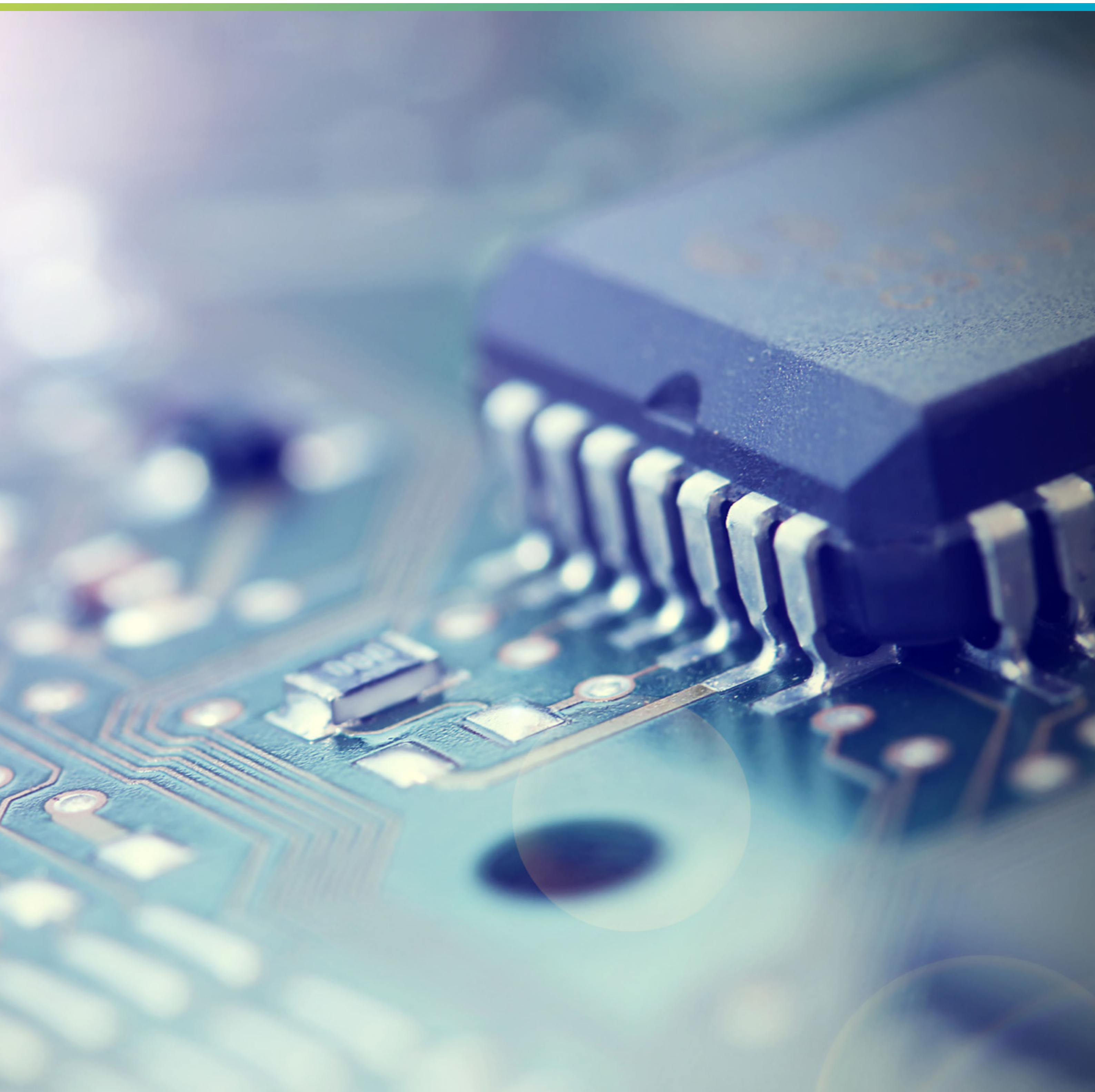
White paper

Accelerating acoustic simulations with high-performance computing

Design better products with faster simulations



Executive summary

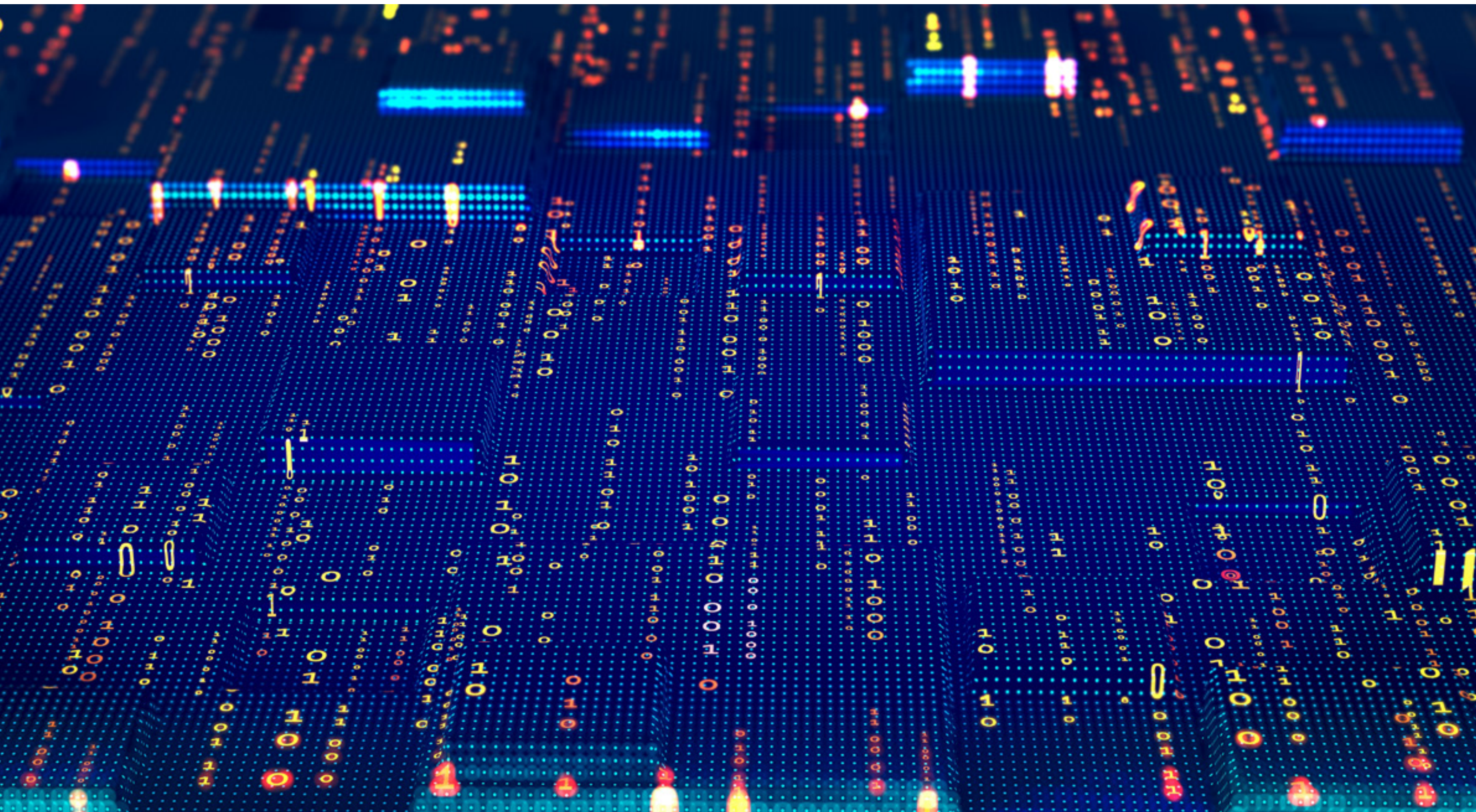
High-performance computing has been considered for engineering physics simulation solvers since their appearance in the 1960s. Due to the high requirements in both computational power and memory for solving the physics equations, developers have always looked at parallelising the solution or implementing new methods for reducing the turnaround time of the simulations, while maintaining a high level of accuracy.

For acoustics, the biggest challenge lies with reaching high frequencies and computing pressure wave propagation with interference from large objects. While different methods exist for solving acoustic wave physics over large distances, the finite element method presents an excellent compromise between accuracy and the computational effort for the most common frequency ranges of interest. On the other hand, reaching high frequencies has an increased computational cost that can be reduced by using high-performance computing, innovative numerical methods, and general-purpose graphics processing unit (GPGPU) technology.

Actran has been built with high-performance scalability from the beginning, resulting in excellent performance and turnaround time across different computer architectures and for various applications. Several example cases will be showcased in this whitepaper for solving various problems with specific requirements. These examples include pass by noise simulation of an electric vehicle, vibroacoustic simulation of a satellite, aircraft engine noise, high frequency vibroacoustic modelling for a vehicle and loudspeaker integration.

Nowadays, high-performance computing has become significantly more available than ever with the democratisation of cloud computing. Engineers can now easily create custom architecture to solve their specific problems. Actran is ready to take advantage of all architectures to empower engineers to design and manufacture quieter and more sustainable products.

Table of contents



1. Introduction

Engineering physics-based simulation has long been associated with high performance computing. From the structural simulations that helped design the original moon lander – where cards were fed to room-size computer – to modern simulations utilising thousands of cores of a modern supercomputer, large machines were needed to run ever-more demanding simulations. Supercomputers became necessary to solve a wide variety of physics including static and dynamic structure performance, fluid flows, weather forecasting, noise phenomena and more. While the numerical methods powering engineering simulation have evolved over the decades, the advances in computational power have had a highly significant impact in making these simulations faster. These advances have allowed simulation models to grow in size as engineers strive for higher fidelity when simulating the physical behaviour of products. At the same time, they enable more rapid design iterations as the time required for solution can be massively reduced using high-performance computing; ensuring shorter product development cycles and a faster go-to-market strategy.

For acoustic simulation, especially when it's based on the Finite Element Method (FEM), the performance of a model is driven by the size of the studied object as well as the frequency range of interest. In a FEM-based acoustic simulation in the frequency domain, the linear system Ax=B is solved for a set of unknowns which are the degrees of freedom (dof) of the problem. These unknowns are related

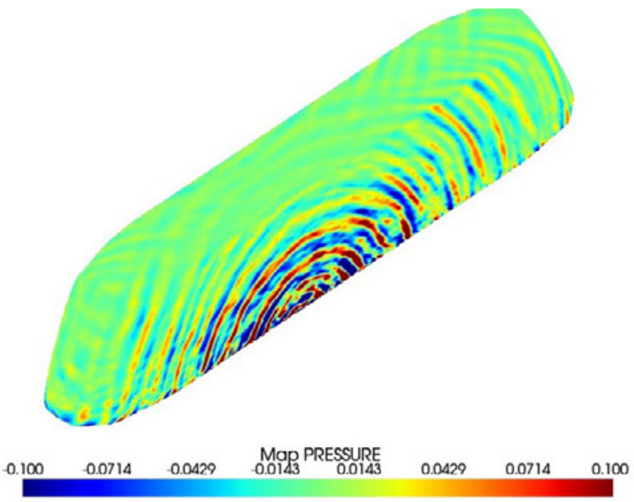


Figure 1. Acoustic radiation around a tram vehicle for pass-by noise evaluation

to the number of nodes in the computational mesh that is used to discretise the simulation domain. The discretisation required for acoustic simulation results in a computational mesh whose element size is driven by the maximum frequency to be solved. A certain number of elements per wavelength – which is defined as the speed of sound divided by the frequency – is required to obtain sufficient accuracy in the solution of the system. Furthermore, for the linear system to be solved the cost of performing the necessary matrix inversion is proportional to the cube of the dimension of the matrix; this cost becomes quite large as the number of dof increases. Numerical methods such as matrix factorisation and back-transformation have been used to reduce this cost, but it still remains quite expensive for large systems. Finally, since the speed of sound depends on the acoustic medium (most commonly air), large objects such as cars, aircraft and trains become difficult to model when extending into higher frequencies.

With advances in high performance computing and numerical methods, larger objects and systems can be modelled efficiently. This has allowed to capture both the system and component level response more accurately in higher frequencies, at a reduced cost and wait time. From a hardware point of view, central processing units (CPUs) have continued to grow in computational power. In the coming years, advances in graphics processor units (GPUs) have the potential to revolutionise engineering simulation thanks to the significant acceleration that they can provide. From a numerical standpoint, new computation methods continue to appear along with continual solver optimisations; such improvements decrease simulation time significantly. Both topics will be covered in this whitepaper as well the scalability of both CPU and GPU hardware.

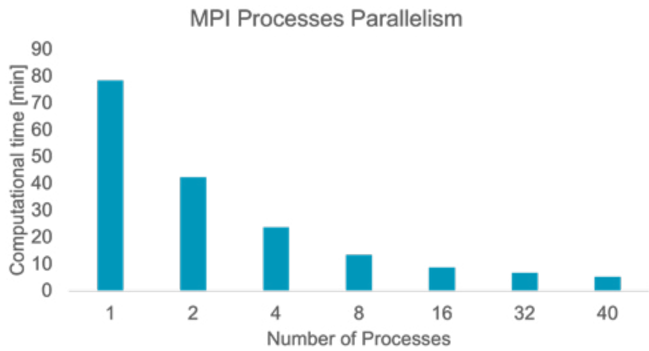


Figure 2. Typical scalability curve

This whitepaper will first introduce a series of definitions, methods and strategies that are required for understanding information presented here. This information will then be applied on real-world examples to highlight the potential of high-performance computing for acoustic simulation. Guidelines will be provided for these specific examples that can be extended to the same type of application. These examples are solved using Actran, the acoustic, vibroacoustic and aeroacoustic simulation software suite.

2. Methods

In this section the methods and strategies for efficiently utilising high-performance computing will be outlined. All of these will focus on both the reduction of computational time and memory. These are important topics to consider as computational time is crucial for rapid design iterations and FEM-based acoustic simulations tend to be memory intensive.

2.1. Hardware definitions

Typically, in scientific computing the software intends to take advantage of the processor of the computer to solve a problem in an appropriate timeframe. The progression of CPU technology over the years has allowed the power of the CPUs to grow exponentially. For scientific computing, these processors are arranged into large clusters (in the order of thousands of CPUs, commonly referred to as cores), which enable the parallelisation of simulation and divide a large problem into many smaller ones that can be solved at the same time; thus, providing results faster. For simulation software, especially when it is solving CPU-intensive problems, the scalability of the software with the increase of the computing cores is an important parameter for its computational performance.

To take full advantage of the hardware, the software needs to be adapted and optimised. To enable parallel computing, there are two strategies which can be used:

1. **Computational process parallelism** based on the Message Passing Interface (MPI) protocol, where the computation tasks are distributed to a team of several different processes and process memory is not shared between the different processes (distributed memory)

2. **Multithreading,** where the work of a given process is distributed on multiple threads and process memory is shared among different threads.

Further to CPU technology, there has been a relatively recent emergence of GPUs for computing. In contrast to a CPU which contains less than one hundred generic computing cores, GPUs consist of thousands of specific cores. Graphics cards like those in Figure 4 have traditionally been used for powering computer graphics; their inherently parallel architecture has the potential to significantly speed up scientific computations such as acoustic simulation. Replacing CPU capacity with GPUs could provide a cost-effective approach to high-performance computing.

2.2 Numerical methods

Like the progress of processor technology, numerical method discoveries are equally important as any improvements in the solver itself can provide an immediate benefit to the end user without having to buy a new computer. Simulation software developers can either construct their own linear solvers optimised for the specific workflows, or they can utilise off-the-shelf linear algebra packages such as Intel's Math Kernel Library (or Intel MKL), Intel's Pardiso or MUMPS (MUltifrontal Massively Parallel sparse direct Solver).

Using such packages can allow developers to focus on the aspects of the software that demand domain expertise while at the same time enabling solid computational performance.

Figure 3. Typical computer cluster

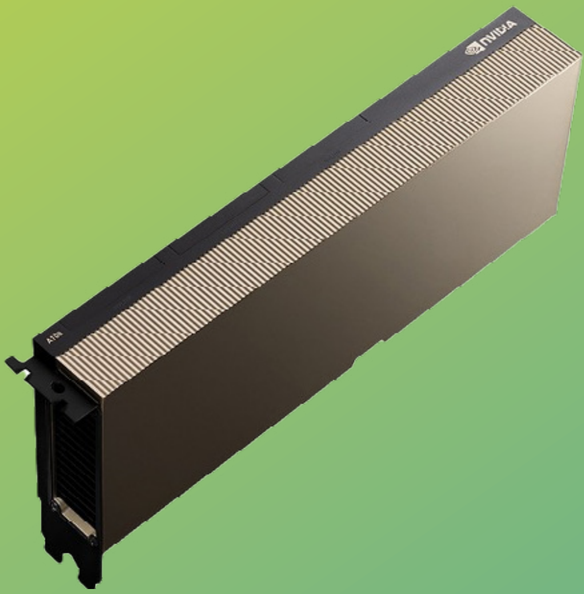


Figure 4. NVIDIA A100 Tensor Core GPU

In this section, parameters will be presented that can have a high impact on the computational performance in terms of both time and memory.

Adaptive meshing

Finite elements for acoustic simulation are all about creating a computational mesh that can accurately capture the wave propagation across the different regions of the model. When defining the mesh, guidelines exist for how many elements per wavelength should be used. However, since all frequencies are solved independently, utilising one mesh refinement based on the highest frequency (where the wavelength is shortest), can lead to a waste of computational resources. The same mesh applied to lower frequencies would be overrefined as the number of elements per wavelength is significantly greater than recommended guidelines.

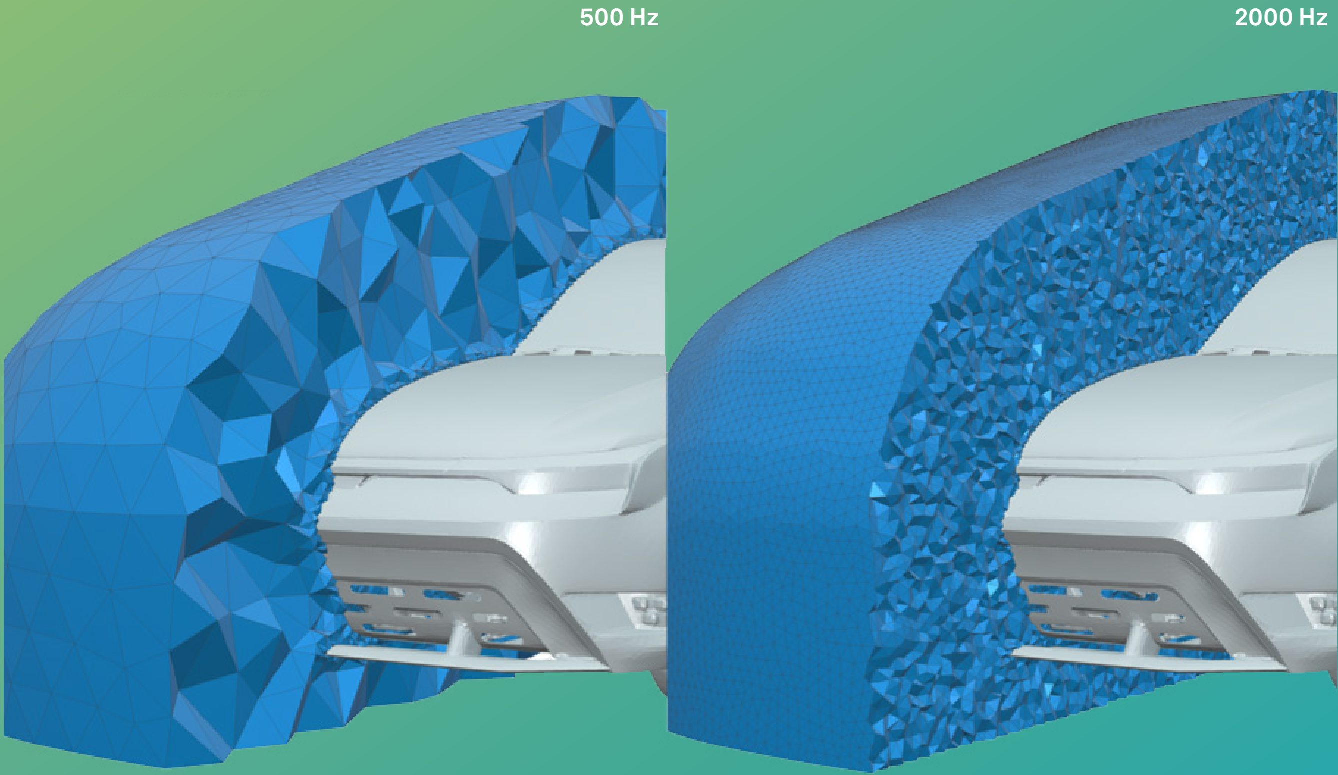


Figure 5. Different meshes are created automatically by the solver to match the frequency of solution

Figure 5 demonstrates how an appropriate computational mesh can vary for two different frequencies, while still achieving the desired level of accuracy.

Since the solution at each frequency is independent from the others, different mesh refinements can be used for each frequency however, this would require a lot of engineering effort to create. An adaptive mesh routine that automatically generates the mesh for each frequency (or bands of frequencies), can significantly speed up the simulation. An example of the computational time required for the base approach, where one mesh is used throughout the simulation frequency spectrum is compared against the computation time of an adaptive approach in Figure 6. The accuracy of the solution remains the same since all frequencies are appropriately resolved.

Single-precision solver

Typically, a double precision level of accuracy is used in the linear algebra matrices as they are assembled and solved. The double precision provides sufficient accuracy when needing to represent very small numbers which are common in both acoustics and dynamics.

However, in many cases using single precision arithmetic will result in a lower memory consumption; this lower precision could result in reduced accuracy in some cases.

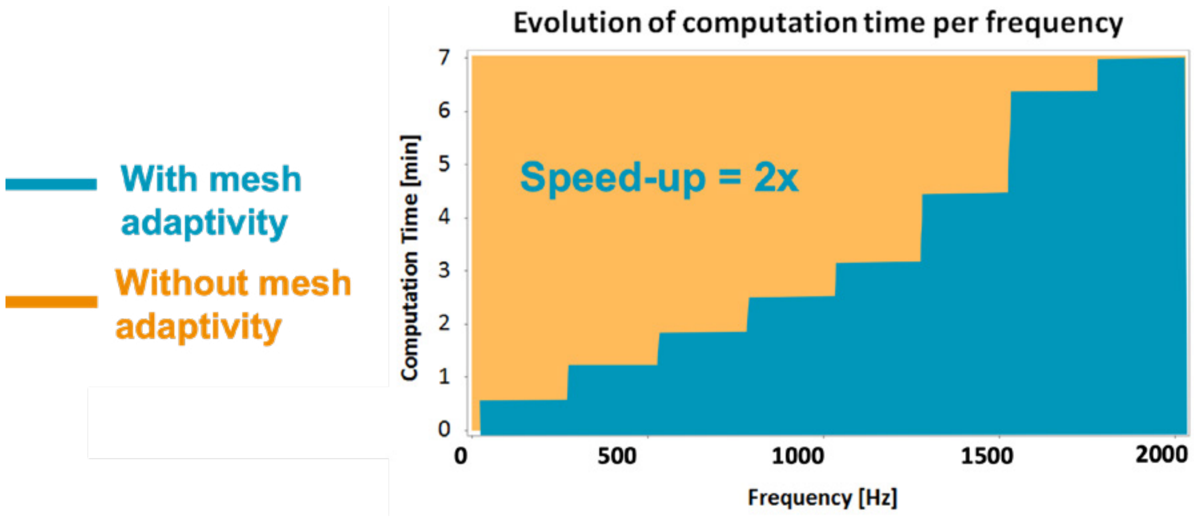


Figure 6. Computational performance comparison with and without adaptive meshing

For purely acoustic applications, such as exterior radiation or interior propagation without structural vibrations, the single precision solver can be readily used with only a negligible loss of accuracy. For vibroacoustic and aeroacoustic simulations, caution may be needed when using the single precision solver as there remains a minimal risk for loss of accuracy. Validation is recommended to ensure the accuracy of the simulation.

The benefits of this feature are notable compared to the minimal risks mentioned above. Memory requirements can be reduced by half and the computational time will also be reduced significantly. The single precision solver can be easily activated within the solver properties of the model for both the MUMPS and Intel Pardiso solvers.

Block low-rank matrix factorisation

The Block Low-Rank (BLR) matrix factorisation technique [2] employs a low-rank approximation of the full-rank matrix. This method provides both memory and performance benefits while keeping the accuracy of operations consistent through an additional parameter. This approach is available within the MUMPS solver and can be used either directly for purely acoustic problems or by tuning the parameter specifically for vibroacoustic and aeroacoustic problems.

Parallelism possibilities in Actran

Actran can parallelise a computation based on different approaches. The use of each approach depends on the computational capabilities of the hardware system. By distributing the workload of a simulation, frequency parallelism remains the best choice as it allows a number of frequencies to be solved simultaneously achieving faster results. In case memory capacity is limited, loadcase or axisymmetric order parallelism can be used, but the computation will not be completed as fast as with frequency parallelism. For tackling large problems which cannot easily fit into the available memory, matrix or domain parallelism can be used. These allow to scale-out the job on several nodes by decomposing the system into smaller parts that can more easily fit into the available memory. Finally, for utilising a large cluster more efficiently, a combination of parallelism techniques can be used simultaneously, such as the combination of domain and frequency parallelism. It is important to use all cores of a processor available as they will allow Actran to run faster and provide results within shorter timeframes, leaving no resources idle through the process.

3. Applications

In this section, a number of different applications will be presented, each with its own challenges which can be overcome by using HPC (High Performance Computing) techniques. For each application, the model and its unique challenge will be presented, numerical methods will be explored, and the scalability of the software will be highlighted with respect to multiple processes and multiple threads.

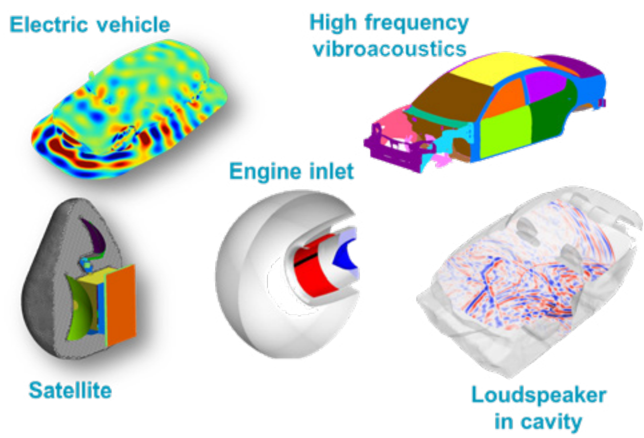


Figure 7. Applications that will be discussed in the following sections

Exterior noise radiation of electric vehicle

As electric vehicles become more prominent, the soundscape of cities is changing. The replacement of conventional internal combustion powertrains by electric propulsion is making vehicles much quieter. The frequency content of an electric motor is vastly different than that of a combustion engine and new source contributions such as road noise become more prominent. At the same time, more stringent regulations are introduced for pass-by noise and simulation can be used to ensure compliance early in the design process, reducing the need for expensive prototype testing.

For conventional pass-by noise testing, the car is driven past a set of microphones under specific conditions. As this is more difficult in simulations using a frequency domain solver and a stationary source, indoor testing can provide useful insight. Indoor pass-by noise testing involves the vehicle being static and a large array of microphones placed on either side and post-processing is used to emulate the movement. Pass-by noise simulation requires the full vehicle to be modelled which translates to a very complex geometry and a large model size. Excitations can come from various sources but most commonly they are equivalent monopole sources which are then recombined at a later step with the actual excitations to recover the real noise.

In this example (see Figure 8), one large wrap mesh surface is used to automatically generate the mesh for the acoustic propagation. A boundary condition (BC) known in Actran as the BC Mesh component is used on the electric engine to apply the acoustic excitation based on available vibration data. The results can be computed for several different loadcases and excitations (such as a varying RPM scenario) in the the same computation and for almost the same cost as a single excitation. Finally, the pass-by noise curves and indicators are output.

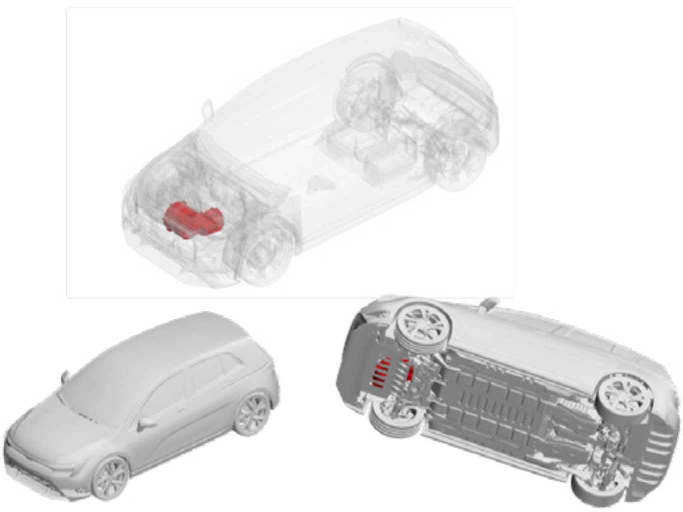


Figure 8. Pass-by noise model of an electric vehicle

Numerical parameters	
Number of DOFs	4.4 million
Number of elements	19 million
Loadcases	12
Frequency range	50 to 2000 Hz, 50 Hz step

Table 1. Numerical parameters for the pass-by noise model

The nominal performance of this case, where one processor and one thread are applied, is 13 hours and 21 minutes with a memory consumption of 78.5 GB. The impact of taking advantage of the several numerical optimisation techniques outlined in section 2 is shown in Figure 9 below.

After applying all numerical methods, the computational time has dropped to 4 hours and 50 minutes; a calculation which is about 2.75 times faster. The peak memory

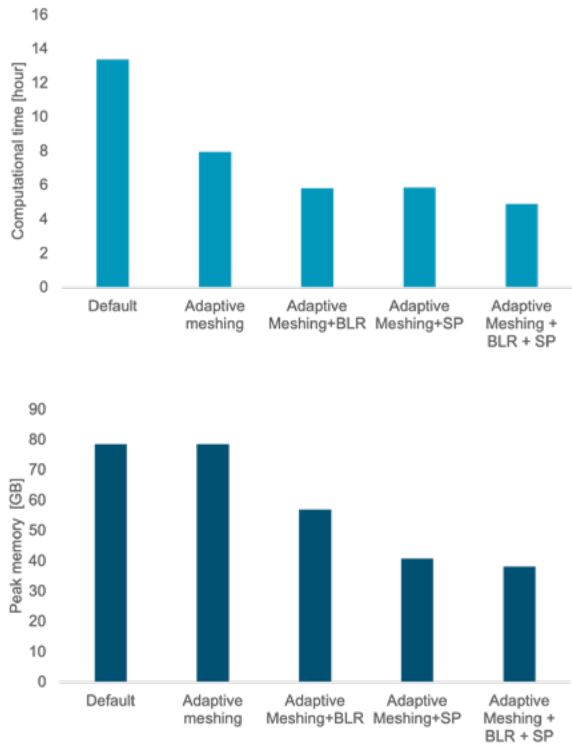


Figure 9. Speed up based on solver parameters. BLR denotes the Block Low-Rank technique and SP the single precision solver.

requirement was also reduced by 52% to 38 GB. This reduction in memory can enable the use of more parallel processes. When applying these modifications, it is important to evaluate where the accuracy of the simulation has been impacted. Reviewing Figure 10 it is evident that

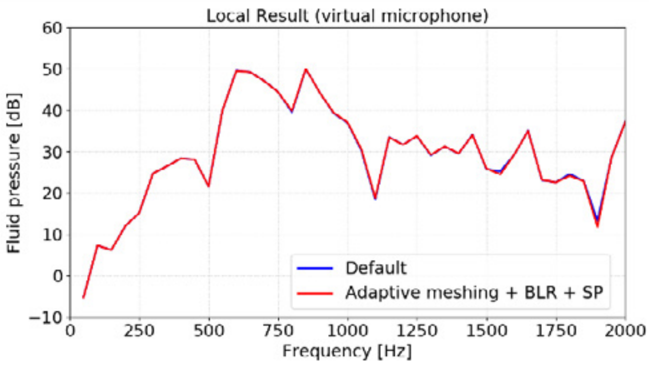


Figure 10. Effect of the solver optimisation techniques on the accuracy of the solution

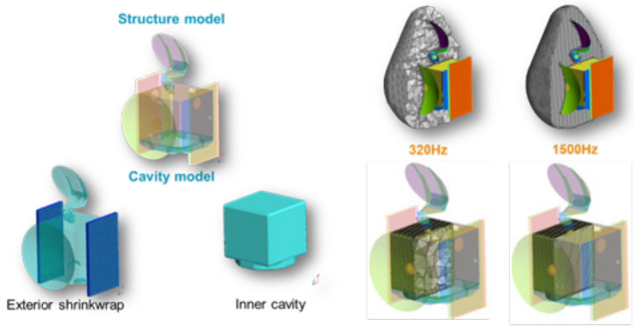


Figure 12. Computational model for satellite vibroacoustics

the accuracy when using these settings has not been significantly impacted, thus it is safe to use them in this context. After reducing the memory requirements significantly, a message passing interface (MPI) process and multithreading HPC strategy can be applied. Since frequency parallelism is used for the process parallelism, the memory requirement per process remains the same.

The simulation time scaled up well as the number of processes was increased up to 40 parallel processes with no obvious saturation point; computation time continued to drop as more parallel processes were applied. This resulted in a simulation time reduction of 96%, from 118 min to 5 min. For multithreading, improvements were observed up to a point, as adding more threads above 16 had a minimal effect to the computational time on the selected node which indicated that saturation point had been passed.

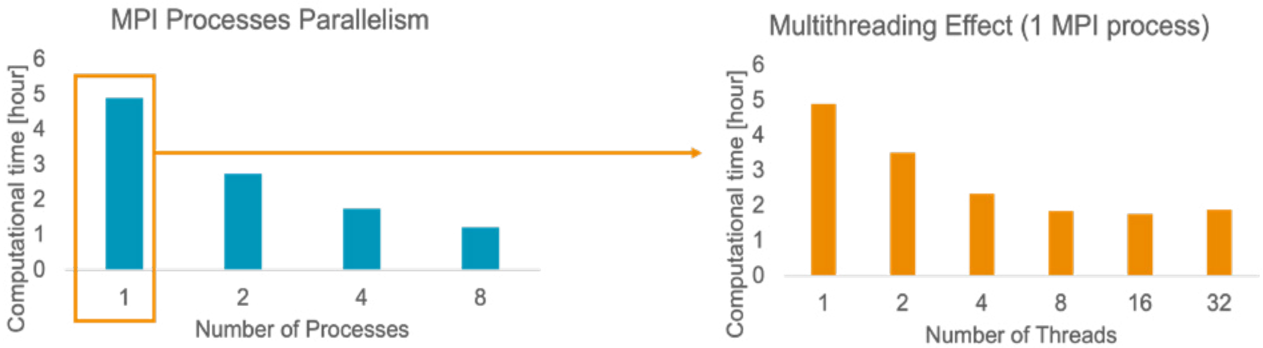


Figure 11. Impact on computational time with an increasing number of processes and threads

Vibroacoustic simulation of satellite

Space launches are famously difficult to achieve correctly. This is because of the adverse conditions to which the launchers are submitted to during lift-off and the various stages involved travelling from the ground to space. An important consideration on the design of a launcher and its payload bay is that the payload, typically a satellite full of sensitive electronic devices, is delivered safe to orbit without any damage despite such conditions. To ensure launcher and payload integrity, engineers require accurate predictions of the acceleration and stress at key parts of the system. Simulation can help them by providing insight on the dynamic behaviour of the system early in the design process. Despite the models for such studies being computationally intensive to solve, high accuracy is required meaning that the impact of any optimisation strategies must be carefully evaluated. In the following model a satellite is subjected to a diffuse sound field, a condition that approximates the excitation during lift-off. The model is split into three parts: the structure of the satellite, its interior cavity and the exterior acoustic domain where the excitation is applied. The model and its characteristics are shown in the Figure 12 and Table 2. It is worth noting that structure is reduced to its modes using MSC Nastran and the performance of this process will not be evaluated further.

Satellite vibration model	
Sequence	Direct Frequency Response
Solver	MUMPS
Number of modes	2,246
Modal extraction up to	3000 Hz
OP2 size	118 GB

Table 2. Numerical parameters for the satellite model

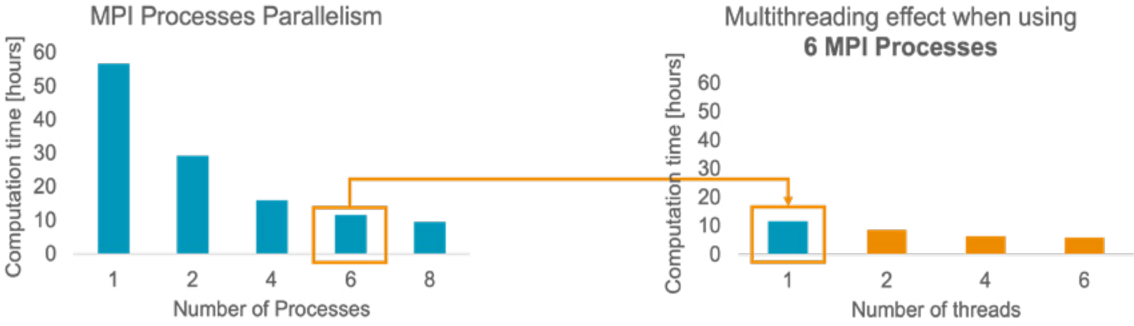


Figure 13. Impact on computational time with an increasing number of processes and threads

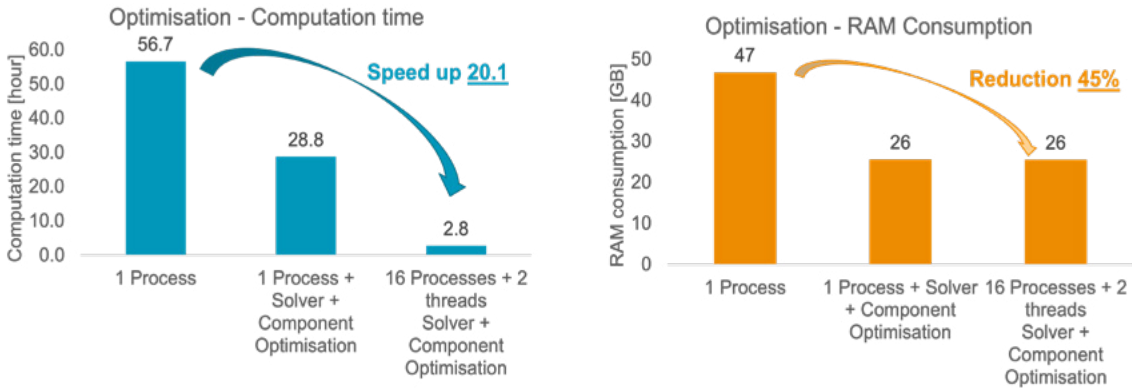


Figure 14. Speed up and memory reduction with parallel processing and solver optimisation

At its nominal form, this case requires about 2 days and 8 hours for solution with a relatively low memory consumption of 47 GB. For reducing the computational time of this model, a combination of frequency parallelism, multithreading parallelism, adaptive components, and the single precision solver will be used. As previously, multiple processes allow to reduce the computational time significantly from 57 hours to 10 hours. The use of 6 processes for parallelism which was complimented with multithreading quickly reached a saturation point as demonstrated in Figure 13.

Using adaptive mesh generation in the interior and the exterior of the satellite allows to reduce both computation time and memory requirements by about 15%. The single precision solver provides a further reduction of 20% for the simulation time and 36% in memory consumption.

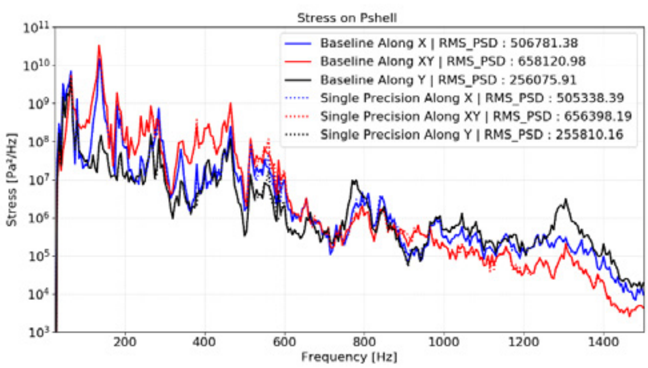


Figure 15. Stress on a shell of interest for comparing the accuracy of the double precision vs the single precision solver.

Overall, by combining all methods and utilising 16 parallel processes with 2 threads per process, the simulation time can be reduced by 20 times, from 57 hours to 2.8 hours on a single computing node. Such reductions allow engineers to do multiple studies per day to evaluate changes in their design. The memory consumption was also reduced by 45% compared to the nominal case.

Finally, as mentioned above, it was important to make sure that the accuracy had not been impacted significantly using these settings. Looking at the plot in Figure 14, acceleration and stress at critical points of the structure indicated only a minimal impact and verified that these settings could be safely used for such configurations.

Aircraft engine noise

During take-off and landing, fan noise is a dominant noise source of modern turbofans with high by-pass ratio which is commonly used in modern aircraft. To respect more and more stringent noise regulations around airports, aircraft manufacturers need to design and optimise acoustic liners and predict accurately the radiated noise from the nacelle. The main frequencies of interest are those associated with the blade passing frequency (BPF), its harmonics, and sub-harmonics. The blade passing frequency is defined as the frequency of rotation times the number of blades of the fan. Since engine fans have multiple blades and rotate at a high speed, this can be in the order of 2000 Hz, which makes simulations computationally intensive.

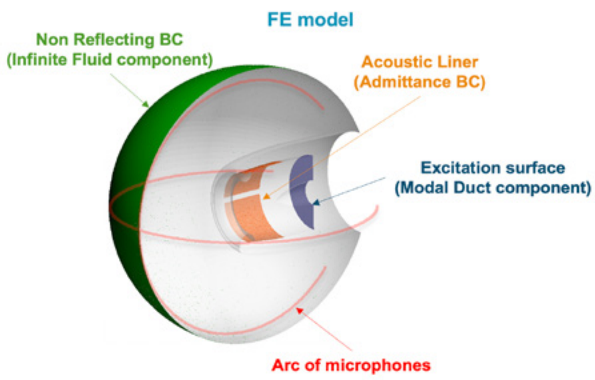


Figure 16. Computational model for engine inlet noise radiation model

For such simulations only a limited number of frequencies need to be solved, therefore a different parallelism method will be presented developed to reduce the size of a system via parallelisation. A nacelle of a turbofan was used as an example case, excited by analytical modes at the fan location and allowed to radiate noise into the free field, see Figure 16.

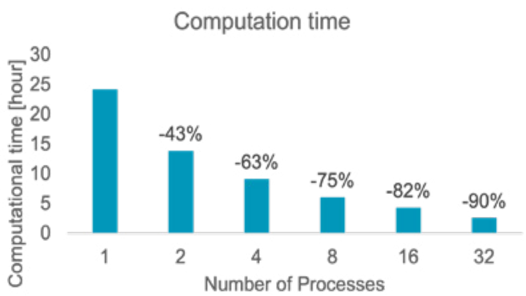
The baseline results, which were solved by a sequential process, required a day of processing to be solved and consumed a staggering 1.01 TB of RAM. The acoustic field was designed to target a frequency of 1.5 times the BPF.

Matrix parallelism was used to parallelise the solution across processes. Contrary to frequency parallelism where different frequencies are solved over different processes, matrix parallelism solves the matrix of the system over different processes. In this way, each process solves only a small part of the matrix and at the end of each frequency the solution matrix of the system is recombined. However, unlike frequency parallelism where increasing the number of parallel processes will increase the total memory consumed, both the memory consumption per process and the total simulation time are expected to reduce as a result of matrix parallelism, as demonstrated in Figure 17.

Engine inlet noise radiation	
Sequence	Direct Frequency Response
Solver	MUMPS
Number of DOF	8.2 million
Number of elements	19 million
Loadcases	177
Frequency range	3750 Hz (1.5 BPF)

Table 3. Numerical parameters for the engine inlet radiation model

By using 32 parallel processes, simulation time can be slashed by 90 % and memory consumption by 96%, which highlights that matrix parallelism can be used to distribute memory intensive simulations over several nodes with lower memory requirements.



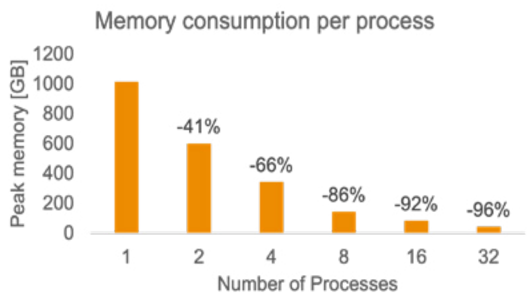


Figure 17. Impact on computational time with an increasing number of processes with matrix parallelism

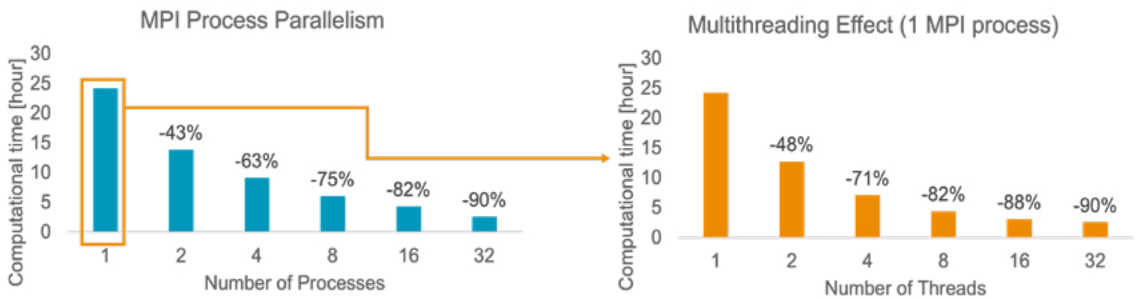


Figure 18. Impact on computational time with an increasing number of processes and threads with matrix parallelism

Similarly, multithreading can also be used for reducing the simulation time. As shown in Figure 17, the efficiency of multithreading is slightly better than the efficiency of the matrix parallelism. Using 16 threads for a sequential process with multithreading can help reduce the simulation time by 88% while matrix parallelism will reduce the simulation time by 82%.

Loudspeaker integration in vehicle cabin

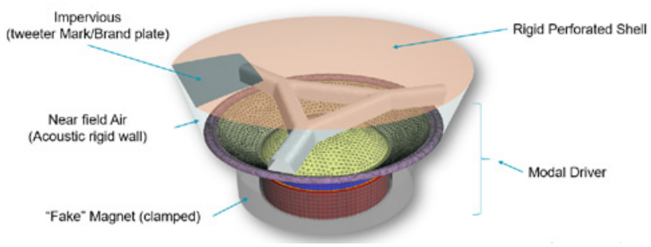
The sound system and interior comfort have become focal points in new vehicle design for which customers appreciate and are willing to pay a premium. A sound system needs to provide excellent sound quality while masking the noise from the other parts of the vehicle. The design should be well integrated within the structure of the vehicle so as not to create further noise due to the loudspeaker vibration.

Simulation can help sound system engineers to evaluate a loudspeaker's performance as both an isolated component and when integrated with the door of a vehicle. Since the human auditory spectrum ranges from 20 Hz to 20,000 Hz, it is important to evaluate the loudspeaker's ability to convey the sound appropriately throughout the whole spectrum. For simulation, this presents a unique challenge as the cavity of a vehicle is large in size – in the order a few cubic meters – and thus traditional FE methods become expensive. Using a different modelling technique, the Discontinuous Galerkin method or DGM offers a high-order, time-domain solution. The linearised Euler equations allow for tackling high frequencies more efficiently in large systems. The DGM solver can also leverage GPU technology to massively accelerate simulation times.

To showcase these capabilities a loudspeaker case study is presented below in Figure 19. The model comprised of 540 million degrees of freedom for the cavity, and it was solved in two steps. As a first step, a loudspeaker model of a tweeter, a stiffener and a grille were solved in the frequency-domain by the FEM solver. The output of this first simulation was used as an input for the cavity simulation where the sound propagation within the vehicle cabin was solved. Table 4 shows the model characteristics for the two steps. It is important to note the first step was negligible in terms of computational effort in comparison to the second step so it will not be discussed further here.

Due to the large size of this case, it was nominally solved with 240 processes (using AMD EPYC 7V12 @ 2.40 GHz) within about 3 hours, with a memory consumption of about 820 GB of RAM. When using a single NVIDIA® A100 Tensor Core GPU, a reduction both in computation time and memory was achieved by 10% and 96% respectively. By using two NVIDIA A100 GPUs, the calculation was consistently sped up by 40%, and with 4 GPUs the computation was solved 3 times faster than with 240 CPU processes. When also factoring the cost of acquisition of the processor and the GPU, it is obvious that the GPU allows for faster simulations with a similar cost, see Figure 20.

Tweeter model



Cavity model

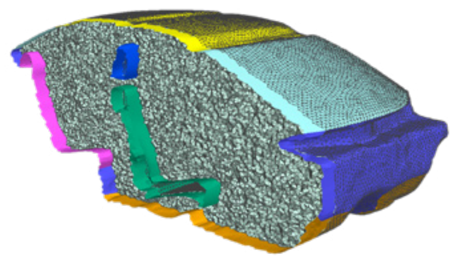


Figure 19. Computational models the loudspeaker integration model

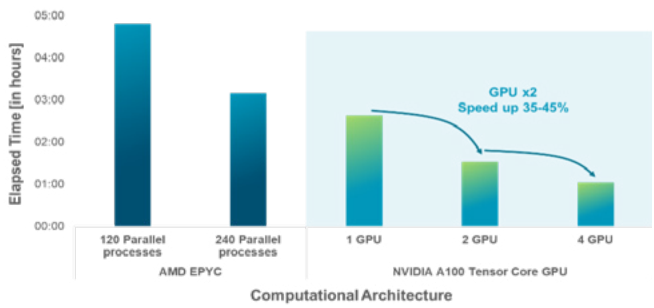


Figure 20. Impact on computational time with GPU acceleration

Loudspeaker in cavity	
Sequence	Actran DGM
Cavity volume	2.84 m³
Number of DOF	540,694,708
Number of elements	1,493,037
Average element order	6.1
Maximum frequency	20,000 Hz

Table 4. Numerical parameters for the loudspeaker integration model

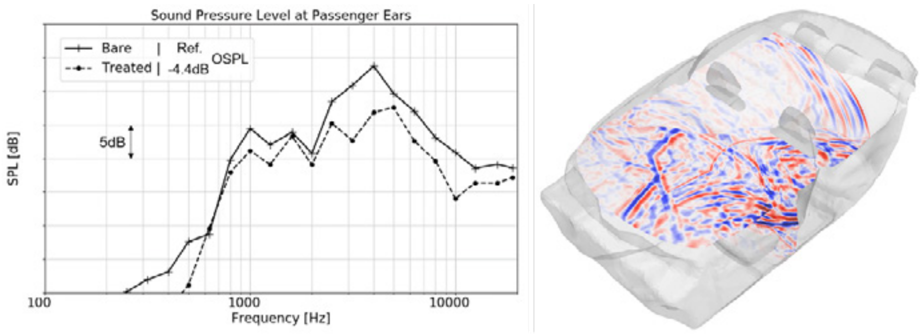


Figure 21. Comparison of sound pressure level between a bare and a treated cavity (left) and sound propagation of a loudspeaker within a vehicle cavity (right)

High frequency response of a vehicle

While the DGM method can solve large, purely acoustic problems in high frequencies, for the moment it cannot be used to solve vibroacoustic problems. Instead, statistical energy analysis (SEA) methods have been traditionally used for solving mid and high frequency vibroacoustic problems. The SEA approach splits the structure into subsystems and calculates the energy flux between the regions to identify vibration levels. Virtual SEA is a particular SEA method where the SEA model is created based on an existing finite element (FE) model [2]. The Virtual SEA workflow will average results in both space and frequency, looking at the response on a subsystem level as opposed to the FEM approach. The influence of trimmed surfaces can still be considered via analytical computations and coupling of structure components with the cavity is based on the Statistical modal Energy distribution Analysis (SmEdA). Excitation is provided through injected power, consistent with an energy-based model. Despite this requirement, deterministic excitations can still be defined however, as Actran will automatically convert these sources into the equivalent injected power, facilitating comparisons with finite element methods. The following case study demonstrates a complete vehicle cavity and structure defined with trim components, identified in Figure 22. The baseline case was solved in 31 hours with a memory consumption of 65 GB.

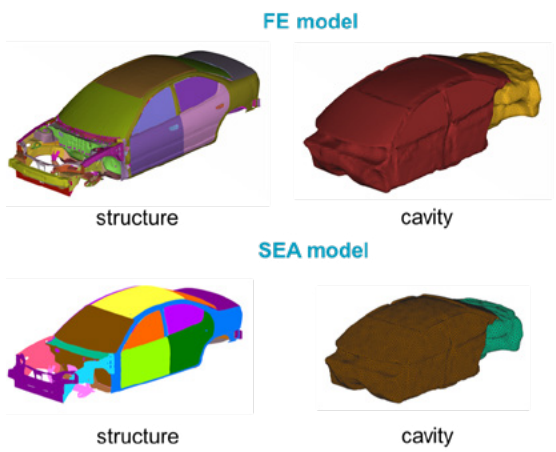


Figure 22. High frequency vibroacoustic computation model

Virtual SEA model	
Sequence	Actran Virtual SEA
Number of modes (structure)	6170
Number of modes (cavity)	836
Modal extraction up to	1120 Hz
OP2 size (structure)	188 GB
Number of subsystems	47
Maximum frequency	1000, third octave band

Table 5. Numerical parameters for the high frequency vibroacoustic model

Using 6 parallel processes, the model was solved three times faster, where the simulation time was reduced by 71%. Multithreading was used to further reduce the runtime, however its efficiency remained limited on the selected node.

Virtual SEA features a frequency extension tool, allowing to extrapolate results to higher frequencies. Accuracy at higher frequencies can be further refined using an intermediate interpolation, based on results extracted over a small frequency range illustrated as a ‘Result Point’ in Figure 24. This feature allows the user to compute energetic results well beyond the frequency range covered by the modal basis database, reducing both simulation time and memory consumption.

Using this scheme and 4 parallel processes, it was possible to reduce the simulation time from 31 hours to 4 hours and the memory consumption from 65 GB to 22 GB. In other words, this simulation was solved 10x faster with 3x less memory required.

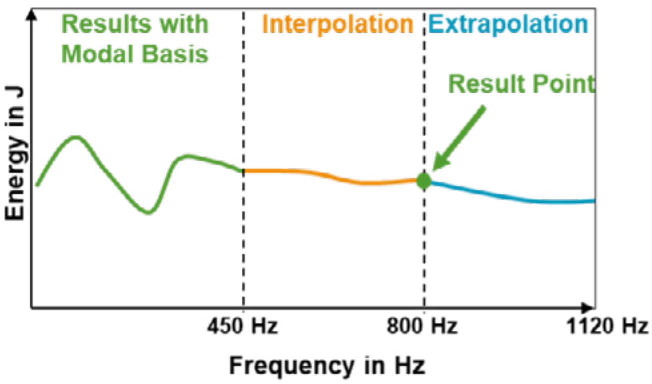
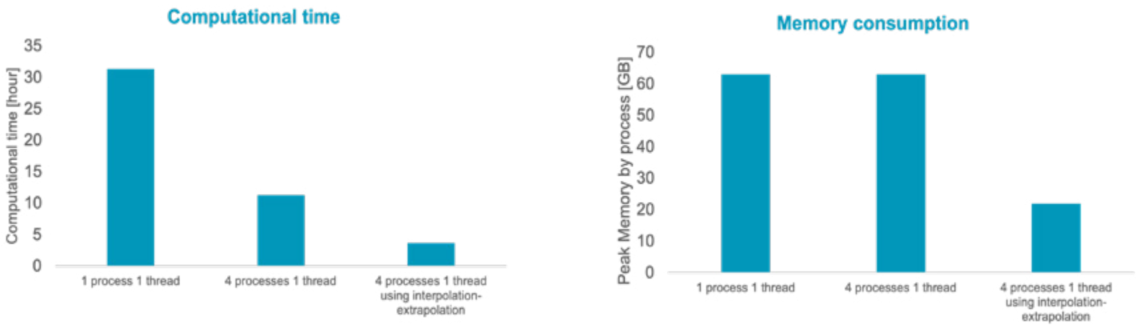


Figure 24. Illustration of the interpolation-extrapolation scheme

Since the interpolation-extrapolation method is an approximation, there will inevitably be an impact to the overall accuracy. However, considering the reduction in the required resources and regarding the minor differences observed in Figure 26, it can be concluded that the accuracy kept was exceptional.

4. Cloud computing

Traditionally, engineering simulations were run in local clusters or workstations. Recently however, cloud computing has emerged as an enabling platform that can allow engineers and companies to have more flexibility and computing power when performing simulations. By using the infrastructure of a cloud provider, whether dedicated to a specific application or a more generic provider, simulations on the cloud can be sped up by utilising a larger HPC capacity instantly. Engineers can run simulations whenever and wherever they want. There are different requirements for different physics solvers in an HPC environment and cloud computing can provide the best resources for numerically massive simulations without having to acquire additional hardware. By utilising cloud resources, organisations can eliminate the cost of both HPC hardware acquisition and maintenance, while having a tailored platform to meet their needs. The level of cloud resources available can vary with respect to cost, capability, scalability, and capacity required.



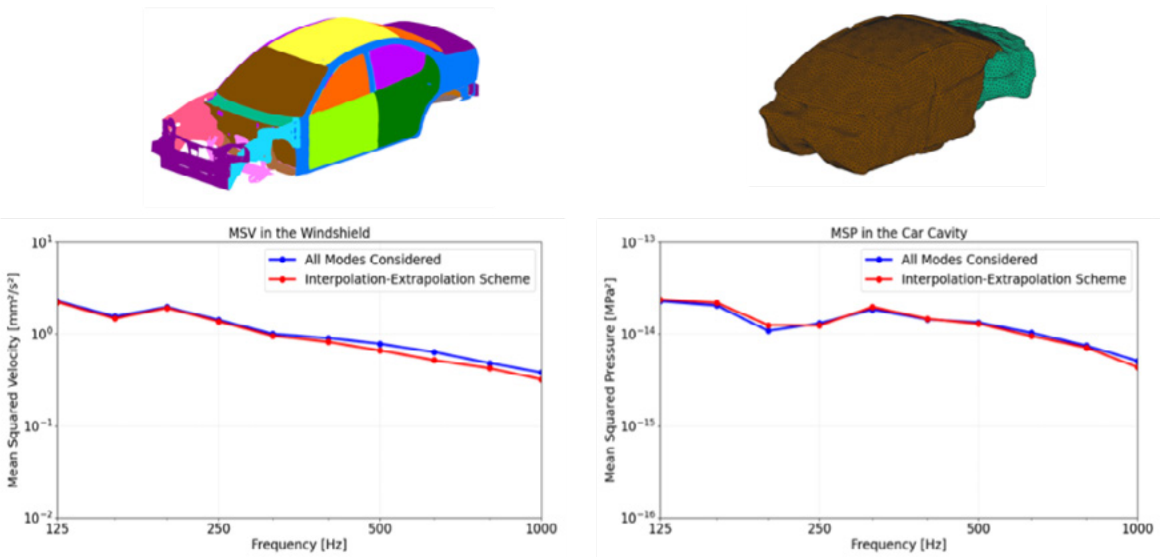


Figure 26. Effect of the interpolation-extrapolation scheme on accuracy of the simulation

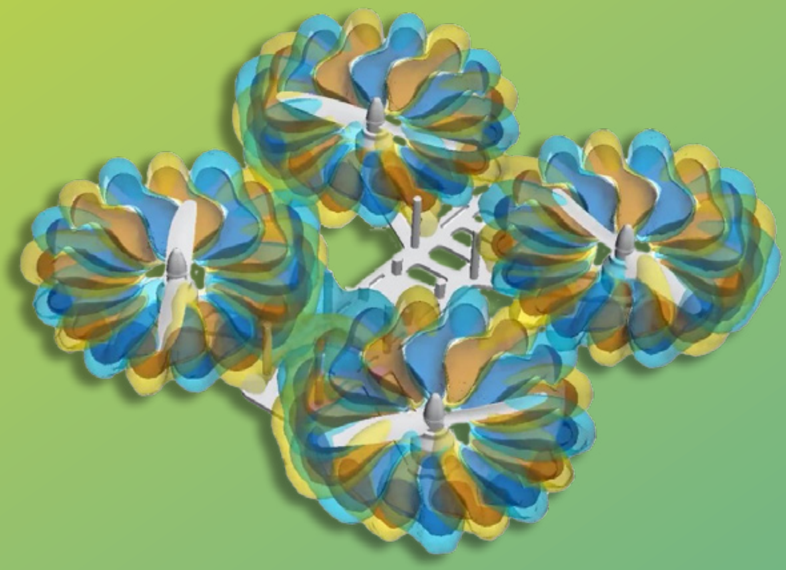


Figure 27. Drone noise simulation performed in the Microsoft Azure

Actran has been developed with high-performance computing in mind and is currently cloud-ready for use in **Amazon Web Services (AWS), Rescale,** and **Microsoft Azure,** where a drone noise project taking advantage of cloud computing was recently completed, demonstrated Figure 27.

Computing on GPUs is also more accessible than before, being available across all major cloud providers. GPU computing can enable compatible solvers to achieve extremely fast simulation times compared to CPUs at a fraction of the cost. Actran is at the forefront of this revolution with Actran DGM being the only GPU-accelerated acoustic solver on the market, continuously improving for GPU computations since 2016.

An important note on cloud computing concerns data security. Since engineering simulation is very often used to design cutting-edge products, consideration is given to ensure all sensitive information and data remain secure and out of prying eyes. Cloud providers, like Rescale, Amazon Web Services (AWS) and Microsoft Azure, have made significant investments in reinforcing their security protocols so that they comply with the strictest industry standards for security and data protection.

5. Summary

Creating high performance products that sound better, vibrate less, and make less noise has become easier than ever with the help of high-performance computing. Engineers can now iterate faster in the design and development process and solve large problems with low turnaround times. This allows for greater insight that can be converted into engineering solutions for solving the world's most important problems related to noise, vibration and harshness (NVH).

Modern solvers can take advantage of the latest iteration of CPUs and GPUs, fast and scalable numerical techniques, and flexible cloud infrastructure to provide answers to product concerns faster. Specific applications have been addressed including pass-by noise, space component integrity, loudspeaker integration and vehicle high-frequency vibration, where simulations have sped up the design and development process.

Actran is at the forefront of high-performance computing providing excellent scalability. Multiple possibilities exist for accelerating simulations based on the available resources and a high level of accuracy under all conditions, empowering organisations to design a quieter and more sustainable world.

References

[1] Mary, T., Block Low-Rank multifrontal solvers: complexity, performance, and scalability, Doctoral Thesis, Université de Toulouse, 2017.

[2] Sipos, D., Brandstetter, M., Guellec, A., Jacqmot, J., Feszty, D., Extended solution of a trimmed vehicle finite element model in the mid-frequency range, 10th International Styrian Noise, Vibration and Harshness Conference, 2020.

Hexagon is a global leader in digital reality solutions, combining sensor, software and autonomous technologies. We are putting data to work to boost efficiency, productivity, quality and safety across industrial, manufacturing, infrastructure, public sector, and mobility applications.

Our technologies are shaping production and people-related ecosystems to become increasingly connected and autonomous – ensuring a scalable, sustainable future.

Hexagon's Manufacturing Intelligence division provides solutions that use data from design and engineering, production and metrology to make manufacturing smarter. For more information, visit hexagonmi.com.

Learn more about Hexagon (Nasdaq Stockholm: HEXA B) at hexagon.com and follow us @HexagonAB.