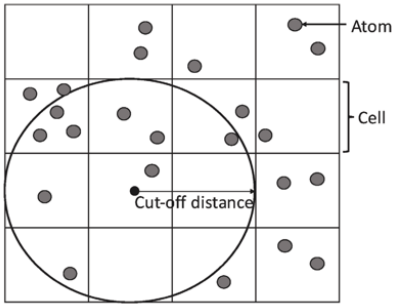



**Fig.7**: Division of the simulation box in to cells.

There are several techniques available to reduce the computation cost and to accelerate non-bonded force computation. MD simulation is done for a system that is usually represented by a box of atoms. To reduce the computation complexity, the box is divided in to multiple cells.

The figure 7 shows a 2-D representation of the cell division. A cut-off distance is set between two atoms and the neighboring cell-pairs within the cut-off distance are extracted to a cell-pair list. Non-bonded force computation is done for the atoms of the cell-pairs in the list. As a result, all atom-pair combinations for the force computation do not have to consider. Since the atoms move in the box, the cell-pair list is updated in each iteration. A periodic boundary condition is used when an atom leaves the box. Usually the same box is replicated at the boundaries so that an atom leaves from the box reappears from the opposite direction. Using this method, a large system can be simulated by using only a small number of atoms. Even with these techniques, MD simulation takes a huge amount of processing time. Non-bonded force computation occupies most of the total processing time. Therefore, the Non-bonded force computation is accelerated using FPGA.

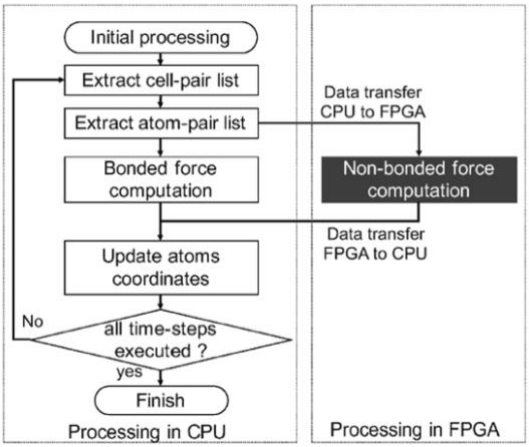



**Fig.8**: Flow-chart of the CPU-FPGA for MD algorithms in heterogeneous architecture.

In order to implement a parallel pipeline architecture for FPGA, the force computation is separated from the atom-pair selection. First the complete list of atom-pairs based on the cell-pair list is extracted. Then the force computation is performed for each atom-pair in the list. The atom pair-list extraction is just a searching procedure that does not contain heavy computations. On the other hand, force computation contains many multiplications and divisions. Therefore, the host computer is used for atom-pair list extraction and transfer the list to the FPGA for force computation.
Once the list is extracted, only a single loop is sufficient for the force computation of all the atom-pairs in the list. As a result, loop-pipeling can be implemented on FPGA to accelerate the computation. The figure 8 shows the flow-chart of the CPU/FPGA heterogeneous architecture. Bonded-force computation is done on CPU while non-bonded force computation is done on FPGA. After computing all the forces, the atom coordinates are updated using the Newton's equations. Then a new atom pair-list is extracted. In this method, we there are two overheads, atom-pair-list data transfer to FPGA and force data transfer from FPGA. The FPGA board is connected to the CPU through a PCI express bus. Initially, the atom-pair list and the atom coordinates are transferred from the host computer to the global memory (DRAM) of the FPGA board. After the computations are done on the FPGA board, force data are read by the host computer. This data transfer is done through the PCIe port of the host computer motherboard. The FPGA accelerator read the input data from the global memory and performs the computation. The outputs are written back to the global memory. The data read, computation and write-back is fully pipelined, so that force data are written to the global memory in every clock cycle after the pipeline is filled. Since the proposed architecture is completely designed by software, the same program code can be reused by recompiling it for any OpenCL capable FPGA board. Any future algorithm change can be also implement by just updating the software and recompiling it by using just few hours of design time. However, the data transfers between CPU and FPGA is still a problem. This problem can be solved by future SoC based FPGA boards that contain a multicore CPU and an FPGA on the same chip. Therefore, PCI express based data transfers can be replaced by much faster on-board data transfers. To use shared memory may be also able to completely eliminate data transfers. The heterogeneous computing system can contain multiple FPGAs and they can be connected to build a scalable computing cluster.

# 4 The Quantum Computing era

As envisioned by the great 20$^{th}$ century physicist Richard Feynman, hope for simulating increasingly complex quantum systems lies in a new paradigm for information processing: quantum computation. Such computers can store and process information about simulated quantum systems natively, reducing the computational resource scaling with the size of the system to just polynomial growth, in principle. In the thirty years since Feynman's forecast, the field of quantum information science has emerged and has made tremendous strides: experimental advances in controlling quantum systems have brought quantum computers to the brink of outperforming classical computation; the synergy between quantum chemistry and quantum algorithm development has continued to equip and refine the toolbox of disruptive applications for quantum computation; and at the interface between computational chemistry and quantum information science, researchers are poised to carry Feynman's vision to fruition. Richard Feynman's original idea was to simulate quantum many-body dynamics – a notoriously hard problem for a classical computer – by using another quantum system [14].

**Theory of Quantum Computing**

A quantum computer, on the other hand, uses quantum bits, or qubits. A qubit is a quantum system that encodes the zero and the one into two indistinguishable quantum states. Using Dirac notation [15], a

qubit can be in the state $|0\rangle$ or $|1\rangle$, or (more importantly) a superposition (linear combination) of these states. Mathematically, the state of a single qubit $|\psi\rangle$ is:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \qquad (1)$$

such that $\alpha, \beta \in \mathbb{C}$. The coefficients also must follow a normalization condition of $|\alpha|^2 + |\beta|^2 = 1$. In the above state, the complex numbers $\alpha$ and $\beta$ are known as *amplitudes*. The states $|0\rangle$ and $|1\rangle$ are known as basis states. Importantly, given any state $|\psi\rangle$, it is impossible to extract the amplitudes of any basis state. Commonly used is the vector notation for states. The basis states $|0\rangle$ and $|1\rangle$ are vectors that form an orthonormal basis for that qubits state space. The standard representation is:

$$|0\rangle = \begin{pmatrix}1\\0\end{pmatrix}, \qquad |1\rangle = \begin{pmatrix}0\\1\end{pmatrix}, \qquad (2)$$

Following from this, the state $|\psi\rangle$ can be represented as a unit vector in the two-dimensional complex vector space,

$$|\psi\rangle = \begin{pmatrix}\alpha\\\beta\end{pmatrix} \qquad (3)$$

The concepts here generalize to quantum systems containing many qubits. Since a single qubit has two distinct basis states, an n qubit system has $2^n$ distinct basis states. In quantum computing, a multiple qubit system is known as a register. To combine the states of two individual qubits, the Kronecker/tensor product must be used. For example, to combine the states two qubits $|\psi_1\rangle$ and $|\psi_2\rangle$,

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{pmatrix}\alpha_1\\\beta_1\end{pmatrix} \otimes \begin{pmatrix}\alpha_2\\\beta_2\end{pmatrix} = \begin{pmatrix}\alpha_1\alpha_2\\\alpha_1\beta_2\\\beta_1\alpha_2\\\beta_1\beta_2\end{pmatrix} \qquad (4)$$

When basis vectors are combined, it is convention to say $|1\rangle \otimes |0\rangle = |10\rangle$ or $|2\rangle$ (as *'10'* is *2* in binary. More generally, An *n* qubit register is described by a unit vector $|\phi\rangle$ in the $2^n$ dimensional complex vector space,

$$|\phi\rangle = \begin{pmatrix}\alpha_0\\\alpha_1\\\vdots\\\alpha_{2^n-1}\end{pmatrix} \qquad (5)$$

This is equivalent to a linear combination of the basis states:

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \alpha_j \, |j\rangle \qquad (6)$$

Where $|j\rangle$ is the *jth* basis vector, and $\sum_{j=0}^{2^n-1} |\alpha_j|^2 = 1$.

There are some things note from this. Consider the vector $|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. It was stated before that individual qubits can be combined using the Kronecker/ tensor product. Yet, there is no solution for the vectors $|a\rangle$ and $|b\rangle$ to the equation $|a\rangle \otimes |b\rangle = |\phi\rangle$. That is because $|\phi\rangle$ entangled, which means the state cannot be separated into individual qubit states. This is important, as it is the entanglement that makes the simulation of quantum computers hard, as it means the number of amplitudes that need to be stored grows exponentially rather then linearly.

**Model of Quantum Computing**

Although several theoretical models of quantum computation exist and are well studied, such as quantum circuits, topological quantum computation, dissipative quantum computing, quantum walks, and the adiabatic quantum computing model, each model has its pros and cons in the context of an actual hardware implementation.

The space of possible quantum computational models is far from fully charted, and developing models in a co-design approach with quantum hardware development may benefit both in:

– *Adiabatic Quantum Computing (AQC)*: The principles of Adiabatic QC are rooted in the so-called quantum annealing protocol, suggested for finding the global minimum of a given objective function over a given set of candidate solutions by exploiting quantum fluctuations. In quantum annealing, the system is initialized in an equal-weight superposition of all possible states and then left free to evolve according to its, usually time-dependent, Hamiltonian. Annealing is obtained introducing a slow transverse-field, slow enough for the system to stay close to the ground state of the instantaneous Hamiltonian, i.e. to evolve adiabatically. If the rate of change of the transverse field is then accelerated, the system may leave the ground state temporarily but is likely to arrive in the ground state of the final problem Hamiltonian, i.e., adiabatic evolution. The transverse field is finally switched off, and the system is expected to finally lands in the ground state of the classical model corresponding to the solution to the original optimization problem. In AQC the Hamiltonian of interest is that with a ground state describing the solution to the problem of interest. This Hamiltonian may be involved but another system with a simple Hamiltonian is prepared and initialized to the ground state. Then, the simple Hamiltonian is adiabatically evolved to the desired Hamiltonian. Since the system remains in the ground state (so the term adiabatic), at the end the state of the system describes the solution to the problem. AQC has been shown to be polynomial equivalent to conventional quantum computing in the circuit model and it is robust against dissipation since the system is always in its ground state. An experimental demonstration of the success of quantum annealing for random magnets was reported immediately after the initial theoretical proposal. Current implementations of AQC are the only commercial devices available (Dwave). They are based on Josephson junctions qubits and contain CPUs made of approximately 512 qubits in the first generation (now 2000), the number of functional qubits varying significantly from chip to chip, due to flaws in manufacturing.

  Machines performing QA at the hardware level such as the D-Wave computer minimizes the following Ising Hamiltonian function [16]:

$$\mathcal{H}(\boldsymbol{h}, \boldsymbol{J}, \boldsymbol{s}) = \sum_i h_i s_i + \sum_{i<j} J_{ij} s_i s_j \qquad (6)$$

  This is closely related to the Ising model energy function as a problem Hamiltonian, where spin variables $s_i \in \{-1, +1\}$ are subject to local fields $h_i$ and pairwise interactions with coupling strengths $J_{ij}$. Each qubit‘s behaviour is governed by the laws of quantum mechanics, enabling qubits to be in a “superposition” state – that is, both a “-1” and a “+1” at the same time, until an outside event causes it to collapse into either a “-1” or a “+1” state. The output of an anneal is a low-energy ground state $\boldsymbol{s}$, which consists of an Ising spin for each qubit where $s_i \in \{-1, +1\}$. This is the basis upon which a quantum computer is constructed which gives the ability to quickly solve certain classes of NP-hard complex problems such as optimization, machine learning and sampling problems. On the D-Wave device the connectivity between the binary variables $s_i$ is described by a fixed sparse graph $G = (V,E)$ called the Chimera graph (Figure 9). Nodes in $V$ as qubits represent

problem variables with programmable weights, and edges as couplers in $E$ have programmable connection strengths. There are weights $h_i$ associated with each qubit $s_i$ and strengths $J_{ij}$ associated with each coupler between qubits ($s_i$ and $s_j$). A quantum machine instruction (QMI) solves the objective function given the weights, strengths, and qubits.

Physical constraints on current D-Wave platforms such as limited precision/control error and range on weights and strengths, sparse connectivity, and number of available qubits have an impact on the problem size and performance. Embedding algorithms are required to map or fit a problem graph onto the hardware. Strictly quantum approaches are limited by the number of graph nodes that can be represented on the hardware. Larger graphs require hybrid classical-quantum approaches.

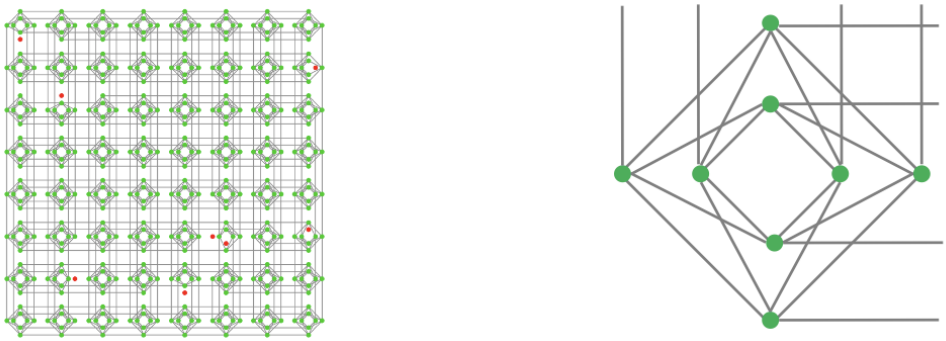


**Fig.9:** Schematic representation of the "Chimera" hardware graph of the D-Wave Two X (DW2X) housed at the Information Sciences Institute (ISI) at the University of Southern California (USC). DW2X was upgrade to 1098qubits from 512 shown in the figure. Green circles represent active qubits, red circles represent inactive qubits and lines represent couplings between qubits.

– *Circuit-Based Quantum Computing (CBQC)*: to solve a particular problem, computers, be it classical or quantum, follow a precise set of instructions that can be mechanically (or quantum-mechanically) applied to yield the solution to any given instance of the problem. CBQC uses qubits, i.e. the intrinsic spin-1/2-like degree of freedom of any bi-stable quantum system, to encode information and unitary operations to process them. The interest raised by CBQC and its development had been fostered by the precise identification of criteria that should be fulfilled by any architecture suitable for a scalable quantum computer, i.e.:

  i. It should be possible to initialize an arbitrary N-qubit quantum system to a known state;
  ii. A universal set of quantum operations must be available to manipulate the initialized system and bring it to a desired correlated state;
  iii. The technology must have the ability to reliably measure the quantum system;
  iv. It must allow much longer qubit lifetimes than the time of a quantum logic gate.

The second requirement encompasses multi-qubit operations; thus, it implies that a quantum architecture must also allow for sufficient and reliable communication between physical qubits. Ordinarily, in a classical computer, the logic gates other than the NOT gate are not reversible. In CBQC quantum logic gates are reversible. However, classical computing can be performed using only reversible gates. For example, the reversible Toffoli gate can implement all Boolean functions. This gate has a direct quantum equivalent, showing that quantum circuits can perform all operations performed by classical circuits.

Just as the state of a system of qubits was defined using vectors, the way they change can be described also. The state of a qubit (or multiple qubits) is changed by quantum logic gates, or just gates. When representing the state of qubits as vectors, quantum gates are represented using matrices. These matrices must comply with certain rules in order to be valid quantum gates.

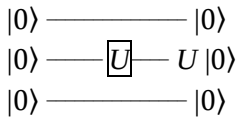
For a matrix to represent a quantum gate, it must be *unitary*. A matrix $U$ is *unitary* if it satisfies the property that it's conjugate transpose $U^{\dagger}$ is also its inverse, thus $U^{\dagger}U=U\,U^{\dagger}=I$, where $I$ is the identity matrix. In quantum computing, all gates have a corresponding *unitary* matrix, and all *unitary* matrices have a corresponding quantum gate. Gates that act on a single qubit are represented by a $2 \times 2$ matrix. More generally, an $n$ qubit gate is represented by a $2n \times 2n$ matrix. A single qubit gate can be applied to a quantum register with an arbitrary number of qubits. For a gate $U$ to act on the *jth* qubit in an $n$ qubit register, the full gate is formed by:

$$U = \underbrace{I \otimes I \otimes \ldots.\otimes}_{j-1\ times} U \otimes \underbrace{\ldots.\otimes I}_{n-j\ times} \qquad (7)$$

or equivalent to:

$$U_t = \bigotimes_{j=1}^{n} \begin{cases} U & j = t \\ I & otherwise \end{cases} \qquad (8)$$

In matrix form, gates are applied to registers using matrix multiplication. Multiple gates can be applied to a register. This is called a circuit. The gates being applied to a register can be detailed using a circuit diagram. In a circuit diagram, each line across represents a qubit, and each of the blocks on the lines) represents gates or other operations such as measurement. An example circuit diagram for three qubits, applying the gate U to the second qubit is shown below.




Quantum computers which are programmed using quantum circuits can be constructed out of any quantum technology that allows for defining qubit elements, and can implement single- and multi-qubit gate operations with high-fidelity. At present, architectures based on superconducting circuits (developed by IBM, figure 10), trapped-ions, semiconducting quantum-dots, photons, and neutral atoms, are actively being developed, and many are accessible to users over the internet. IBM-Q is the cloud solution of IBM to Quantum computing simulation. A IBM toolkit, *Qiskit*, [17] can compile a quantum circuit to match the entangling gate topology of a quantum device, map the circuit instructions into the native gate set of the device, and optimize the resulting quantum circuit for enhanced fidelity. The IBM toolkit, *Qiskit*, is installed on the HPC *XCRESCO*, the GPU cluster in operation.

**Fig.10**: *A view inside the IBM Quantum System One*

IBM Q has successfully built and tested the most powerful universal quantum computing processors it is developing a suite of scalable, increasingly larger and better processors, with a 1,000-plus qubit

device, called IBM Quantum Condor, targeted for the end of 2023. IBM Q devices are available for public use by developers, researchers and programmers via the IBM Cloud at no cost (more than hundred thousand of quantum experiments have been run by users on the IBM Cloud).

**Assessing quantum computers**

As quantum computers become available, it will be critical to ascertain whether, in fact, the performance of quantum computers exceeds that of conventional computers and if so, whether that gain is due to quantum effects that can be expected to scale. Developing clear metrics for assessing quantum computers now is therefore an important exercise to understand how to fairly compare resource usage between quantum and "classical" computers, and how to even compare different types of quantum computers. The main items are:

- Establish benchmarking quantum computing tests to validate and verify performance.
- Develop methods for emulating features of quantum computers with classical computers.
- Develop automatic methods for estimating resource consumption of a given quantum program (quantum algorithm), most notably in terms of quantum device implementation requirements such as number of qubits and quantum gates.

**Computational Support for Quantum Algorithm Development**

The quantum algorithms research lacks many of the computational tools available to conventional efforts. Future development of quantum algorithms will benefit from both exploratory mathematics research and development of computational tools for testing implementations.

It remains possible to use abstract machine models for algorithm testing purposes. The benefits derived from quantum algorithms can then be measured relative to the forecasted impact on a model HPC system. Abstract machine models can offer representations of both the quantum processor and the hybrid HPC system level. These representations can provide meaningful feedback to algorithm developers on which architectural constraints and issues must be addressed. Ultimately, how system architecture constrains algorithm implementations is likely to be a key bottleneck for quantum algorithm performance in future HPC platforms. Along side architectural issues, we expect that programming and execution models for hybrid HPC systems will also play a role in shaping quantum algorithms for applied mathematics. Programming models define the means by which end users make use of quantum algorithm implementations.

Execution models that define the order and precedence with which resources are used and the methods by which execution is negotiated must be specified. Even in an abstract setting, these execution models can provide insight into the best choices for algorithm implementation. There are currently few conventions for developing quantum algorithmic libraries but it is clear that future adoption of these libraries will need to reconcile design features with HPC system concerns.

**Quantum Simulation**

Quantum simulation is the emulation by a controlled quantum system of another quantum system of interest in the physical sciences. Much recent progress has been made, especially in trapped ion and trapped atom systems, in the analogue simulation of physical systems. Digital simulation then uses well-established techniques, principally Trotter formulae, to implement a simulated time evolution under a given Hamiltonian as a sequence of elementary gates. Given such a gate sequence, one can then apply quantum error correction to it, so that once the physical device has reached the error correction threshold, simulations that exceed the decoherence time of the device can be performed. There are four proposed approaches to digital quantum simulation (DQS) of physical systems: *i)* a grid to discretize space, represent the position of each particle on this grid by the binary expansion of its components,

and evolve forward in time according to the Hamiltonian; *ii)* based directly on the second quantized Hamiltonian; *iii)* take the Configuration-Interaction (CI) matrix of a fermionic system as the starting point for the simulation; *iv)* different approach is needed for the simulation of the quantum dynamics of fields.
Quantum algorithms can be used more generally, for example, as subroutines that support a broad range of applications or numerical solvers. In this applied mathematics setting, quantum and classical algorithms work together, perhaps in parallel, and may exhibit non-trivial dependencies on each other. Their use is not application specific but rather driven by the varying demands of HPC end users. Quantum algorithms for applied mathematics therefore represent a very broad and revolutionary approach to algorithmic development for high performance computing. In particular, quantum algorithms achieving exponential speedup over known classical algorithms have been discovered for certain problems in linear algebra and combinatorial optimization. In addition, quantum algorithms achieving polynomial speedup have been discovered for integration and summation, extraction of certain graph-theoretic properties, and optimization on structured objective functions. Adiabatic quantum computation and quantum annealing also show promise for optimization problems.
Quantum processors based on superconducting qubits can now perform computations in a Hilbert space of dimension $2^{53} \approx 9 \times 10^{15}$, beyond the reach of the fastest classical supercomputers available today. Quantum processors have thus reached the regime of quantum supremacy. Their computational power will continue to grow at a double-exponential rate: the classical cost of simulating a quantum circuit increases exponentially with computational volume, and hardware improvements will probably follow a quantum-processor equivalent of Moore's law, doubling this computational volume every few years.

# 5 Conclusions

The CRESCO HPC systems during the pandemic year have proved to be a useful tool for the users community involved in the research of solutions to fight the Covid-19, in term of drug design as well as to provide numerical simulations of virus spread by means fluid dynamics and statistic models.
Tanks to the exascale HPC class, much more computing resources is going on the availability of the community of modellers and new challenges can be achieved. The quantum computing is transitioning from a research topic to a technology that unlocks new computational capabilities and it shall be the beginning of the future generation of the computing era.

# Acknowledgements

The work presented here has been carried out within the following EU projects:
H2020-EU.2.1.1.2.: Extreme scale computing and data driven technologies - Grant Agreement n. 956831;
H2020-INFR AEDI-2018-1: HPC Centres of Excellence -Grant Agreement n: 824158;
H2020-EU.1.4.1.3.: Development, deployment and operation of ICT-based e-infrastructures - Grant Agreement n. 823964.

# References

[1] D. van der Spoel, E. Lindahl, B. Hess, G. Groenhof, A.E. Mark, and H.J.C. Berendsen. GROMACS: fast, flexible, and free. Journal of Computational Chemistry, 26:1701–1718, 2005
[2] S. Plimpton, Fast Parallel Algorithms for Short-Range Molecular Dynamics, J Comp Phys, 117, 1-19 (1995).

[3] Thomas D. Kühne et. Al. CP2K: An electronic structure and molecular dynamics software package - Quickstep: Efficient and accurate electronic structure calculations. J. Chem. Phys. 152, 194103 (2020).
[4] Macchiagodena M, Pagliai M, Karrenbrock M, Guarnieri G, Iannone F, Procacci P. Virtual Double-System Single-Box: A Nonequilibrium Alchemical Technique for Absolute Binding Free Energy Calculations: Application to Ligands of the SARS-CoV-2 Main Protease. J Chem Theory Comput. 2020 Nov 10;16(11):7160-7172.
[5] https://www.openfoam.com
[6] N. Azizi, I. Kuon, A. Egier, A. Darabiha, and P. Chow. Reconfigurable molecular dynamics simulator. In Proceedings of IEEE Symposium on Field Programmable Custom Computing Machines (FCCM), pages 197–206, 2004.
[7] T. Hamada and N. Nakasato. Massively parallel processors generator for reconfigurable system. Proceedings of IEEE Symposium on Field Programmable Custom Computing Machines (FCCM), pages 329–330, 2005.
[8] R. Scrofano and V. Prasanna. Preliminary investigation of advanced electrostatics in molecular dynamics on reconfigurable computers. In Proceedings of ACM/IEEE International Conference for High Performance Computing, Networking, Storage and Analysis (SC), 2006.
[9] V. Kindratenko and D. Pointer. A case study in porting a production scientific supercomputing application to a reconfigurable computer. In Proceedings of IEEE Symposium on Field Programmable Custom Computing Machines (FCCM), pages 13–22, 2006.
[10] S.R. Alam, P.K. Agarwal, M.C. Smith, J.S. Vetter, and D. Caliga. Using FPGA devices to accelerate biomolecular simulations. Computer, 40(3):66–73, 2007.
[11] M. Chiu and M.C. Herbordt. Molecular dynamics simulations on high performance reconfigurable computing systems. ACM Transaction on Reconfigurable Technology and Systems (TRETS), 3(4):1–37, 2010.
[12] J. Cong, Z. Fang, H. Kianinejad, and P. Wei. Revisiting FPGA Acceleration of Molecular Dynamics Simulation with Dynamic Data Flow Behavior in High-Level Synthesis. arXiv preprint arXiv:1611.04474, 2016.
[13] J.M. Haile. Molecular Dynamics Simulation. Wiley, New York, NY, 1997.
[14] Richard P. Feynman, “Simulating physics with computers,” Int. J. Theor. Phys. 21, 467–488 (1982).
[15] P. A. M. Dirac. A new notation for quantum mechanics. In Mathematical Proceedings of the Cambridge Philosophical Society, volume 35, pages 416–418. Cambridge University Press, 1939.
[16] Ushijima-Mwesigwa, H., Negre, C., Mniszewski, S. (2017). Graph Partitioning using Quantum Annealing on the D-Wave System. arXiv:1705.03082, 1–20.
[17] https://www.research.ibm.com/ibm-q/

# COMPUTATIONAL METHODS APPLIED TO THE DETECTION OF SARS-COV-2 INHIBITORS TARGETING THE SPIKE GLYCOPROTEIN

Alice Romeo[1], Federico Iacovelli[1] and Mattia Falconi[1*]

[1]*Department of Biology, University of Rome Tor Vergata, Via della Ricerca Scientifica 1, 00133 Rome, Italy*

**ABSTRACT.** Given the pivotal role of surface glycoproteins in mediating recognition and fusion of enveloped viruses to host cells, targeting these proteins is a promising strategy for blocking the viral life cycle already at the early entry phase. Considering previous data reported for the respiratory syncytial virus, we identified a highly conserved internal cavity of the SARS-CoV-2 Spike as a new possible therapeutic target due to its key structural role in the viral membrane fusion process. To evaluate this hypothesis, a virtual screening was carried out on a set of FDA drugs to identify possible fusion inhibitors strongly binding to this region, and molecular dynamics simulations were performed for the two top-ranking complexes to characterize compounds interactions with the cavity using the MM/GBSA method. In parallel, considering the well-known antiviral properties of Lactoferrin, a glycoprotein present in all human secretions, molecular docking and molecular dynamics simulations have been carried out to evaluate its potential to interfere with host cells recognition acting as a competitive inhibitor of Spike's binding to ACE2, and the promising results obtained were also supported by *in vitro* evidences.

[*] Corresponding author. E-mail: *falconi@uniroma2.it*

## 1 Introduction

Despite the recent release of highly effective vaccines against the novel coronavirus SARS-CoV-2, the causative agent of the COVID-19 pandemic, more than one year after its outbreak there is still a need for potent, safe, and broad-spectrum antiviral drugs to treat infected patients and, possibly, to control any future outbreak of similar viruses [1]. The SARS-CoV-2 Spike glycoprotein represents one of the most promising therapeutic targets and is composed by a large extracellular domain, which is divided into two functional subunits and mediates the key steps of the virus entry process: the S1 subunit (residues 1-685) is involved in the recognition of the ACE2 cell receptor, while the S2 (residues 686-1273) mediates viral and cell membranes fusion and the subsequent release of viral genome inside the cell [2]. Fusion is mediated by trimeric α-helical regions called heptad repeats 1 (HR1) and 2 (HR2) that, upon receptor binding and protease cleavage, undergo huge conformational changes determining the transition of the Spike from a metastable prefusion conformation to a postfusion conformation, providing the energy requirements to drive viral and cell membranes fusion [3]. The S2 domain and its interaction modes are highly conserved among human coronaviruses and enveloped viruses in general [2]. Recently, structural and experimental data reported for the human respiratory syncytial virus (RSV) fusion (F) glycoprotein showed that specific small molecules can bind within a central cavity of this protein and act as fusion inhibitors, preventing the protein transition to the postfusion state [4]. Given the structural similarity of the RSV F protein with the Spike, we hypothesized that the internal cavity of the Spike in prefusion conformation could represent a possible new broad-spectrum therapeutic target that would allow to interfere with coronavirus infection at an early stage. To evaluate this hypothesis and to suggest possible fusion inhibitor compounds, we performed a virtual screening (VS) within this internal pocket using a library of thousands of FDA drugs [5].

In parallel, we also characterized the antiviral potential of the glycoprotein Lactoferrin (Lf) against SARS-CoV-2. Lf is a member of the transferrin family, is present in all human secretions and is part of the innate immunity [6]. Several studies showed that this protein can interfere with receptor recognition and prevent viral entry into cells either by obscuring cellular receptors or by directly binding to surface viral particles, like the RSV F, the HIV gp120 and the HCV E1 and E2 fusion glycoproteins [6]. To evaluate if this protein could also interfere with Spike-ACE2 recognition, we performed protein-protein molecular docking and molecular dynamics (MD) simulations to determine the presence of binding sites for the bovine (bLf) or human (hLf) form of Lf, sharing about 70% of sequence identity, on the Spike surface [7].

# 2 Methods

### 2.1 Virtual screening of FDA drugs

A cryo-EM structure of the SARS-CoV-2 Spike glycoprotein in prefusion conformation (PDB ID: 6VSB) [8] was used as receptor for the VS procedure (Fig. 1, left) after modelling several non-terminal missing loops using the SWISS-MODEL webserver [9]. A drug library of 8755 FDA-approved, experimental, or investigational drugs, was obtained from the DrugBank database [10]. Molecular docking simulations have been performed using an in-house parallelized version of the Autodock Vina program [11] implemented on the ENEA HPC cluster CRESCO6. Parallelization was obtained through in-house written scripts employing the *mpi4py* Python3 library, which allowed to run 4 different Vina processes at the same time on each node of the CRESCO6 cluster. The use of 12 nodes of the cluster (576 CPUs) allowed to perform about 1200 molecular docking simulations per day, extremely reducing the computational time. The simulation box was placed over one of the HR1 internal lobes (residues 897–920) of the Spike, selecting 15 receptor sidechains inside the box as flexible. Binding energies of the 10 top-ranking compounds were calculated as an average of the best poses obtained in three repeated molecular docking simulations. The top-10 ranking was then reweighted using an additional scoring procedure, based on: binding energies, known side effects, physiological effects, and antiviral properties of the compounds, which resulted in a final score (named S-final) for each drug [5]. Due to the reciprocal size of cavity and ligand and hypothesizing the simultaneous binding of three molecules within the cavity, sequential molecular docking simulations have been performed for the two top-ranking drugs placing the simulation boxes over the other two internal lobes of the cavity and, again, selecting 15 receptor side chains as flexible. Each docking simulation was repeated three times and binding energies were calculated as an average of the interaction energies obtained for the best poses. Contacts between the ligands and the Spike were analysed using the LigPlot+ program [12]. Pictures were generated using the PyMOL 2.1.0 [13] and the UCSF Chimera programs [14].

### 2.2 Molecular dynamics simulations and trajectory analyses

Complexes obtained after sequential molecular docking simulations of phthalocyanine and hypericin were simulated using classical MD. Topologies and coordinates files of the input structures were generated using the tLeap module of the AmberTools 19 package [15], parametrizing the Spike with the ff19SB force field [16] and the ligands with the antechamber module of AmberTools 19 [15] and the general Amber force field [17]. Each complex was solvated using a box filled with TIP3P water molecules and 0.15 mol/L of NaCl, setting a minimum distance of 12.0 Å from the box sides. Four minimization cycles were performed for both systems, each composed by 500 steps of steepest descent followed by 1500 steps of conjugate gradient method, slowly decreasing the constraints applied on

protein and ligand atoms, starting from 20.0 kcal · $mol^{-1}$ · $Å^{-2}$. Systems were gradually heated from 0 to 300 K in a NVT ensemble over a period of 2.0 ns using the Langevin thermostat [18], imposing a starting restraint of 0.5 kcal · $mol^{-1}$ · $Å^{-2}$ on protein and ligand atoms, which was decreased every 500 ps. Simulations were then carried out in an isobaric-isothermal (NPT) ensemble for 2.0 ns, imposing a pressure of 1.0 atm using the Langevin barostat [19] and fixing the temperature at 300 K. A production run of 30 ns was finally performed for both systems with a timestep of 2.0 fs, using the pmemd.cuda module of the AMBER16 software [18]. Covalent bonds involving hydrogen atoms were constrained using the SHAKE algorithm [19], the PME method [20] was used for calculating long-range interactions and a cut-off of 9.0 Å was set for short-range interactions. Hydrogen bonds (Hbonds) have been evaluated using the Hbonds plugin of VMD 1.9.3 [21]. Interaction analysis has been performed using the molecular mechanics energies combined with generalized Born and surface area continuum solvation (MM/GBSA) method [22], coupled to per-residue decomposition, using the MMPBSA.py.MPI program of the AMBER16 software [18] on 3 nodes of the CRESCO6 HPC cluster.

### 2.3 Protein-protein molecular docking and molecular dynamics simulations of Spike-Lactoferrin complexes

Protein-protein molecular docking simulations were carried out between the Spike protein and the lactoferrin structures using the FRODOCK webserver [23]. Structure of the Spike in prefusion conformation was extracted from a previously performed trajectory [5], while that of the bLf and hLf were obtained from the PDB Database (PDB IDs: 1BLF [24] and 1B0L [25]). Topology and coordinate files for MD simulations were generated using the tLeap module of the AmberTools 19 package [15] and the ff19SB force field [16], inserting the proteins into a box of TIP3P water molecules and 0.15 mol/L of NaCl ions and setting a minimum distance of 12.0 Å from the box sides. The same simulation protocol described in the previous paragraph was used and production runs of 30 ns were performed for both systems on the CRESCO6 HPC cluster, using the NAMD 2.13 MD package [26]. Hbonds were analysed using the hbonds module of the GROMACS 2019 analysis tools [27], while hydrophobic contacts were identified using the mdtraj Python library [28]. MM/GBSA analysis [22] was performed as previously described.

## 3 Results

### 3.1 Virtual screening of FDA drugs and evaluation of top-ranking compounds

The VS procedure allowed to rapidly evaluate the affinity of 8755 drugs against one of the internal lobes (residues 897-920) of the Spike cavity (Fig.1, left). The ten top-ranking compounds, obtained after performing a literature-based ranking reweighting [5], are reported in Table 1.
Considering their high interaction energies with the Spike, subsequent analyses were focused only on phthalocyanine (PHT) and hypericin (HYP). HYP and different PHT derivatives have been previously evaluated as antiviral and, in particular, anti-HIV compounds for their capability to interfere with viral gp120 glycoprotein binding and fusion to the host cell, and both drugs also possess other targets on different viruses, including the avian coronavirus infection bronchitis virus (IBV) [5]. Recently, the use of a PHT-containing mouthwash has also helped reduce symptoms and hospitalization time for COVID-19 patients [29]. Considering the trimeric structure of the Spike, we performed sequential molecular docking simulations, proposing an inhibitory mechanism by which at least three molecules could arrange inside the inner cavity, binding the three HR1 internal lobes (Fig. 1A, B).

| Compound | Binding energy (kcal/mol) | Final score S-final |
|---|---|---|
| phthalocyanine | -16.3 | 62.6 |
| hypericin | -15.1 | 55.3 |
| TMC-647055 | -12.5 | 49.5 |
| quarfloxin | -12.6 | 35.2 |
| tepotinib | -12.0 | 24.1 |
| laniquidar | -12.8 | 23.0 |
| tadalafil | -12.4 | 21.4 |
| ergotamine | -13.2 | 20.1 |
| JNJ-10311795 | -13.1 | 19.1 |
| TZ2PA6 | -12.7 | 12.7 |

**Table 1**: Final reweighted ranking of the VS.

Results showed that PHTs and HYPs arrange in a shifted and tilted orientation within the pocket. PHTs mainly establish hydrophobic contacts with surrounding residues and within themselves, while HYPs establish both hydrophobic contacts and Hbonds within themselves and with other residues.

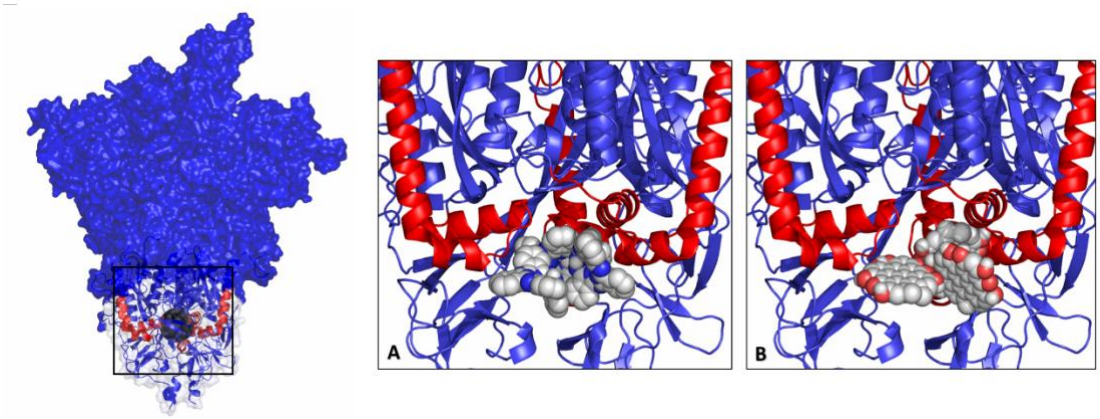



**Fig.1**: On the left, the Spike glycoprotein modelled structure. The black sphere placed within the structure highlights the internal cavity, selected for the VS, shown in cartoon representation surrounded by a transparent surface. The region surrounded by the black square is magnified in the images on the right, showing the trimeric Spike-PHT (**A**) and Spike-HYP (**B**) complexes after three sequential molecular docking simulations. HR1 regions of the Spike are highlighted in red.

To further assess the stability of the obtained binding poses, we also performed 30 ns MD simulations of the trimeric Spike-PHT and Spike-HYP complexes. Free energies of binding, calculated using the MM/GBSA method [22], confirmed the high interaction energies observed through molecular docking simulations and highlighted the presence of a predominant hydrophobic contribution within the pocket during both simulations, indicated by the presence of highly favourable Van der Waals energies for all six compounds (Table 2).
Furthermore, HYP interactions are also characterized by a negative electrostatic contribution (Table 2) and Hbonds analysis showed that the three HYPs can establish from of 1 to 4 Hbonds during the simulation time, while no valuable Hbond was observed for PHTs.

| Compound | VdW (kcal/mol) | Electrostatic (kcal/mol) | Nonpolar solvation (kcal/mol) | Polar solvation (kcal/mol) | $\Delta G_{binding}$ (kcal/mol) |
|---|---|---|---|---|---|
| PHT #1 | -84.8 ± 4.7 | -5.6 ± 3.0 | -6.8 ± 0.2 | 30.6 ± 2.2 | -66.6 ± 4.7 |
| PHT #2 | -76.5 ± 3.0 | -0.7 ± 2.3 | -6.7 ± 0.2 | 24.7 ± 2.0 | -59.1 ± 3.0 |
| PHT #3 | -54.3 ± 2.7 | 3.0 ± 2.6 | -5.0 ± 0.3 | 20.7 ± 2.9 | -35.6 ± 2.6 |
| HYP #1 | -55.3 ± 3.6 | -18.4 ± 6.4 | -4.7 ± 0.3 | 34.7 ± 5.5 | -43.6 ± 4.4 |
| HYP #2 | -53.3 ± 3.0 | -19.4 ± 5.2 | -5.2 ± 0.2 | 33.8 ± 4.0 | -44.1 ± 3.5 |
| HYP #3 | -37.4 ± 2.4 | -14.4 ± 5.5 | -4.3 ± 0.2 | 28.8 ± 4.7 | -27.3 ± 3.3 |

**Table 2**: MM/GBSA results for the three PHT and HYP molecules.

Per-residue decomposition analyses [22] showed that each drug can establish from 9 to 15 contacts, with interaction energies ranging from -0.5 to -5.1 kcal/mol. During the simulation time, the three PHTs and HYPs created a clustered arrangement within the pocket, contacting residues from the HR1 internal lobes and from upward and downward regions of the Spike. This is expected to generate a hard impairment to the motion of the internal regions and supports our hypothesis that the strong network of interactions established, and the steric hindrance generated should be sufficient for the compounds to block the Spike transition to the postfusion state.

### 3.2 Molecular docking and molecular dynamics simulations of Spike-Lactoferrin complexes

Protein-protein molecular docking simulations of bLf and hLf targeting the Spike glycoprotein, indicated that both proteins mainly interact with the Spike receptor binding domain (RBD) in "up" conformation (Fig. 2 B-C). The stability of the best obtained complexes was further evaluated through 30 ns of classical MD simulations. Simulations showed that close and stable contacts are established between the proteins' interfaces, and MM/GBSA [22] interaction analyses also showed that both proteins interact with the Spike RBD domain with high free energies of binding. In particular, bLf shows an interaction energy of -28.0 ± 9.0 kcal/mol, while the hLf reaches an almost two-fold higher energy of -48.3 ± 17.0 kcal/mol. This is explained by the higher number of interactions established by the human protein in comparison with the bovine form (45 and 28, respectively). A detailed analysis of the interaction types showed that the bLf can arrange 20 hydrophobic contacts, 3 salt bridges and 5 Hbonds with the Spike RBD, while the hLf sets up 23 hydrophobic contacts, 12 salt bridges and 10 Hbonds. Furthermore, only two Spike residues (Gly502 and Tyr505) are shared between the ACE2 [30] and lactoferrins binding interfaces, despite the three proteins bind to close sites on the RBD surface (Fig. 2). The obtained results allowed us to hypothesize that one of the many beneficial effects of Lf could involve the competitive binding of the Spike RBD, preventing Spike attachment to the human ACE2 receptor and, in this way, blocking host cell recognition and virus entry. Indeed, results obtained *in vitro* support our computational results showing that pre-incubation of bLf with SARS-CoV-2 reduced the infection of two different cell lines [7].

## 4 Conclusions

The VS carried out using the ENEA HPC cluster CRESCO6, and the following MD simulations, allowed us to characterize a novel SARS-CoV-2 therapeutic target, setting the path for future research that could be useful also against other threatening enveloped viruses sharing a similar entry mechanism. Furthermore, MD simulations of two Spike-Lf complexes, performed on the CRESCO6 cluster, allowed

us to suggest a further and unknown protective role exerted by Lf against SARS-CoV-2, that binding the Spike RBD domain *de facto* prevents virus attachment and entry into cells.

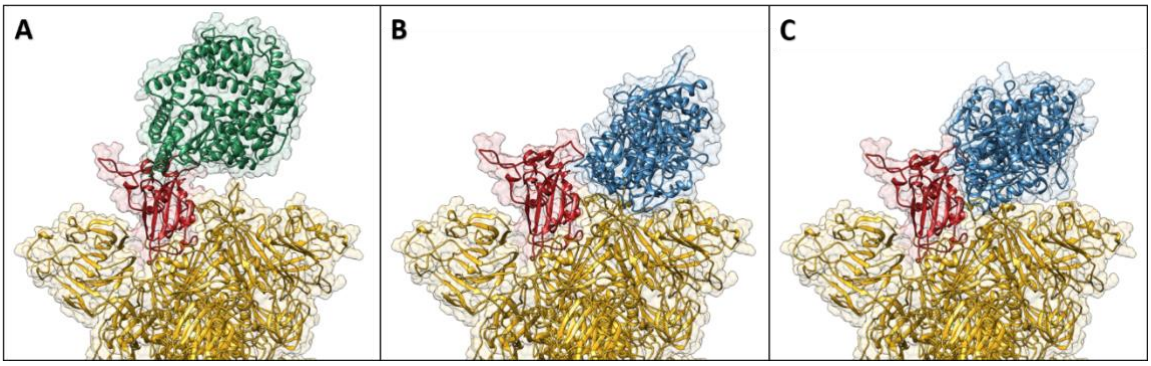



**Fig. 2**: Spike-ACE2 complex (PDB ID: 6LZG) (**A**), compared with the best FRODOCK binding poses obtained between the Spike and the bLf (**B**) or hLf (**C**), represented as ribbons surrounded by a transparent surface. The Spike is represented in yellow and the RBD domain is highlighted in red. ACE2 is shown in green, while both lactoferrins are shown in blue.

## References

[1] M. Mei and X. Tan. Current Strategies of Antiviral Drug Discovery for COVID-19. *Front. Mol. Biosci.* **8**, p. 671263, (2021).
[2] T. Tang, M. Bidon, J.A. Jaimes, et al. Coronavirus membrane fusion mechanism offers a potential target for antiviral development. *Antiviral Res.* **178**, p. 104792, (2020).
[3] A.C. Walls, M.A. Tortorici, J. Snijder, et al. Tectonic conformational changes of a coronavirus spike glycoprotein promote membrane fusion. *Proc. Natl. Acad. Sci. U. S. A.* **114**, pp. 11157–11162, (2017).
[4] M.B. Battles and J.S. McLellan. Respiratory syncytial virus entry and how to block it. *Nat. Rev. Microbiol.* **17**, pp. 233–245, (2019).
[5] A. Romeo, F. Iacovelli and M. Falconi. Targeting the SARS-CoV-2 spike glycoprotein prefusion conformation: virtual screening and molecular dynamics simulations applied to the identification of potential fusion inhibitors. *Virus Res.* **286**, p. 198068, (2020).
[6] E. Campione, T. Cosio, L. Rosa, et al. Lactoferrin as protective natural barrier of respiratory and intestinal mucosa against coronavirus infection and inflammation. *Int. J. Mol. Sci*. **21**, pp. 1–14, (2020).
[7] E. Campione, C. Lanna, T. Cosio, et al. Lactoferrin against SARS-CoV-2: in vitro and in silico evidences. *Front. Pharmacol.* **12**, p.1524,(2021).
[8] D. Wrapp, N. Wang, K.S. Corbett, et al. Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. *Science* **367**, pp. 1260-1263, (2020).
[9] A. Waterhouse, M. Bertoni, S. Bienert, et al. SWISS-MODEL: Homology modelling of protein structures and complexes. *Nucleic Acids Res.* **46**, pp. W296–W303, (2018).
[10] D.S. Wishart, Y.D. Feunang, A.C. Guo, et al. DrugBank 5.0: A major update to the DrugBank database for 2018. *Nucleic Acids Res.* **46**, pp. D1074–D1082, (2018).
[11] O. Trott and A.J. Olson. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem.* **31**, pp. 455–461, (2010).
[12] R.A. Laskowski and M.B. Swindells. LigPlot+: Multiple ligand-protein interaction diagrams for drug discovery. *J. Chem. Inf. Model.* **51**, pp. 2778–2786, (2011).
[13] The PyMOL Molecular Graphics System, Version 2.1.0 Schrödinger, LLC.

[14] E.F. Pettersen, T.D. Goddard, C.C. Huang, et al. UCSF Chimera - A visualization system for exploratory research and analysis. *J. Comput. Chem.* **25**, pp. 1605–1612, (2004).
[15] R. Salomon-Ferrer, D.A. Case and R.C. Walker. An overview of the Amber biomolecular simulation package. *WIREs Computational Molecular Science* **3**, pp. 198–210, (2013).
[16] C. Tian, K. Kasavajhala, K.A.A. Belfon, et al. Ff19SB: Amino-Acid-Specific Protein Backbone Parameters Trained against Quantum Mechanics Energy Surfaces in Solution. *J. Chem. Theory Comput.* **16**, pp. 528–552, (2020).
[17] J. Wang, R.M. Wolf, J.W. Caldwell, et al. Development and testing of a general amber force field. *J. Comput. Chem.* **25**, pp. 1157–1174, (2004).
[18] N. Goga, A.J. Rzepiela, A.H. de Vries, et al. Efficient algorithms for Langevin and DPD dynamics. *J. Chem. Theory Comput.* **8**, pp. 3637–3649, (2012).
[19] K.M. Aoki, M. Yoneya and H. Yokoyama. Constant pressure Md simulation method. *Mol. Cryst. Liq. Cryst.* **413**, pp. 109–116, (2004).
[18] D.A. Case, R.M. Betz, D.S. Cerutti, et al. AMBER 2016. University of California, San Francisco (2016).
[19] J.P. Ryckaert, G. Ciccotti and H.J.C. Berendsen. Numerical integration of the cartesian equations of motion of a system with constraints: molecular dynamics of n-alkanes. *J. Comput. Phys.* **23**, pp. 327–341, (1977).
[20] T. Darden, D. York and L. Pedersen. Particle mesh Ewald: An N·log(N) method for Ewald sums in large systems. *J. Chem. Phys.* **98**, pp. 10089–10092, (1993).
[21] W. Humphrey, A. Dalke and K. Schulten. VMD: Visual molecular dynamics. *J. Mol. Graph.* **14**, pp. 33–38, (1996).
[22] S. Genheden and U. Ryde. The MM/PBSA and MM/GBSA methods to estimate ligand-binding affinities. *Expert Opin. Drug Discov.* **10**, pp. 449–461, (2015).
[23] E. Ramírez-Aportela, J.R. López-Blanco and P. Chacón. FRODOCK 2.0: fast protein–protein docking server. *Bioinformatics* **32**, pp. 2386–2388, (2016).
[24] S.A. Moore, B.F. Anderson, C.R. Groom, et al. Three-dimensional structure of diferric bovine lactoferrin at 2.8 Å resolution. *J. Mol. Biol.* **274**, pp. 222–236, (1997).
[25] X.L. Sun, H.M. Baker, S.C. Shewry, et al. Structure of recombinant human lactoferrin expressed in Aspergillus awamori. *Acta Crystallogr. Sect. D Biol. Crystallogr.* **55**, pp. 403–407, (1999).
[26] J.C. Phillips, R. Braun, W. Wang, et al. Scalable molecular dynamics with NAMD. *J. Comput. Chem.* **26**, pp. 1781–1802, (2005).
[27] M.J. Abraham, T. Murtola, R. Schulz, et al. Gromacs: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX* **1–2**, pp. 19–25, (2015).
[28] R.T. McGibbon, K.A. Beauchamp, M.P. Harrigan, et al. MDTraj: A Modern Open Library for the Analysis of Molecular Dynamics Trajectories. *Biophys. J.* **109**, pp. 1528–1532, (2015).
[29] P.S.S. Santos, B.F. Orcina, R.R.G. Machado, et al. Beneficial effects of a mouthwash containing an antiviral phthalocyanine derivative on the length of hospital stay for COVID-19: Randomised trial. *Research Square*, (2021)
[30] Q. Wang, Y. Zhang, L. Wu, et al. Structural and Functional Basis of SARS-CoV-2 Entry by Using Human ACE2. *Cell* **181**, pp. 894-904, (2020).

# HPC-DRIVEN HIT-TO-LEAD PROCESS FOR SARS-COV-2 MAIN PROTEASE INHIBITION

Piero Procacci[1*], Marina Macchiagodena[1],Maurice Karrenbrock[2] , Marco Pagliai[1], Guido Guarnieri[3], Francesco Iannone[3]

[1]*University of Florence, Chemistry Department, Via Lastruccia 3, Sesto F.no 50019 (Italy)*

[2]*University of Geneva, Pharmaceutical Sciences, 30, quai Ernest-Ansermet CH-1211 Genève (Switzerland)*

[3]*ENEA, Energy Technologies & Renewable Sources Department - Information Communication Technologies Division, Lungotevere Thaon di Revel, 76, 00196 Rome Italy*

**ABSTRACT.** In this contribution, we present a molecular dynamics-based technique for the calculation of the absolute binding free energies (ABFE) in drug-receptor systems. The technique, called virtual double system single box (vDSSB), is a versatile nonequilibrium variant of the so-called alchemical approach for ABFE calculation, specifically tailored for homogenous and heterogenous high-performing computing platforms. The technique has been applied to the calculation of potential non-covalent inhibitors of the main protease of the SARS-CoV-2 virus. We report here the results obtained in the summer 2020 on the CRESCO6 facilities using the program ORAC [M. Macchiagodena et al., J. Chem. Theory Comput. 16, 7160 (2020), P. Procacci et al., Chem. Comm., 56, 8854 (2020)] on a series of ligands selected as docking hits from a previous virtual screening study. The algorithm has been further adapted to the GROMACS code and tested successfully on the Marconi100 heterogeneous architecture at CINECA.

* Corresponding author. E-mail: *piero.procacci@unifi.it*

## 1 Introduction

Ligand-receptor binding free energy (BFE) prediction is one of the main research topics in computational chemistry today due to the potential impact on drug discovery and public health. Powered by the exponential growth in computer speed of the last two decades, modern supercomputing architectures are now affording high-throughput screening (HTS) with an efficiency largely outperforming experimental HTS, at a much lower cost and accessing a much larger and unrestricted chemical space domain. According to the Scopus database, more than 6% of all peer-reviewed Covid-19-related scientific output in 2020 involved computational approaches, mostly based on molecular docking. A contemporary high-end HPC platform is capable of screening via docking many millions of compounds per day on a given biological target [1]. Docking techniques have benefited in the last years of knowledge-based and machine learning (ML) methods. Scoring functions (SF) have gradually evolved towards the use of empirically weighted simplified physical descriptors heavily trained on ever-extending databases of ligand-receptor binding free energy. For example, in the widely popular Vina program [2], a recent evolution of the Autodock code, electrostatic interactions based on atomic charges have been replaced by piecewise linear functions for H-bonded atoms, with weighted steric atom-atom functions accounting for hydrophobic effects. The Vina code was found to significantly improve the

binding mode prediction accuracy on the well-established virtual screening benchmark called the Directory of Useful Decoys [3] with respect to its precursor Autodock relying on a complex, more physically-grounded SF. The performance of docking SFs in terms of BFE predictions is generally measured using ranking-based binary metrics such as the area under the Receiver Operating Characteristic (ROC) curve (AUC) or the Enrichment Factor (EF). These methods assume that ligands can be classified in two groups, namely active and inactive compounds according to some affinity threshold. Performances are measured with respect to a random decision (e.g., the random-hit rate in case of the EF or the random guess in case of the ROC-AUC).

While powerful in screening large datasets of compounds, docking is unable to provide secure affinity predictions. This is especially true when the target/ligand pair is deflecting significantly from the chemical and structural traits of the ML training sets [4]. Vina, exhibiting ROC-AUC from 0.70-0.90 in the DUD-E derived test sets [5]. When tested on a simple host-guest database, it yields a prediction efficiency only slightly higher that based on the flipping of a coin with ROC-AUC of only 0.55 [6].

The main problem in docking technology lies in the high number of false positives produced in screening campaigns, a limit that is shared with experimental HTS. False negatives are unavoidable in HTS processes, whether virtual or experimental. False positives, on the other hand, are one of the key factors that currently restricts the discovery potential of HTS techniques, as they require time, energy, and high cost to be identified in wet-lab low-throughput protocols by medicinal chemists [7]. Docking hits urgently need, prior to wet-lab confirmation, a further computational assessment using advanced techniques, based on atomistic MD simulations with explicit solvent, using increasingly accurate all-atom force fields [8]. In this regard, in the last two decades, the so-called alchemical method has emerged as one of the most powerful and promising approaches to the calculation of binding free energies in ligand-receptor systems. The alchemical protocol evaluates the BFE in a thermodynamic cycle as a difference (see Fig. 1) of the solvation energy of the ligand in the bound state and in the bulk solvent [9].

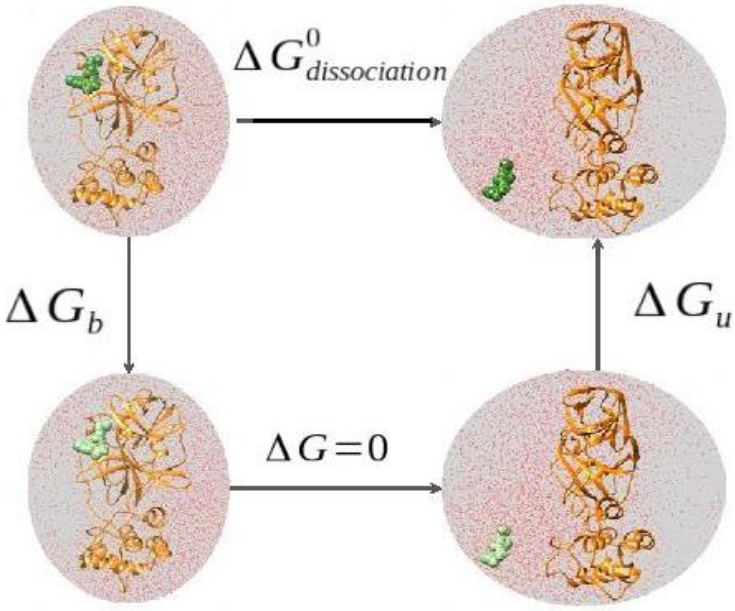



**Fig 1:** Thermodynamic cycle in alchemical transformations. Left and right legs of the cycle represent the solvation free energy of the ligand in the binding site and in the bulk solvent, respectively. The ligand is shown using van der Waals spheres; dark and light green: fully interacting and decoupled (gas-phase) ligand.

These solvation energies are computed independently by connecting the ligand end-states through a stratification of discrete intermediate states where the ligand-environment interactions are progressively turned off via a λ coupling parameter. The solvation free energies (namely the reversible work to bring the ligand from the gas phase to the condensed phase) are given by the sum of the free energy contributions along the stratification, computed using the free energy perturbation (FEP) approach [9], requiring equilibrium MD simulations for each of the intermediate λ-states. Alchemical techniques are costly and generally limited to the calculations of relative binding free energies (RBFE) of strictly congeneric series of compounds. This is a severe limitation since hits form docking campaigns can be chemically distant and not easily amenable to RBFE calculations through, e.g., intricate "perturbation graphs" gradually connecting the chemically distant interesting molecules, thereby spending computational resources in determining RBFE's between uninteresting intermediate decoys [10]. Quoting Yutong Zhao [11], lead engineer at Relay Therapeutics, FEP-based absolute binding free energy (ABFE) calculations may easily (absolutely) end up in tears due to daunting sampling challenges emerging at low coupling, when the ligand in the binding site experiences an order-disorder transition from low enthalpy to high entropy states [12]. Paradoxically, the advent of GPUs in scientific calculations, that allows simulating a typical drug-receptor system for up to hundreds ns/day, has strengthened the illusion that a single sufficiently long MD trajectory can achieve correct sampling in FEP applications of complex biomolecular systems. Conformational transitions are in fact sudden events that occur rarely in the time scale that can be attained in a simulation of a single typical ligand-receptor system (microseconds at most).

An important tightly connected limiting factor in ABFE calculations using FEP is the lack of a reliable confidence interval (CI) determination and the sensibility to the initial conditions of the ABFE prediction [13]. Credible CIs can be only assessed by repeating the FEP calculation several times, thereby expanding considerably the computational cost. From an implementation point of view, while the calculations on the windows of the stratification can be run independently, the convergence rate of these concurrent simulations is strongly λ-dependent [9,13], posing severe challenges to an efficient parallelization of the algorithm on HPC systems.

In the last years, in the context of the alchemical approach we have been developing a massively parallel computational variant based on enhanced sampling methodologies of the end states only and on the production of nonequilibrium (NE) alchemical trajectories rapidly connecting the end-states. The ABFE is recovered by computing the work done in these driven NE trajectories exploiting the Crooks [14] and Jarzynski [15] NE theorems on the resulting work distribution. At variance with FEP, these NE-based techniques strictly require that the concurrent NE trajectories are run according to a common alchemical time schedule, hence automatically satisfying a perfect load balance in the embarrassingly parallel implementation. Besides, in the bound leg of the alchemical thermodynamic cycle, enhanced sampling is applied only to the low enthalpy ordered fully coupled ligand states, according to a powerful torsional tempering scheme involving the so-called "hot zone" comprising the ligand and residues of the binding site. During 2020, in the midst of the Covid-19 crisis, we set up an automated workflow for the calculations of MD-based accurate ABFEs starting from the knowledge of the docking pose. The method, termed virtual Double System Single Box (vDSSB) allows to screen on a dedicated HPC platform such as CRESCO6 tens of compounds per day, hence potentially providing the missing link for chemically distant docking hits assessment (usually few tens or hundreds over millions of tested compounds) and false positive filtering. Compound refinement on surviving hits can be effectively and rapidly achieved using unidirectional NE technique applied to RBFE calculations of congeneric series.

This contribution is organized as follows. In the method section we briefly describe the theoretical background of the vDSSB approach discussing its parallel implementation on the CRESCO6 and M100/CINECA HPC platforms. In Section 3 we report on some recent results obtained with this

technique on the inhibition of the $3CL^{pro}$ protein, the main protease of the SARS-CoV-2 virus. Concluding remarks and perspectives are drawn in the last section.

# 2 Non-equilibrium Virtual Double System Single Box: theoretical background and methodological workflow

vDSSB is an inherently parallel approach composed of two distinct computational stages for HPC execution with ideal parallel efficiency applied to both legs of the alchemical thermodynamic cycle (see Fig. 1), namely the HREX stage and, in sequence, the NE stage, followed by the fast post-processing of the resulting work distribution.

## 2.1 HREX stage: equilibrium sampling of the fully coupled bound end-state and of the decoupled unbound state

This stage involves the canonical sampling of the bound end-state (top left panel in Fig. 1) and of the ligand in the gas-phase (bottom right panel in Fig. 1) embedded as a ghost molecule in equilibrated solvent. These two equilibrium simulations are performed by launching multiple batteries of Hamiltonian Replica Exchange (HREX) simulation with a torsional tempering scheme [16]. Parallel tempering or Replica exchange schemes are paradigmatic low-communication weak scaling parallel algorithms in the MD simulations of complex systems. The HREX torsional variant, by selectively scaling the crucial degrees of freedom for binding, allows to keep the replica number (and hence the computational cost) to a minimum while affording an effective enhanced sampling of the ligand-receptor rugged conformational free energy landscape. In each HREX battery, the torsional potential of a selected subset of degrees of freedom of the system is scaled along a progression of n replicas up to a maximum scaling factor 1. The lowest *s* factor sets the temperature of the system subset in the highest replica to$T_t = \frac{T_0}{s}$ (where $T_0$ is the target temperature, 300 K), while the rest of the degrees of freedom of the system remains cold at $T_0$. In the bound state, the system subset includes the ligand and the residues of the binding site, implemented in a progression of 12 replica states with torsional scaling decreasing from 1 to 0.2 corresponding to a torsional temperature of 1500 K. In the gas-phase state, the starting end-states of the decoupled ligand are efficiently obtained by combining HREX sampling of a single molecule (*s*=0.1 and 8 replicas) in vacuo with pre-equilibrated samples of pure explicit solvent. The HREX sampling of the unbound leg end-state can be performed on a local workstation in a matter of minutes. Higher torsional and nonbonded scaling is needed in the unbound leg due to the strong and unscreened electrostatic interactions that may occur between polar moieties of flexible ligands.
The HREX computational stage starts from a docking-generated ligand-receptor structure, and includes a series of complex operations consisting in i) the generation of the MD engine topology files using appropriate software tools (e.g. PrimaDORAC [17], LigParGen [18], Paramchem [19]) for the ligand force field parameterization; ii) a preliminary minimization of the complex with the user-selected force field; iii) the NPT equilibration in standard conditions of the resulting solvated receptor-ligand structure in a MD box of optimal size; iv) the setting-up of the input files with the definition of the hot-zone in the bound state and specification of the scaling protocols to perform HREM simulations for bound and unbound ligand state on the HPC platforms. These error-prone steps, which are central in the vDSSB approach, have been fully automatized by the middleware [20], HPC_Drug, an effective tool for the guided submission of vDSSB jobs on a HPC system. HPC_Drug is a python application distributed under the GPL that can be cloned from the GitHub repository https://github.com/MauriceKarrenbrock/HPC_Drug. All prerequisite software (GPL) for using HPC_Drug and installation instructions are detailed at the GitHub site. The generation of ready-to-use

HREX input files for HPC batch submission is achieved by HPC_Drug in a matter of minutes on the HPC front-end or on a local workstation. Currently, HPC_Drug supports the ORAC MD code [12] and the GROMACS MD suite [20]. ORAC has been designed to run on homogeneous CPU-based architecture such as the CRESCO6 platforms [21]. In ORAC, HREX is implemented using an hybrid OpenMP-MPI approach, whereby scaling factors rather than system configurations are exchanged on the MPI layer during the simulation with a minimal impact on the communication overhead. For GROMACS, HPC_Drug generates the HREX input files exploiting the versatile open-source Plumed library [22].
For a typical drug-receptor systems (50K atoms), the HPC execution of the HREX stage of the bound state using ORAC may typically involve from 32 to 64 Skylake 48-cores nodes on CRESCO6 with 8 to 16 HREX batteries producing a total simulation time of the order of 1-2 microseconds simulation time with 30/60 ns on the target state in one wall clock day. GROMACS/Plumed may produce on the M100/CINECA platform equipped with 4 Volta100 GPU per node up to 200 ns a day on the target state engaging 36 nodes. The HREX output consists in uncorrelated end-state snapshots, sampled at regular time intervals on the target state, providing a high-quality starting set of equilibrium system configurations for the subsequent NE stage.

### 2.2 NE stage: production of the NE alchemical trajectories

The NE stage is launched, for each leg of the cycle, in a single parallel job starting from the configurations generated by the preceding HREX stage. In the annihilation job (bond-state leg) and in the growth job (unbound-state lef), each MPI instance read its own equilibrium starting configuration and then proceeds independently (no communication) completing the NE alchemical simulation in the *common* prescribed time schedule,[1] with a perfect load balance by design. The output of these independent runs consists in files where the work done on the driven system by the switching off (bound leg) or on (unbound leg) of the ligand-environment alchemical coupling parameter is saved at regular intervals. Examples of work time recorded during the NE trajectories are reported in Fig. 2. The main output of NE stage growth and annihilation jobs is hence a $N_{MPI}$-sample of *final* growth and annihilation work values. The final growth work values refer to processes where we start with the equilibrated ligand in the gas-phase and we end in a non-equilibrium state with a fully coupled ligand in the solvent. The final annihilation work values refer to processes where an equilibrated ligand is brought into the gas-phase ending up in a NE conformation. Typically, on the CRESCO6 cluster, for Covid-19 related drug-receptors pairs, we launched $N_{MPI} = 640$NE trajectories in each leg engaging 80 Skylake nodes. Using the ORAC code, the annihilation job (bound state) for a typical system (50K atoms) is completed on CRESCO6 in few wall clock hours, while the growth job (unbound state) for a drug-size ligand (5K atoms) takes few tens of wall clock minutes. The total simulation times produced by the jobs is of the order of 0.3 microseconds for each leg of the cycle. Using GROMACS on M100/Cineca, wall-clock times for the NE stage are cut by a factor of 4 to 5.

[1] For the annihilation job, the atomic charges of the ligand are first turned off in about 100-200 ps, followed by ligand-environment Lennard-Jones interaction in, typically, 300-500 ps.

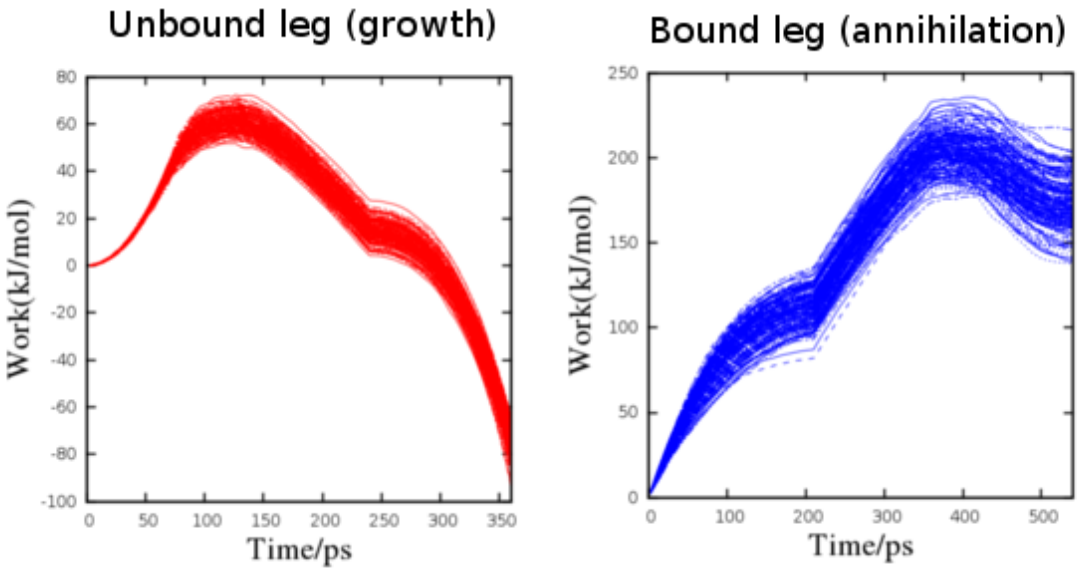



**Fig. 2:** Alchemical work trajectories for a typical ligand of the SARS-CoV-2 main protease.

### 2.3 Post processing stage: production of ligand decoupling equilibrium sampling of the fully coupled bound end-state and of the decoupled unbound state

The bound and unbound work values are two sets of independent random variables. We can hence emulate a "double-system-single-box" by combining each annihilation work with each growth work hence producing a sample total of $N_{MPI}^2$work values for a vDSSB NE process where the ligand is annihilated on the protein and is simultaneously grown in the bulk. In terms of work distribution, this operation corresponds to the *convolution* of the annihilation and growth work distributions, as it is shown in equation 1:

$$P_b(W) * P_u(W) = \int P_b(W) P_u(W - w) dw \qquad (1)$$

In Fig.3 we show the work distributions obtained for the final work values reported in Fig.2. The estimate of the dissociation free energy $\Delta G_{b+u}$can be readily calculated from the combined growth and annihilation work values by exploiting the Jarzynski identity [15] or, equivalently, by assuming that the convolution distribution can be represented as a combination of Gaussian mixture and using the Crooks theorem [13]. The number and weights of the normal components can be evaluated using the Expectation-Maximization algorithm [23]. The so-computed dissociation free energy must be corrected[9] by adding a standard state dependent term of the form $\Delta G_{vol} = RTln\left(\frac{V_{site}}{V_0}\right)$, where $V_0 =$ 1661 Å$^3$is the standard state volume (corresponding to 1M concentration) and $V_{site} = 4\pi\frac{(2\sigma)^3}{3}$is the binding site volume estimated from the variance $\sigma^2$of the distance between the centers of mass of the ligand and the receptor in the HREX simulation of the bound state. The ligand-receptor dissociation constant (in M units) is finally given by equation 2:

$$K_d = exp\left[\frac{-\left(\Delta G_{b+u} + RTln\left(\frac{V_{site}}{V_0}\right)\right)}{RT}\right] \qquad (2)$$

The confidence interval on $\Delta G_{b+u}$ can be reliably estimated by bootstrapping the $N_{MPI}$ uncorrelated growth and annihilation work values *before* performing the convolution.

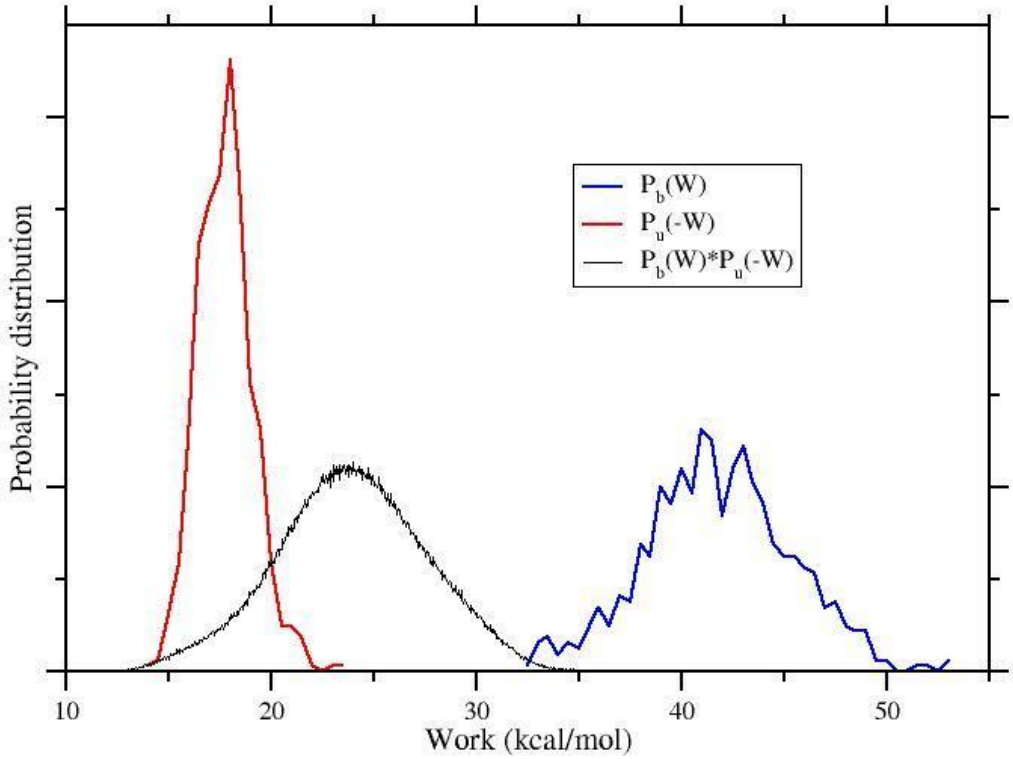



**Fig. 3:** Growth and annihilation final work distribution for a typical Covid-19 related ligand of the main protease.

The post-processing stage has been fully automated by simple application scripts to be executed on the front-end once the HREX stage and the subsequent NE stage have been completed. Further technical details of the vDSSB methodology are provided in Refs. 23 and 24.

## 3 Assessing docking hits for the inhibition of the SARS-CoV-2 main protease using vDSSB on CRESCO6.

The SARS-CoV-2 main protease ($3CL^{pro}$) is a non-structural protein that cleaves the pp1a and pp1ab polyproteins expressed by the viral m-RNA upon host cell entry. $3CL^{pro}$ is generated by self-excision from the pp1a polyprotein following dimerization. The catalytic activity of $3CL^{pro}$ is expressed by the $(3CL^{pro})2$ dimer. The $3CL^{pro}$ monomer is in turn composed of two loosely coupled units, the chymotrypsin-like domains I+II (residues 1-197), harbouring the catalytic site, and the cluster of helices domain III (residues 198-304), regulating dimerization via two intertwined salt bridges involving ARG4(A)-GLU290(B) and GLU290(A)-ARG4(B) of the A and B protomers. The dimer is characterized by two symmetric extended clefts for pp1a, pp1ab adhesion. Each dimer cleft ends at the solvent exposed catalytic site with the CYS145-HIS41 proteolytic dyad. The two catalytic dyads, on opposite sides of the dimer and far from the monomer-monomer adhesion surface, very likely act independently for maximizing the catalytic efficiency. To identify an effective $3CL^{pro}$ inhibitor, only

the I+II chymotrypsin catalytic domain needs to be considered in the ligand receptor simulation of the bound state. In Fig. 4, we show, as an example, the chymotrypsin domain of $3CL^{pro}$ bound to hydroxychloroquine [26].

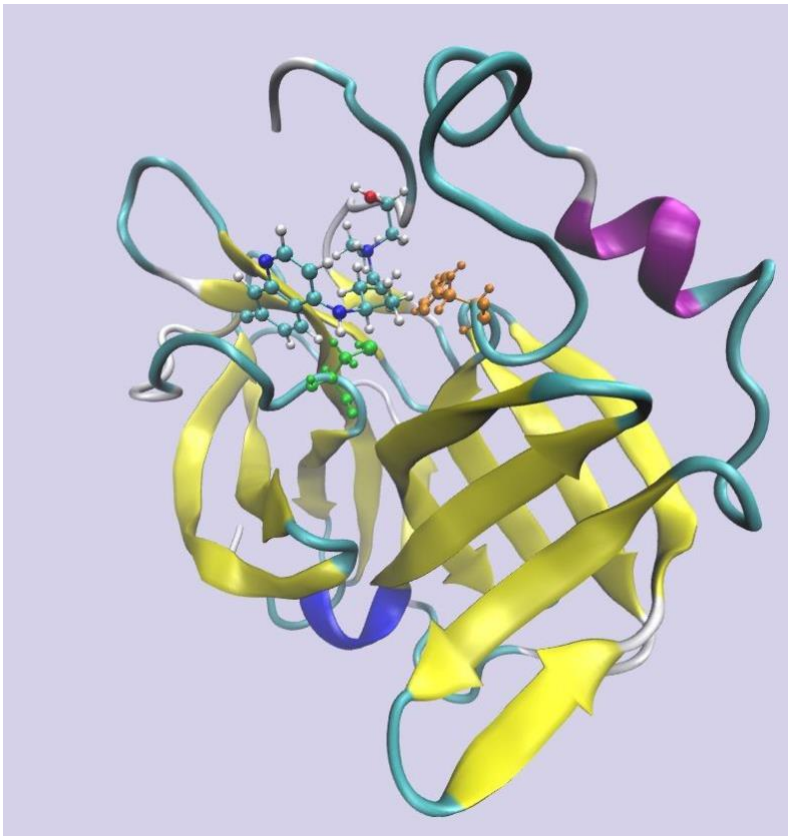
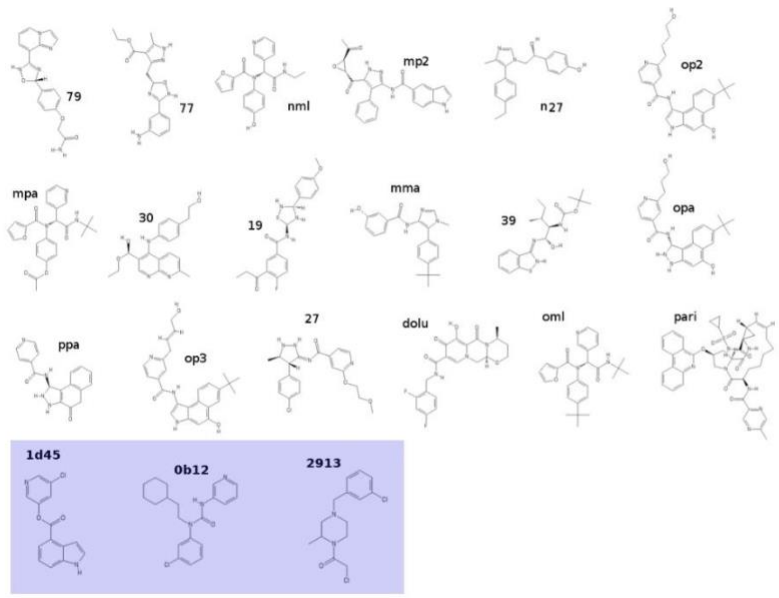


**Fig. 4:** Chymotrypsin domain of the SARS-CoV-2 main protease in complex with hydroxychloroquine. The catalytic dyad CYS145-HIS41 is in orange color.

In Ref. 25, we have identified via multimodal structure-based design and molecular docking several possible inhibitors of the SARS-CoV-2 $3CL^{pro}$ protease. Most of these putative chemically distant inhibitors are not commercially available. The experimental validation hence requires the synthesis of these non-congeneric compounds. In Fig. 5, we present the results obtained on CRESCO6 using the vDSSB approach for the dissociation free energies of these docking hits. The compounds within the light blue background in the left panel have been identified in the context of the Covid-19 Moonshot initiative as possible $3CL^{pro}$ inhibitors. Their activities were experimentally determined using fluorescence techniques. vDSSB-predicted dissociation free energies for these compounds are in fair agreement with the experimental data (within 1:1.5 kcal/mol), except for the 1d45 compound where the predicted value for the dissociation free energy is significantly smaller than the experimental counterpart. However, while 1d45 is labelled as a non-covalent $3CL^{pro}$-binder according to the Covid-19 Moonshot activity data [27], the same compound was found to be a potent covalent inhibitor of the highly homologous SARS-CoV-1 main protease with approximately the same dissociation free energy ($\Delta G_{exp.}$ = 10.3 kcal/mol) [28]. Covalent binding (that is not accounted for in vDSSB of FEP-based techniques) may explain the observed difference between experimental and calculated dissociation free energy for 1d45-$3CL^{pro}$ interaction.

Assuming as a threshold for activity $K_d = 1\mu M$, only compound 27 among the docking hits identified in Ref. 25 survives after vDSSB assessment. The compounds marked in red color in the right panel of Fig. 5 are hence false positives according to our calculations. With the availability of the entire

CRESCO6 cluster (~500 nodes), these false positives could have been identified in less than 2 wall-clock days with an estimated cost of less than 100K Euros (assuming a cost of 0.1 euro cent per core hours), a time and a cost incomparably smaller than that needed in low-throughput medicinal chemistry involving in all cases the synthesis of the compound.




| Ligand | $\Delta G_j^0$(kcal/mol) | $\Delta G_{exp.}$(kcal/mol) |
|---|---|---|
| pari | $10.7 \pm 0.6$ | - |
| opa | $9.4 \pm 0.5$ | - |
| ml | $9.0 \pm 0.7$ | $7.9^{(a)}$ |
| op3 | $8.8 \pm 0.8$ | - |
| 27 | $8.3 \pm 0.5$ | - |
| 39 | $7.6 \pm 1.5$ | - |
| mma | $7.4 \pm 0.8$ | - |
| ppa | $7.3 \pm 0.8$ | - |
| op2 | $7.1 \pm 2.0$ | - |
| dolu | $6.7 \pm 1.6$ | - |
| 19 | $6.5 \pm 0.7$ | - |
| 30 | $6.1 \pm 0.9$ | - |
| mp2 | $5.8 \pm 0.6$ | - |
| mpa | $4.3 \pm 1.2$ | - |
| 77 | $4.0 \pm 0.4$ | - |
| n27 | $3.9 \pm 0.6$ | - |
| 79 | $3.7 \pm 0.9$ | - |
| nml | $3.1 \pm 0.4$ | - |
| 1d45 | $5.4 \pm 0.8$ | 10.0 |
| 0b12 | $9.3 \pm 0.9$ | 7.46 |
| 2913 | $5.8 \pm 0.8$ | 7.0 |

(a) BFE for 3CLpro of SARS-CoV-1 [Jakobs, J. Med. Chem 56, 534-546, 2013]

**Fig: 5** Dissociation free energies computed on CRESCO6 with the vDSSB approach.

# 4 Conclusion

Most of the (successful) coordinated efforts by governments and big-pharma industry in Covid-19 therapy has been directed toward the SARS-CoV-2 viral structural proteins (spike (S), nucleocapsid (N), membrane (M), and envelope (E)). Pfizer and Moderna mRNA vaccine, based on the translation of the prefusion Spike protein, are the chief achievement of this extraordinary effort. Despite the success, we must nonetheless remain vigilant as SARS-CoV-2 can evolve to wane vaccine efficacy or new zoonotic pandemics can arise in the future. On the other hand, research for antiviral therapeutic agents have been erratic and poorly coordinated. During the pandemic, drug discovery has disorderly dispersed in plethora of biological targets including human proteins such as ACE2, bromodomain, sigma receptors, immunophilins, kinases, while a well-coordinated collaborative effort in targeting the highly conserved nonstructural proteins involved in the viral life cycle (such as the precursor SARS-CoV-2 $3CL^{pro}$ and $PL^{pro}$ proteases) could have maybe delivered an effective and specific antiviral agent. In this context, HPC-driven vHTS can provide an effective tool for a rationalization of drug-discovery process with the potential to substantially increase the productivity of pharmaceutical research, especially in the field of antiviral design. We do believe that the paradigm based on non-equilibrium thermodynamics, relying on rigorous and general enhanced sampling approaches for the end-states and on the equal-time condition for the NE swarm of alchemical trajectories, is effective in by-passing the well-known sampling-related pitfalls and entanglements of conventional equilibrium based MD-based technologies such as FEP+ [29] that have prevented so far its widespread use in industrial setting. NE-based vDSSB has the potential to provide, through fully automated procedures, a firm framework for

an HPC based measurement of $K_d$ in realistic thermodynamic conditions with minimal end-user intervention/tweaking. Accuracy of the prediction via SF-vDSSB can further benefit from the progress in force field parameterization (e.g., polarizable force fields, QM/MM schemes, Car-Parrinello MD schemes) from better algorithm for force kernels and from increased capacity of the hardware technology. While the competence for the setting up of the Docking/NE-vDSSB hit-to-lead framework on a HPC system are high and with a strong interdisciplinary character, the required skills for its bare use are limited. vDSSB may hence provide the missing link for the setting up of an automated HPC-based instrument for hit-to-lead in drug discovery.

## References


[1] A. Acharya *et al.* Supercomputer-Based Ensemble Docking Drug Discovery Pipeline with Application to Covid-19, J. Chem. Inf. Model. **60**, pp. 5832-5852, (2020).
[2] O. Trott and A. J. Olson. Autodock vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multi-threading. J. Comput. Chem., **31,** pp. 455-461, (2010).
[3] M. M. Mysinger, M. Carchia, J. J. Irwin, and B. K. Shoichet. Directory of useful decoys, enhanced (dud-e): Better ligands and decoys for better benchmarking. J. Med. Chem., **55**, pp. 6582-6594, (2012).
[4] M. Wójcikowski, P. Ballester, and P. Siedlecki. Performance of machine-learning scoring functions in structure-based virtual screening. Scientific Reports, **7**, pp. 1-10 (2017).
[5] L. Masters, S. Eagon, and M. Heying. Evaluation of consensus scoring methods for AutoDock Vina, smina and idock. J. Mol. Graph. Model., **96,** 107532, (2020).
[6] L. Casbarra, and P. Procacci, J. Comput. Aided Mol. Des., *in press* DOI:10.1007/s10822-021-00388-4 (2021).
[7] J. Bibette. Gaining confidence in high-throughput screening. Proc. Natl. Acad. Sci., **109**, pp. 649-650 (2012).
[8] F. Palazzesi, M. K. Prakash, M. Bonomi, and A. Barducci. Accuracy of current all-atom force-fields in modeling protein disordered states, J. Chem. Theory Comput., **11**, pp. 2-7 (2015).
[9] A. Pohorille, C. Jarzynski, and C. Chipot. Good practices in free-energy calculations. J. Phys. Chem. B, **114**, pp.10235-10253 (2010).
[10] L. F. Song, T.-S. Lee, C. Zhu, D. M. York, and K. M. Merz. Using amber18 for relative free energy calculations. J. Chem. Inf. Model., **59**, pp. 3128-3135, (2019).
[11] https://twitter.com/proteneer/status/1376314932732571658
[12] R. K. Pal and E. Gallicchio. Perturbation potentials to overcome order/disorder transitions in alchemical binding free energy calculations. J. Chem. Phys., **151**, pp. 124116 (2019).
[13] P. Procacci. Methodological uncertainties in drug-receptor binding free energy predictions based on classical molecular dynamics. Curr. Op. Struct. Biol. Biology **67**, pp. 127-134 (2021).
[14] G. E. Crooks. Nonequilibrium measurements of free energy differences for microscopically reversible Markovian systems. J. Stat. Phys. **90**, pp. 1481–1487 (1998).
[15] C. Jarzynski. Nonequilibrium Equality for Free Energy Differences. Phys. Rev. Lett., **78**, 2690-2693 (1997).
[16] S. Marsili, G. F. Signorini, R. Chelli, M. Marchi, and P. Procacci. Orac: A molecular dynamics simulation program to explore free energy surfaces in biomolecular systems at the atomistic level. J. Comput. Chem., **31**, pp. 1106-1116 (2010).
[17] P. Procacci. PrimaDORAC: A Free Web Interface for the Assignment of Partial Charges, Chemical Topology, and Bonded Parameters in Organic or Drug Molecules J. Chem. Inf. Model. **57**, 1240-1245 (2017).

[18] L. S. Dodda, I. Cabeza de Vaca, J. Tirado-Rives, W. L. Jorgensen LigParGen web server: An automatic OPLS-AA parameter generator for organic ligands. Nucleic Acids Research, Volume 45, Issue W1, 3 July 2017, Pages W331-W336.
[17] K. Vanommeslaeghe, E. Prabhu Raman, and A. D. MacKerell, Jr. Automation of the CHARMM General Force Field (CGenFF) II: Assignment of bonded parameters and partial atomic charges. J. Chem. Inf. Model. **52**, pp. 3155-3168 (2012).
[19] M. Karrenbrock, HPC_DRUG: A middleware python tool for computational drug discovery on HPC architectures, https://github.com/MauriceKarrenbrock/HPC_Drug (2021).
[20] S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess, E. Lindahl. GROMACS 4.5: a High-Throughput and Highly Parallel Open Source Molecular Simulation Toolkit. Bioinformatics, **29**, 845-854 (2013).
[21] F. Iannone, F. Ambrosino, G. Bracco, M. De Rosa, A. Funel, G. Guarnieri, S. Migliori, F. Palombi, G. Ponti, G. Santomauro, P. Procacci. CRESCO ENEA HPC clusters: a working example of a multifabric GPFS Spectrum Scale layout. 2019 International Conference on High Performance Computing & Simulation (HPCS), pp. 1051-1052 IEEE Dublin (2019).
[22] G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi. PLUMED2: New feathers for an old bird, Comp. Phys. Comm. **185**, 604-613 (2014).
[23] J. A. Bilmes. A gentle tutorial of the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden Markov models. Tech Rep. International Computer Science Institute, Berkeley CA (1998).
[23] M. Macchiagodena, M. Karrenbrock, M. Pagliai, G. Guarnieri, F. Iannone, P. Procacci. Virtual Double-System Single-Box: A Nonequilibrium Alchemical Technique for Absolute Binding Free Energy Calculations: Application to Ligands of the SARS-CoV-2 Main Protease. J. Chem. Theory Comput. **16**, pp. 7160-7172 (2020).
[24] M. Macchiagodena, M. Karrenbrock, M. Pagliai, G. Guarnieri, F. Iannone, P. Procacci. Chapter: Nonequilibrium Alchemical Simulations for the Development of Drugs Against Covid-19, in Methods in Pharmacology and Toxicology, Springer, New York, NY (2021).
[25] M. Macchiagodena, M. Pagliai, P. Procacci, Identification of potential binders of the main protease 3CLpro of the COVID-19 via structure-based ligand design and molecular modeling. Chem. Phys. Lett. **750**, pp. 137489 (2020).
[26] P. Procacci, M. Macchiagodena, M. Pagliai, G. Guarnieri, F. Iannone. Interaction of hydroxychloroquine with SARS-CoV2 functional proteins using all-atoms non-equilibrium alchemical simulations. Chem. Comm. **56**, pp. 8854-8856 (2020).
[27] J. Chodera, A. A. Lee, N. London, F. von Delft. Crowdsourcing drug discovery for pandemics. Nature Chemistry, **12**, pp. 581-583 (2020).
[28] A. K. Ghosh, G. Gong, V. Grum-Tokars, D. C. Mulhearn, S. C. Baker, M. Coughlin, V. S. Prabhakar, K. Sleeman, M. E. Johnson, A. D. Mesecar. Design, synthesis and antiviral efficacy of a series of potent chloropyridyl ester-derived SARS-CoV 3CLpro inhibitors. Bioorg. Med. Chem. Lett. **18**, pp. 5684–5688 (2008).
[29] L. Wang, Y. Wu, Y. Deng, B. Kim, L. Pierce, G. Krilov, D. Lupyan, S. Robinson, M. K. Dahlgren, J. Greenwood, D. L. Romero, C. Masse, J. L. Knight, T. Steinbrecher, T. Beuming, W. Damm, E. Harder, W. Sherman, M. Brewer, R. Wester, M. Murcko, L. Frye, R. Farid, T. Lin, D. L. Mobley, W. L. Jorgensen, B. J. Berne, R. A. Friesner, R. Abel. Accurate and Reliable Prediction of Relative Ligand Binding Potency in Prospective Drug Discovery by Way of a Modern Free-Energy Calculation Protocol and Force Field. J. Am. Chem. Soc. **137**, pp. 2695–2703 (2015).

# ENHANCING CFD SIMULATIONS OF COVID-19 DIFFUSION BY COUGHING AND SNEEZING USING DATA ASSIMILATION

Rossella Arcucci[1,3,*], César Quilodrán Casas[1], Aniket Joshi[2], Laetitia Mottet[2], Asiri Obeysekara[2], Yi-Ke Guo[1,4], Christopher Pain[2]

[1] *Data Science Institute, Department of Computing, Imperial College London, UK*
[2] *Department of Earth Science and Engineering, Imperial College London, UK*
[3]*Leonardo Centre, Imperial College Business School, Imperial College London, UK*
[4]*Department of Computer Science, Hong Kong Baptist University, Hong Kong*

**ABSTRACT.** Coughing is one of the most effective methods for SARS-CoV-2, the coronavirus strain that causes COVID-19, to spread. Coughing is a natural reaction that serves to protect the lungs and airways from irritants and infections by expelling droplets at speeds of up to 50 miles per hour. Unfortunately, it's also one of the most efficient methods for infections to spread, particularly respiratory viruses that require host cells to replicate. CFD (Computational Fluid Dynamics) is an useful tool for simulating droplets ejected by the mouth and nose during coughing and sneezing. Coughing and sneezing models, like any numerical models, add uncertainty through the choice of scales and parameters. Any numerical simulation must take these uncertainties into account in order to be accepted. In the medium to long-term analysis, numerical forecasting models frequently use Data Assimilation (DA) approaches for uncertainty quantification. DA is the ideal combination of time-distributed data with a dynamic model to approximate the true state of a physical system at a particular time. In order to improve numerically forecast results, DA adds observational data into a prediction model. We use a Variational Data Assimilation model to assimilate direct observation of the physical mechanics of droplet generation at the mouth's exit during coughing in this research. We employ high-speed imaging to analyse the fluid fragmentation at the exit of the lips of healthy participants in a sneezing scenario, which we learned about from previous studies. We demonstrate the impact of the suggested approach on the accuracy of CFD simulations.

[*] Corresponding author. E-mail*: r.arcucci@imperial.ac.uk*

## 1 Introduction

Infections of the respiratory tract, like as influenza, are transmitted when a healthy person comes into touch with respiratory droplets from an infected person's cough, sneeze, or breath [1]. Coughing is one of the most effective methods for SARS-CoV-2, the coronavirus strain that causes COVID-19, to spread. Coughing is a natural reaction that protects the lungs and airway from irritants and germs by expelling droplets at speeds of up to 50 miles per hour. Unfortunately, it's also one of the most effective methods for infections to spread, particularly for respiratory viruses that require host cells to proliferate. CFD (Computational Fluid Dynamics) is an useful tool for simulating droplets ejected by the mouth and nose while coughing or sneezing. CFD models for coughing and sneezing, like any numerical models, introduce uncertainty through the choice of scales and parameters. Any numerical simulation must take these uncertainties into account in order to be accepted. In the medium to long-term analysis,

numerical forecasting models frequently use Data Assimilation (DA) approaches for uncertainty quantification.

DA is the ideal combination of time-distributed data with a dynamic model to approximate the true state of a physical system at a particular time. In order to improve numerically forecasted results, DA adds observational data into a prediction model. It allows for problems like redundancy and unequal spatial and temporal data distribution to be handled, allowing models to assimilate information more efficiently.

"What can be said about the value of an unknown variable x that characterises the evolution of a system if we have some measured data y and a model M of the underlying mechanism that created the data?" DA seeks to answer questions like this. This is the Bayesian setting, in which we seek a quantification of the uncertainty in our parameter information, which, according to Bayes' rule takes the form:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

Here, the physical model is represented by the conditional probability (also known as the likelihood) p(y | x), and the prior knowledge of the system by the term p(x). The denominator is considered as a normalising factor and represents the total probability of y. Many DA approaches have been created based on this formulation [2], with the majority of them being custom-built for the forecasting model with which they are integrated. The variational DA (VarDA) approaches [3] based on the minimisation of a function that estimates the discrepancy between numerical results and observations, assuming that the two sources of information, forecast and observations, have errors that are adequately described by error covariance matrices, have gained acceptance as powerful methods in the last ten years. In order to apply a DA approach to a CFD model for coughing and sneezing, real observations are needed.

### 1.2 Related works and contribution of the present work

The travel characteristics of evaporating droplets discharged into the vented chamber were investigated using a CFD study with an Eulerian-Lagrangian model in [4]. With the use of Bayesian Data Assimilation, this study tries to explain the transport and dispersal of droplets created by coughing in a ventilated environment. The beginning velocity and duration of a coughing burst were measured in experiments. Instead of a CFD study, [5] proposes an analytical technique. To explore the dispersion and deposition of expiratory droplets in a room during coughing, the authors analyse the detailed processes of cough jet flow, including droplet evaporation and motion, turbulent flow surrounding jet, and particle tracking. The authors report the results of a combined experimental and theoretical examination of the fluid dynamics of such violent expiratory events in [6]. Sneezing and coughing events are multiphase turbulent buoyant clouds with suspended droplets of varying sizes, according to direct observation. The creation of a theoretical model of pathogen-bearing droplets interacting with a turbulent buoyant momentum puff is guided by observations. The transport characteristics of saliva droplets produced by coughing are investigated in [7] in a peaceful indoor setting. The Lagrangian equation is used to investigate the dispersion processes of saliva droplets of various sizes expelled while coughing. The findings show that the size affects the transport characteristics of saliva droplets caused by coughing.

The authors disclose firsthand observation of the physical mechanics of droplet generation at the mouth's outflow during sneezing in [8]. They use high-speed imaging to analyse the fluid fragmentation at the mouth exits of healthy people in particular. In this paper, we use the genuine photos from [8] to improve the CFD models of coughing and sneezing. We used a 3D Variational DA model with an ideal parameter to balance the weight of the errors covariance matrices to achieve this goal.

In summary, we employ data assimilation to improve the accuracy of the CFD models to simulate droplet and aerosol size distributions [10] using real-world data. This will provide us more precise information on the evolution of the particle size distribution around the mouth.

## 2 Data Assimilation

Data Assimilation (DA) is an approach for fusing data (observations) with prior knowledge (e.g., mathematical representations of physical laws; model output) to obtain an estimate of the distribution of the true state of a process [9].
In order to perform DA, one needs observations (i.e., a data or measurement model), a background (i.e., a priori state or process model), and information about the distribution of the errors on these two.
DA merges the estimated state $x_t \in R^n$ of a discrete-time dynamic process at time t:

$$x_{t+1} = M_{t+1}x_t + w_t$$

with an observation $y_t \in R^m$:

$$y_t = H_t x_t + v_t$$

where $M_t$ is a dynamic system and $H_t$ is called observation operator. The vectors $w_t$ and $v_t$ represent the process and observation errors, respectively. They are usually assumed to be independent white-noise processes with Gaussian probability distributions.

$$w_t \sim \mathcal{N}(0, Q_t), \quad v_t \sim \mathcal{N}(0, R_t)$$

where $Q_t$ and $R_t$ are called error covariance matrices of the model and observation respectively.
DA is a Bayesian inference that combines the state $x_t$ with $y_t$ at each given time. The Bayes theorem conducts to the estimation of $x_t^a$ which maximise a probability density function.
given the observation $y_t$ and a prior from $x_t$. This approach is implemented in one of the most popular DA methods which is the three-dimensional Variational (3DVar) DA. The goal of 3DVar is to compute an optimal solution, $x_t^a$, that minimises a weighted difference between the actual measurement, $y_t$, and the measurement prediction.
If the error covariance matrices $Q_t$ and $R_t$ are designed to be correlated by a parameter α and such that $Q_t = \alpha I$ and $R_t = (1 - \alpha)I$ with $0 < \alpha < 1$ and where $I$ is the identity matrix [10], we can decide the degree of fidelity we want to give to the observations with respect to the CFD simulation by setting a proper values of the parameter $\alpha$. As the weight of the covariance matrices in the DA process is given by the inverse of the matrices [2], with this choice of covariance matrices we can chose a bigger value of α if we assume that the observations are very reliable or a smaller value to α if the CFD model is a high-fidelity model. With this choice the data assimilation process can be described as following:

$$\delta x_t^a = argmin_{\delta x} J(\delta x)$$

with

$$J(\delta x) = \frac{\delta x^T \delta x}{2\alpha} + \frac{(H_t \delta x - d_t)^T (H_t \delta x - d_t)}{2(1-\alpha)}$$

where $\mathrm{d_t} = [\mathrm{y_t} - \mathrm{x_t}]$ is the misfit and $\delta \mathrm{x} = \mathrm{x} - \mathrm{x_t}$ is the increment.

As an important issue in Data Assimilation is to provide a result in real-time, the choice of an efficient method to compute the minimum of the functional J is a fundamental topic.
In this paper, we compute the minimum of the functional J by the minimisation method proven to be faster for optimisation problems [11], i.e., the L-BFGS (Limited-Broyden Fletcher Goldfarb Shanno) method. The L-BFGS method is a Quasi-Newton method that can be viewed as extension of conjugate-gradient methods in which the addition of some modest storage serves to accelerate the convergence rate.

# 3 Test case

The test case represents an idealised case used to test the ability of our 3DVar to be used to assimilate real images of sneeze or cough emissions in coughing and sneezing CFD models. The images of real sneeze emissions were obtained from [8]. In order to assimilate sneezing images with a coughing CFD, we are here assuming that, in terms of velocity and mass fraction, the sneeze at later time step corresponds to a cough at earlier time step. The data was pre-processed using OpenCV. OpenCV (Open Source Computer Vision Library) is an "open source computer vision and machine learning software library built to provide a common infrastructure for computer vision applications" [12].
Two images of real observations were used: sneeze emissions after 5 ms recorded at 2000 fps; and sneeze ejecta at 8 ms recorded at 8000 fps. The images represent the observed data in the DA function. The background data from the coughing CFD model are two images of simulated sneeze emissions: at an angle of 24 degrees and horizontal. This data represents the state vector in the DA function. The images from the CFD simulation were scaled between 0 and 0.01 (for sneezing) and 0 and 0.02 (for coughing) with respect to their water mass fraction. The images from the simulations were cropped between 1.53 m and 1.63 m of height and a 0.12 m width. After this pre-process, the dimensions of the images from the CFD simulation and the observations match. All images were set at the same resolution and the observations (interpolation) operator $H_t$ is the identity function. The observation images were also scaled between 0 and 0.01 (for sneezing) and 0 and 0.02 (for coughing) with respect to their water mass fraction to be consistent to the simulation images.
The four images in were transformed to grayscale (ranged from 0 to 255 in one channel) using OpenCV. Additionally, a mask was drawn by hand on all images to eliminate the nose and mouth from the observations, and the inlets from the simulations. We chose a non-grayscale colour (blue) for the mask to avoid eliminating useful information. The mask allows us to perform the data assimilation only on the sneeze emissions and ejecta [10]. The backgrounds of all four images were set to white. Since the observations do not include a water mass fraction associated to them, we assumed a value of 0.01 (for sneezing) and 0.02 (for coughing) for pixels in black and 0 for pixels in white, to complement the simulations. The execution time of the algorithm for assimilating y in x is approx. 0.38 to 0.42 seconds. These values of the execution times have been computed as mean values of 50 runs of the algorithm on the same machine and the same data set and for different values of $0<\alpha<1$. Table 1 shows value of Mean Square Error (MSE) defined as

$$MSE(x) = \frac{\|x - x_C\|_2}{\|x_C\|_2}$$

where $x_C$ denotes a control variable. The MSE is here computed with respect to the observed data before and after the assimilation process for different values of the parameter α. As expected, for bigger values of α, the result of the assimilation come closer to the observations, and it presents a smaller value of MSE.

**Table 1**: Values of MSE, for the sneezing CFD simulations, computed with respect to the observed data before and after the assimilation process for different values of the parameter α

| α | couching | sneezing |
|---|---|---|
| 0.1 | 0.94 | 1.68 |
| 0.2 | 0.89 | 1.60 |
| 0.3 | 0.83 | 1.48 |
| 0.4 | 0.74 | 1.32 |
| 0.5 | 0.62 | 1.12 |
| 0.6 | 0.49 | 0.88 |
| 0.7 | 0.35 | 0.63 |
| 0.8 | 0.22 | 0.40 |
| 0.9 | 0.10 | 0.18 |

Fig. 1 shows how the DA technology merges the two data for different values of 0< α <1. The results confirm that for small values of α the solution $x^a$ of the assimilation process is closer to the CFD simulation. The assimilation of these data could have a significant impact on real-world applications for determining safe distances. In the scenario at hand, the CFD estimates that the dispersion of the droplets after 5 ms has a radius of about 4 cm and a distance of nearly 6 cm from the mouth. The observation shows a smaller radius near the mouth, but the droplets reach 12 cm from the mouth. The combination of this data is critical in determining safe distances for human interactions. In fact, the technology and model we gave are generic and may be used to simulate other scenarios using other types of computational fluid dynamic systems.

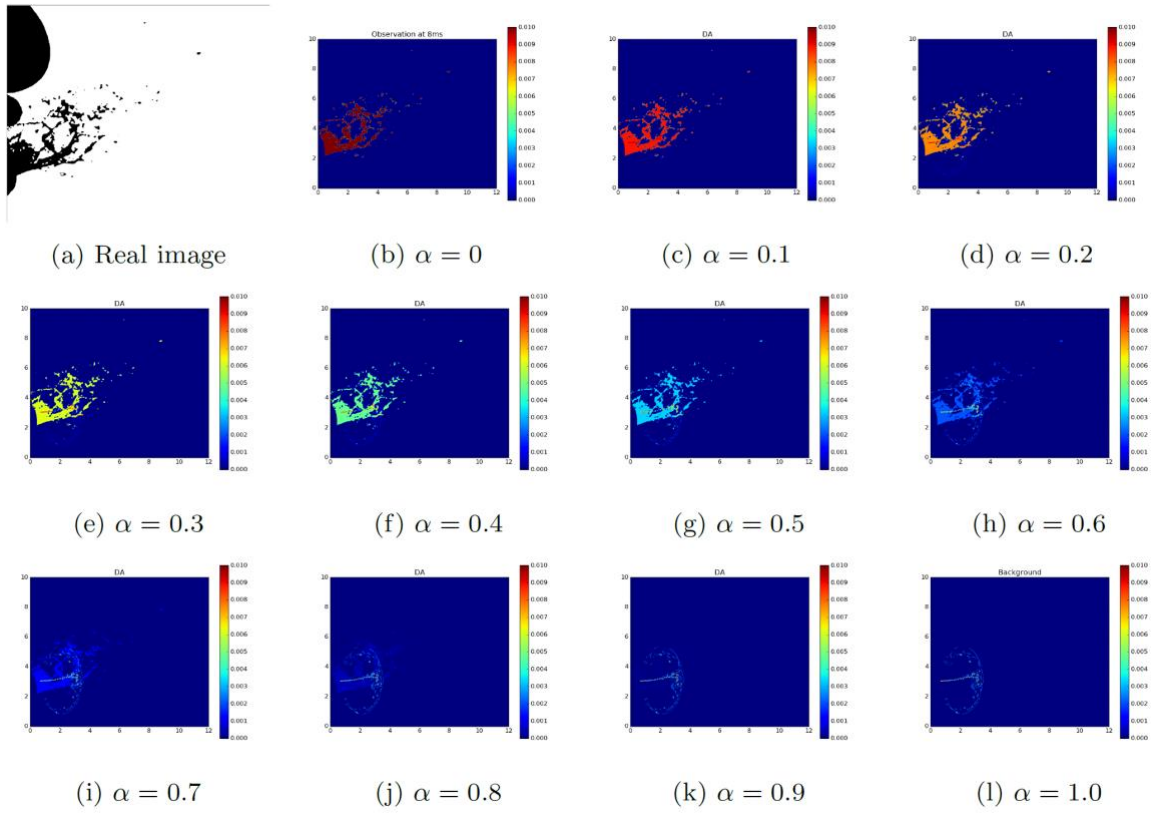



**Fig.1**: Results of the assimilation of the real image y in the CFD coughing simulation x for different values of α.

## Conclusions and future work

In this study, we compare experimental data from [8] of sneezing with results from a coughing CFD simulation with various inlet geometries. We've only considered the mouth as a source of ejecta in our first test case. We also hypothesised that comparing a sneeze simulation to a cough simulation at a later timestep would yield equivalent and interchangeable findings, allowing us to assess the adaptability of the DA approaches. We used a 3D Variational DA model with an optimal parameter to balance the weight of the error covariance matrices in the assimilation function.

The adaptability of DA in efficiently using the experimental sneezing results in the coughing CFD simulation highlights the benefits of this work. This adaptability may be used to create a comprehensive technique that can aid in the modelling of various COVID-19 spread situations that incorporate ventilation airflows (indoors or outdoors), while also leaving room for modelling crowd airflow dynamics.

The CFD simulations in domain such as a room aim to show the dispersion in real case scenarios. These simulations need High Performance Computing infrastructure to run. The computing resources and the related technical support used for this future work have been provided by CRESCO/ENEAGRID High Performance Computing infrastructure and its staff [13]. CRESCO/ENEAGRID High Performance Computing infrastructure is funded by ENEA, the Italian National Agency for New Technologies, Energy and Sustainable Economic Development and by Italian and European research programmes, see http://www.cresco.enea.it/english for information.

## Acknowledgements

This work is supported by the EP/V036777/1 Risk EvaLuatIon fAst iNtelligent Tool (RELIANT) for COVID19.

## References

[1] Mahesh Jayaweera, Hasini Perera, Buddhika Gunawardana, and Jagath Manatunge. Transmission of covid-19 virus by droplets and aerosols: A critical review on the unresolved dichotomy. Environmental Research, page 109819, 2020.

[2] Mark Asch, Marc Bocquet, and Maffelle Nodet. Data assimilation: methods, algorithms, and applications, volume 11. SIAM, 2016.

[3] Rossella Arcucci, Laetitia Mottet, Christopher Pain, and Yi-Ke Guo. Optimal reduced space for variational data assimilation. Journal of Computational Physics, 379:51-69, 2019.

[4] Wei Sun and Jie Ji. Transport of droplets expelled by coughing in ventilated rooms. Indoor and Built Environment, 16(6):493-504, 2007.

[5] Jianjian Wei and Yuguo Li. Enhanced spread of expiratory droplets by turbulence in a cough jet. Building and Environment, 93:86-96, 2015.

[6] Lydia Bourouiba, Eline Dehandschoewercker, and John WM Bush. Violent expiratory events: on coughing and sneezing. Journal of Fluid Mechanics, 745:537-563, 2014.

[7] Shengwei Zhu, Shinsuke Kato, and Jeong-Hoon Yang. Study on transport characteristics of saliva droplets produced by coughing in a calm indoor environment. Building and environment, 41(12):1691-1702, 2006.

[8] BE Scharfman, AH Techet, JWM Bush, and L Bourouiba. Visualization of sneeze ejecta: steps of fluid fragmentation leading to respiratory droplets. Experiments in Fluids, 57(2):24, 2016.

[9] Christopher K Wikle and L Mark Berliner. A bayesian tutorial for data assimilation. Physica D: Nonlinear Phenomena, 230(1-2):1-16, 2007.
[10] Rossella Arcucci, César Quilodrán Casas, Aniket Josh, Asiri Obeysekara, Laetitia Mottet, Yi-Ke Guo, Christopher Pain - Merging Real Images with Physics Simulations via Data Assimilation- EuroPar 2021, Lecture Notes in Computer Science (in print)
[11] Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. Mathematical programming, 45(1-3):503-528, 1989.
[12] G. Bradski. The OpenCV Library. Dr. Dobb's Journal of Software Tools, 2000.
[13] F. Iannone et al., "CRESCO ENEA HPC clusters: a working example of a multifabric GPFS Spectrum Scale layout," 2019 International Conference on High Performance Computing & Simulation (HPCS), Dublin, Ireland, 2019, pp. 1051-1052, doi: 10.1109/HPCS48598.2019.9188135.

# LONG TIMESCALE MOLECULAR DYNAMICS AND ONE TRILLION VIRTUAL SCREENING ON HPC5

Francesco Frigerio[1*], Silvia Pavoni[1], Alessandro Grottesi[2], Neva Bešker[2], Andrew Emerson[3], Federico Ficarelli[3], Giorgia Frumenzio[3], and Carmine Talarico[4]

[1]*Eni SpA, Physical Chemistry Department, via Maritano 26, IT-20097, San Donato Milanese (Mi), Italy*

[2]*CINECA, HPC Department, via dei Tizii 6, IT-00185 Roma, Italy*

[3]*CINECA, HPC Department, via Magnanelli 6/3, IT-40033 Casalecchio di Reno (Bo), Italy*

[4]*Dompé Farmaceutici SpA, via Campo di Pile, IT-67100, L'Aquila, Italy*

**ABSTRACT**. The EXSCALATE4CoV project, funded by the European Union and coordinated by Dompé Farmaceutici, set up a platform for an immediate response to COVID-19. Its main goal is the identification of the most promising safe-in-man drugs and de-novo small molecules to be active against the SARS-CoV-2 virus. The computational pipeline implemented on the Eni HPC5 GPU-accelerated supercomputing facility comprises: 1) very long Molecular Dynamics simulations of all SARS-CoV-2 proteins, with selection of a few conformational clusters for each (performed by GROMACS); 2) massively parallel Virtual Screening of a huge molecular library on the best selected protein clusters (by LIGEN). The following experimental pipeline takes on in-silico molecule candidates throughout laboratory tests and ends up with clinical trials at Italian hospitals. The key achievement so far is the identification of the osteoporosis drug Raloxifene as possibly active against COVID-19. It is already registered and generic, with known safety characteristics (e.g. dosage and side effects). Application has been made to the European Medicines Agency and clinical trials have started

* Corresponding author. E-mail: *Francesco.Frigerio@eni.com*

## 1 Introduction

At the beginning of 2020 the European Union (EU) funded the EXSCALATE4CoV (E4C) project within Horizon 2020. This action set up a platform for an immediate response to COVID-19 in the search of new molecules as potential drugs against the new Coronavirus SARS-CoV-2. E4C (exscalate4cov.eu) is coordinated by Dompé Farmaceutici and brings together 18 partners among institutions and top research centers from 7 European countries, including CINECA from Italy.


**Fig.1**: the Eni HPC5.

In March 2020 Eni offered its HPC5 facility (Figure 1) to support the big computational effort required in the project. With 58 PetaFLOPS and hybrid CPU/GPU architecture, HPC5 was the most powerful supercomputer for industrial use in the world. Since then the computing power made available by Eni joined the 32 PetaFLOPS of M100 - contributed by CINECA - in the infrastructure deployed for the E4C platform. Molecular Dynamics (MD) and Virtual Screening (VS) experiments were performed on both HPC resources following the project timeline.

# 2 Eni Collaboration with E4C

### 2.1 Main goal of the project

E4C aims at the identification of the most promising safe-in-man (SIM) drugs and de-novo small molecules to be active against the SARS-CoV-2 virus. This is to be accomplished along a clearly defined mixed pipeline (Figure 2) with computational actions, biochemical studies, in vitro and in vivo experiments and finally hospital trials.

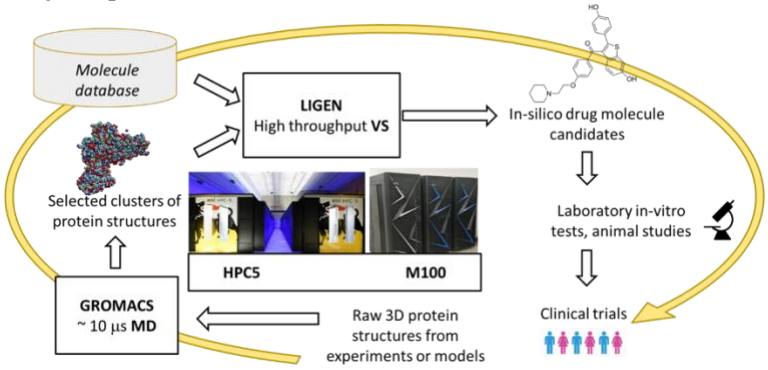



**Fig.2**: the E4C computational/experimental pipeline.

Within the E4C platform, two main computational items were implemented by research scientists of CINECA and Eni on M100 and HPC5 (Table 1):

- Very long MD simulations of the experimental crystal structures and the homology models of all SARS-CoV-2 proteins. They were followed by the selection of a few representative clusters of conformations for each protein.
- Massively parallel VS of very large SIM molecular libraries through docking into carefully identified sites of the best selected protein clusters.

**Table 1**: Hardware resources.

| System name | Eni HPC5 | CINECA M100 |
|---|---|---|
| Nodes | 3400 | 980 |
| Processors (node) | 2x24 Intel Cascade Lake @ 2.1 GHz | 2x16 IBM P9 @ 3.1 GHz |
| Accelerators (node) | 4 x Nvidia Tesla V100 | 4 x Nvidia Tesla V100 (NVLink CPU-GPU) |
| Memory (node) | 56 GB | 200 GB |
| Peak Performance | 51.7 PFlops | 32PFlops |
| TOP500 (06/2020) | 6$^{th}$ | 9$^{th}$ |

Details of both computational items are described in the following.

### 2.2 MD simulations

For every SARS-CoV-2 protein the initial structure was taken from published X-ray crystallography determinations or from available homology models (an example in Figure 3).

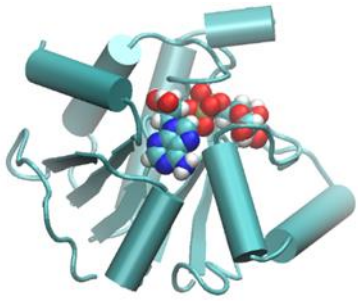


**Fig.3**: The structure of the SARS-CoV2 protein NSP3 in complex with its ligand [1].

The initial dataset contained 30 unique structures and, due to known variants and models, a total of 46 simulations were started according to a supervised MD protocol. High, medium or low priorities were assigned according to their role in viral infection and replication. The AMBER ff99 forcefield [2] was adopted and the GROMACS program suite [3] was used for simulation and analysis. The GPU acceleration characterizing HPC5 and M100 was fully applied to GROMACS MD simulations.
The system preparation involved the assignment of forcefield parameters, the protein soaking into water, the possible addition of inorganic ions for electrical neutrality, the energy minimization and structure relaxation to remove any close contacts. This was followed by long simulations in the NVT ensemble at ambient pressure and temperature with a timestep of 2 femtoseconds. For most proteins 10 microseconds long trajectories were produced, while in a few specific cases 20 microseconds were reached. All results are being periodically deposited in a publicly available cloud for the whole scientific community (see further).
The structure equilibration was checked with Root Mean Square Deviation analysis. Each trajectory was subjected to Principal Component Analysis and Cluster Analysis. These treatments allowed to identify major clusters and to extract only a few highly representative protein conformations for the subsequent VS treatment with LiGen [4].

### 2.3 VS experiments

The VS platform (Figure 4) is based on the adaptation of the LiGen program to the GPU accelerated HPC5 and M100 facilities. Through extremely efficient docking evaluation of protein/ligand complementarity a high-performance in silico screening of huge databases of chemical structures was made possible.

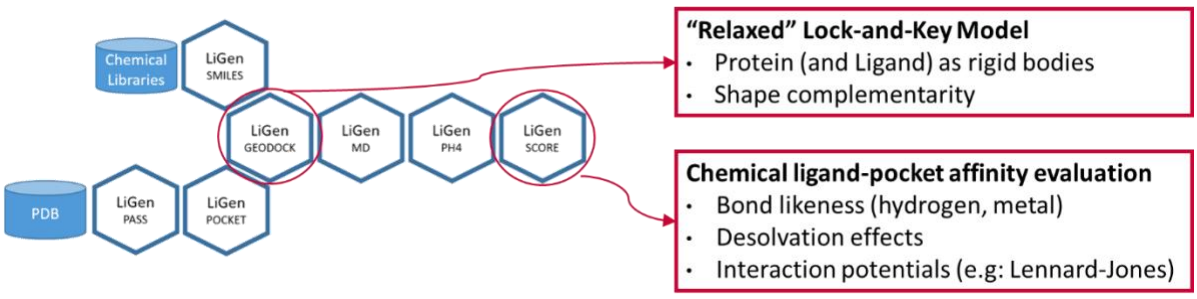



**Fig.4**: the E4C VS software.

The LiGen code base is owned by Dompè and was co-developed by CINECA and Politecnico di Milano along a production process lasting for about 20 years. Its main goal is the exploration of a tangible chemical space, collecting known chemical structures, whose synthesis is considered achievable in one reaction step from commercial reagents.
The main effort was the need to port all relevant algorithms (Figure 5) to GPU-accelerated HPC platforms using the CUDA parallel computing platform.

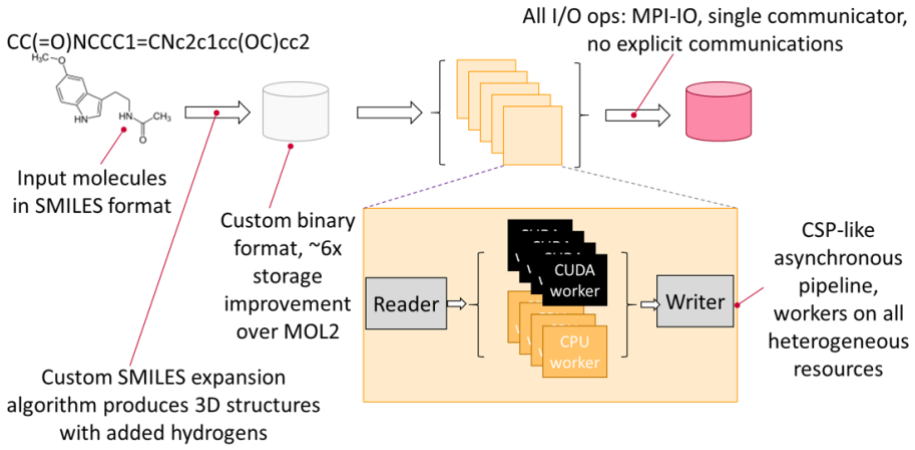



**Fig**. **5**: the E4C VS application architecture.

The input molecules are defined in the SMILES format. A custom SMILES expansion algorithm produces 3D structures with added hydrogens in binary format, with ~6 times storage improvement over reference method MOL2. All I/O works as MPI-IO, single communicator, with no explicit communications.
The resulting VS scale up (Figure 6) typically obtained a sustained single-node throughput of ~1600 ligands/s (~16000 poses/s, 10 best poses kept for scoring) on the full M100 system.

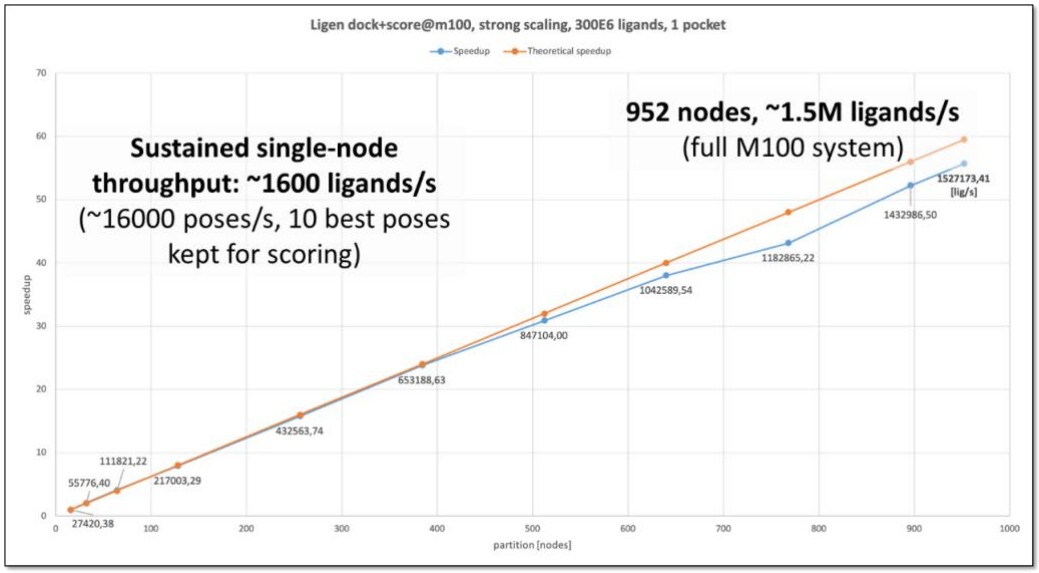



**Fig**.**6**: VS scale up on M100.

The first VS run of the E4C studied a database containing 400000 molecules. This in silico test selected ~7000 candidates while, throughout following in vitro tests, a total of ~100 structures were deemed

interesting. At the end of this VS run ~40 candidates were experimentally found to be effective in limiting the SARS-CoV-2 virus replication.
A key achievement was then obtained during the 2020 spring: the identification of the osteoporosis drug Raloxifene (Figure 7) as possibly active against COVID-19.

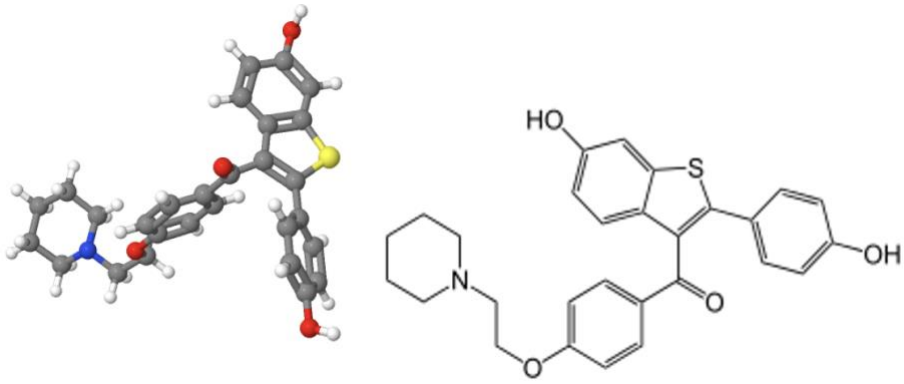



**Fig.7**: Two depictions of the chemical structure of Raloxifene.

This is a registered and generic drug with known safety characteristics (e.g. dosage and side effects). As a consequence, an application was made to the European Medicines Agency (EMA) and clinical trials have started.
The second run of the described simulation pipeline resulted in the largest VS experiment ever run. In November 2020 a library of 71 billion ligands was docked and scored in 15 active sites of 12 SARS-CoV-2 proteins: PLPRO, SPIKEACE, NS12thumb, NS12Palm, NSP12ortho, 3CL, NSP13allo, NSP13ortho, NSP3, NSP6, NSP9, NSP14, NSP15, NSP16, Nprot. Its workflow (Figure 8) is essentially the same as in the first run and it is composed of three steps: the input data preparation, the docking and scoring, the data post-processing.

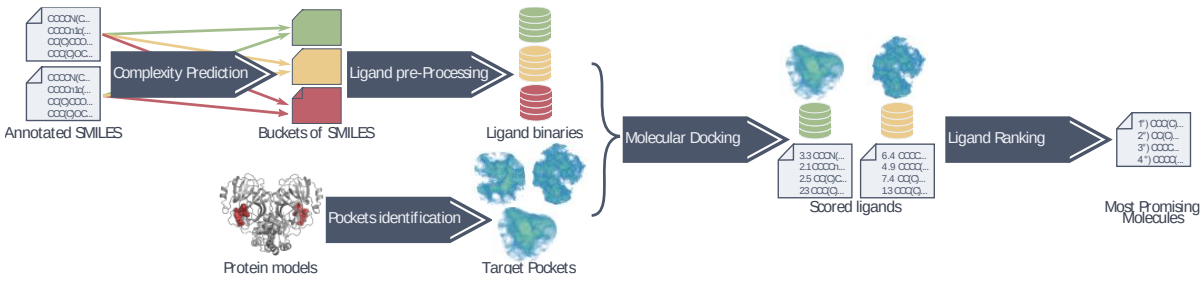



**Fig.8**: The VS workflow.

This meant that one trillion interactions were evaluated in 60 hours on the CINECA M100 (800 nodes) and the Eni HPC5 (1500 nodes). In details, that was accomplished by 2 K ligands/second/HPC-node and 5 M ligands/second overall. The post processing of the 65 TB of data generated by the big experiment is still ongoing.
A similar experiment run in June 2020 on the SUMMIT supercomputer at Oak Ridge National Lab evaluated less than 3 Billion interactions (1.4 B ligands in 2 sites). Therefore, the E4C experiment was 300 times larger and 500 times faster.

## Conclusions

All scientific results are being published (https://www.exscalate4cov.eu/contribute.html#papers). Moreover, all produced data are being made public on mediate.exscalate4cov.eu (Figure 9), that is the

Web portal of the MEDIATE initiative and was designed in collaboration with SAS. Arieh Warschel (Nobel prize winner for Chemistry in 2013), Rossen Apostolov, Igor Tetko and Yang Ye sit in its Scientific Committee. That is the starting point for any scientific and practical collaboration and it declares the following purpose: "to collect the best possible chemical library of novel inhibitors of SARS-COV-2 targeting the most relevant viral proteins combining as much as possible VS simulations and Artificial Intelligence (AI) predictions generated by the best drug hunters worldwide".

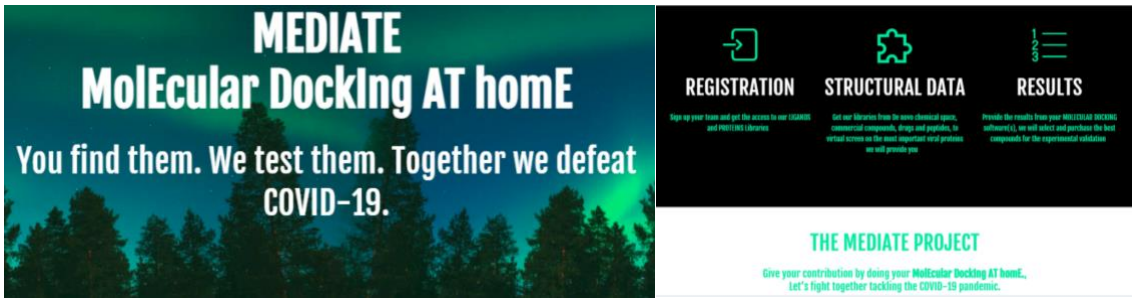



**Fig**.**9**: the MEDIATE initiative for collaboration on the E4C results.

After registration, it is possible to participate by taking at least one of five actions: submission and data collection of crowdsourced simulations, compound selection (generation of a single final ranking by using machine learning and AI technologies, thanks to the SAS platform), compounds acquisition (purchase of the best candidates selected in phase 2 of the VS), testing (the libraries will be tested in all E4C COVID-19 assays and the most promising could be crystallized), data sharing (results on the compounds will be published in public portals and made available).
The first VS phase of the E4C project selected Raloxifene as the best candidate for the hospital trials, that are currently ongoing. A new list of candidate drugs is going to come out after concluding the analysis of the second VS phase results. Some example can be found in the MEDIATE homepage (Figure 10).
Our acknowledgements go to the Project "EXaSCale smArt pLatform Against paThogEns for Corona Virus - Exscalate4CoV" funded by the EU's H2020-SC1-PHE-CORONAVIRUS-2020 call under the grant N. 101003551. The two E4C computational teams are composed by:

- Carmine Talarico (Dompè), Alessandro Grottesi (CINECA), Andrew Emerson (CINECA), Neva Besker (CINECA), Giorgia Frumenzio (ABD) and Francesco Frigerio (Eni S.p.A.) for MD;
- Federico Ficarelli (CINECA), Chiara Latini (CINECA), Davide Gadioli (Politecnico di Milano), Emanuele Vitali (Politecnico di Milano), Gianluca Palermo (Politecnico di Milano), Carlo Cavazzoni (Leonardo S.p.A.) for VS.

Dedicated and efficient contributions to hardware operation and software installation and maintenance by Eni and CINECA support teams are kindly acknowledged.

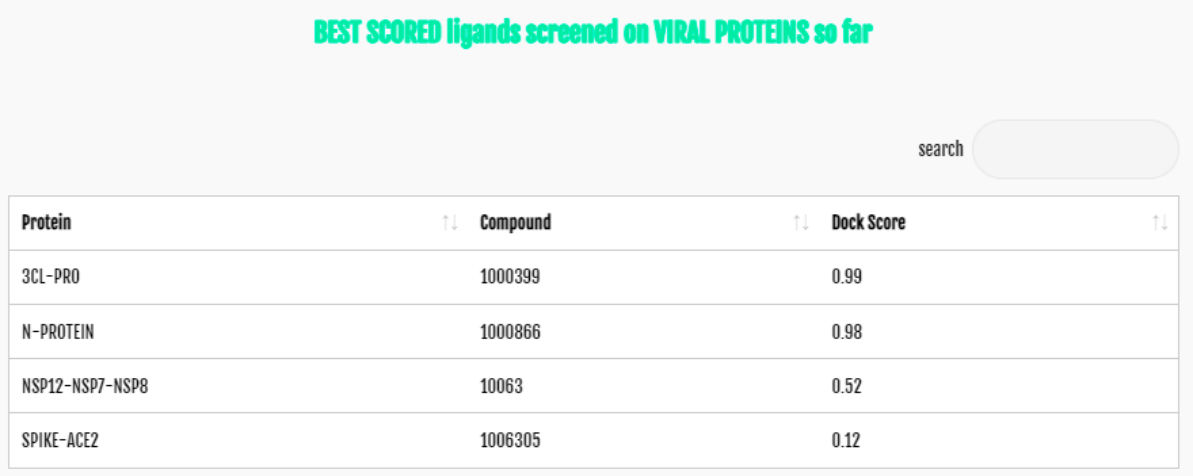



**Fig.10**: MD and VS simulation results published within MEDIATE.

# References

[1] K. Michalska, Y. Kim, R. Jedrzejczak, N. I. Maltseva, L. Stols, M. Endres and A. Joachimiak. Crystal structures of SARS-CoV-2 ADP-ribose phosphatase: from the apo form to ligand complexes. *International Union of Crystallography Journal* **7**, pp. 814-824, (2020).

[2] J. Wang, P. Cieplak and P. A. Kollman. How well does a restrained electrostatic potential (RESP) model perform in calculating conformational energies of organic and biological molecules? *Journal of Computational Chemistry* **21**, pp. 1049-1074, (2000).

[3] M.J. Abraham, T. Murtola, R. Schulz, S. Páll, J.C. Smith, B. Hess and E. Lindahl. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, **1–2** pp. 19–25 (2015).

[4] A.R. Beccari, C. Cavazzoni, C. Beato and G. Costantino. LiGen: A High Performance Workflow for Chemistry Driven de Novo Design. *J. Chem. Inf. Model.* **53**, pp. 1518–1527 (2013).

# A MULTI-SCALE APPROACH FOR MODELING SALIVA DROPLETS AIRBORNE TRANSPORT IN RELATION TO SARS-COV-2 TRANSMISSION

Valerio D'Alessandro*, Matteo Falone, Renato Ricci

*Università Politecnica delle Marche, Dipartimento di Ingegneria Industriale e Scienze Matematiche, Via Brecce Bianche, 12, 60131, Ancona (Italy)*

**ABSTRACT.** It is well known that SARS–CoV–2, can be transmitted through airborne diffusion of saliva micro-droplets which travels into atmospheric air through a thermo-fluid dynamic interaction with it.
In order to limit SARS-CoV-2 spread, social distancing is crucial. However, in this context is really important to emphasize that available knowledge is largely inadequate to make predictions on the airborne diffusion of infectious droplets emitted during a cough and/or sneezing. Consequently, it is very difficult to achieve proper guidelines for social distancing rules based on scientific evidences. For this reason many research groups have been devoted their efforts in order to gain new insight into the transport of fluids and particles originated from human respiratory tracts.
The main aim of our research activity is to provide a contribution to thermo-fluid dyamic modelling of saliva droplets diffusion produced by coughing. Thus, a campaign of HPC computations were performed on ENEA CRESCO in order to give a better understanding on: (i) saliva droplets diffusion physics; (ii) the effectiveness of the social distancing rules adopted in Italy during the pandemic.

* Corresponding author. E-mail: v.dalessandro@univpm.it

## 1 Introduction

It well known that several viruses, as well as SARS-CoV-2, can be transmitted through airborne diffusion of saliva micro-droplets.
Typical infection mechanisms are the discussed in Mittal et al. [1] and they can be classified as follows: (i) direct transfer of large droplets to the receiver's conjunctiva, mouth, or nose; (ii) physical contact with droplets deposited on the surface and subsequent absorption to the nasal mucosa of the receiver; and (iii) inhalation of respiratory ejected aerosolized droplet nuclei. For this reason, many countries in the world have imposed variable social distances to be maintained between persons.
The physical phenomena involved in all the droplets' transmission processes are very complex. Indeed, after their emission, saliva droplets travel as a result of their inertia and their aerodynamic interaction with moist air. Furthermore, the mass of each droplet can vary due to evaporation, which is strictly connected to environmental temperature and relative humidity. In most cases, droplets' evaporation can produce a supersaturated solution of sodium chloride (NaCl) in water which are the main components of human saliva; this condition triggers crystallization nuclei and their growth that is finalized in the solid dry nuclei (also known as droplet nuclei). Therefore, it is very straightforward to understand that the study of thermo-fluid dynamics aspects related to a saliva cloud deriving from coughing or sneezing

is really complex. However, droplets' dynamics is the key ingredient to determine the guidelines on social distancing, face mask wearing, and the implementation of new practices in daily social life. This is the reason why after the unprecedented COVID-19 pandemic, several research groups have started research efforts in order to gain new insight into the transport of fluids and particles produced by human respiratory tracts.

The aim of this research work is the development of a new computational model, relying on the well-established OpenFOAM library [2], for the evaluation of saliva droplets' dynamics during coughing. In addition, it is also our intention to develop a mathematical model to predict crystallization kinetics, within a saliva droplet, triggered from NaCl/water solution supersaturation. To this end, this research is intended to advance the state of the art by developing a multi-scale mathematical model that can predict droplet nuclei generation and their space evolution.

Finally, we have also focused on the possibility to reduce SARS-CoV-2 transmission potential by means of ultraviolet-C (UV-C) radiation. Nevertheless, for the sake of compactness, we will invite the interested readers to our former paper, [3].

This paper is organized as follows: the governing equations are reported in Sec. 2, while the numerical approximations are discussed in Sec. 3. Numerical results are shown in Sec. 4. Finally, Sec. 5 contains the conclusions.

# 2 Governing equations

Our numerical computations are developed using an Eulerian–Lagrangian framework described in the following. In particular, the particle-source-in-cell (PSI-Cell) method [4] is adopted to couple Eulerian and Lagrangian phases, while Population Balance Equation (PBE) is solved at droplet level to take into account salt crystallization kinetics related issues.

### 2.1 Eulerian phase

Eulerian phase was modeled using compressible Reynolds Averaged Navier-Stokes (RANS) equations:

$$\begin{aligned}
&\frac{\partial \bar{\rho}}{\partial \mathrm{t}} + \frac{\partial}{\partial x_j}\left(\bar{\rho}\tilde{u}_j\right) = s_m, \\
&\frac{\partial}{\partial t}\left(\bar{\rho}\tilde{u}_i\right) + \frac{\partial}{\partial x_j}\left(\bar{\rho}\tilde{u}_i\tilde{u}_j\right) = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial \hat{\tau}_{ij}}{\partial x_j} + \bar{\rho} g \delta_{i3} + s_{m,i}, \\
&\frac{\partial}{\partial t}\left(\bar{\rho}\tilde{E}\right) + \frac{\partial}{\partial x_j}\left(\bar{\rho}\tilde{u}_j\tilde{H}\right) = -\frac{\partial q_j}{\partial x_j} + \frac{\partial}{\partial x_j}\left(\tilde{u}_i\hat{\tau}_{ij}\right) + s_e, \\
&\frac{\partial}{\partial t}\left(\bar{\rho}\tilde{Y}_k\right) + \frac{\partial}{\partial x_j}\left(\bar{\rho}\tilde{u}_j\tilde{Y}_k\right) = -\frac{\partial m_{k,j}}{\partial x_j} + s_{Y_k},
\end{aligned} \tag{1}$$

where $\bar{\rho}$, $\tilde{u}$, $\bar{p}$, and $\tilde{Y}_k$ denote density, velocity component in $x_i$ direction, pressure, temperature and chemical specie *k* mass fraction. $\tilde{E}$ and $\tilde{H}$ are, respectively, the total internal energy and enthalpy. It is important to note that the in the above equations overbar and the tilde are filtering operators which are introduced for unweighted and density–weighted averages.

As regards the unclosed terms reported in eq. 1, they are handled using standard constitutive equations, *i.e.* Fick law, Fourier law and rheological equation for Newtonian fluids. Turbulence modeling is performed using standard SST k-ω model developed by Menter [5] not described here for the sake of compactness. The source terms appearing in eq. (1) right hand side correspond to coupling between Lagrangian and Eulerian phases.

### 2.2 Lagrangian phase

Saliva particles are handled using a Lagrangian frame throughout the computational domain. It is worth noting that, within OpenFOAM Lagrangian libraries, for trivial efficiency reasons, the concept of computational parcel is adopted. This means that particles are organized in groups, called parcels, and each parcel represents the center of mass of a small cloud of particles having the same properties.
Non-collisional spherical parcels are employed in our approach. Thus, position and velocity are the results of the trajectory and momentum equations. The particles considered in this research work are sufficiently small to neglect pressure and virtual mass forces and sufficiently large to neglect Brownian forces [6,7]; hence the only forces acting on the parcels are the following: gravity, aerodynamic drag and buoyancy. Aerodynamic drag coefficient is obtained from Putnam correlation, [8]. Mass and energy equations were also solved for each parcel; the convective heat transfer coefficient and the mass transfer one are obtained from the Ranz-Marshall correlation for Nusselt and Sherwood numbers, [9].
The Rosin–Rammler distribution is used for representing initial parcels' diameter. The curve parameters calibration are the same discussed in D'Alessandro et al. [3].

### 2.3 PSI-PBE approach

In this study we propose a coupling strategy between PSI-Cell method and PBE in order to simulate also the nucleation and growth of NaCl crystals within a saliva droplet. It is worth emphasizing that PBE is considered within a Lagrangian frame; this approach allows to model micro-scale particles behavior induced by meso-scale thermo-fluid dynamic phenomena. Therefore, in our approach PSI-Cell method is used in order to predict the particles interaction with Eulerian phase, while PBE is solved at parcel level to consider droplet nuclei generation.
It is also important to note that the PBE of a parcel includes the time derivative and particle dimensions and can ignore the spatial dimension, which removes the convective and diffusive terms. A similar formulation increases the numerical stability and reduces the computational costs without eliminating relevant mathematical details.
PBE for spatially in-homogeneous crystallization processes can be written as follows, [10]:

$$\frac{\partial N_j}{\partial t} + \boldsymbol{\nabla} \cdot \left(N_j \boldsymbol{u}_p\right) - \boldsymbol{\nabla} \cdot \left(D_t \boldsymbol{\nabla} N_j\right) = \sum_j \frac{\partial \left(G_j N_j\right)}{\partial r_j} + B \prod_j \delta\left(r_j - r_{j0}\right) + h \tag{2}$$

where $N_j$ is the NaCl crystals number density within a parcel, $D_t$ is the local turbulent diffusivity, $G_j$ is the growth rate, $r_j$ is the particle internal coordinate, $r_{j0}$ is the particle internal coordinate for a crystal nucleus, $\delta$ is the Dirac function, $B$ is the nucleation rate, and h is the creation or destruction of particles due to aggregation, agglomeration, and breakage.
In this study we refer only to crystallization phenomenon and this is the reason why PBE needs to take into account only nucleation and growth of particles. Moreover, the convective and diffusive terms for the particles disappear because we assume each parcel to be well mixed and tracked independently in the Lagrangian frame.
The following semi-discrete PBE is obtained after integrating eq. (2) over $r$:

$$\frac{\partial f_j}{\partial t} = -\frac{1}{\Delta r}\left[G_{j+1/2}\left(f_j + \frac{\Delta r}{2}\left(f_r\right)_j\right) - G_{j-1/2}\left(f_{j-1} + \frac{\Delta r}{2}\left(f_r\right)_{j-1}\right)\right] \tag{3}$$

where $f_j$ is the parcel-averaged population density. Note that, the nucleation term is included in the cell, corresponding to the nuclei size, by averaging the nucleation rate.
Starting from eq. (3) solution the parcel-averaged crystal mass can be evaluated as in Woo et al., [10]:

$$N_{w,j} = -\frac{1}{4}\rho_c k_v f_j \left(r^4_{j+1/2} - r^4_{j-1/2}\right). \tag{4}$$

Therefore, semi-discrete PBE can be formulated as follows:

$$\frac{\partial N_{w,j}}{\partial t} = -\frac{\rho_c k_v}{\Delta r}\left(r^4_{j+1/2} - r^4_{j-1/2}\right) max\,(sign\,\Delta c, 0) \times \left[G_{j+1/2}\left(f_j + \frac{\Delta r}{2}(f_r)_j\right) - G_{j-1/2}\left(f_{j-1} + \frac{\Delta r}{2}(f_r)_{j-1}\right)\right] + B|_{j=0}. \tag{5}$$

In the above equation $\Delta c = c - c^*$ is supersaturation, while $c$ and $c^*$ are the NaCl concentration and its solubility in pure water. Nucleation and growth rates coefficients present in eq. (5) are obtained from literature experimental data for NaCl/water solutions [16,17].
From eq. (5), we can also calculate the total mass related to crystallization process as follows:

$$m_{cr} = \pi \frac{D_p^3}{6} \sum_j N_{w,j}. \tag{6}$$

Lastly, the radius of the (dry) solid part of the droplets, $r_N$, is obtained as follows:

$$r_N = \frac{\sum_j N_{w,j} r_j^4}{\sum_j N_{w,j}\, r_j^3}. \tag{7}$$

# 3 Numerical approximation

The governing equations solution relies on the unstructured, collocated, cell-centered finite volume approach available within OpenFOAM library. An implicit, three level, second-order scheme was used for the time integration together with the dynamic adjustable time stepping technique for guaranteeing a local Courant (Co) number less than a user-defined value.
Face interpolation of convective fluxes is handled by the linear upwind scheme, whereas diffusive terms are discretized by a standard second-order central scheme. Furthermore, pressure–velocity coupling is handled through the Pressure-Implicit with Splitting Operators (PISO) procedure. As regards linear solvers, a preconditioned conjugate gradient (PCG) method with a diagonal incomplete-Cholesky preconditioner was used to solve the pressure equation. A preconditioned bi-conjugate gradient (PBiCG) method with the Diagonal Incomplete Lower Upper (DILU) preconditioner was adopted instead for the remaining equations. In particular, a local accuracy of $10^{-7}$ was established for the pressure, whereas other linear systems were considered as converged when the residuals reached the machine precision.
In the present work, a 3D computational domain was considered. It consists of an air volume starting from the mouth print of a standing coughing person. A length L = 4 m, a width W = 1 m, and a height H = 3 m were adopted, in accordance with Dbouk and Drikakis, [11]. A suite of three fully structured

grids, having hexahedral cells were built in order to discretize the domain. All the related details as well as the space-time convergence study can be found in our former paper [3].
Lagrangian phase momentum and mass equations were solved using a standard Euler scheme for time-integration. Energy equation was solved analytically. A special care was devoted to PBE which was time-integrated using an explicit Strong Stability Preserving Runge-Kutta (SSPRK) having 9 stages and 5-th order of accuracy, [15], to avoid blow-up of the computations. Moreover, $(f_r)_j$ terms, are approximated by the min-mod limiter [10].

### 3.1 Initial and boundary conditions

A stepped velocity inlet at the mouth boundary was applied to mimic the human cough over 0.12 s. The velocity inlet value was deduced on the basis of measurements carried by Scharfman et al., [12] and it is equal to 8.5 m/s in the stream-wise direction both for the carrier fluid and injected parcels.
Saliva droplets were also injected into the domain from the mouth print boundary. Specifically, the initial total mass of saliva droplets laden into the domain for a single cough event is 7.7 mg, according to the experimental measurements performed by Xie et al., [13] and CFD simulations of Dbouk and Drikakis, [11]. All the other relevant details can be found in D'Alessandro et al., [3].
The initial temperature of the carrier fluid is 20° C with relative humidity fixed at 50%. The ground is at 25° C, while the air and droplets ejected by human mouth are at 34° C. No background flow is included in the presented computations. However, initial fields consistent with atmospheric conditions were adopted. These fields are obtained through a preliminary computation which does not provide Lagrangian particles into the domain. It is worth noting that cloud evolution is strongly influenced by this conditions, [3].
A key role is played by saliva chemical composition which in general a complex fluid. However, in many CFD oriented papers it is modeled as pure water. In this work we take into account the presence of NaCl within saliva droplets. In particular, we consider NaCl completely diluted in water and related concentration value are obtained from Rosti et al. [14].

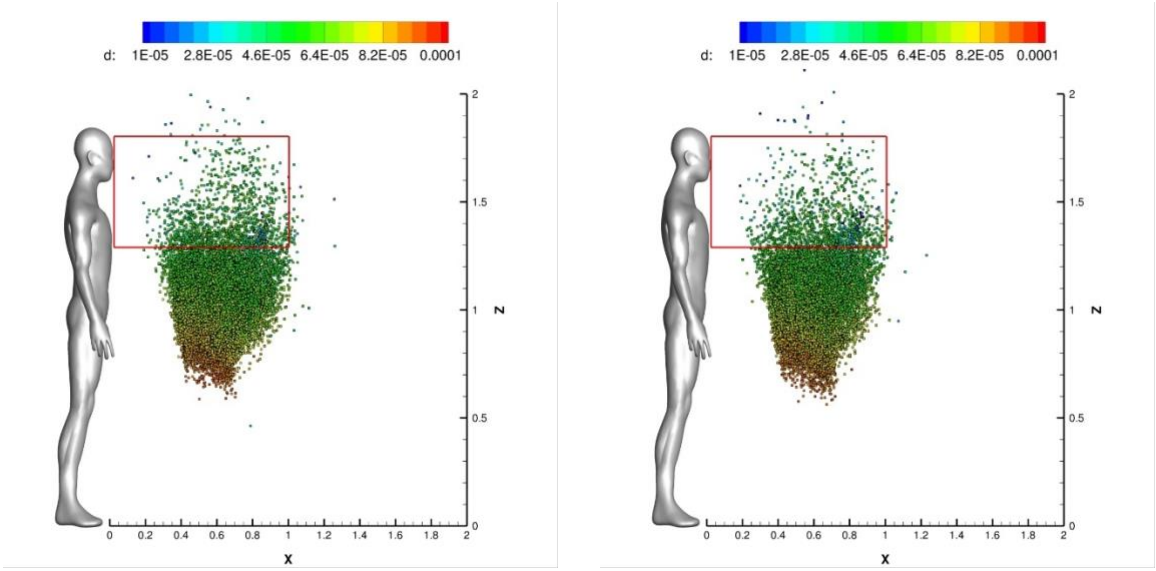
## 4 Results

In this section we present the obtained numerical results referred to the saliva droplets produced during coughing.
Some cloud characteristics are computed in order to investigate its diffusion, i.e., (i) the cloud center of mass; (ii) fraction of particles present in a reference volume.
The cloud center of mass is defined as follows:

$$\boldsymbol{G} = \frac{\sum_{i=1}^{N_p(\Omega_0)} m_{P,i}\boldsymbol{x}_{P,i}}{\sum_{i=1}^{N_p(\Omega_0)} m_{P,i}} \tag{8}$$

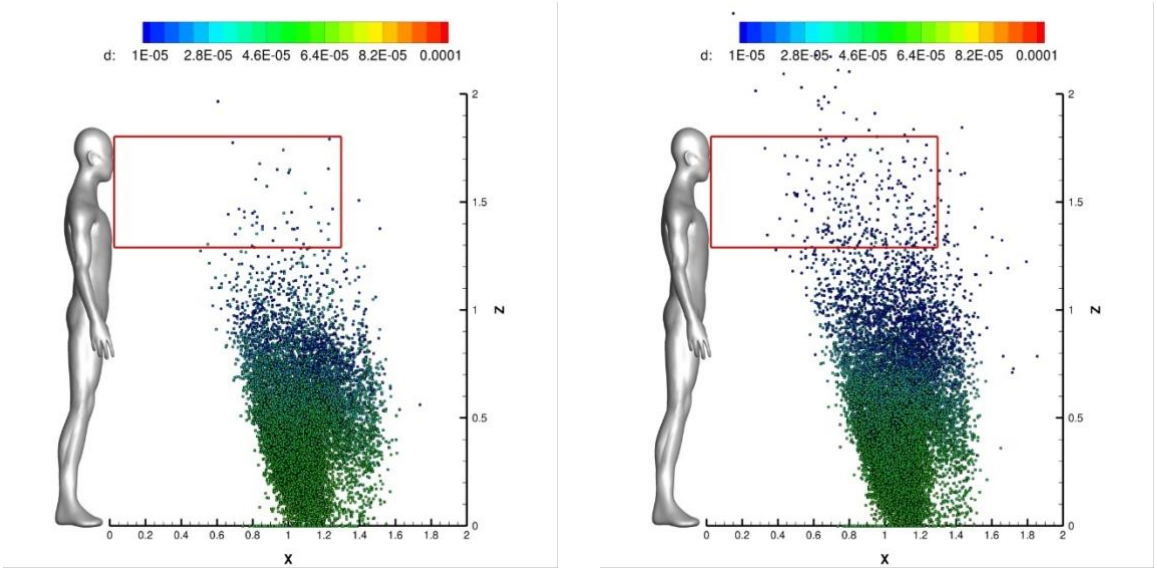
where $N_p(\Omega_0)$ is the number of parcels laden in the overall domain, $\Omega_0$, in a given time-instant. In the following, $\boldsymbol{G} = (x_G, y_G, z_G)$ is considered as the center of mass components. It is worth noting that x-axis is associated to stream-wise direction, y-axis represent the transverse direction, while z is the vertical one. On the other hand, the ratio between the number of particles present in a reference volume, $\Omega_i$, and the total number of particles in $\Omega_0$ (in a given time instant) is used to track the droplets' population distribution in risk zone. The reference index is defined as:

$$\Phi_{\Omega} = \frac{\sum_{k=1}^{N_p(\Omega_i)} N_{p,k}}{\sum_{k=1}^{N_p(\Omega_0)} N_{p,k}} \tag{9}$$




(a) Pure water droplets (b) Salty droplets

**Fig. 1:** Cloud representation at t = 4 s. Parcels are coloured with the particle diameter.




(a) Pure water droplets (b) Salty droplets

**Fig. 2:** Cloud representation at t = 10 s. Parcels are colored with the particle diameter.

The aforementioned reference volume, $\Omega_i$, is parallelepipedal type having the following features:

$$\Omega_i = [0, \alpha_i] \times [-0.5, 0.5] \times [1.3, 1.8] \tag{10}$$

the parameter $\alpha_i$, appearing in eq. (10), spans the following values: 1.0m and 1.3 m. $\Omega_i$ stream-wise dimension was selected in order to investigate the effectiveness of 1 m distance, which is the safety distance adopted in Italy. The transverse direction range is considered in order to completely cover the

domain. The z-axis interval is defined for acting on a sufficiently wide range of possible virus receivers' heights.

In this section we mainly address saliva chemical composition effect. More in depth, we discuss the impact of NaCl presence within droplets, rather than pure water particles, on the related cloud space-time evolution. In this framework, looking at Fig. 1 (a) and Fig. 1 (b), it is very easy to perform a qualitative analysis about clouds behavior. Indeed, at t = 4 s the number of particles contained within red lines are almost the same as also corroborated by Fig. 3. However, it is important to stress that the clouds shape is clearly different. A similar condition is mainly related to the different density of the droplets laden into the domain. Actually, being the initial parcels' diameter distribution the same for both pure water and salty water based droplets, trajectories are significantly affected by a similar condition. Moreover, up to t = 4 s the crystallization kinetics is still in a premature stage to have notably effect on particles evolution.

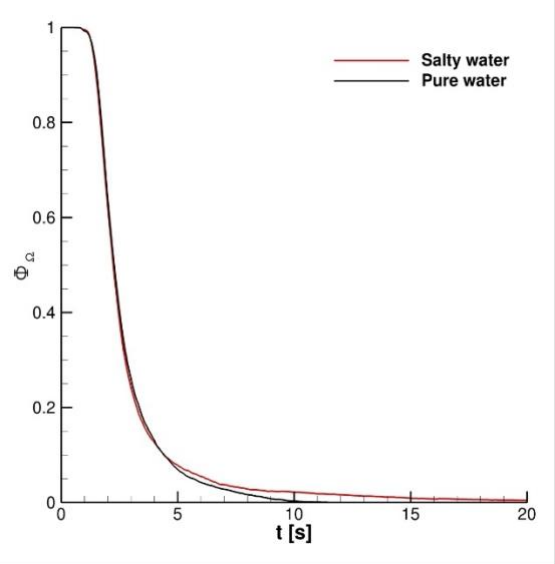



**Fig. 3:** Fraction of particles present in the volume $\Omega_i$

As can be noted in Fig. 2 (a) and Fig. 2 (b), at t = 10 s the effect of NaCl crystallization process produces an evident impact on the parcels' space distribution. In particular, the risk area is sensibly more populated, even for a stream-wise length greater than 1 m, when salty droplets are considered. This evidence is also expressed quantitatively from $\Phi_\Omega$ time-history represented in Fig. 3. It is very important to put in evidence that the particles contained into $\Omega_i$ domain are dry and they have diameters approximately around 10 μm. For this reason a fly time having an order of magnitude of 60 s is expected. The center of mass trajectory on the x–z plane is shown in Fig. 4. For 0.35 m< $x_G$ < 1 m, the $x_G$, $z_G$ curve underlines a peculiar trend. Actually, the trajectory is almost linear due to the cancellation of the inertial term. For $x_G$ > 1 m the impact of droplets chemical composition is evident and it is related to the crystallization effects and the consequent droplet nuclei generation. For $x_G$ > 2 m case the curve $z_G$= f( $x_G$) is extrapolated since the background flow is extinct and the Lagrangian particles have uniform velocity. It is really evident that dry nuclei are able to reach distance sensibly longer than pure water droplets which completely evaporates from the domain in less than 20 s.

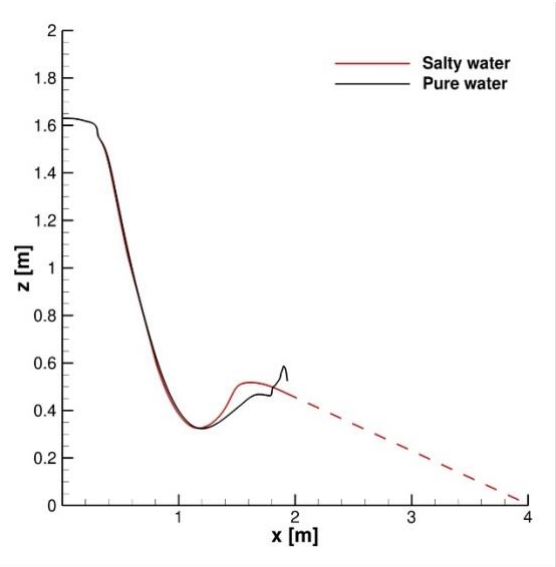



**Fig. 4:** Saliva cloud center of mass evolution

## 5 Conclusions

This paper addresses the development and application of an Eulerian–Lagrangian model for saliva droplets' cloud deriving from coughing. This work focuses on no-wind configurations, thus a field initialization strategy was also developed.
In addition, we have also introduced a multi-scale mathematical model to predict NaCl crystallization kinetics. In particular, a coupling approach between PSI-Cell method and PBE was proposed. In this context, it was showed that the chemical composition of saliva droplets laden into CFD domain strongly affects the obtained solution. Specifically, the NaCl crystallization produces dry particles having small diameters which can be transported for time longer than pure water droplets.
Future work will be devoted to more realistic configuration in several relative humidity conditions since this parameter strongly affect evaporation phenomena hence crystallization kinetics.

## Acknowledgments

The authors want to acknowledge "Associazione Nazionale Big Data" that awarded this research work within COVID-19–Fast access to the HPC supercomputing facilities program. We acknowledge ENEA for awarding us access to CRESCO6 based at Portici.

## References

[1] R. Mittal, R. Ni, and J.-H. Seo, "The flow physics of COVID-19," J. Fluid Mech. 894, F2 (2020).
[2] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby, "A tensorial approach to computational continuum mechanics using object-oriented techniques," Comput. Phys. 12, 620–631 (1998).
[3] V. D'Alessandro,M. Falone, L. Giammichele, and R. Ricci, "Eulerian–Lagrangian modeling of cough droplets irradiated by ultraviolet–C light in relation to SARS-CoV-2 transmission", Physics of Fluids 33, 031905(2021).

[4] C. T. Crowe, M. P. Sharma, and D. E. Stock, "The particle-source-in cell (PSI-Cell) model for gas-droplet flows," J. Fluids Eng. 99, 325–332 (1977).
[5] F. R. Menter, "Two-equation eddy-viscosity turbulence models for engineering applications," AIAA J. 32, 1598–1695 (1994).
[6] G. Busco, S. R. Yang, J. Seo, and Y. A. Hassan, "Sneezing and asymptomatic virus transmission," Phys. Fluids 32, 073309 (2020).
[7] M. Abuhegazy, K. Talaat, O. Anderoglu, and S. V. Poroseva, "Numerical investigation of aerosol transport in a classroom with relevance to COVID-19," Phys. Fluids 32, 103311 (2020).
[8] A. Putnam, "Integratable form of droplet drag coefficient," ARS J. 31, 1467–1468 (1961).
[9] W. E. Ranz and W. R. Marshall, "Evaporation from drops," Chem. Eng. Prog. 48, 141–146 (1952).
[10] Woo, X.Y., Tan, R.B.H., Chow, P.S., Braatz, R.D., 2006. Simulation of mixing effects in antisolvent crystallization using a coupled CFD-PDF-PBE approach. Cryst. Growth Des. 6, 1291–1303.
[11] T. Dbouk and D. Drikakis, "On coughing and airborne droplet transmission to humans," Phys. Fluids 32, 053310 (2020).
[12] B. E. Scharfman, A. H. Techet, J. W. M. Bush, and L. Bourouiba, "Visualization of sneeze ejecta: Steps of fluid fragmentation leading to respiratory droplets," Exp. Fluids 57, 24 (2016).
[13] X. Xie, Y. Li, H. Sun, and L. Liu, "Exhaled droplets due to talking and coughing," J. R. Soc. Interface 6, S703 (2012).
[14] Rosti, M.E., Olivieri, S., Cavaiola, M. *et al.* Fluid dynamics of COVID-19 airborne infection suggests urgent data for a scientific design of social distancing. *Sci Rep* **10,** 22426 (2020).
[15] **S. J. Ruuth** and **R. J. Spiteri.** High-Order Strong-Stability-Preserving Runge-Kutta Methods with Downwind-Biased Spatial Discretizations. SIAM J. Numer. Anal., 42(3), 974–996, 2004.
[16] J. Dedsarnau, H. Derluyn, J. Carmeliet, D. Bonn, N. Shahidzedeh. Metastability Limit for the Nucleation of NaCl Crystals in Confinement *J. Phys. Chem. Lett.* 2014, 5, 5, 890–895
[17] A. Naillon, P. Joseph, M. Prat, Sodium chloride precipitation reaction coefficient from crystallization experiment in a microfluidic device. *J. Cryst. Growth, 463,* 201-210.
crystallization experiment in a microfluidic device, Journal of Crystal Growth, Vol. 463, 2017.

# BRINGING AI PIPELINES ONTO CLOUD-HPC: SETTING A BASELINE FOR ACCURACY OF COVID-19 AI DIAGNOSIS

Iacopo Colonnelli[1*], Barbara Cantalupo[1], Concetto Spampinato[2], Matteo Pennisi[2], Marco Aldinucci[1]

[1]*University of Torino, Computer Science Dept., Corso Svizzera 185, 10149, Torino, Italy*

[2]*University of Catania, Electrical Engineering Dept., Viale Andrea Doria 6, 95125, Catania, Italy*

**ABSTRACT.** HPC is an enabling platform for AI. The introduction of AI workloads in the HPC applications basket has non-trivial consequences both on the way of designing AI applications and on the way of providing HPC computing. This is the leitmotif of the convergence between HPC and AI. The formalized definition of AI pipelines is one of the milestones of HPC-AI convergence. If well conducted, it allows, on the one hand, to obtain portable and scalable applications. On the other hand, it is crucial for the reproducibility of scientific pipelines. In this work, we advocate the StreamFlow Workflow Management System as a crucial ingredient to define a parametric pipeline, called "CLAIRE COVID-19 Universal Pipeline", which is able to explore the optimization space of methods to classify COVID-19 lung lesions from CT scans, compare them for accuracy, and therefore set a performance baseline. The universal pipeline automatizes the training of many different Deep Neural Networks (DNNs) and many different hyperparameters. It, therefore, requires a massive computing power, which is found in traditional HPC infrastructure thanks to the portability-by-design of pipelines designed with StreamFlow. Using the universal pipeline, we identified a DNN reaching over 90% accuracy in detecting COVID-19 lesions in CT scans.

[*] Corresponding author. E-mail: *iacopo.colonnelli@unito.it*

## 1 Introduction

The ability of AI-related techniques to transform raw data into valuable knowledge is growing at a breakneck pace. Among these techniques, Deep Learning (DL) has benefited from crucial results in Machine Learning (ML) theory and the large availability of data. The accuracy of the process is strictly related to the quality and size of available datasets. Usually, more data means more accurate predictions, but also more computing power needed to train the model.

In particular, in the last decade, Deep Neural Networks (DNNs) became larger and larger, and nowadays, a reasonably sized DL workload cannot prescind from the availability of heterogeneous (GPU-equipped) computing resources. For this reason, High-Performance Computing (HPC) is undoubtedly an enabling platform for AI, and, in turn, AI enables success in scientific challenges where traditional HPC techniques have failed. For their part, supercomputers are shifting more and more to heterogeneous hardware, both because of their better energy efficiency and to satisfy the ever-increasing need for GPU-enabled workloads pushed by DL.

Despite this potential, supercomputers are rarely used for standard AI workloads. This is mainly due to technical barriers, i.e., user-unfriendly SSH-based remote shells and queue-based job submission mechanisms, which prevent AI researchers without a strong computer science background from effectively unlocking their computing power. In practice, HPC centres are not designed for general

purpose applications. Only scalable and computationally demanding programs can effectively benefit from the massive amount of processing elements and the low-latency network interconnections that characterize HPC facilities, justifying the high development cost of HPC-enabled applications. Moreover, some seemingly trivial features are not supported by HPC facilities, e.g., exposing a public web interface for data visualization in an air-gapped worker node.

On the other hand, the technical barriers to Cloud-based infrastructures lowered substantially with the advent of the *-as-a-Service model. Alas, the multiple layers of virtualization that characterize modern cloud architectures introduce significant processing overheads and make it impossible to apply adaptive fine-tuning techniques based upon the underlying hardware technologies, making them incompatible with performance-critical HPC applications.

In this work, we advocate the combination of two distinct approaches as an effective way to lower the technical barriers of HPC facilities for AI researchers [1]:

- A *cluster-as-accelerator* design pattern, in which cluster nodes act as processing elements of user-defined tasks sent by a Cloud-based, general-purpose host executor. This paradigm can be used to offload computation to HPC facilities in a more intuitive way, as it mimics the GPGPU paradigm used by DL applications in a GPU-equipped machine;
- *Hybrid workflows*, i.e. workflows whose steps can be scheduled on independent and potentially not intercommunicating execution environments, as a programming paradigm to express this design pattern while ensuring portability and reproducibility of complex AI workloads.

In the evaluation part, we apply these two principles to a real AI application, i.e., a DNN training pipeline for COVID19 diagnosis from Computed Tomography (CT) scan images. The StreamFlow [2] Workflow Management System (WMS), which supports hybrid Cloud-HPC workflows as first-class citizens, is used as the underlying runtime system.

## 2 The StreamFlow toolkit

A workflow is commonly represented as an acyclic digraph $G = (N, E)$, where nodes refer to different portions of a complex program and edges encode *dependency relations* between nodes. In this representation, a direct edge connecting a node $m$ to a node $n$ means that $n$ must wait for $m$ to complete before starting its computation.

The workflow abstraction has already been explored for offloading computation to HPC facilities transparently [3]. Many of the existing WMSs come with a diverse set of *connectors*, some of them addressing Cloud environments and some others more HPC-oriented. Nevertheless, a far smaller percentage can deal with *hybrid workflows*, offering the possibility to seamlessly assign each portion of a complex application to the computing infrastructure that best suits its requirements.

Hybrid workflows can strongly reduce the necessary tradeoffs in relying on such high-level abstraction, both in terms of performance and costs. Indeed, complex applications usually alternate computation-intensive and highly parallelizable steps with sequential or non-compute-bound operations. When scheduling such applications to an HPC centre, only a subset of steps will effectively take advantage of all the available computing power, resulting in a low cost-benefit ratio.

Moreover, HPC facilities do not support some common operations that are instead trivial on Cloud-based infrastructures, such as exposing web interfaces for data visualization.

The StreamFlow toolkit[2] [2], whose logical stack is depicted in Fig.1, has been specifically developed to orchestrate hybrid workflows on top of heterogeneous and geographically distributed architectures.

Written in Python 3, it supports the CWL coordination standard [4] for expressing workflow models through a declarative JSON or YAML syntax. The translation of these declarative semantics into an

[2] https://streamflow.di.unito.it/

executable workflow model is delegated to `cwltool`, the CWL reference implementation. The StreamFlow runtime layer is then able to efficiently execute such a model by translating it into a dataflow graph, identifying independent steps and running them in parallel whenever possible.

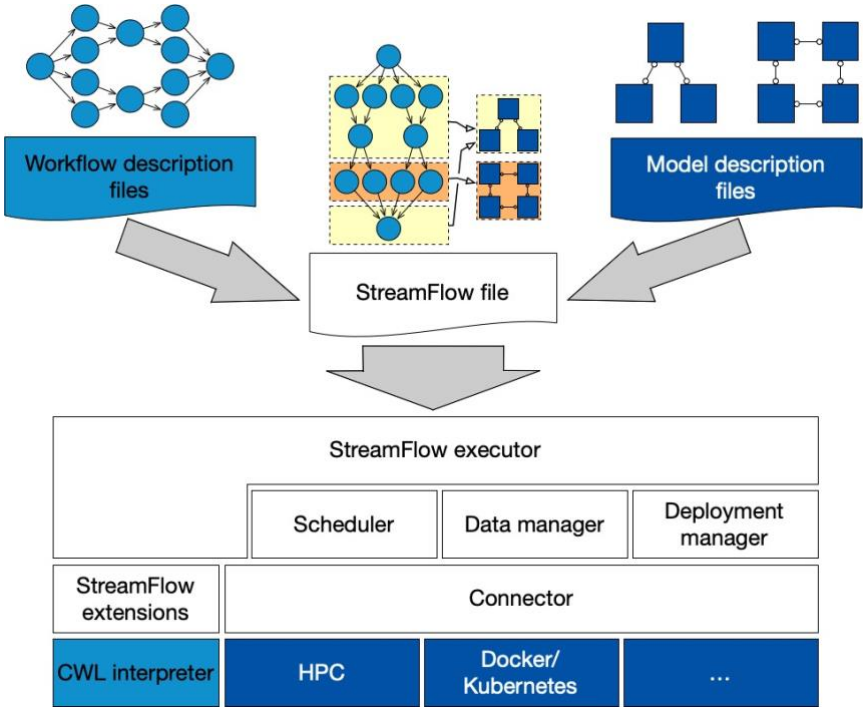



**Fig.1**: The StreamFlow toolkit's logical stack.

Alongside, one or more execution environments can be described in well-known external formats, e.g., Helm charts for Kubernetes deployments or Slurm files for HPC workloads. A `streamflow.yml` file constitutes the entry point of a StreamFlow run, relating each workflow step with the best suitable execution environment. This feature actually plugs the hybrid layer in the workflow design process.
Another distinctive feature of the StreamFlow WMS is the possibility to manage complex, multi-agent execution environments, ensuring the *co-allocation* of multiple heterogeneous processing elements to execute a single workflow step. The same interface can then be used to describe a diverse ecosystem of distributed applications, ranging from MPI clusters running on HPC facilities to microservices architectures deployed on Kubernetes. To provide enough flexibility, StreamFlow adopts a three-layered hierarchical representation of execution environments:

- A `model` is an entire multi-agent infrastructure and constitutes the *unit of deployment*, i.e., all its components are always co-allocated when executing a step;
- A `service` is a single agent in a model and constitutes the *unit of binding*, i.e., each step of a workflow can be offloaded to a single service for execution;
- A `resource` is a single instance of a potentially replicated service and constitutes the *unit of scheduling*, i.e., each step of a workflow is offloaded to a configurable number of service resources to be processed.

All communications and data transfer operations are started and managed by the StreamFlow controller, removing the need for bidirectional channels between the management infrastructure and the target resources and allowing tasks to be offloaded to HPC infrastructures with air-gapped worker nodes. Moreover, StreamFlow does not need any specific package or library to be installed on the target resources, other than the software dependencies required by the host application. As a consequence, virtually any target infrastructure reachable by a practitioner can serve as a target model, as long as a compatible connector implementation is available.

# 3 The CLAIRE COVID-19 universal pipeline

To demonstrate how StreamFlow can help bridge HPC and AI workloads, enabling reproducibility and portability across different platforms, we present the COVID-19 universal pipeline, developed by the Confederation of Laboratories for Artificial Intelligence Research in Europe (CLAIRE)[3] task force on AI & COVID-19 during the first COVID-19 outbreak. The group, composed of fifteen researchers in complementary disciplines (Radiomics, AI, and HPC) and led by Prof. Marco Aldinucci [5], investigated the diagnosis of COVID-19 pneumonia assisted by Artificial Intelligence (AI).
At the start of the pandemic, several studies outlined the effectiveness of chest radiology imaging for COVID-19 diagnosis. Even if X-Ray scans represent a cheaper and most effective solution for large scale screening, their low resolution led AI models to show lower accuracy than those obtained with CT scans. Therefore, the latter has become the gold standard for the investigation of lung diseases.
Several research groups worldwide began to develop DL models for the diagnosis of COVID-19, mainly in the form of deep Convolutional Neural Networks (CNN). As is especially the case in the medical field, reproducibility of the results was an important issue to address. Providing AI pipelines for COVID diagnosis with reproducible steps should not be an option to ensure the goodness of the results. This is particularly important when dealing with DL models, which are obscure by definition.
Furthermore, such a significant number of proposals was not accompanied by any baseline of the accuracy expectation. A comprehensive study of the proposed solutions highlighted that it would not be easy to evaluate the most promising approaches due to the adoption of different architectures, pipelines and datasets. Therefore, instead of proposing yet another hopefully better solution, the task force commitment was to organize the knowledge so far to consolidate and formalize all or the most state-of-the-art deep learning models to diagnose COVID-19.
The result of such commitment was the distillation of a reproducible workflow, the *CLAIRE COVID-19 universal pipeline* represented in Fig.2, capable of automating the comparison of the proposed DL models and supporting the definition of a baseline for any further evaluation.
The pipeline is basically designed by composing the main steps in a standard AI workflow. TC scan images are pre-processed to insulate the region of interest, the lungs in this case, and then used to train a classifier to recognize the typical lesions of interstitial pneumonia caused by COVID-19, specifically consolidation, crazy paving and grown glass.

The pipeline is composed of two main parts:

- A *data preparation* phase (yellow elements), comprising *pre-processing*, where standard techniques for cleaning the training images are applied, and *segmentation*, for extracting and selecting the region of interest for the next training. This step is performed just once for each dataset;
- The *core training* phase (blue elements), composed of standard AI steps such as *data augmentation*, to generate image variants, model *pre-training*, to generate an initial set of weights for initialization, and eventually *classification*, which labels each image with a class identified with a kind of lesion that is typical of the disease. The final steps are *cross-validation*, which increases the pipeline robustness by applying the training on different portions of the dataset, and *performance metrics*, obtained by collecting and comparing all the measures from the different pipelines.

The effectiveness of the DL approach depends on many parameters, e.g., the input dataset, the pre-processing steps, the chosen DL model, and the hyperparameters of the training algorithm, such as learning rate, weight decay, learning rate decay. It is worth noting that, in the universal pipeline, the

[3] https://claire-ai.org/

DNN itself, which is the model used for classification, is just one of the variables that can be set for training. Different variants of existing networks (such as AlexNet, ResNet, DenseNet) can be plugged into the pipeline, but any future network could be included in principle.
Finally, the pipeline is modelled as a workflow where every single step is a kind of container without any dependency on external libraries or vendor-specific technology. This choice enables portability on different platforms allowing to run the same pipeline in different platforms or even across different ones.

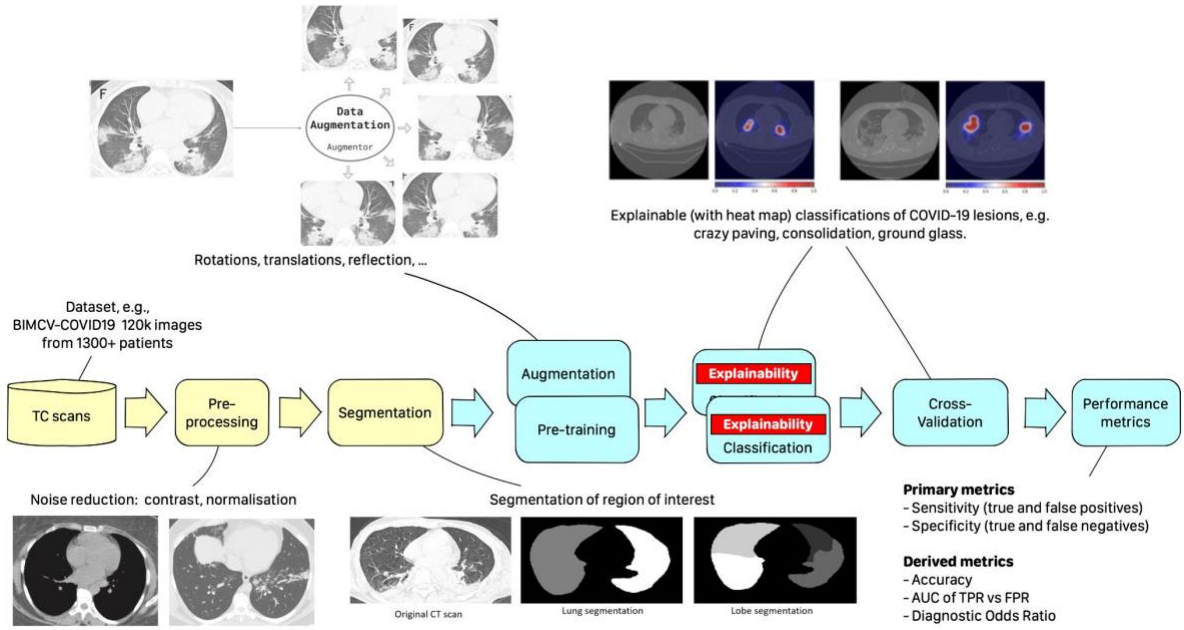



**Fig.2**: The CLAIRE COVID-19 universal pipeline.

The first set of experiments has been already completed comparing about 1% of the variants (11 of 990) and applying different segmentation types. Results show that the pipeline can generate models with excellent accuracy in classifying typical interstitial pneumonia lesions due to COVID-19, with sensitivity and specificity metrics over 90% in the best cases. More detailed information on the experiment's results are available in [6] and demonstrate that the proposed approach is able to carry out the same task with an accuracy that is at least on par with, or even higher than, human experts, thus showing the potential impact that these techniques may have in supporting physicians in decision making.

# 4 Experimental evaluation

On the move from the design to the implementation, the universal pipeline takes advantage of the StreamFlow technology. The workflow is defined using CWL, using traditional parallel computing operators (such as scatter, broadcast, gather, reduce) to explicitly annotate the parallelizable portions of the pipeline (Fig. 3). The search space is composed of all the combinations of network models, training hyperparameters, and one or more datasets. Each point in such space is a tuple, which constitutes the input for a single pipeline instance. As each pipeline instance is independent of each other, the overall execution is a typical embarrassingly parallel problem, whose parallelism can be exploited by distributing the input tuples to the work units through the “scatter” operator.

In turn, every single pipeline instance can further exploit parallelism, distributing different partitions of the dataset (generated by the cross-validation algorithm) to as many instances of the classification step through another scatter operator. In this setting, the initial weights of the DNN are broadcasted to all the classifiers of a specific pipeline instance. Performance metrics are then collected through a combination of reduce operators, first reducing internally in the single pipeline instance and then globally collecting results from all the pipelines. The baseline performance of the analyzed pipelines for COVID-19 diagnosis is finally obtained.

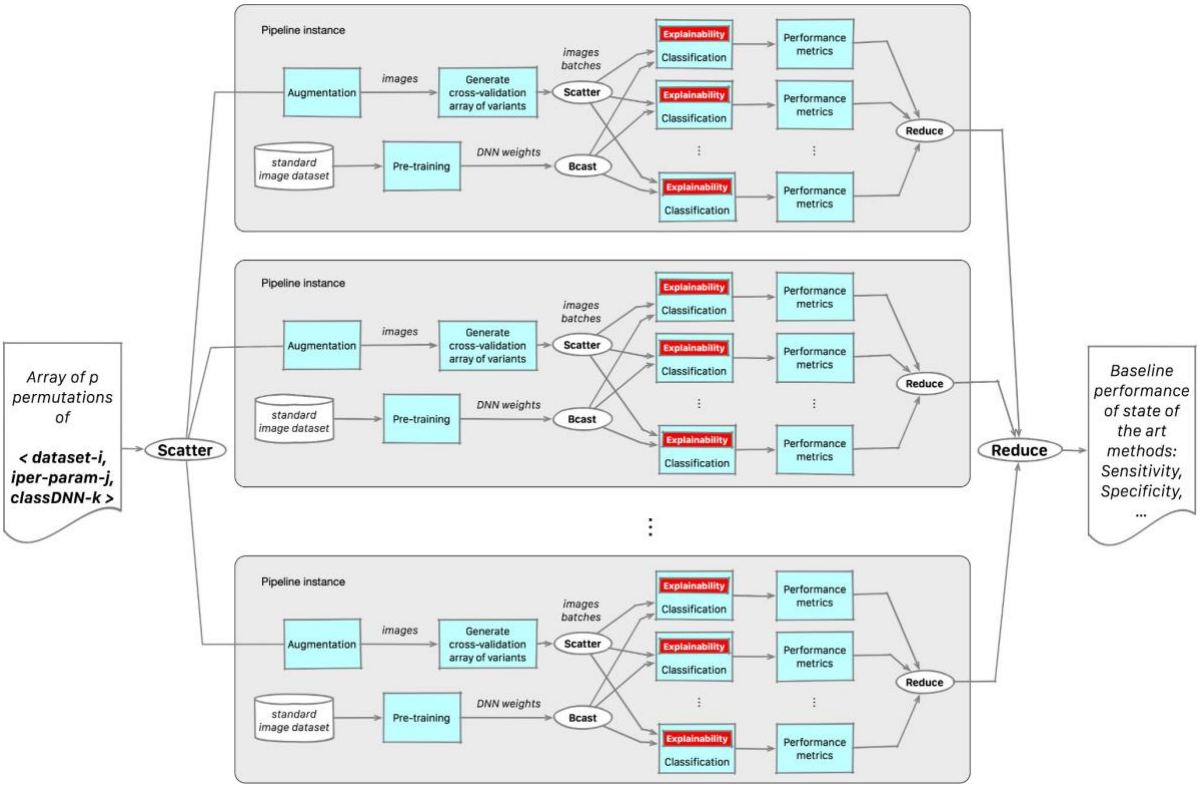



**Fig**.**3**: Unfolded implementation of the pipeline training components

For evaluation purposes, we ran the pipeline on the BIMCV-COVID19 dataset, with more than 120k images from 1300 patients. Assuming to train each pre-trained model for 20 epochs on such dataset, a single variant of the pipeline takes over 15 hours on a single NVidia V100 GPU, one of the most powerful accelerators in the market. Therefore, exploring all the 990 pipeline variants we have selected would take over two years. Fortunately, as we already pointed out, the universal pipeline has an embarrassingly parallel structure, and therefore using a supercomputer could reduce the execution time down to 15 hours in the best case (i.e., when 990 GPUs are available at the same time).
Post-training steps, as performance metrics extraction and comparison, are better suited for a Cloud infrastructure. Indeed, they do not require much computing power and can significantly benefit from web-based visualization tools. Given that, we used StreamFlow to model the pipeline as a hybrid workflow, offloading the training portions to HPC nodes and collecting the resulting networks on the host execution flow for visualization purposes. In particular, different portions of the training spectrum have been offloaded to three different heterogeneous architectures:

- The ENEA CRESCO D.A.V.I.D.E. cluster, composed of 45 nodes with 2 IBM POWER8 sockets, 256 GB of RAM and 4 NVidia P100-SMX2 GPUs each, that can be used on-demand through bare SSH connections;

- The CINECA MARCONI100 cluster, a SLURM-managed HPC facility with 32 IBM POWER9 cores, 256 GB of RAM and 4 NVidia V100 GPUs per node;
- The High-Performance Computing for Artificial Intelligence (HPC4AI) infrastructure at the University of Torino, a multi-tenant hybrid Cloud-HPC system with 80 cores and 4 GPUs per node (T4 or V100-SMX2) managed by OpenStack [7].

As an interface towards Cloud-HPC infrastructures, StreamFlow seamlessly manages data movements and remote step execution with each of these infrastructures, automatically transferring back the training results to the Cloud-based host node to perform post-training steps.

# 5 Conclusion and future work

Presenting the work on the universal COVID-19 pipeline, we demonstrated that AI can be an effective support for human activity, that HPC is crucial to perform complex tasks in a useful time, and last but not least, that the convergence of different platforms is the next big thing. In this scenario, the general adoption of hybrid infrastructures from the scientific communities can only be obtained by leveraging advanced software tools like StreamFlow to enable portability and reproducibility.

The computing power required by the largest AI training runs has been increasing exponentially with a 3.4-month doubling time in the last 10 years [8]. A need that today can be only matched by way of accelerated computing provided by specialized processors such as GPUs and TPUs. With this background, the next era of HPC will inevitably see a further increase of heterogeneous hardware, with general-purpose CPUs flanked by highly parallel co-processors as GPUs and special-purpose hardware as neuromorphic chips and quantum annealing. Even if each architecture comes with its peculiar programming paradigm for local computations, the accelerator pattern is becoming a de-facto standard for moving computation away from CPUs.

We believe that a sound and stable system software part is crucial for the mainstream industrial adoption of HPC, enabling technology to transform applications into easily usable services hence into innovation. While in scientific computing the modernization of HPC applications is a scientific desideratum required to boost industrial adoption, the shift toward the Cloud model of services is a must in AI. AI applications are already modern, and they will not step back.

We advocate StreamFlow as an intuitive programming paradigm to foster the design of portable and scalable AI pipelines and reduce technical barriers to HPC facilities for domain experts without a strong computer science background.

Such paradigms can be further extended and improved. From one side, support for specific hardware (e.g., quantum processors) can be added to the list of connectors offered by hybrid WMSs. Moreover, more intuitive and user-friendly technologies (e.g., Jupyter Notebook) can be augmented with hybrid workflow semantics to evolve them from prototyping technologies to production-ready toolchains. Both these challenges are essential parts of the StreamFlow roadmap, together with further applications in the domains of deep learning, bioinformatics and molecular dynamics simulations.

# Funding & Acknowledgement

We gratefully acknowledge the support of Francesco Iannone from ENEA and the CRESCO/ENEAGRID High Performance Computing infrastructure and its staff. This work has been partially supported by the DeepHealth project, which has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 825111, by the HPC4AI project funded by Regione Piemonte (POR FESR 2014-20 - INFRA-P). Access to CINECA resources has been possible thanks to the CINI-CLAIRE-ABD MoU 2020 supporting research on COVID-19.

We want to thank Emanuela Girardi and Gianluca Bontempi, who are coordinating the CLAIRE task force on COVID-19, for their support, and the group of volunteer researchers who contributed to the development of the CLAIRE COVID-19 universal pipeline, they are: Marco Calandri and Piero Fariselli (Radiomics & medical science, University of Torino, Italy); Marco Grangetto, Enzo Tartaglione (Digital image processing Lab, University of Torino, Italy); Simone Palazzo, Isaak Kavasidis (PeRCeiVe Lab, University of Catania, Italy); Bogdan Ionescu, Gabriel Constantin (Multimedia Lab @ CAMPUS Research Institute, University Politechnica of Bucharest, Romania); Miquel Perello Nieto (Computer Science, University of Bristol, UK); Inês Domingues (School of Sciences University of Porto, Portugal).

## References

[1] I. Colonnelli, B. Cantalupo, R. Esposito, M. Pennisi, C. Spampinato, and M. Aldinucci. HPC Application Cloudification: The StreamFlow Toolkit. *12th Workshop on Parallel Programming and Run-Time Management Techniques for Many-core Architectures and 10th Workshop on Design Tools and Architectures for Multicore Embedded Computing Platforms*, Dagstuhl, Germany, pp. 5:1–5:13, (2021). doi:10.4230/OASIcs.PARMA-DITAM.2021.5

[2] I. Colonnelli, B. Cantalupo, I. Merelli, and M. Aldinucci. StreamFlow: cross-breeding Cloud with HPC. *IEEE Transactions on Emerging Topics in Computing*, (2020). doi:10.1109/TETC.2020.3019202

[3] R.F. da Silva, R. Filgueira, I. Pietri, M. Jiang, R. Sakellariou, and E. Deelman. A characterization of workflow management systems for extreme-scale applications. *Future Generation Computer Systems*, Vol. 75, pages 228–238, October 2017. doi:10.1016/j.future.2017.02.026

[4] P. Amstutz, M. R. Crusoe, N. Tijanić, B. Chapman, J. Chilton, M. Heuer, A. Kartashov, J. Kern, D. Leehr, H. Ménager, M. Nedeljkovich, M. Scales, S. Soiland-Reyes, and L. Stojanovic. Common workflow language, v1.0, (2016). doi:10.6084/m9.figshare.3115156.v2

[5] High-performance computing and AI team up for COVID-19 diagnostic imaging. https://aihub.org/2021/01/12/high-performance-computing-and-ai-team-up-for-covid-19-diagnostic-imaging/, (2021). Accessed: 2021-06-30

[6] M. Pennisi, I. Kavasidis, C. Spampinato, V. Schinina, S. Palazzo, F.P. Salanitri, G. Bellitto, F. Rundo, M. Aldinucci, M. Cristofaro, P. Campioni, E. Pianura, F. Di Stefano, A. Petrone, F. Albarello, G. Ippolito, S. Cuzzocrea, S. Conoci. An Explainable AI System for Automated COVID-19 Assessment and Lesion Categorization from CT-scans. *Artificial Intelligence in Medicine*, (2021). doi:10.1016/j.artmed.2021.102114

[7] M. Aldinucci, S. Rabellino, M. Pironti, F. Spiga, P. Viviani, M. Drocco, M. Guerzoni, G. Boella, M. Mellia, P. Margara, I. Drago, R. Marturano, G. Marchetto, E. Piccolo, S. Bagnasco, S. Lusso, S. Vallero, G. Attardi, A. Barchiesi, A. Colla, and F. Galeazzi. HPC4AI, an AI-on-demand federated platform endeavour. *ACM Computing Frontiers*, Ischia, Italy, (2018). doi:10.1145/3203217.3205340

[8] OpenAI. AI and Compute. https://openai.com/blog/ai-and-compute/, (2018). Accessed: 2021-06-30

# MULTISCALE MODELING OF THE WILD-TYPE AND ALPHA VARIANT SARS-COV-2 SPIKE PROTEIN

Marco Lauricella [1*], Letizia Chiodo [2], Fabio Bonaccorso [3,4], Mihir Durve [3], Andrea Montessori [1], Adriano Tiribocchi [1], Alessandro Loppini [2,3], Simonetta Filippi [2], and Sauro Succi [3,1]

[4] *Consiglio Nazionale delle Ricerche, Istituto per le Applicazioni del Calcolo IAC-CNR, 00185, Rome Italy*

[2] *Campus Bio-Medico University, Engineering Department, 00128, Rome, Italy*

[3] *Istituto Italiano di Tecnologia, Center for Life Nano- & Neuro-Science@Sapienza – IIT, 00161, Rome, Italy*

[4] *Department of Physics and INFN, University of Rome "Tor Vergata", 00133 Rome, Italy*

**ABSTRACT.** Physiological solvent flows act in such a way as to promote collective motions of biological structures. In virus/host-cell interactions, the solvent flow may facilitate the virus adhesion on the target receptors and drive the hierarchy of multivalent adhesion mechanisms.
To elucidate functional interactions between flows and molecules, we couple the all-atom atomistic molecular dynamics (for proteins) with the computational Lattice Boltzmann fluid dynamics (for solvent flows).
Preliminary results are presented for SARS-CoV-2 viral spike glycoprotein S in implicit solvent, as well as for its Alpha variant. Our multiscale simulations are performed with the LAMMPS classical molecular dynamics code, used within a customized installation on Cresco6 at ENEA.
The mesoscopic solvent description is critically compared to the all-atom solvent model, to quantify advantages and limitations of the multiscale description.

[*] Corresponding authors. E-mail: *marco.lauricella@cnr.it*; *l.chiodo@unicampus.it*

## 1 Introduction

In the last decades, computational tools have undergone spectacular methodological and technological progress, which can play a decisive role in fighting contagious diseases, by providing *in silico* simulations of biological molecules and drugs design. Concurrently, computing infrastructures endure great technological progress, largely fueled by the computing power of the graphics processing units (GPUs). The confluence of such major advances spawns opportunities to develop a new multiscale modelling tool by coupling a mesoscale solvent representation to the molecular dynamics to obtain an efficient computational biomedicine tool able to boost the simulation power and the understanding of the biological mechanisms at the atomistic level.
Nowadays, the molecular dynamics (MD) method has shown its massive potential in describing at the atomistic level the biological mechanisms underlying the activities of several proteins. Remarkable examples in computational biomedicine are the recent simulations of an entire cell organelle, a photosynthetic chromatophore vesicle from a purple bacterium [1] or the study of the N-Methyl-D-

[4] Corresponding authors. E-mail: marco.lauricella@cnr.it; l.chiodo@unicampus.it.

Aspartate (NMDA) neuroreceptor by the DE Shaw research group [2]. As of today, an enormous scientific effort has been spent to investigate in-silico the molecular behaviour of SARS-CoV-2 proteins, both for drug repurposing [3] and for antibody design. Standard MD simulations have been used, for example, to estimate binding free energies of spike in interaction with the human angiotensin-converting enzyme 2 (ACE2) receptor [4,5,6,7] alongside with their interaction scores [8].

Nonetheless, the long time scales of the movements associated with the allosteric and functional response of biological mechanisms normally lie in several microseconds, beyond the actual high-performance computational limits to obtain a statistically meaningful description [9,10], even by exploiting optimised codes for GPUs clusters [11]. Thus, in the last three decades, the development of coarse-grained models has shown great scope in overcoming these limits. The coarse-grained strategy usually aims at reducing the details of the protein structures alongside their aqueous solvent. Such a reduction shall be made with particular care to preserve the detailed description, where necessary to appropriately describe the protein structure and function.

The coexistence of different time- and length- scales necessarily requires the use of a multiscale approach in new computational biomedicine tools, that shall be pursued with prompted concern in the actual pandemic scenario [12**,**13].

The present work exploits a multiscale description based on the Lattice Boltzmann (LB) method for the solvent fluid of the aqueous combined with all-atom molecular dynamics (MD) description for the protein structures. Several multiscale approaches to coupling LB/MD are already available in the literature. Here, the coupling of the LB velocity field with MD objects is implemented via a Stokes friction term in the overlay region of the two descriptions to realise the multiscale description [14**].**

As a test case, we use the SARS-CoV-2 spike protein S, a heavily glycosylated protein anchored in the viral membrane. It is constituted by three chains, each one made of an identical primary sequence of more than 1200 amino acids of which 1146 form the extracellular domain. Each chain of the trimer is composed of two fragments: the receptor-binding fragment S1, containing the receptor-binding domain (RBD), interacting with ACE2, and the fusion fragment S2 [15]. The S protein is cleaved by a furin-like protease at residue 686 into the S1 and S2 fragments [16], initiating the membrane fusion process. We also study the Variant of Concern 202012/01 (lineage B.1.1.7), also commonly referred to as Alpha variant (α-Spike).

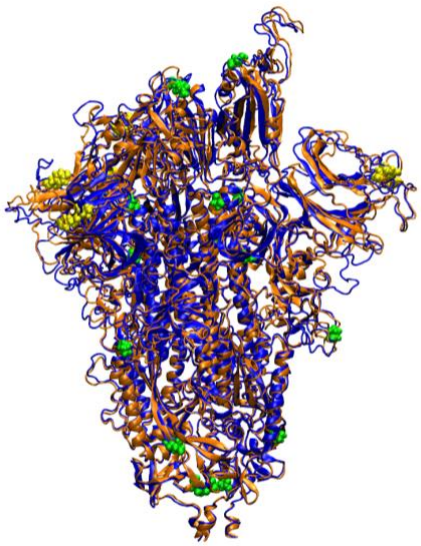


**Fig**.**1**: Secondary structure of the spike protein, wild type (blue) and Alpha variant (orange) as obtained from experimental pdb (6vsb.pdb) and homology modelling, respectively. In yellow and green, deletions and mutations of the wild type giving the Alpha variant are highlighted.

Equilibrium properties, such as rigidity and elasticity of specific sites, are pivotal for binding and other functional activities of the viral protein [15,17], therefore they must be properly described within the multiscale model. We compare equilibrium state properties from LB/MD data obtained by all-atom molecular dynamics, to assess the quality and efficiency of the multiscale description. The key role of water molecules for protein structure and function is highlighted [18].
In perspective, our multiscale approach could permit us to overcome the statistical sampling limitation affecting current explicit solvent atomistic description, in particular providing a unique tool to investigate large conformational changes and dynamics solvent flow effects.

# 2 Methodology

### 2.1 Simulated systems

Based on available cryo-EM data [19,20], we built the two models, hereafter called Spike and α-Spike. By relying on the 6vsb.pdb structure [10], we added the missing loops in the receptor-binding domain (RBD) as obtained from the 6m17.pdb structure [20]. The glycans from the 6vsb.pdb structure are used in these simulations. The α-Spike has been modelled via I-Tasser [21], using the wild-type as template, by including in the sequence [22, 23] three deletions: ΔH69/ΔV70 and ΔY144, and six mutations; N501Y, A570D, P681H, T716I, S982A, and D1118H (Fig. 1).
The protonation states have been calculated via Playmolecule webserver [24], based on PROPKA 3.1 [25] to determine protein pKa values, and on PDBTOPQR 2.1 [26] to optimize the protein for favourable hydrogen bonding.
The cell of the LB/MD systems is a cubic box of 19.2 nm side length, surrounding the all-atom glycosylated-protein (53k atoms). The same box size is used for the all-atom simulations (676k atoms). The cell is built and neutralized (100 nM solution) via the Solvate and Ionize plugins included in Visual Molecular Dynamics (VMD) [27].

### 2.2 Coupled Lattice-Boltzmann and Molecular Dynamics (LB/MD)

LB/MD simulations have been performed with LAMMPS (stable release 3 March 2020) [28] on the CRESCO-6 cluster based on Intel(R) Xeon(R) Platinum 8160 with 24 cores and two CPUs per computing node.
The two systems (wild-type and mutant) endured an initial 10000 steps of conjugate gradient minimization and were afterwards equilibrated in an NVT ensemble, with the temperature increased to 310 K over 2.0 nanoseconds of MD simulations. Hence, both systems were evolved in time for 500 nanoseconds (NVT, 310 K). The direct summation method was exploited to assess the Coulomb interactions. In particular, the additional screening of the solvent was modelled by a Coulomb correction for implicit solvent interactions which exploits a distance-dependent dielectric permittivity, scaling with an additional 1/r term included in the Coulomb formula. The cut-off of the intermolecular interactions was set to 7 Å corresponding to the Bjerrum length in water [29]. The CHARMM36 force field [30] has been used to model inter- and intra-molecular interactions, including the glycan and N-linked glycan bond descriptions.
The aqueous solvent is described by a specific mesoscale technique, known as the Lattice Boltzmann (LB) method, namely a minimal lattice version of the Boltzmann equation. The LB approach allows modelling the dynamic behaviour of fluid flows without directly solving the Navier-Stokes equations of continuum fluid mechanics. In this framework, the solvent is treated via a fictitious ensemble of particles, whose motion and interactions are confined to a regular space-time lattice. The dramatic reduction of the degrees of freedom associated with the velocity space is the main advantage of the LB

approach. Thus, the solvent is described in terms of probability to find a certain quantity of solvent particle at position $\vec{r}$ and time $t$ moving with velocity $\vec{c_i}$ along a possible grid direction. In the LB approach, the particle collisions are represented through a relaxation to the local equilibrium. Here, we rely on the simplest form of the collision operator that is the celebrated Bhatnagar-Gross-Krook operator, where the operator is a simple single-time relaxation term [31].
The standard LB scheme in single-relaxation time (BGK) form reads as follows:

$$f_i(\vec{r} + \vec{c_i}\Delta t, t + \Delta t) = f_i(\vec{r}, t) - \omega\left[f_i(\vec{r}, t) - f^{eq}{}_i(\vec{r}, t)\right] + S(\vec{r}, t) \quad (1)$$

where $f$ is the discrete Boltzmann distribution associated with the discrete velocity $\vec{c_i}$, i = 0, b running over the discrete lattice, in our case 19-speed lattices, commonly denoted D3Q19. In Eq. 1, the relaxation frequency $\omega$ is used to set the kinematic viscosity $\nu$ of the fluid by the relation $\omega = 2/(6\nu + 1)$, while $f^{eq}{}_i$ is the lattice local equilibrium, basically the local Maxwell-Boltzmann distribution truncated to the second order in the Mach number. mass density and mass flow are obtained in terms of moments of the distribution functions:

$$\rho = m\sum_{i=0}^{b} f_i(\vec{r}, t) \qquad (2)$$

$$\rho\vec{u} = m\sum_{i=0}^{b} f_i(\vec{r}, t)\vec{c_i} \quad (3)$$

where $m$ denotes a scaling mass factor set to obtain the correct water density of 993.4 kg/m$^3$ at 1 atm and 310 K. The fluid and particles are coupled as follows. The effect of the particle on the surrounding fluid is modelled via a friction force term, $\overrightarrow{F_{nj}} = \gamma\left(\vec{v}_n - \vec{u}_j\right)$, where $\vec{v}_n$ denotes the particle velocity and $\vec{u}_j$ the fluid velocity at the particle position obtained by a linear interpolation over the nearest eight lattice points. The coupling force is then added to Eq. 1 by the extra force term $S(\vec{r}, t)$. Hence, an equal and opposite force is applied to the particle to model the counterpart of the coupling term (from the fluid to the particle). Following the literature [32], the friction coefficient $\gamma$ is taken equal to $0.1\ fs^{-1}$, while the kinematic viscosity was set equal to 0.07 Å$^2$ /*fs* corresponding to the water kinematic viscosity at 310 K. The LB scheme is evolved in time step by step with the MD integration scheme, with a timestep equal to 2.0 femtoseconds.

**2.3 All-atom molecular dynamics (AA-MD)**

AA-MD simulations have been performed with GROMACS-2020.6 [33] on the Cineca Marconi100 cluster, based on IBM Power9 architecture and Volta NVIDIA GPUs. Both Spike and α-Spike endured an initial 10000 steps of conjugate gradient minimization. Hence, the systems were equilibrated up to 310 K over 2.0 nanoseconds, as in the LB/MD simulations. The MD simulations last 400 nanoseconds (NPT, 310 K, 1 atm). The simulation timestep is 1.6 femtoseconds. Periodic boundary conditions were used, with particle-mesh Ewald long-range electrostatics, using a grid spacing of 1.5 Å along with a fourth-order B-spline charge interpolation scheme. Both the Coulomb and Lennard-Jones interactions use a cut-off of 12 Å with a force switching function acting from 10 Å [34]. The CHARMM36 force field [30] has been used to model inter- and intra-molecular interactions, including the glycan and N-linked glycan bond descriptions. The water is described via the TIP3P water model as implemented in CHARMM [35] which specifies a 3-site rigid water molecule with Lennard-Jones parameters and charges assigned to each of the 3 atoms.

## 3 Results and Discussion

Using the AA-MD and LB-MD trajectories for Spike and α-Spike, we calculated the Root Mean Square Deviation (RMSD) and the Root Mean Square Fluctuations (RMSFs) (for chain A, that has the RBD in open conformation). We also performed the principal component analysis (PCA) of the internal motion involving the linked S1 and S2 fragments and evaluated the cross-correlation matrix (CCM) between pairs of Cα. Comparison with experimental or MD data, when available, is reported.

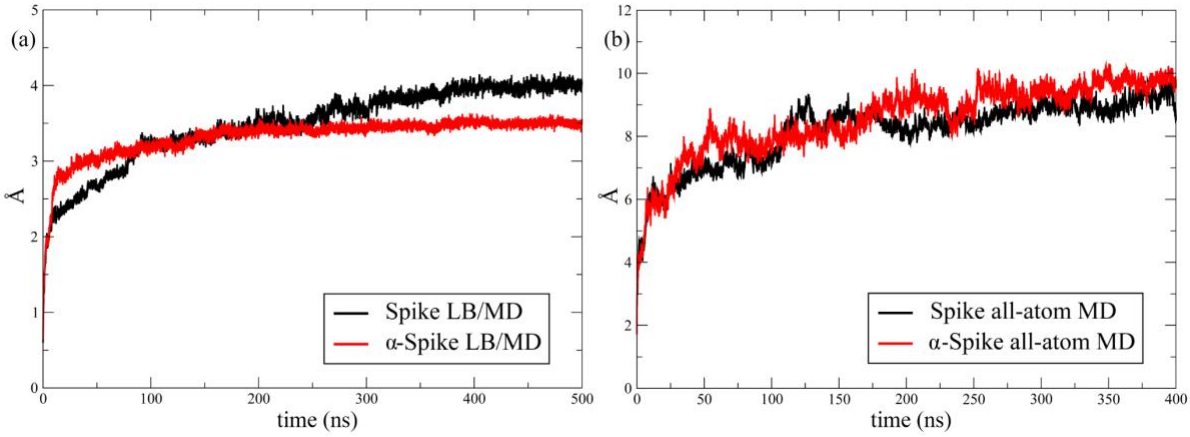



**Fig.2**: Root Mean Square Deviation of the Cα atoms, for LB/MD (left panel) and all-atom MD (right panel) simulations, for the 400 ns long trajectories. Slightly differences are observed between wild and mutated. The intensity decrease for the LB/MD case is due to the low mobility of the protein.

The RMSD (Fig.2) shows that the equilibration of the AA/MD is quite slow, with large changes from the initial models as obtained from Cryo-EM data and homology modelling. The relaxation is much faster in LB/MD, and changes are smaller with respect to starting structure. Also, RMSD fluctuations are larger in AA/MD. Part of the observed differences are ascribable to the intrinsic rigidity of the LB/MD model, but we cannot exclude a role of the starting Cryo-EM structure.

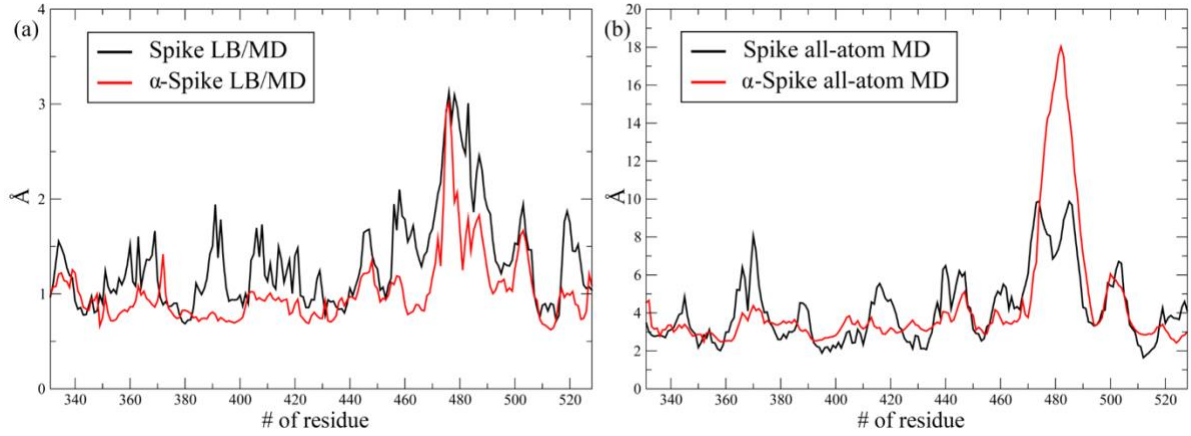



**Fig.3**: Root Mean Square Fluctuations (Cα atoms, RBD) for LB/MD (left panel) and AA/MD (right panel) simulations [15]. Slightly differences are observed between wild and mutated. The significant intensity decrease in the LB/MD is due to the low protein mobility. This result, expected on qualitative grounds, to the best of our knowledge was never inspected on quantitative grounds before.

The protein rigidity in the LB/MD description is confirmed by the RMSFs (Fig.3). However, despite the intensity differences, the LB/MD is able to obtain a qualitative agreement with AA/MD, with fluctuations observed in the same regions. We highlight that the RBD (res 331-528) is less flexible in

the α-Spike than in the wt-Spike, both in LB/MD and AA/MD simulations, apart from the recognition loop L3 (at residue 480). L3 is a loop of interest because its rigidity has been related to a stable interaction with ACE2 in an MD study [17].
The PCA (Figs. 4-5) is also quite indicative, for the sake of methods' comparison. Focusing on Spike/α-Spike comparison, their description is quite similar in the framework of the same method. Obviously, due to the lower flexibility of the LB/MD protein, the size of the modes is almost 3-4 times larger in AA/MD. Moreover, also the weight of the various modes is different, meaning that different regions of the protein show different flexibility changes when the solvent is described all-atom or at the mesoscale.

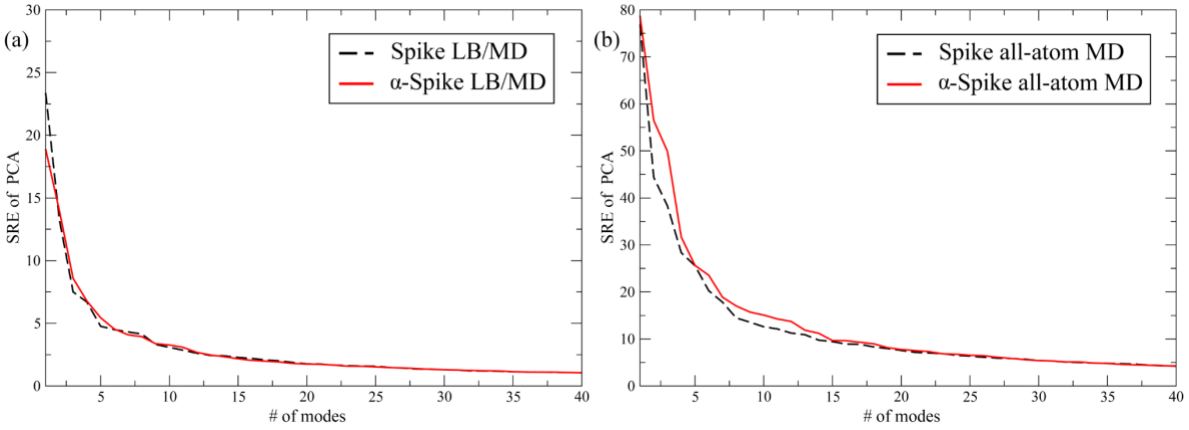



**Fig.4**: Square root of the eigenvalues (SRE) of the largest 40 modes from the PCA, for chain A of LB/MD (left panel) and AA/MD (right panel) simulations. Slightly differences are observed between wild and mutated. A strong intensity decrease is observed for the LB/MD cases, corresponding to the reduced mobility of the protein.

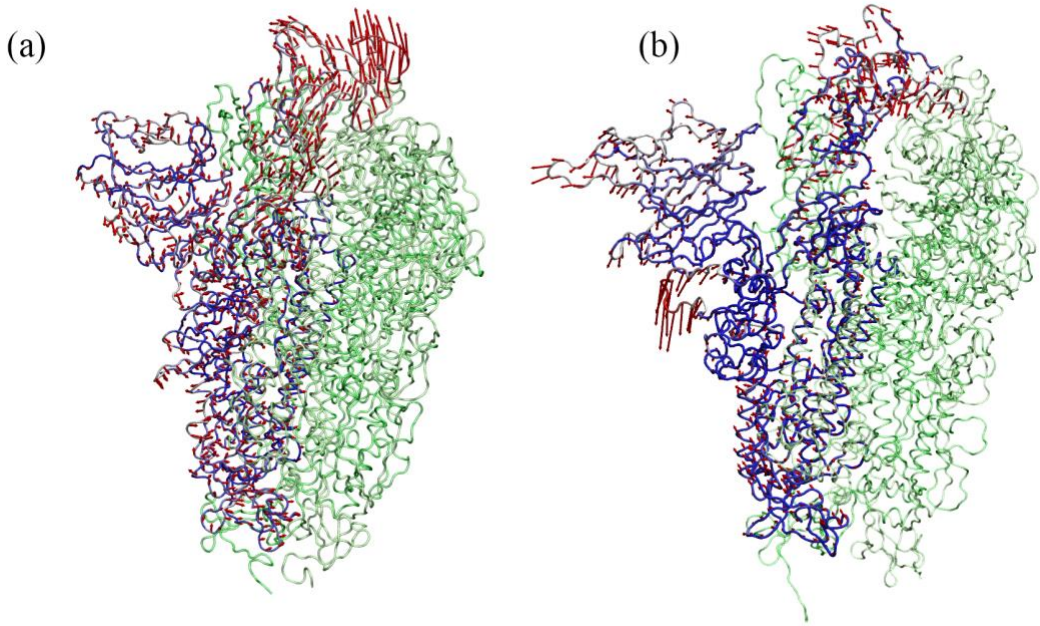



**Fig.5**: The main variation mode in principal component analysis (PCA) for chain A of the wild type Spike protein in the LB/MD simulation (a) and in the AA/MD simulation (b).

Finally, the map of the CC (Fig. 6) shows something comforting, because, overall, correlations between residues are reasonably described by LB/MD with respect to AA/MD. This result is due to the fact that

intramolecular local interactions, not mediated by solvent, are kept and well described in the LB/MD method. To note that, in both methods, the α-Spike presents higher correlation values with respect to the wild type, denoting that mutations indeed affect the protein structure and interaction propensity, and that these fundamental features are kept in the mesoscale description.

Overall, from this preliminary investigation, two results emerge a) the *in nuce* potential of the multiscale approach to describe large systems, with the possibility of including flow motion effects in the protein dynamics; b) the needing for a more refined coupling scheme at the protein/solvent interface, possibly a scheme where the solvent is explicitly described only close to the protein interface, providing a reliable description of the interaction among water molecules and amino acids, a pivotal interaction in determining both equilibrium and dynamic properties of biological molecules.

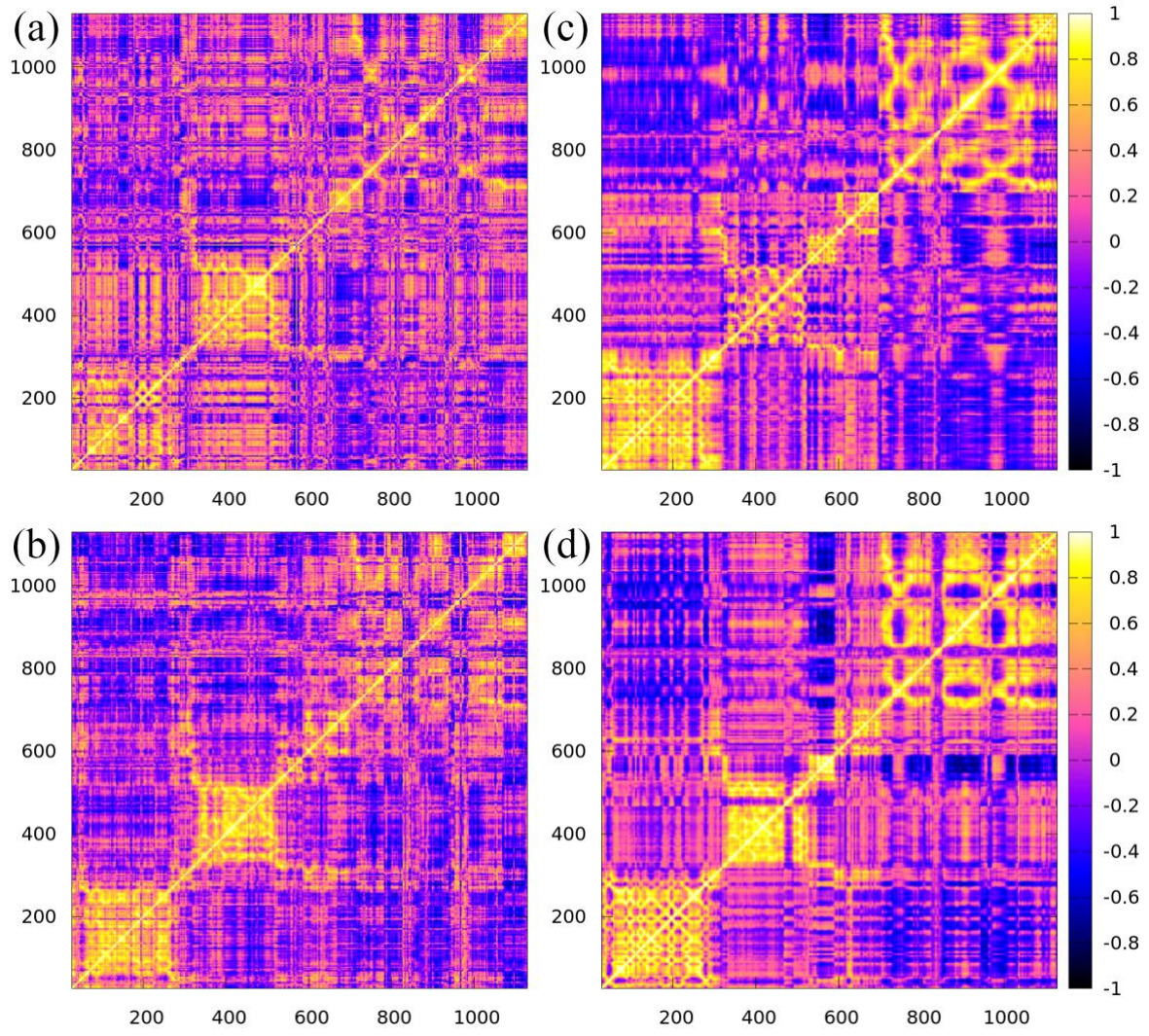



**Fig.6**: Cross-correlation map of the cross-correlation matrix for chain A of Spike and α-Spike in LB/MD simulations (panels a, b), and in AA/MD simulation (panels c, d). All cases show a significant correlation block associated with residues from the RBD (residues 331-528). The higher mobility of the protein in the all-atom simulations is reflected by the richer structure of the CCM (panels c,d), yet the qualitative structure of the patterns is preserved in LB/MD.

# 4 Conclusions

To conclude, on one side our simulations demonstrate and quantitatively estimate the pivotal role of the explicit water molecules treatment to obtain a statistically reliable characterization of biological molecules. In this respect the use of the LB solvent, while computationally advantageous, does not deliver quantitatively accurate information. On the other hand, our study highlights that many structural features, important in biological activities, are preserved in the LB/MD mesoscale solvent description, as shown for instance by the CC map and the RMSFs. The RMSFs, despite being quantitatively different, show similar qualitative behaviour when comparing Spike and α-Spike. In particular, we observe an increased rigidity of the RBD in the α-Spike compared to the wild type Spike. The main correlations inside the sub-domains are preserved, and the large correlation of the RBD block in the CC map for the α-Spike reinforces the information on the stability of the mutated structure. Indeed, the capability of the α-Spike RBD to maintain a rigid structure can be related, from the statistical point of view, with its remarkable human-to-human transmissibility [17], since the smaller fluctuations are limiting the sampling of possible RBD configuration basins to the subregion where the structure more efficiently binds to ACE2. This behaviour is somehow similar to what is observed for RBD rigidity of SARS-CoV-2 with respect to the Sars virus [17]. Moreover, the same correlation trends over Spike and α-Spike structures are found in the LB/MD simulations, showing the capability of the multiscale approach to preserve essential information even if at a lower description level of the solvent.

Future work should be directed to the development of a new class of LB models capable of supporting larger fluctuations than presently possible. A promising direction along this line is the resort to higher-order sets of discrete velocities.

## Acknowledgements

M. L., F. B., M. D., A. M., A. T. and S. S. acknowledge funding from the European Research Council under the European Union's Horizon 2020 Framework Programme (No. FP/2014-2020) ERC Grant Agreement No.739964 (COPMAT). L. C., A. L. and S. F. acknowledge the support of the International Center for Relativistic Astrophysics Network (ICRANet) and of the Italian National Group for Mathematical Physics (GNFM-INdAM). All the Authors gratefully acknowledge ENEA for the availability of high-performance computing resources and support on the HPC CRESCO facility used in the LB/MD simulations. Also, we acknowledge CINECA Project ABRISP under the ISCRA initiative, for the availability of high-performance computing resources and support used in the all-atom MD simulations.

## References

[1] J. D. Rochaix. (2019). Dynamic Modeling of a 100-Million-Atom Organelle at the Source of Life. Cell, 179 (5), 1012-1014.
[2] X. Song, M. Ø. Jensen, V. Jogini, et al. (2018). Mechanism of NMDA receptor channel block by MK-801 and memantine. Nature, 556(7702), 515-519.
[3] https://www.exscalate4cov.eu/eu
[4] E. Taka, S. Z. Yilmaz, M. Golcuk, C. Kilinc, U. Aktas, A. Yildiz and M. Gur. (2020). Critical Interactions Between the SARS-CoV-2 Spike Glycoprotein and the Human ACE2 Receptor. bioRxiv.
[5] J. He, H. Tao, Y. Yan, S.-Y. Huang and Y. Xiao. (2020). Molecular Mechanism of Evolution and Human Infection with SARS-CoV-2. Viruses, 12, 428.

[6] V. Armijos-Jaramillo, J. Yeager, C. Muslin and Y. Perez-Castillo. (2020). SARS-CoV-2, an evolutionary perspective of interaction with human ACE2 reveals undiscovered amino acids necessary for complex stability. Evol. Appl., 13, 2168– 2178.
[7] J. Zou, J. Yin, L. Fang, M. Yang, T. Wang, W. Wu, M. A. Bellucci and P. Zhang. (2020). Computational Prediction of Mutational Effects on SARS-CoV-2 Binding by Relative Free Energy Calculations. J. Chem. Inf. Model. 60, 5794– 5802.
[8] E. S. Brielle, D. Schneidman-Duhovny and M. Linial. (2020). The SARS-CoV-2 exerts a distinctive strategy for interacting with the ACE2 human receptor. Viruses, 12, 497.
[9] A. Ali and R. Vijayan. (2020). Dynamics of the ACE2–SARS-CoV-2/SARS-CoV spike protein interface reveal unique mechanisms. Scientific reports, 10(1), 1-12.
[10] W. Zheng, H. Wen, G. J. Iacobucci and G. K. Popescu. (2017). Probing the structural dynamics of the NMDA receptor activation by coarse-grained modeling. Biophys. J., 112(12), 2589-2601.
[11] B. Lev, S. Murail, F. Poitevin, B. A. Cromer, M. Baaden, M. Delarue and T. W. Allen. (2017). String method solution of the gating pathways for a pentameric ligand-gated ion channel. PNAS, 114(21), pp. E4158-E4167.
[12] P. V. Coveney, A. Hoekstra, B. Rodriguez and M. Viceconti. (2021). Computational biomedicine. Part II: organs and systems.
[13] W. Edeling, H. Arabnejad, R. Sinclair, et al. (2021). The impact of uncertainty on predictions of the CovidSim epidemiological code. Nature Computational Science, 1(2), pp.128-135.
[14] M. Bernaschi, S. Melchionna and S. Succi. (2019). Mesoscopic simulations at the physics-chemistry-biology interface. Reviews of Modern Physics, 91(2), p.025004.
[15] Y. Cai, J. Zhang, T. Xiao, et al. (2020). Distinct conformational states of SARS-CoV-2 spike protein. Science, 369(6511), pp.1586-1592.
[16] B. J. Bosch, R. Van der Zee, C. A. De Haan and P. J. Rottier. (2003). The coronavirus spike protein is a class I virus fusion protein: structural and functional characterization of the fusion core complex. Journal of virology, 77(16), pp.8801-8811.
[17] A. Spinello, A. Saltalamacchia and A. Magistrato. (2020). Is the Rigidity of SARS-CoV-2 Spike Receptor-Binding Motif the Hallmark for Its Enhanced Infectivity? Insights from All-Atom Simulations. J. Phys. Chem. Lett. 2020, 11, 12, 4785–4790
[18] M. C., Bellissent-Funel, A. Hassanali, M. Havenith, et al. (2016). Water determines the structure and dynamics of proteins. Chemical reviews, 116(13), 7673-7697.
[19] D. Wrapp, N. Wang, K. S. Corbett, et al. (2020). Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation. Science, 367(6483), 1260-1263.
[20] R. Yan, Y. Zhang, Y. Li, L. Xia, Y. Guo, & Q. Zhou, (2020). Structural basis for the recognition of SARS-CoV-2 by full-length human ACE2. Science, 367(6485), 1444-1448.
[21] J. Yang, R. Yan, A. Roy, D. Xu, J. Poisson and Y. Zhang, (2015). The I-TASSER Suite: protein structure and function prediction. Nature methods, 12(1), pp.7-8.
[22] A. Rambaut, N. Loman, O. Pybus, et al., (2020). Preliminary genomic characterisation of an emergent SARS-CoV-2 lineage in the UK defined by a novel set of spike mutations Virological. org
[23] V. Kovacova, K. Boršová, E.D. Paul, et al., (2021), A novel, room temperature-stable, multiplexed RT-qPCR assay to distinguish lineage B.1.1.7 from the remaining SARS-CoV-2 lineages, medRxiv, doi: https://doi.org/10.1101/2021.02.09.21251168
[24] G. Martínez-Rosell, T. Giorgino and G. De Fabritiis, (2017). PlayMolecule ProteinPrepare: A web application for protein preparation for molecular dynamics simulations. Journal of chemical information and modeling, 57(7), pp.1511-1516.
[25] M.H. Olsson, C.R. Søndergaard, M. Rostkowski and J.H., Jensen, (2011). PROPKA3: consistent treatment of internal and surface residues in empirical pKa predictions. Journal of chemical theory and computation, 7(2), pp.525-537.

[26] T.J. Dolinsky, J.E. Nielsen, J.A. McCammon and N.A. Baker, (2004). PDB2PQR: an automated pipeline for the setup of Poisson–Boltzmann electrostatics calculations. Nucleic acids research, 32(suppl_2), pp.W665-W667.
[27] W. Humphrey, A. Dalke and K. Schulten, (1996). VMD: visual molecular dynamics. Journal of molecular graphics, 14(1), pp.33-38.
[28] S. Plimpton, (1995). Fast parallel algorithms for short-range molecular dynamics. Journal of computational physics, 117(1), pp.1-19.
[29] U. Micka, C. Holm, and K. Kremer, (1999). Strongly charged, flexible polyelectrolytes in poor solvents: molecular dynamics simulations. Langmuir, 15(12), pp.4033-4044.
[30] J. Huang, and A.D. MacKerell Jr, (2013). CHARMM36 all-atom additive protein force field: Validation based on comparison to NMR data. Journal of computational chemistry, 34(25), pp.2135-2145.
[31] S. Succi, (2018). The lattice Boltzmann equation: for complex states of flowing matter. Oxford University Press.
[32] F. Sterpone, P. Derreumaux, and S. Melchionna, (2015). Protein simulations in fluids: Coupling the OPEP coarse-grained force field with hydrodynamics. Journal of chemical theory and computation, 11(4), pp.1843-1853.
[33] M.J. Abraham, T. Murtola, R. Schulz, S. Páll, J.C. Smith, B. Hess, and E. Lindahl, (2015). GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. SoftwareX, 1, pp.19-25.
[34] P. J. Steinbach and B.R. Brooks, (1994). New spherical-cutoff methods for long-range forces in macromolecular simulation. Journal of computational chemistry, 15(7), pp.667-683.
[35] A.D. MacKerell Jr, D. Bashford, M.L.D.R. Bellott, et al., (1998). All-atom empirical potential for molecular modeling and dynamics studies of proteins. The journal of physical chemistry B, 102(18), pp.3586-3616.

# LIST OF AUTHORS

Marco Aldinucci
Rossella Arcucci*
Neva Bešker
Fabio Bonaccorso
Barbara Cantalupo
César Quilodrán Casas
Letizia Chiodo*
CRESCO team: D.Alderuccio, F.Ambrosino, G.Baldassarre, T.Bastianelli, R.Bertini, G.Bracco, L.Bucci, F.Buonocore, M.Caiazzo, B.Calosso, G.Cannataro, M.Caporicci, G.Carretto, M.Celino, M.Chinnici, R.Clemente, M.De Rosa, D.Di Mattia, G.Formisano, S.Ferriani, G.Ferro, A.Funel, D.Giammattei, S.Giusepponi, G.Guarnieri, M.Gusso, W.Lusani, M.Marano, A.Mariano, S.Migliori, M.Mongelli, P.Ornelli, S.Pagnutti, P.Palazzari, F.Palombi, S.Pecoraro, A.Perozziello, G.Ponti, M.Puccini, G.Santomauro, , A.Scalise, F.Simoni, M.Steffè, D.Visparelli
Iacopo Colonnelli*
Valerio D'Alessandro*
Mihir Durve
Andrew Emerson
Mattia Falconi*
Matteo Falone
Federico Ficarelli
Simonetta Filippi
Francesco Frigerio*
Giorgia Frumenzio
Alessandro Grottesi
Guido Guarnieri
Yi-Ke Guo
Federico Iacovelli
Francesco Iannone*
Aniket Joshi
Maurice Karrenbrock
Marco Lauricella*
Alessandro Loppini
Marina Macchiagodena
Andrea Montessori
Laetitia Mottet
Asiri Obeysekara
Marco Pagliai
Christopher Pain
Silvia Pavoni
Matteo Pennisi
Samuele Pierattini
Piero Procacci*
Renato Ricci
Alice Romeo

Concetto Spampinato
Sauro Succi
Carmine Talarico
Adriano Tiribocchi

ENEA CRESCO IN THE FIGHT AGAINST COVID-19

ISBN: 978-88-8286-414-6