Supercomputing Frontiers and Innovations

2025, Vol. 12, No. 1

Scope

- Future generation supercomputer architectures
- Exascale computing
- Parallel programming models, interfaces, languages, libraries, and tools
- Supercomputer applications and algorithms
- Novel approaches to computing targeted to solve intractable problems
- Convergence of high performance computing, machine learning and big data technologies
- Distributed operating systems and virtualization for highly scalable computing
- Management, administration, and monitoring of supercomputer systems
- Mass storage systems, protocols, and allocation
- Power consumption minimization for supercomputing systems
- Resilience, reliability, and fault tolerance for future generation highly parallel computing systems
- Scientific visualization in supercomputing environments
- Education in high performance computing and computational science

Editorial Board

Guest Editor's Introduction for Special Issue on

Computational Experiment in Aeroacoustics and Aerodynamics

Tatiana Kozubskaya, DSc, Keldysh Institute of Applied Mathematics of RAS, Moscow, Russia

I am pleased to introduce this special issue of Supercomputing Frontiers and Innovations. It is mainly composed of selected papers presented at the Tenth Russian conference "Computational Experiment in Aeroacoustics and Aerodynamics" (CEAA2024) which took place in September 16–21, 2024, in Svetlogorsk, Russia. Until 2018, these Russian conferences were held jointly with International workshops on the same topics.

The goal of these bi-annual scientific events is to discuss the state-of-the-art of computational aeroacoustics and related nonstationary aerodynamics, as well as their challenges from the viewpoint of aerospace and other industries dealing with high-speed flows and associated sound generation. The conferences provide an opportunity for fruitful interdisciplinary discussions between researchers, engineers, software developers, and industry managers in order to foster the development and promotion of computational experiments in aeroacoustics and aerodynamics as an effective tool for applied research and design.

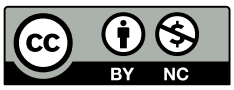
For publication in this special issue, we selected six papers whose topics are directly related to supercomputing and thus best suited the journal scope. According to the agreement with the Editorial board, all these papers went through the rigorous review process used by the journal.

Two papers of those six, namely by Alexey Duben with Viacheslav Sapozhnikov and with Andrey Gorobets, are devoted to the development of special low-cost techniques to reproduce and study computationally unsteady effects in turbomachinery. The paper by Ilya Abalakin, Vladimir Bobkov *et al.* focuses on combining the complex turbulent flow simulation with modeling the articulated blade mechanics for hinged helicopter rotors. The paper by Gleb Plaksin, Pavel Rodionov *et al.* is aimed to recover distributed acoustic sources formed on the airframe of a model supersonic business jet through analyzing the numerical data of scale-resolving simulations of turbulent flows around the jet body. The papers of the authors' teams, represented by Anna Posudnevskaya and Aleksey Yatskikh present the results of fundamental numerical studies of the turbulence development in Kolmogorov type flows and of the growth of localized disturbances in supersonic boundary layers, respectively. All of the above problems have very high computational complexity and require large resources of modern supercomputers to solve them.

I would like to thank the journal Editors, Technical editors and all the involved reviewers for preparing this special issue. Thanks to their kind assistance and earnest efforts, the JSFI special issues collecting selected papers of the regular CEAA conferences are becoming a good tradition.

Besides, I would like to bring your attention to the upcoming CEAA2026 Conference scheduled for September 2026, in Svetlogorsk.

Contents



DOI: 10.14529/jsfi250101

Recent Technology Trends in High-Performance Computing: Quantifying the Divergence of Top and the Rest

Sergey S. Konyukhov[1], *Alexander A. Moskovsky*[1,2]

Exploratory analysis methods were used to study basic characteristics of computing systems from TOP500_Lists. One of the peculiarities of the distribution of computing systems by performance is that it sufficiently well obeys an analog of the empirical Zipf's law, in which logarithm of performance is reciprocal to the rank of computing system. Based on this observation we can divide all systems from the lists into several performance classes: top, high, base, and entry levels. Our analysis also revealed differences between these classes in other characteristics besides, the computational performance, e. g., such as power consumption. For all performance classes, trends in evolution of the basic characteristics of TOP500 computing systems were described and, where possible, comments were provided to explain their behavior. Performance and energy efficiency of the TOP500_List computing systems in the next 5–10 years were estimated using simple linear models obtained by the least-square method. We have found that energy consumption needed for entry-level supercomputers to surpass the threshold value of performance and to enter into TOP500_List will decrease during this period.

Keywords: TOP500_List, *energy efficiency, supercomputer technology trends, Zipf's law.*

Introduction

Data archives of the world-wide known TOP500_List present an indispensable source of information about technology trends, international politics, economy, sociology, and other subjects concerning the area of high-performance computation. The GREEN500 rating emerged more recently with such important aspect as measurement of energy consumed by large high-performance computing systems. As average energy consumption increased, this aspect became one of the most important characteristics of computing systems along with their performance, so the latest TOP500_Lists contain information about energy efficiency of computing systems. Our work is dedicated to the combined analysis of performance and energy efficiency for computing systems from TOP500_Lists.

TOP500_List consists of 500 entries sorted by maximal performance in HPL test, R_{max}. The rating has been updated twice a year since 1993. In each edition, Erich Strohmaier, Jack Dongarra, and other rating authors present an analysis of new data and trends. They analyze long-term trends of R_{max} on TOP1, TOP100, TOP500, or other sublists, and different aggregates such as sums of R_{peak} or R_{max} of all TOP500 entries.

While R_{max} of Rank_1 position for both GREEN500 and TOP500 lists demonstrate sharp, step-like increase over the last years, most aggregate numbers change more smoothly. Decades-long trends are clearly visible in the analysis of R_{max} (R_{peak}) of Rank_1 or Rank_500 positions what allowed to make the long-term predictions such as the development of petaflops machines n the early 2000s or the emergence of the exascale supercomputer by 2018 back in 2007–2008.

By the mid 2010s the fast pace of the performance increasing slowed down, driven by slower progress of microelectronics[3] and budget limitations. By the end of the decade, the number of new systems in each TOP500_List edition also declined dramatically to 40–50 from more than 150 in the 2000s.

[1]RSC Group, Moscow, Russian Federation
[2]Lomonosov Moscow State University, Moscow, Russian Federation
[3]Proclaimed Moore's Law "death".

During the same period, energy efficiency of the Rank_1 computing systems from GREEN500_List grew by the factor of 20 over the last 10 years. However, this rapid progress was enrolled with diminishing returns from new microelectronic technology generations compared to six decades of development of computer architectures. We think that it would be useful to characterize quantitatively how such progress of the high-end technology is supported by a "common" supercomputer system from TOP500_List because the most performant TOP500 systems cannot rely on bleeding edge technology – they need mature solutions that can be deployed at scale. At the same time GREEN500 rating is a better indicator of new technology trends as it demonstrates the best solutions viable enough to run efficiently the complex program stack of LINPACK benchmark. In other words, we could suppose energy efficiency became the most important characteristic of computing system.

In our work we noted and measured the recent decline in the "TOP500 entry power ticket", i. e., minimal energy consumption between all TOP500 systems, $Min(P)$. We suppose that it demonstrates an important trend in the current technology development: divergence of the most energy efficient architectures from a "general case" represented in low-rank TOP500 systems. In other words, the cutting edge advances in CPU/GPU architecture become available to the majority of practical users only after a significant period of time.

The article is organized as follows. Section 1 is devoted to the review of related works. In Section 2 we describe our method by which we analyze data from TOP500_Lists. Section 3 contains the discussion of obtained results. Conclusion summarizes the study and points directions for further work.

1. Related Works

TOP500_List was initiated as a source of information to identify the current trends in the field of high-performance computing and continue Mannheim List statistics proposed by Hans Meuer [8]. Since 2001 in addition to brief analyses performed by the rating authors in each TOP500 issue, papers have been regularly published, in which their authors tried also to reveal technology trends and tendencies by statistical methods [1–4, 6, 7].

The scope of analysis was broad: from low-level detailed analysis of various subsystem designs to market share of HPC vendors and application areas. In our paper we focus on energy efficiency of supercomputers, or more specifically, on how much power would be enough to run supercomputer having minimal performance sufficient to appear on TOP500_List.

The work [7] presents "strawman" design solutions of future supercomputers and uses the TOP500 trends to estimate relative performance and energy efficiency of future designs, and especially to carry out detailed analysis of future design of memory and interconnect subsystems. Although in this work physical and infrastructure limits were taken into consideration, economic limitations were underexplored. In our work we analyze relations between performance and energy efficiency in a more formal way.

Out of the most recent publications, the papers [4] and [6] seem to be the most relevant to the subject of our study. The former article overviews the performance and energy efficiency trends and tries to give reasons for them by observing features of new hardware architectures. For these purposes heterogeneous systems and top-rank systems were carefully studied.

The authors of the later paper made a unique comparison of actual trends of the 2010s with the past assessment of "Exascale Report-2008" [5], and the "Frontier" supercomputer characteristics were compared with "strawman" models presented in 2015. As a result this work

gives an idea how accurate such long-term projections can be. While the works [4] and [6] focus mostly on top-rank systems, our work investigates trends in the entry-level systems of the TOP500_Lists.

Continuing a series of their works, Abramov and Abramov [1] analyze the TOP500_Lists of the latest years from different points of view. Together with other researchers they determine the concentration of the most part of the aggregate performance by the top-ranks systems in complete agreement with our observations.

2. Method

In this work we have analyzed data collected from *top500.org* website back to June 2013 since the first GREEN500 data became available. We used methods of exploratory data analysis and try to figure out evolution trends for the basic characteristics of TOP500 systems.

We used the following primary data (i. e., measured data):

- maximal performance, R_{max};
- peak performance, R_{peak};
- energy consumption, power, P;
- total number of cores, N_{cores};
- number of accelerator/co-processor cores, $N_{accores}$;

and secondary data (i. e., derived quantities):

- resource usage efficiency, $RUE = R_{max}/R_{peak}$;
- energy efficiency, $EE = R_{max}/P$.

First of all, we were interested in total energy consumption of a computational system having performance compared with the performance of Rank_500 system because the cost of electricity and the need to create special infrastructure required for the operation of computational equipment with high energy consumption can limit the usage of high-performance systems in practice.

Due to the specifics and quality of the used data[4], we applied the following procedure for our analysis:

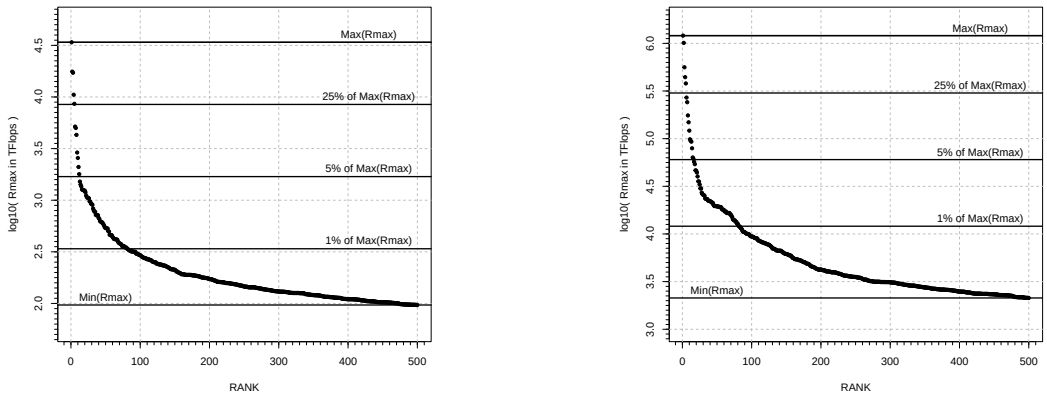
1. divide computational systems from the TOP500_Lists by R_{max} into several classes;
2. for each class deduce extrapolation formulae (models) describing behavior of R_{max} and P as functions of time (more precisely, as functions of TOP500_List number);
3. for two classes with strongly correlated data, transfer the model from the class having more reliable statistical description to the class with less reliable data.

3. Discussion

The distinctive feature of the TOP500 systems is a significant difference between characteristics such as maximal performance, energy consumption, or total cores number for Rank_1 computing systems and the vast majority of other computing systems in a list.

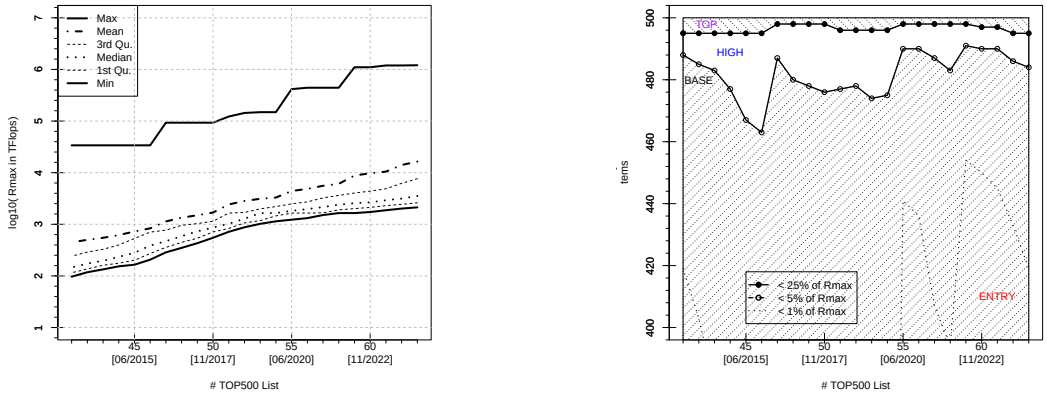
The distribution of R_{max} by system ranks on a semi-logarithmic scale can be described as an analogue of Zipf's empirical law (see Fig. 1) and this behaviour is more or less the same for all studied lists (see Fig. 2a).

[4]While the publication of R_{max} is evidentally mandatory for TOP500 systems, such data as energy consumption are practically absent for the most part of TOP500_List.



(a) List#41 [June/2013] (b) List#63 [June/2024]

Figure 1. Stratification of TOP500 systems by maximal performance



(a) Statistical distribution of R_{max} (b) Total number systems in different classes

Figure 2. Distribution of TOP500 systems by R_{max}

The data presented in Fig. 1 show that it is quite difficult to provide an objective criterion to divide all systems from TOP500_List into separate classes, so we formally divided the dynamic range of $\log(R_{max})$ values into 4 approximately equal intervals, giving preference to the ease of perception of information compared to the exact values. Thus, denoting R_{max} for Rank_1 system as $Max(R_{max})$ we use as class boundries $Max(R_{max})/4$, $Max(R_{max})/20$, and $Max(R_{max})/100$, (25%, 5%, and 1% of $Max(R_{max})$), instead of

$$\log\big(Max(R_{max})\big) - \frac{\log\big(Max(R_{max})\big) - \log\big(Min(R_{max})\big)}{4} \times K, \quad K = 1, 2, 3;$$

what, as we believe, should not greatly change our results and conclusions. Moreover, at $R_{max} = Max(R_{max})/4$ the distribution changes its character, the density of points noticeably increases and the slope of the curve decreases, i. e., the division of systems by this boundary seems to us more or less objective and justified.

So, for the purpose of our analysis we distinguish the following types of computing systems:

- top class with $R_{max} > 25\%$ of $Max(R_{max})$;
- high class with $R_{max} > 5\%$ of $Max(R_{max})$ and $R_{max} < 25\%$ of $Max(R_{max})$;

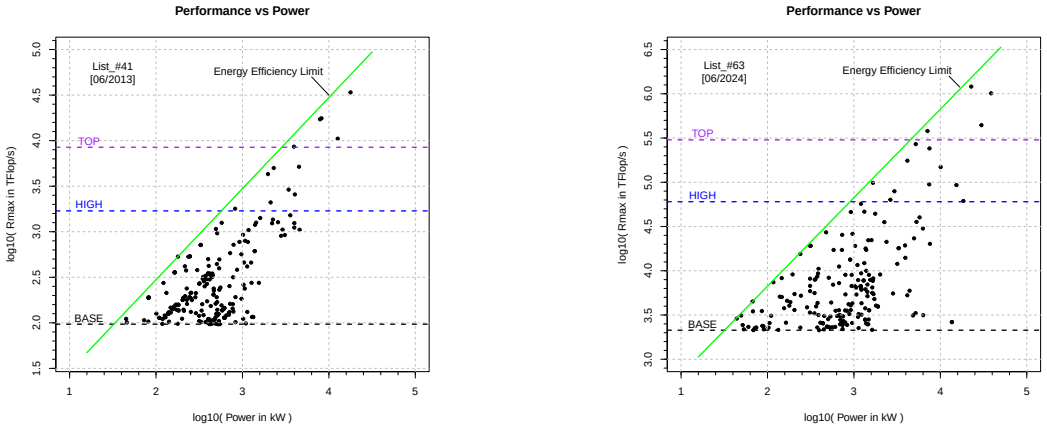
- base class with $R_{max} > 1\%$ of $Max(R_{max})$ and $R_{max} < 5\%$ of $Max(R_{max})$;
- entry class with $R_{max} < 1\%$ of $Max(R_{max})$.

We refer further to these classes as TOP, HIGH, BASE, and ENTRY classes, respectively.

The TOP class usually comprised 2–5 computing systems (see Fig. 2b). In the HIGH class approx. 7–33 computing systems resided. Total number of computing systems in the BASE and ENTRY classes, i. e., with $R_{max} < 5\%$ of $Max(R_{max})$, was never less than 460 during the reviewed period (June 2013 – June 2024), so the BASE and ENTRY classes were never less than 92% of the total number of computing systems in TOP500_List.

The ENTRY class is not only the most widespread class, but also the most diverse one in its design and technical characteristics (see, e. g., Fig. 3). Since the number of computing systems from the ENTRY class reached up to 90% of the total number of systems within the BASE and ENTRY classes, they are described further as one BASE class (as it denoted in Fig. 2b), and the term *entry-level systems* is reserved for computing systems with R_{max} close to R_{max} of Rank_500 systems, which we will denote further as $Min(R_{max})$.

Each of the aforementioned classes has its own statistical characteristics. The TOP and HIGH classes include relatively few computing systems; therefore, to describe their typical properties, it is necessary to use such robust estimators as median. Although the BASE class is the most representative class in terms of the number of computing systems, their properties change over a wide range of values what from a statistical point of view can be interpreted as the presence of "outliers", and the usage of robust estimators like median is also preferable in this case.

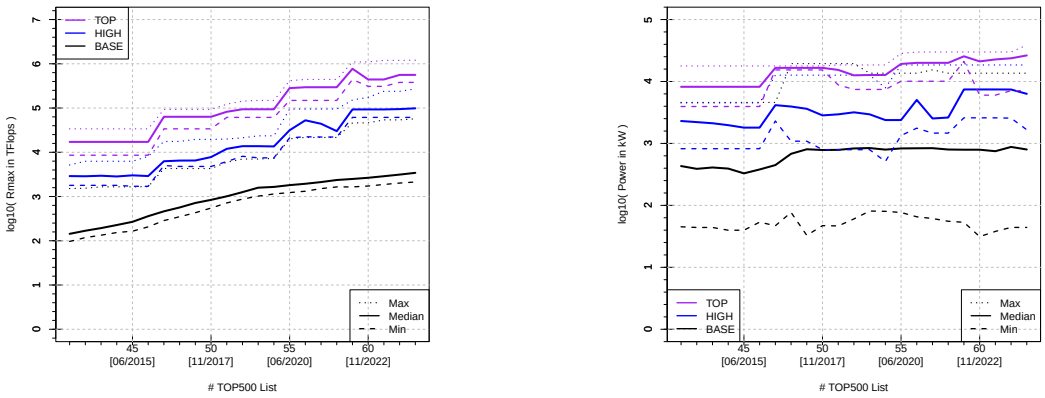


(a) List#41 [June/2013] (b) List#63 [June/2024]

Figure 3. Scatterplots based on energy consumption and maximal performance data for TOP500 systems

Segregation of computing systems from TOP500_List into three classes is observed not only by their maximum (or peak) performance, but also by other extensive characteristics, first of all, by energy consumption and total number of computing cores (see Fig. 4–Fig. 6).

The behavior of intensive quantities defined either as ratio of two different extensive quantities (e. g., energy efficiency) or as fraction (e. g., resource usage efficiency) is more complex. For each TOP500]_List maximum energy efficiency among all systems almost always was attained by not top-level, but basic-level systems (see Fig. 5a); which is understandable because the most

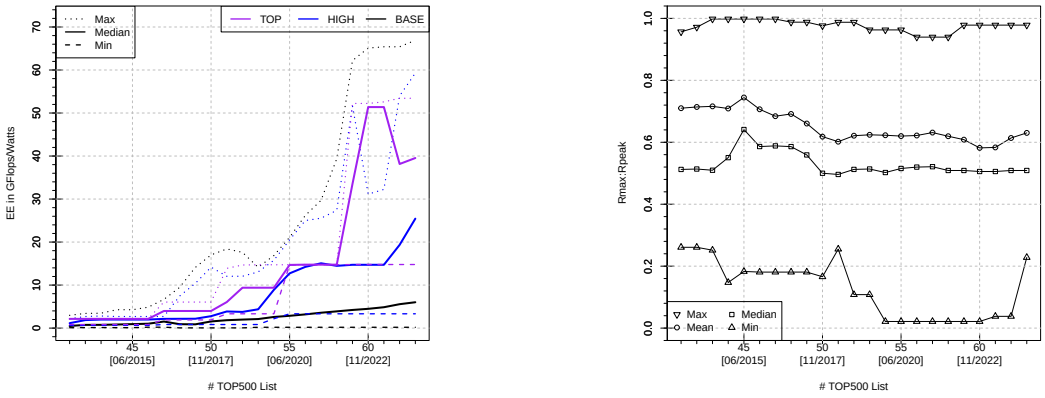


(a) Maximal performance, R_{max} (b) Power, P

Figure 4. Basic characteristics of TOP500 systems
for different performance classes
[Max/Median/Min – maximal/median/minimal within a class]

energy-efficient technologies, due to their novelty, are first tested on relatively small computing systems.

During the observed period the ratio of co-processor cores to total number of cores was approximately the same for all three classes (see Fig. 6b) and was ≈ 0.7 until mid-2016, and then increased more or less abruptly to 0.9. Therefore, it can be assumed that the difference between classes is due to absolute value of total core number and performance per core, rather than a fundamental difference in architecture.



(a) Energy efficiency, EE (b) Resource usage efficiency, RUE

Figure 5. Computational efficiency of TOP500 systems
[Max/Median/Min – maximal/median/minimal within a class]

Similarly, the resource usage efficiency for all three classes has the same distribution with no significant difference between the mean and median values (see Fig. 5b). Possibly it can be explained by the fact that this characteristic depends more strongly on software settings and HPL test running conditions rather than on hardware parameters and specifics. More precisely, the increasing of GPU cores number in computing system hinders their usage efficiency. Indirectly

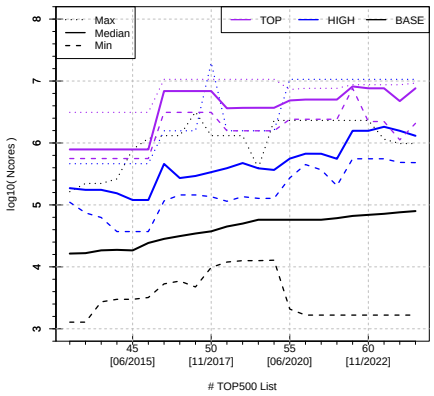
it is confirmed by the fact that starting from Lists#48–52 there was a significant growth in the proportion of co-processor cores used in computing systems accompanied by a fall in the resource usage efficiency (compare Fig. 5b and Fig. 6b).

Considering the evolution of the main extensive characteristics we may conclude that:

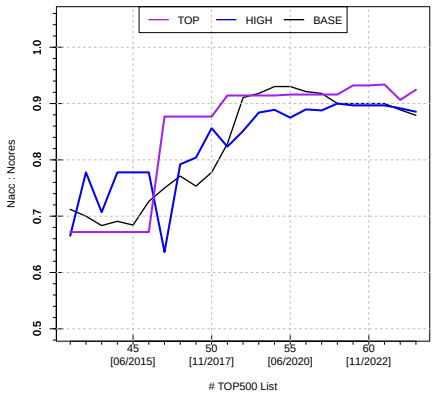
- all characteristics have revealed growth except the minimal energy consumption, $Min(P)$, and energy consumption of BASE class, $P(BASE)$, the behavior of which is more complicated;
- gaps between characteristics of the top and base or entry level systems also have widened during the observation time.

Data given in Tab. 1 and Tab. 2 illustrate these trends. Meanwhile some comments would probably be appropriate.

For the TOP and HIGH classes the average number of cores in computing system correlated with the average performance (see Fig. 6a). However, for the BASE class the dependence between these two characteristics was extremely complicated. Indeed, the maximum number of cores for the base-level systems was comparable to the minimum number of computing cores for the top-level systems, meanwhile the performance between the base-level and top-level systems could be different by 2–3 orders of magnitude (compare Fig. 4a and Fig. 6a).



(a) Total cores number, N_{cores}



(b) Accelerator/Total cores numbers ratio, $N_{accores}/N_{cores}$

Figure 6. Formal performance characteristics of TOP500 systems [Max/Median/Min – maximal/median/minimal within a class]

While the maximum energy consumption has increased steadily over time, the minimum energy consumption could both increase and decrease (see more details later, Fig. 7b). It can be arguably explained as follows. Usually the computing system with the maximum performance has energy consumption near to maximum values, and therefore it sets the value of the maximum energy consumption for the lists during its almost entire service life. Here, the most prominent example are Tianhe-2/Tianhe-2A supercomputers.

Meanwhile, for the computing systems with minimal energy consumption and performance the time of presence in the TOP500_Lists is mainly determined by growth rate of the entry threshold value of R_{max}[5].

[5] On average, energy consumption was not reported for 250–300 systems out of 500, what evidently complicates the analysis; one can assume technical difficulties in the reliable measuring of this parameter but by and large we have no explanation for this fact, because, e. g., List#49 contains data for all systems.

Table 1. Median values of basic characteristic for TOP500 systems

		TOP500_List	
		41 June/2013	**63 June/2024**
Total Cores, N_{cores}	TOP	786 432	7 630 848
	HIGH	186 368	1 305 600
	BASE	16 388	79 524
Maximal Performance, R_{max}, TFlop/s	TOP	17 173.2	561 200
	HIGH	2 897	98 510
	BASE	143.4	3 430
Power, P, kW	TOP	8 209	26 343
	HIGH	2 301	6 316
	BASE	431.3	798.3
Energy Efficiency, EE, GFlop/s per W	TOP	2.14	39.54
	HIGH	1.15	25.44
	BASE	0.49	6.01
$N_{cores}(TOP)/N_{cores}(BASE)$		48	96
$R_{max}(TOP)/R_{max}(BASE)$		120	164
$P(TOP)/P(BASE)$		19	33
$EE(TOP)/EE(BASE)$		4.37	6.58

Table 2. Growth factors of basic characteristics for TOP500 systems between two time snapshots: List#41 [June, 2013] and List#63 [June, 2024]

		Growth
Total Cores, N_{cores}	TOP	9.7
	HIGH	7.0
	BASE	4.8
Maximal Performance, R_{max}, TFlop/s	TOP	32.68
	HIGH	34.0
	BASE	23.92
Power, P, kW	TOP	3.2
	HIGH	2.7
	BASE	1.8
Energy Efficiency, EE, GFlop/s per W	TOP	18.5
	HIGH	22.1
	BASE	12.2
		Gap Growth
$N_{cores}(TOP)/N_{cores}(BASE)$		2.0
$R_{max}(TOP)/R_{max}(BASE)$		1.4
$P(TOP)/P(BASE)$		1.7
$EE(TOP)/EE(BASE)$		1.5

The data in Tab. 1 and Tab. 2 show that computing systems from different classes evolved differently; however, some general patterns common for all classes can be identified. For example, the comparison of growth factors for number of cores in computing system and energy consumption shows that, on average, energy consumption per computing core has decreased: more for the TOP and HIGH class systems and less for the BASE class systems. For example, theoretical growth in $EE(TOP)$ can be roughly estimated as $32.68/3.2 = 10.2$, but empirical growth is 18.5, i. e., 1.8 times more. For $EE(BASE)$ these values are $23.92/1.8 = 13.3$ and 12.2, respectively.

The comparison of the growth in the total number of cores with the growth in performance shows that the greatest increase in performance per one core has occurred in the BASE class, $32.68/9.7 = 3.37$ and $34.0/7.0 = 4.85$ vs. $23.92/4.8 = 4.98$ (see Tab. 2). Complementing this observation with the contrary tendency for energy consumption per computing core, it can be assumed that the greater rate of the growth in the performance per one core for the BASE class systems in comparison with this growth for the TOP class systems was due to the usage of computational cores from the top-level systems of the previous generations in the present base-level systems because, this hardware is usually already available on the market and can be used widely. While the growth in the performance per one core for the top-level systems is possibly connected with the usage of new types of computing cores, which require more time for their development and production.

Thus, it seems that the growth in the total performance was mainly related with the growth of the total number of cores for the TOP/HIGH class systems and with the growth of the performance per core for the BASE class systems.

Studying the behavior of energy efficiency, EE, we can make the following observations (see Fig. 5a):

- maximum energy efficiencies for the computing systems from all three classes were relatively close, which indicates, the simultaneous emergence of the most energy efficient platforms in all three classes (green lines on Fig. 3 and dotted lines on Fig. 5a);
- minimum energy efficiency of the computing systems of a higher performance class was close to average energy efficiency of a lower performance class what probably reflects the natural "aging" of computing systems when less energy-efficient computing systems are replaced by more energy-efficient ones and pass into a lower performance class.

Distribution of resource usage efficiency, RUE, remained almost without significant changes except for the sharp drop in the minimum values in the last few years. This is probably caused by an appearance in the Top500_Lists of computing systems not initially intended for massively parallel calculations, or perhaps due to the fact that HPL test benchmarking was executed not using all computing resources available in such systems (see Fig. 5b).

To summarize, it can be argued that the gap in performance between the TOP and BASE class systems was mainly widening due to the usage in the TOP class systems of large number of increasingly more powerful computing equipment having increasingly greater energy efficiency. No matter how the facts listed above might be trivial, taken together, they show the degree of heterogeneity in the evolution of computing systems from TOP500_Lists.

The trends of the basic characteristics presented above show that the maximum/minimum values of the characteristics have the greatest variability and the tendency to sharp transitions that are difficult to predict, so more reliable information can be obtained by considering the median values. In addition, it should be noted that extrapolation of derived quantities such

as, for example, energy efficiency, EE, has an even lower degree of reliability compared to extrapolation of primary (i. e., measured) characteristics, since derived quantities depend on implementation of not one, but several (at least two) random variables.

On the basis of the presented information, we try to predict the energy consumption of a typical computing system from three main classes in 5 and 10 years, as well as what the lowest value of the energy consumption of a system, the computational performance of which corresponds to the threshold level to enter to TOP500_List.

Our choice of a forecast horizon is based on the Bell's empirical law [8], which states that complete mass replacement of computing technologies occurs within 10 years on average.

The data from the TOP500_Lists on computing performance and energy consumption show that, as a rule, changes of these two quantities had complex abrupt nature, so for their extrapolation we used the simplest linear extrapolation designed first of all to highlight the general global trend.

In case of the TOP or HIGH classes, such approximation seems quite sufficient. In case of our particular interest, i. e., for the data on computational performance and energy consumption of the BASE and ENTRY class systems, it seems reasonable for the sake of more accurate description to take the nonlinear component into account in addition to the global linear trend (see Fig. 7).

Thus, for quantities $R = Max(R_{max}), R_{max}(TOP), R_{max}(HIGH)$, in TFlop/s, the following formula was used:

$$Y = A \cdot X + B, \tag{1}$$

where $Y = \log_{10}(R)$ and X is TOP500_List number, and similarly, for quantities $P = Max(P), P(TOP), P(HIGH)$, in kW, the following formula was used:

$$\tilde{Y} = A \cdot X + B, \tag{1'}$$

where $\tilde{Y} = \log_{10}(P)$ and X is TOP500_List number.

To take into account the nonlinear effects having local nature, a nonlinear term was added to the linear trend (1/1'):

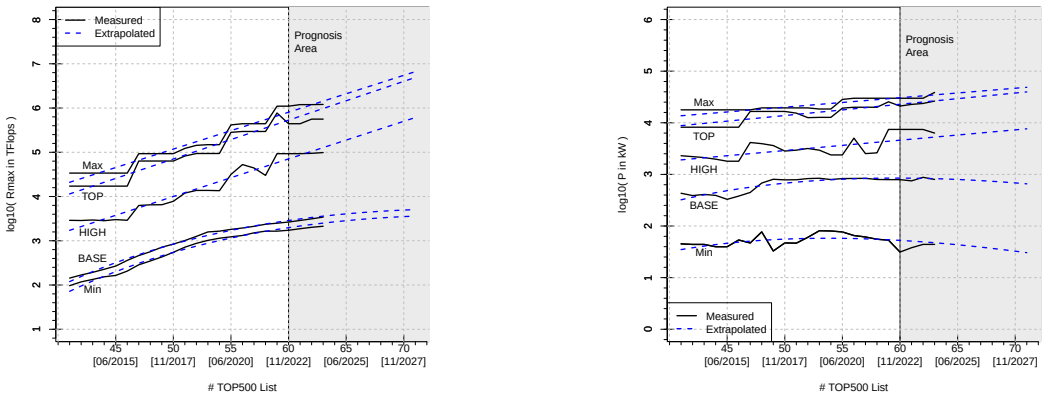
$$Y = C \cdot \log_{10}(X) + D \cdot X + E, \text{ and} \tag{2}$$

$$\tilde{Y} = C \cdot \log_{10}(X) + D \cdot X + E, \text{ respectively.} \tag{2'}$$

We used the least squares method to obtain the coefficient values for all our models with data from Lists#41–60 as training data and data from Lists#61–63 as test data (see Tab. 3).

In Tab. 4 and Tab. 5 the results of predicting the values of R_{max} and P using the formulas (1–2) are given. Figure 8 shows the energy efficiency values calculated as the ratio R_{max}/P, where R_{max} and P were estimated by formulas (1) and (2), along with the corresponding measured values. The dotted lines in Fig. 8 indicate the hypothetical boundaries of energy efficiency (the red line is for the minimum and the green line is for the maximal values) estimated by the following formulae:

$$Max(EE)_{estimated} = \left(f_{pol}(X) \cdot f_{exp}(X)\right)^{1/2},$$
$$f_{pol}(X) = 458.5 - 20.3X + 0.2269X^2,$$
$$f_{exp}(X) = 10^{(0.064X - 2.13)};$$
$$Min(EE)_{estimated} = 0.014X - 0.62, \text{ where } X \text{ is TOP500\_List number.}$$



(a) Maximal performance, R_{max}

(b) Power, P

Figure 7. Measured and extrapolated values for basic characteristics of TOP500 systems

Table 3. The coefficients for models (1/1') and (2/2')

	A	B	C	D	E
$Max(R_{max})$	0.08337	0.90365	—	—	—
$R_{max}(TOP)$	0.08773	0.46187	—	—	—
$R_{max}(HIGH)$	0.08499	−0.24883	—	—	—
$R_{max}(BASE)$	—	—	24.4634	−0.1402	−31.6292
$Min(R_{max})$	—	—	25.2348	−0.1439125	−32.9403375
$Max(P)$	0.01406	3.61809	—	—	—
$P(TOP)$	0.02465	2.89387	—	—	—
$P(HIGH)$	0.01878	2.52791	—	—	—
$P(BASE)$	—	—	15.6505	−0.1141	−18.0546
$Min(P)$	—	—	15.0905	−0.1219405	−17.79493

To derive this formulae, we performed a standard statistical trial-error procedure including:

1. the choice basis function to cope with nonlinearity;
2. the least-square procedure.

The similar behavior of the calculated energy efficiency for all three performance classes can imply the reasonable correctness of the values of R_{max} and P obtained by formulae (1) and (2). Therefore, it can be assumed that model (2) for $Min(R_{max})$ and $Min(P)$ also correctly describes the behavior of R_{max} and P for systems of the entry level. As the main result we could suppose that energy consumption for the computing system with the performance of the TOP500 entry threshold will decrease in the next 5–10 years (see Tab. 5).

Conclusion

1. The emphasis placed in this work on the difference between the top-level and base-level systems made it possible not only to show the heterogeneity of the development of the HPC area, but also served as a methodological basis for identifying the reasons for such heterogeneous development; whereas the knowledge of such causes and relationships between various characteristics allows to build more accurate models to describe their behavior.

Table 4. Measured and predicted by formula (1) and (2) maximal performance for TOP500 systems

TOP500 List	Maximal Performance, R_{max}, TFlop/s				
	Max(R_{max})	**R_{max}(TOP)**	**R_{max}(HIGH)**	**R_{max}(BASE)**	**Min(R_{max})**
	Measured				
61	1 194 000	442 010	93 015	2 878	1 872
62	1 194 000	561 200	94 640	3 131	2 015
63	1 206 000	561 200	98 510	3 430	2 130
	Predicted				
61	975 483.7	650 728.8	86 210.47	3 117.726	2 154.970
62	1 181 925.2	796 397.7	104 845.55	3 360.430	2 332.026
63	1 432 056.0	974 675.4	127 508.75	3 599.048	2 507.113
64	1 735 122	1 192 861	155 070.8	3 830.921	2 678.261
65	2 102 326	1 459 890	188 590.6	4 053.448	2 843.518
70	5 489 718	4 008 390	501 729.9	4 945.058	3 519.391
	Relative Errors, %				
61	−18	47	−7	8	15
62	−1	42	11	7	16
63	19	74	29	5	18

Table 5. Measured and predicted by formula (1) and (2) energy consumption for TOP500 systems

TOP500 List	Power, P, kW				
	Max(P)	**P(TOP)**	**P(HIGH)**	**P(BASE)**	**Min(P)**
	Measured				
61	29 899.2	22 703	7 438	749	38
62	29 899.2	23 695	7 421	877.87	44.07
63	38 698.4	26 343	6 316	798.27	44.07
	Predicted				
61	31 905.1	23 997.7	4 814.93	844.51	51.08
62	33 285.1	25 232.5	5 042.43	837.58	49.30
63	34 724.8	26 530.8	5 280.68	827.33	47.40
64	36 226.8	27 895.9	5 530.19	813.99	45.40
65	37 793.8	29 331.2	5 791.49	797.81	43.32
70	46 705.7	37 694.7	7 295.25	684.08	32.56
	Relative Errors, %				
61	7	6	−35	13	34
62	11	6	−32	−5	12
63	−10	1	−16	4	8

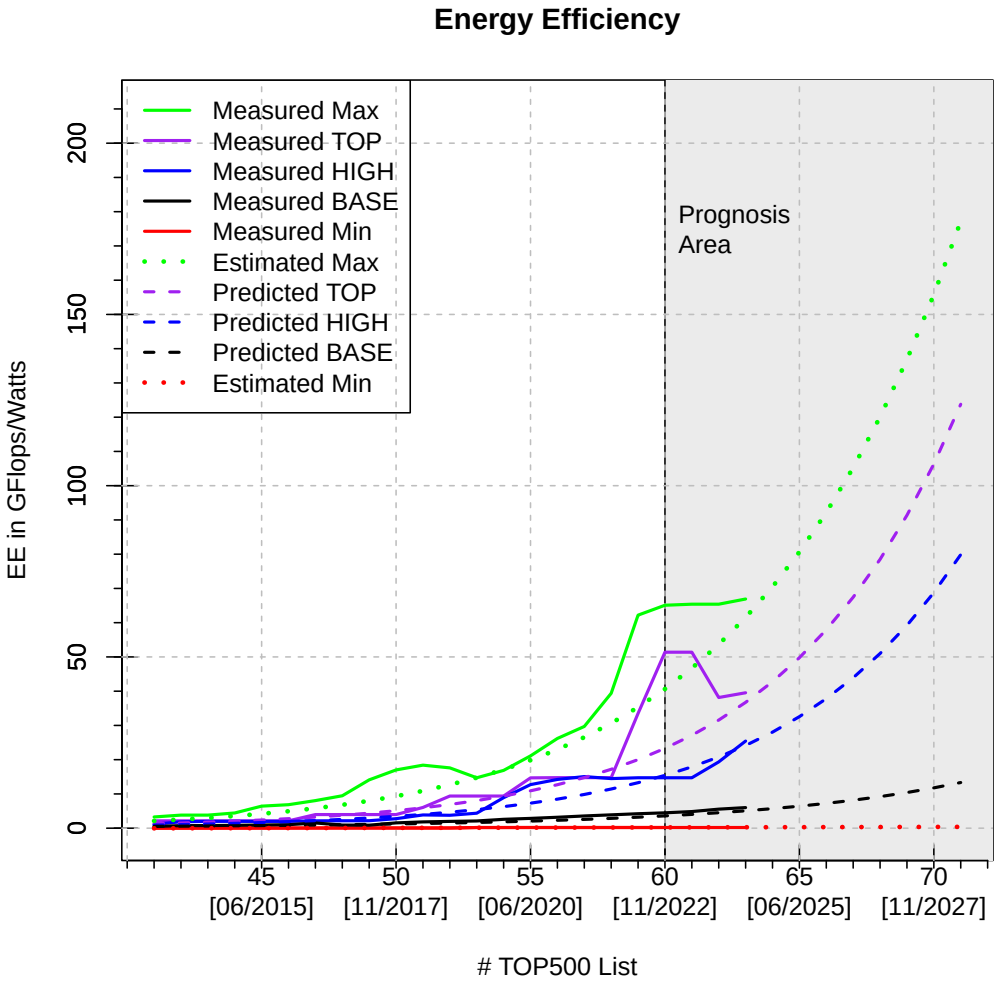


Figure 8. Measured and predicted values along with hypothetical estimations of energy efficiency for computing systems from TOP500_List

2. The proposed procedure for the estimation of energy consumption of Rank_500 system allowed us to make prognosis for the near future. Despite the increase in the threshold value of R_{max} required to include a computing system in TOP500_List, the total cost of the energy entry ticket will decrease due to the rapid growth in the energy efficiency of computing equipment. According to our estimations, it will decrease by 2 times in the next 5–10 years. We expect this trend to continue in the next few years, as adoption of thin-cored, vector and tensor-operation architectures for general numerical simulation. If the trend continues for a few years, we could install Top500 system in facilities such as International Space Station or Antarctic research sites (under 10 kW budget).
3. The decision to limit ourselves to the consideration of only quantitative characteristics, related directly to the functioning of computing systems, allowed us to carry out the analysis at a relatively simple level. However, a more complicated approach based on the inclusion qualitive characteristics describing computing system design (for example, processor/co-processor model, type of interconnect and so on) into consideration can increase the accuracy of the forecast provided that machine learning methods are used that make it possible to uniformly take into account both quantitative and qualitative characteristics of the objects being studied.

Acknowledgements

S.S.K. is especially grateful to Maria S. Timoshik for her incomparable help with the translation of the article.

References

1. Abramov, N.S., Abramov, S.M.: Top500 list overview. Supercomputing Frontiers and Innovations 10(2), 4–17 (2023). https://doi.org/10.14529/jsfi230201

2. Austin, B., Daley, C.S., Doerfler, D., *et al.*: A metric for evaluating supercomputer performance in the era of extreme heterogeneity. In: 2018 IEEE/ACM Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems, PMBS@SC 2018, Dallas, TX, USA, November 12, 2018. pp. 63–71. IEEE (2018). https://doi.org/10.1109/PMBS.2018.8641549

3. Davis, N.E., Robey, R.W., Ferenbaugh, C.R., *et al.*: Paradigmatic shifts for exascale supercomputing. J. Supercomput. 62(2), 1023–1044 (2012). https://doi.org/10.1007/S11227-012-0789-3

4. Khan, A., Sim, H., Vazhkudai, S.S., *et al.*: An analysis of system balance and architectural trends based on Top500 supercomputers. In: Hwang, S., Yeom, H.Y. (eds.) HPC Asia 2021: The International Conference on High Performance Computing in Asia-Pacific Region, Virtual Event, Republic of Korea, January 20–21, 2021. pp. 11–22. ACM (2021). https://doi.org/10.1145/3432261.3432263

5. Kogge, P., Borkar, S., Campbell, D., *et al.*: Exascale computing study: Technology challenges in achieving exascale systems. Defense Advanced Research Projects Agency Information Processing Techniques Office (DARPA IPTO), Techinal Representative 15 (01 2008)

6. Kogge, P.M., Dally, W.J.: Frontier vs the exascale report: Why so long? and are we really there yet? In: IEEE/ACM International Workshop on Performance Modeling, Benchmarking and Simulation of High Performance Computer Systems, PMBS@SC 2022, Dallas, TX, USA, November 13–18, 2022. pp. 26–35. IEEE (2022). https://doi.org/10.1109/PMBS56514.2022.00008

7. Kogge, P.M., Dysart, T.J.: Using the TOP500 to trace and project technology and architecture trends. In: Lathrop, S.A., Costa, J., Kramer, W. (eds.) Conference on High Performance Computing Networking, Storage and Analysis, SC 2011, Seattle, WA, USA, November 12–18, 2011. pp. 28:1–28:11. ACM (2011). https://doi.org/10.1145/2063384.2063421

8. Meuer, H.W.: The TOP500 project: Looking back over 15 years of supercomputing experience. Inform. Spektrum 31(3), 203–222 (2008). https://doi.org/10.1007/S00287-008-0240-6

DOI: 10.14529/jsfi250102

3D Numerical Simulation of Micro-Jet Excitation

Luka S. Volkov[1], ***Aleksandr A. Firsov***[1]

Excitation of round laminar air micro-jet by volumetric force and by pulse-periodic heat source was simulated using the FlowVision software package in 3D formulation at normal conditions. Heat source and volumetric force imitate an influence of electrical discharge. Air jet was formed by a circular cross-section channel with inner size of 1 mm with the Poiseuille velocity profile at inlet boundary, the maximum profile velocity was 5 m/s. The conditions corresponded to the formulation of the problem considered earlier in the experiment. Dependence of large-scale vortex formation from volumetric force frequency and amplitude was obtained. Amplitude of force corresponding to the effect of the discharge on the air jet was determined and was set to 5 μN. For round laminar jet the exited oscillations of the jet was obtained at frequency range 500–2500 Hz, further increase in the frequency of oscillations left the shape of the jet close to the initial. Lack of influence of pulse-periodic heat source on flow structure was discussed. The results obtained demonstrate that the main contribution from the corona discharge to the jet in the experiment is provided by the volumetric force (ionic wind), and not by the heating of the gas.

Keywords: micro-jet, excitation, discharge, instability, numerical simulation.

Introduction

Studies of the stability of laminar jet flows and the mechanisms of their instability and excitation are of significant interest for various applications, including burners and combustion chambers of various types, chemical reactors, and medical devices [1]. For example, micro-jets are used to control the flow pattern and structure of a large diameter jet [2]. A significant part of investigations is dedicated to the so-called micro-jets with a diameter of no more than 1 mm, the excitation of which by volumetric force and heat source will be investigated in this work. The flow in micro-jets differs significantly from macrojets, the conditions for vortex formation change, and the penetration depth increases [3]. Similar studies of the stability of micro-jets are carried out not only in gases, but also in liquids; thus, in work [4] the transition of a laminar jet to newtonian turbulence, elasto-inertial filaments, and elasto-inertial turbulence is described, depending on the addition of polymer.

The velocity profile in the nozzle influences the position of laminar-turbulent transition and/or excitation of various types of instabilities. The closer the profile is to the Poiseuille (parabolic), the further from the nozzle the transition and/or excitation of various types of instabilities occurs. The parabolic profile forms due to the growth of the boundary layer. Therefore, to obtain extended laminar jets, the diameter d of the gas channel should be much smaller than its length L ($L/d > 100$). In the practical applications [5, 6] both the stabilization of laminar jets and the early excitation of instabilities are of interest. The excitation leads to increased jet mixing with the surrounding gas (moving or stationary) and intensification of combustion.

The classical method of jet excitation is to use acoustic vibrations [7]; however, in most works the use of a plane wave is discussed, i.e., non-local influence [8–10]. Thus, in [11] the bifurcation of a propane micro-jet under external acoustic influence and an increase in flame stability are shown. In addition to the excitation of instability or bifurcation of the jet, acoustic radiation contributes to reducing NOx concentration during the combustion of a laminar hydrogen jet in

[1]Joint Institute for High Temperatures of the Russian Academy of Sciences (JIHT RAS), 125412, Moscow, Russia

the air [12]. However, in recent works the local acoustic influence is also considered – but it is discussed in relation to turbulent jets, for example, in [13, 14] the excitation of a turbulent jet is investigated using local input of acoustic radiation at the nozzle edge.

Electrical discharges of various types can provide local effects: form shock waves, be a source of volumetric force and heat generation. For example, spark discharges have been considered in various settings as an actuator that intensifies the mixing of a transverse gas jet with a supersonic air flow, both experimentally [15] and using numerical modeling [16, 17]. In [18], the possibility of excitation and turbulence of a laminar jet using a corona discharge was experimentally investigated; bifurcation of the jet with a rapid transition to turbulence was shown by analogy with works on acoustic excitation [11]. Analysis of the literature showed that it is noteworthy that the stability of micro-jets and the possibility of their excitation using acoustic vibrations through numerical modeling were practically not considered, with rare exceptions [19] where the instability of a plane (two-dimensional) jet was considered. At the same time, such works are known for turbulent jets, for example, in work [20] the instability and bifurcation of a turbulent jet caused by a pulsed micro-jet impact located on the nozzle edge were considered using DNS modeling. In our previous work, the fundamental possibility of modeling the excitation of a flat (two-dimensional) microjet using an acoustic impact or a heat source in FlowVision was considered, and positive results were obtained [21]. FlowVision is a commercial heavy industrial CFD code that is used both for fundamental research and for solving applied problems [22, 23]. Therefore, it looks relevant to consider, using 3D numerical modeling by FlowVision, the stability of the round laminar microjet and the possibility of its local excitation using a pulse-periodic force or energy input simulating a corona discharge under the conditions corresponding to experimental research [18].

The article is organized as follows. Section 1 is devoted to description of a simulation model and boundary conditions. In Section 2 we discuss the amplitude of volumetric force at which the results obtained in the simulation will coincide well with the experiment on the effect of a corona discharge on an air jet. Section 3 represents the results of the simulation for fixed jet velocity 5 m/s at varying the frequency of the discharge. And Section 4 demonstrates the lack of influence of pulse-periodic heat source on flow structure. Conclusion summarizes the study and reveals that the main contribution from the corona discharge to the jet in the experiment is provided by the volumetric force (ionic wind), and not by the heating of the gas.

1. Model and Boundary Conditions

Obtaining detailed information on the distribution of gas-dynamic quantities is significantly difficult in an experiment on excitation of a laminar micro-jet using an electric discharge. Numerical modeling can be used to obtain information about pressure and velocity fields throughout the volume of interest. Previous attempt to describe the influence of discharge by numerical simulation was performed in a 2D formulation [21] to evaluate the capabilities of the software and due to limited computing resources. Having a license of FlowVision for computing on 64 cores and a PC based on AMD Ryzen 9 3900X (12 cores), equipped with 64 GB of memory, as well as the possibility to use the nodes of the “Fischer” supercomputer of JIHT RAS based on AMD Epyc 7301 / AMD Epyc 7551 (16 / 32 cores), it was decided to perform three-dimensional modeling of the excitation of a circular laminar air jet by discharge.

Numerical modeling of the excitation of a jet by the operation of a discharge was performed using the FlowVision 3.14 software package. The simulation is based on solving a three-

dimensional unsteady system of Navier–Stokes equations with second-order central difference scheme of accuracy. The turbulence model was disabled, and taking into account fairly large cell size and schematic viscosity, the simulation performed is closer to LES than to DNS.

The calculation area is a cylinder with a length (along the stream) of 60 mm and a diameter of 50 mm and is presented in Fig. 1. There is a tube 5 mm long with an internal diameter of 1 mm and an external diameter of 1.2 mm located along the x-axis on one of the boundaries. At the distance of 3 mm from the edge of the tube there is an area of volumetric force or volumetric heat, simulating the operation of a discharge. The project used a rectangular computational mesh (Fig. 2) with adaptation: an increased level of adaptation (refinement of the computational mesh) was used in the area of air jet flow. The cells near the boundary conditions have a complex shape because they are cut off by geometry. Thus, inside the tube there are 46 cells per channel diameter, and the total number of cells in the project did not exceed 5 million. The simulation required the use of a small time step ($CFL < 1$) and it was set to 2 μs.

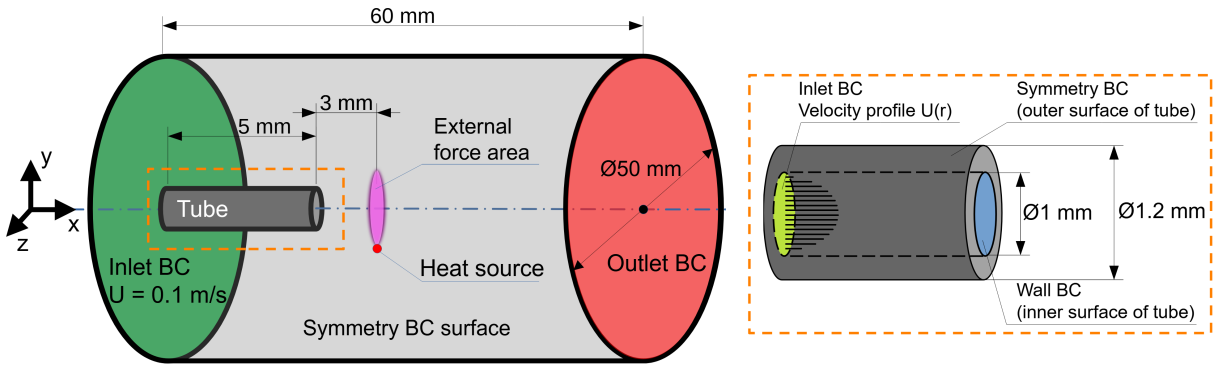


Figure 1. Calculation domain and boundary conditions

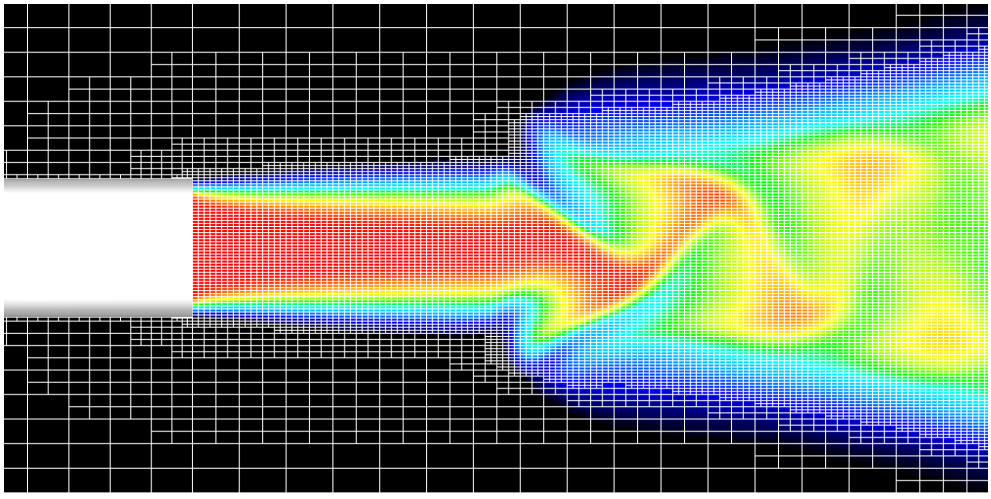


Figure 2. Computational grid on the background of jet passive scalar

The "wall" boundary condition (no-slip boundary) was used on the inner wall of the tube and on the edge; a Poiseuille velocity profile was installed at the entrance of the tube; the maximum profile speed U_0 was set to 5.0 m/s. At the entrance to the computational domain from the side of the tube, a constant velocity boundary condition of 0.1 m/s was set to minimize the mutual influence of pulsations created by the discharge. The working gas was air. In all

calculations, to accurately determine the jet passive scalar, two identical substances were used, both having the properties of air: "Air1" as jet and "Air2" as ambient medium. Jet passive scalar is then defined as $Y = [Air1]/([Air1] + [Air2])$, where [X] is mass concentration of X. This made it possible to visualize the jet similarly to visualization using smoke/spray in the experiment.

On the outer walls of the tube, as well as along the curved surface of the cylindrical computational domain, a symmetry condition was established (a surface with flow slip and zero gradient of all values). At the remaining boundary opposite the tube, zero relative total pressure was established (i.e., the total pressure was equal to the reference pressure). The relative temperature at all permeable boundaries was set to zero. The reference temperature and pressure were set to 273 K and 101000 Pa, respectively. Thus, the simulation parameters corresponded closely to the setup described in the experimental work [18].

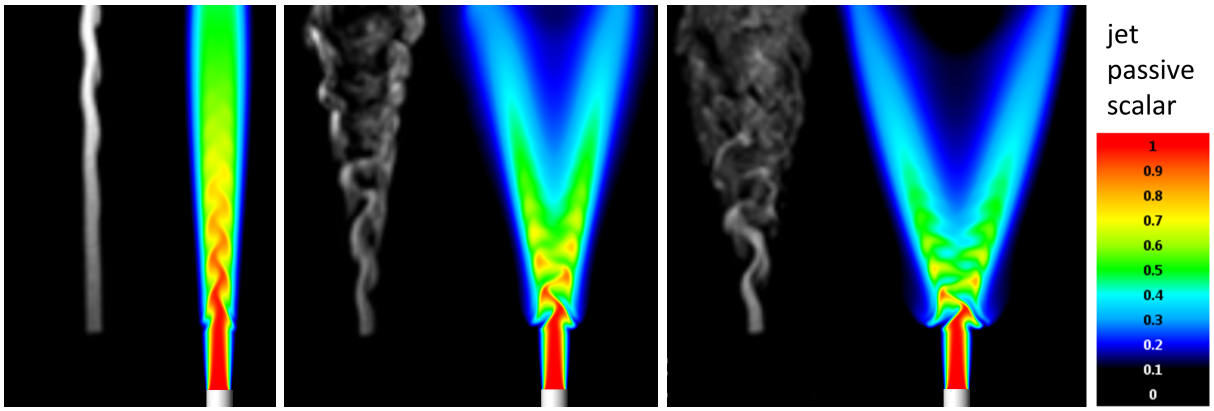
2. Choosing the Amplitude of Volumetric Force

It is well known that a corona discharge causes an ion wind (a directed gas jet) and is also a weak source of heat. To simulate the effect of a corona discharge on a laminar jet by volumetric oscillating force, it is necessary to know the amplitude of the force. For example, in the work [24], the volumetric force characteristic of a dielectric barrier discharge is indicated as a 2–5 μN and the corona discharge is close to it in type [25]. In accordance with these data, the amplitude of the integral force was tested in the range of 1–10 μN. The external force was applied to an elliptical shape region marked in Fig. 1, the chosen shape of the volume force region approximately corresponds to the effect observed in the experiments [24, 25].

In this paper, the dependence of force on time is assumed as sinusoidal. However, it should be noted that this a simplification. Due to the peculiarities of the corona discharge, in reality the force acting on the jet is expected to depend on time in a more complex way. Nevertheless, in this study, it was decided to limit ourselves to the effect of a sinusoidal external force, i.e., corresponding to a narrow spectrum in Fourier space.

The data from [18] were used as a basis for determining the amplitude of the external force: the photographs of a seeded laminar jet excited by a corona discharge. In [18], photographs of the jet are given for several values of the amplitude of the voltage in the discharge at a fixed frequency of 1500 Hz, but data on the volumetric force are missing. For several values of the voltage in the simulation, the corresponding amplitude of the external sinusoidal force was selected.

The comparison of the experiment and the simulation was carried out as follows. The divergence angle of the jet streams was measured. According to the experimental images, this angle was determined by a clearly visible boundary between the dark and seeded areas. In the simulation, straight lines were drawn through areas with a given value of the passive scalar (substance fraction) of the jet to determine the angle. By comparing the divergence angle in the simulation and in the experiment, the corresponding values of the external force amplitude in the simulation were selected for several values of the discharge voltage. Figure 3 shows the photographs of the jet from the article [18] and the results of the computer simulation in a single scale.



(a) 3.6 kV, 2 μN (b) 3.8 kV, 5 μN (c) 4.5 kV, 7.5 μN

Figure 3. Comparison of experiment (with different crown discharge voltage) and simulations (with different force amplitude). The experiment is shown in grayscale (spray visualization) and the simulation is shown as a color scale

3. Results for Fixed Jet Velocity at Varying the Frequency of the Discharge

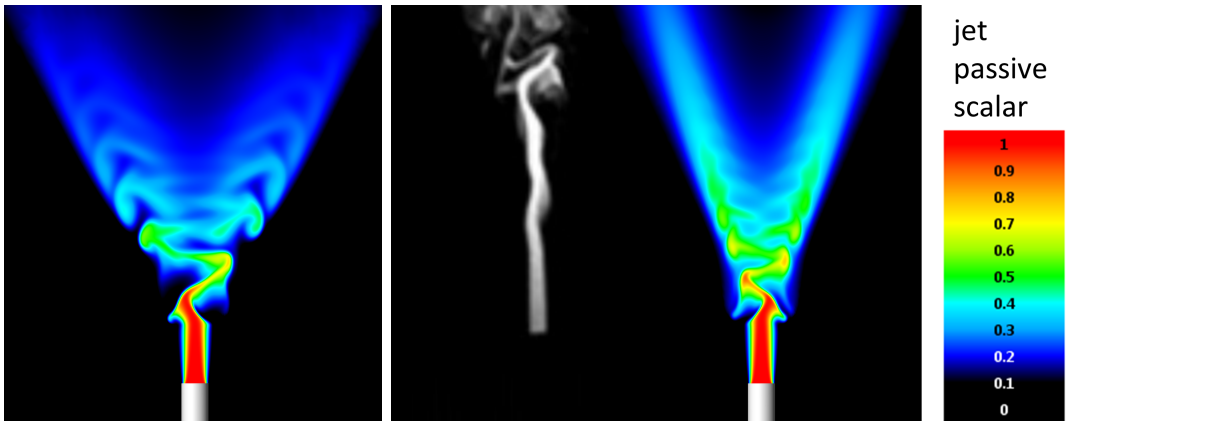
In this part of the research, the amplitude of the force $F = 5\ \mu N$ was selected. It corresponds to the amplitude of the voltage in the discharge of 3.8 kV, because at this voltage the dependence of the flow configuration on frequency was studied in the experiment. The jet was modeled under the action of a periodic external force with a frequency of 500 to 2500 Hz. The goal of this part of the work was a qualitative observation of the jet response to action with different frequencies. A comparison of the obtained distributions of the passive scalar of the jet with the images from the article [18] for several frequencies is shown in Fig. 4.

At the highest frequency of 2500 Hz, visible oscillations similar to those observed in the experiment are formed in the jet. But the jet in the simulated section has only one stream, i.e., in each cross-section of the jet there is only one local maximum of the jet passive scalar.

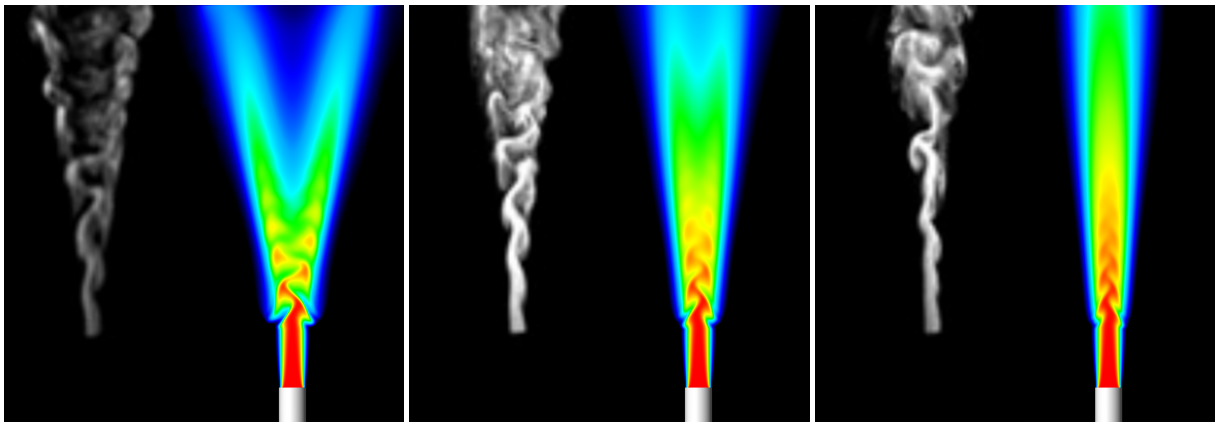
At lower frequencies (this is especially pronounced for 1500 Hz), the jet splits into two streams in the plane formed by the initial velocity vector and external force vector. The jet splitting point is arbitrary, since the passive scalar on the jet axis decreases smoothly. However, this qualitative transition occurs closer to the injector in cases with lower frequencies.

For a frequency of 1000 Hz, a significant discrepancy between the simulation and experimental results begins. This can be explained by the fact that in reality a time dependence of the force differs from the sinusoidal one, and at low frequencies this difference is most pronounced. At a low frequency of the external force, 500 Hz, the jet does not just break up not into two branches, but splits into separate fragments drifting along the flow.

Next, the structure of the jet is studied in detail using the example of one particular case (1500 Hz, 5 μN), and the processes that determine the mixing of the jet with the external flow are analyzed. Figure 5 shows a three-dimensional visualization of the jet structure for the case of 1500 Hz. The lines on two planes mark the position of the jet boundary, drawn at the level of 50% of the jet passive scalar. The ellipsoid marks a position of the area with a volumetric force. The force vector is parallel to y-axis. The jet has almost no oscillations in the direction perpendicular to the force vector. The disturbances caused by the external force are most pronounced in the x-y plane.



(a) 500 Hz, Sr = 0.1 (b) 1000 Hz, Sr = 0.2



(c) 1000 Hz, Sr = 0.3 (d) 1000 Hz, Sr = 0.4 (e) 1000 Hz, Sr = 0.5

Figure 4. Jet passive scalar distribution. For (a) no experimental data provided

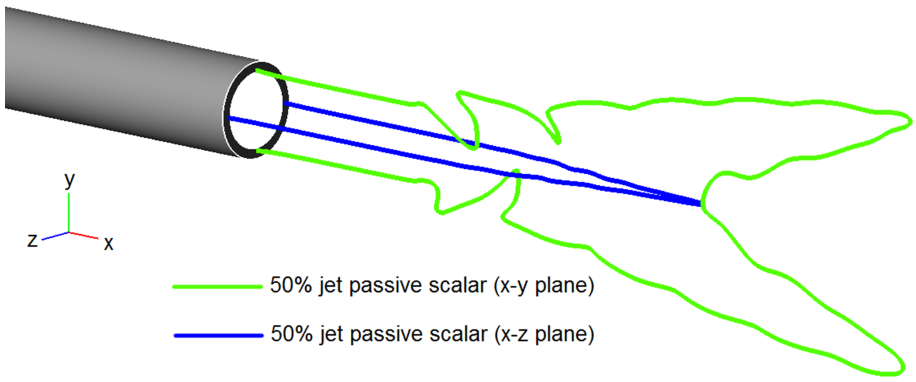


Figure 5. 3D jet visualization: 50% jet passive scalar contours in two planes

The streamlines in the jet and the co-current flow surrounding it are shown in Fig. 6. As follows from the image, the gas surrounding the jet is partially drawn into the area of the force action and the surrounding volume. Thus, mutual penetration of the jet and flow matter occurs, which enhances kinematic mixing.

The layers located far from the jet (at a distance of about 2 injector diameters) bend around the jet without mixing with it. Thus, the jet and the adjacent layers of the external flow form a domain within which the kinematic mixing of the layers occurs. However, the layers of this domain do not undergo kinematic mixing with the layers of the surrounding flow. At a distance

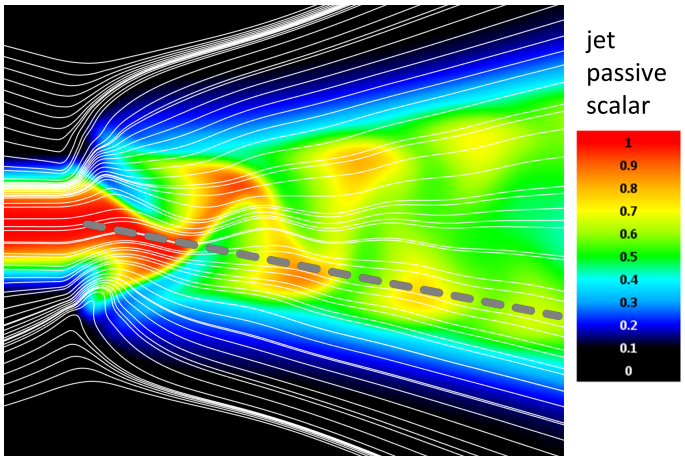
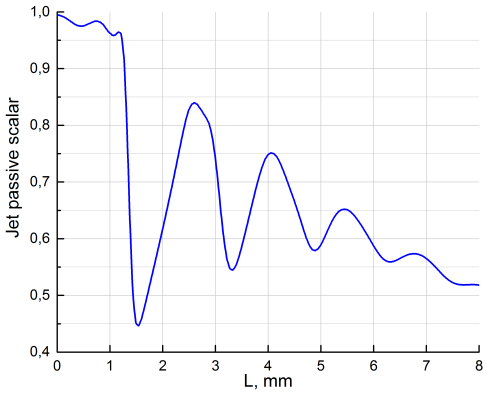


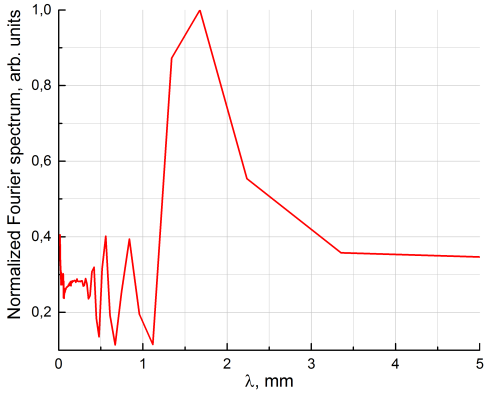
Figure 6. Streamlines in jet and surrounding stream and jet passive scalar

from the point of action of the force (about 5 jet diameters), the streamlines become almost straight. This means that with further propagation of the jet, mixing is caused mainly by molecular processes (diffusion).

The dashed line in Fig. 6 denotes a line directed along one of the two jet streams. The passive scalar profile along this line is shown in Fig. 7a. This profile illustrates how the jet substance mixes with the ambient air: both in the region of curved streamlines and in the region where the streamlines are straight. As follows from the figure, alternating portions of the jet and portions of external air gradually mix, causing the oscillations of the passive scalar to dump. At the same time, a general trend towards a decrease in the passive scalar is observed due to the divergence of the jet.



(a) Profile along a dotted line in Fig. 6



(b) Normalized Fourier transform of a part of the profile

Figure 7. Jet passive scalar distribution

On the right in Fig. 7b the Fourier spectrum obtained using the FFT is presented for the right part of the passive scalar graph (for the right stream). From this spectrum, one can determine the characteristic spatial period λ of the jet structure. In the work [18], a similar (with minor differences) analysis of the spatial frequency of disturbances was carried out based on the experiment for a frequency of 1000 Hz. The wavelength (1.5 ± 0.3) mm obtained in the simulation agrees well with the experiment.

The jet and the external force can be quantitatively compared by the momentum per unit of mass. For jet it is just the velocity $P_{jet} = U_0 = 5\ m/s$. The force momentum for the half period is $P_{force} = F/(\pi\rho f V) \sim 3\ m/s$, where $F = 5\ \mu N$ is the amplitude of volumetric force, $V = 4.6 \cdot 10^{-10}\ m^3$ is the volume of the region to which the force is applied, $\rho = 1.28\ kg/m^3$ is mass density, and $f \sim 1000\ Hz$ is frequency. These quantities, P_{jet} and P_{force}, are of the same order. This is in good agreement with the results of an experimental study of corona discharge [18] – the speed of the jet initiated by the corona is a few m/s with a width of 2 mm.

4. Results for 3D Simulation of Micro-Jet Excitation by Heat Source

Another possible mechanism for jet excitation is the pulse-periodic heating due to the discharge and the resulting pressure pulsations. To test the degree of influence of the thermal effect of the corona discharge on the jet using pulsed local heating, a case with a frequency of 1 kHz was selected. Volumetric heat release was activated in the area marked in red in Fig. 1. The periods of the presence and absence of heating alternated with a duty cycle of 50% (0.5 ms of heating and 0.5 ms of no heating). During the heating periods, the released power was 20 mW, while the maximum temperature rose to 320 degrees Celsius, which approximately corresponds to the average gas temperature in the corona discharge area. The pressure in the area near the pulsed heating site experienced jumps of up to 0.5 Pa at the moments of switching the heating on and off, but at other times the pressure changed insignificantly. Time dependencies of temperature and pressure in the area of discharge are presented in Fig. 8.

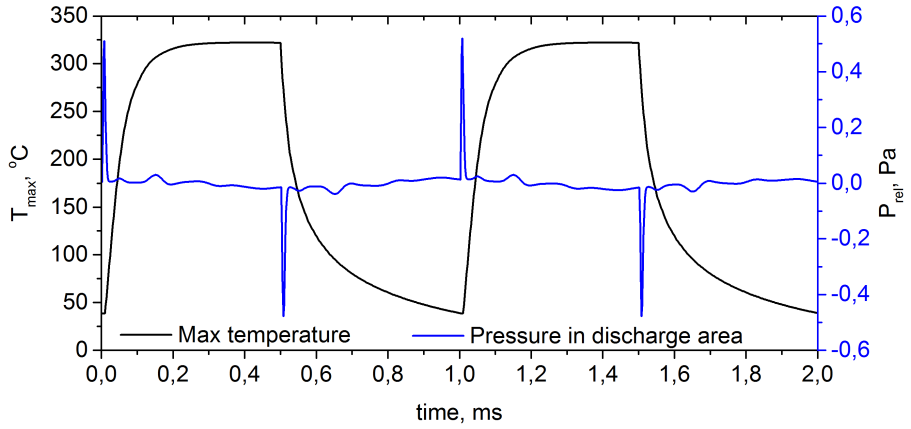
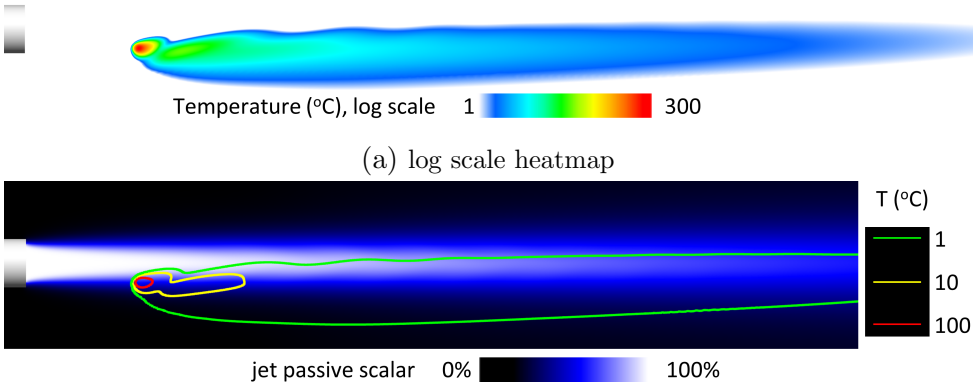


Figure 8. Time dependencies of temperature and pressure caused by pulse-periodic heat source operation at frequency 1000 Hz

The temperature distribution in the jet under the influence of pulse-periodic heating is shown in Fig. 9 on a logarithmic scale and in the form of contours against the background of the distribution of the passive scalar of the jet. Judging by the bends of the heated gas region, oscillations are observed in the jet, but their amplitude is insufficient to make the jet diverge (at least in the simulated section).

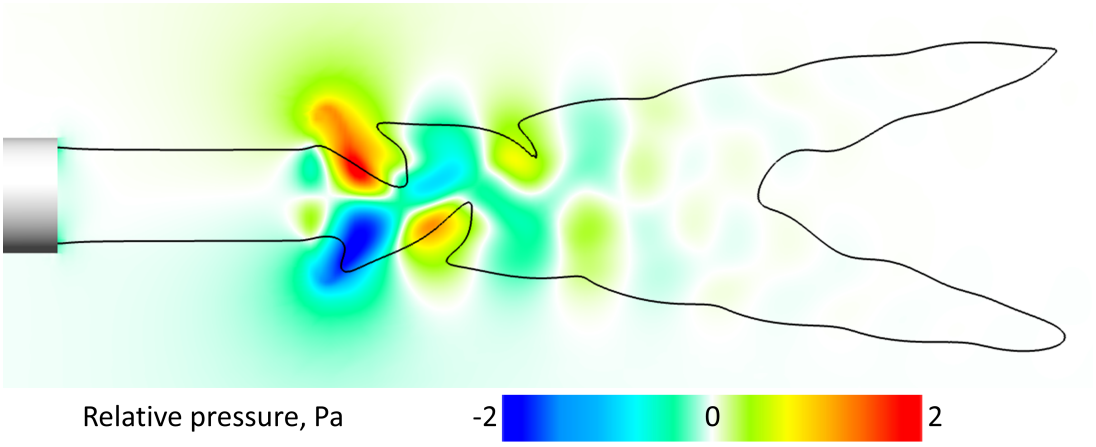
Figure 10 allows us to compare the pressure distribution in the jet in the cases of the action of a periodic volumetric force (a) and a periodic heat source (b). The pressure differences excited by local heating are an order of magnitude smaller than the pressure differences arising under the action of an external force.



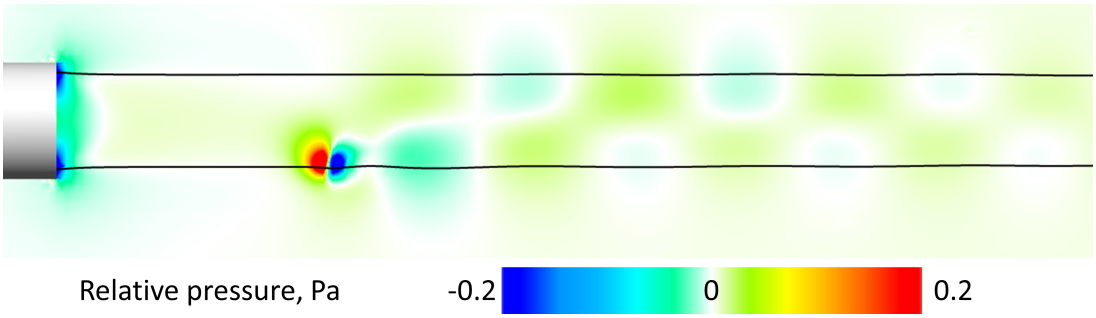
(a) log scale heatmap

(b) lines on the background of jet passive scalar

Figure 9. Distributions of temperature in case of pulsed heating



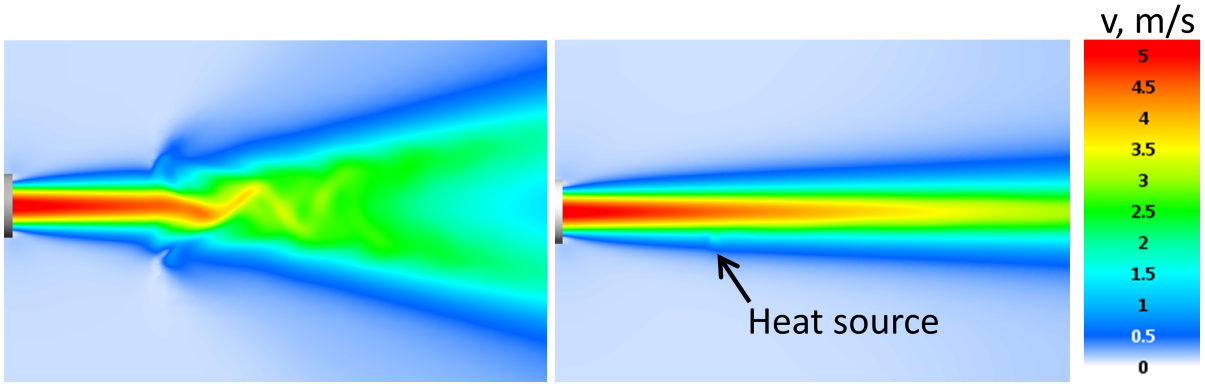
(a) volumetric force



(b) heat source

Figure 10. Pressure distribution in cases of jet excitation by volumetric force and heat source

The velocity distributions in the jet for the two indicated mechanisms of jet excitation are shown in Fig. 11. For the case of periodic volumetric force, the gas velocity drops rapidly: at a distance of about three jet diameters from the point of action, it no longer exceeds half the initial value. And in the case of periodic heat action, the jet retains its velocity longer, and the heat source introduces only a barely noticeable distortion into the velocity distribution.



(a) volumetric force (b) heat source

Figure 11. Velocity distribution in cases of jet excitation by volumetric force and heat source

Conclusion

The volumetric force and pulse-periodic heat source excitation of a round laminar air micro-jet at normal conditions were simulated using the FlowVision software package. The jet diameter was 1 mm, velocity had Poiseuille profile with the maximum of about 5 m/s. For this case, the effect of two mechanisms that could be responsible for the excitation of the jet affected by corona discharge was considered: a volumetric oscillating force with an amplitude of 5 μN and a pulse-periodic heat release with a power of 20 mW.

The numerical modeling shows that the volumetric force (ion wind) is the main mechanism responsible for the excitation of jet oscillations under the influence of a corona discharge. Another supposed mechanism of the discharge action – volumetric heat release – turned out to play no significant role in the effect on the jet.

For several values of the discharge voltage in the experiment from the article [18], the amplitude of the external sinusoidal force was selected, which in computer modeling leads to effects similar to those observed in the experiment when a corona discharge affects a laminar jet. The dependence of the jet shape on the frequency of the external force was investigated. It is shown that, depending on the frequency of the force, the following can occur in the jet:

- perturbations that do not significantly affect the shape of the jet, and partial mutual penetration of the outer layers of the jet and the adjacent layers of the surrounding air (2500 Hz, 2000 Hz) can occur;
- the jet may bifurcate, i.e., split into two streams (1500 Hz, 1000 Hz);
- the jet may break up into separate fragments (500 Hz).

The structure of the jet in case of bifurcation was considered in detail, the structure of the jet streams during bifurcation was described, the main mixing mechanisms and the place of their dominance were indicated. The obtained results of modeling when simulating a discharge using a volumetric force are in satisfactory agreement with the experimental results.

Acknowledgements

This work was performed within the State Assignment No. 075-00270-24-00 by the Ministry of Science and Higher Education of the Russian Federation.

References

1. Reynolds, W.C., Parekh, D.E., Juvet, P.J.D., *et al.*: Bifurcating And Blooming Jets. Annual Review of Fluid Mechanics 35(1), 295–315 (2003). https://doi.org/10.1146/annurev.fluid.35.101101.161128

2. Perumal, A.K., Zhou, Y.: Axisymmetric jet manipulation using multiple unsteady minijets. Physics of Fluids 33(6), 065124 (2021). https://doi.org/10.1063/5.0052275

3. Gau, C., Shen, C.H., Wang, Z.B.: Peculiar phenomenon of micro-free-jet flow. Physics of Fluids 21(9), 092001 (2009). https://doi.org/10.1063/1.3224012

4. Yamani, S., Keshavarz, B., Raj, Y., *et al.*: Spectral Universality of Elastoinertial Turbulence. Physical Review Letters 127(7), 074501 (2021). https://doi.org/10.1103/PhysRevLett.127.074501

5. Yoshida, H., Koda, M., Ooishi, Y., *et al.*: Super-Mixing Combustion Enhanced by Resonance between Micro-Shear Layer and Acoustic Excitation. Int J Heat Fluid Flow 22(3), 372–379 (2001). https://doi.org/10.1016/S0142-727X(01)00103-5

6. Krivokorytov, M.S., Golub, V.V., Volodin, V.V.: The effect of acoustic oscillations on diffusion combustion of methane. Technical Physics Letters 38(5), 478–480 (2012). https://doi.org/10.1134/S1063785012050240

7. Kozlov, G.V., Grek, G.R., Sorokin, A.M., *et al.*: Influence of initial conditions at the nozzle exit on the structure of round jet. Thermophysics and Aeromechanics 15(1), 55 (2008). https://doi.org/10.1134/S0869864308010046

8. Kozlov, V.V., Grek, G.R., Litvinenko, Y.A., *et al.*: Subsonic Round and Plane Jets in the Transversal Acoustic Field. Siberian Journal of Physics 5(2), 28–42 (2010). https://doi.org/10.54362/1818-7919-2010-5-2-28-42

9. Krivokorytov, M.S., Golub, V.V., Moralev, I.A.: The evolution of instabilities in gas microjets under acoustic action. Technical Physics Letters 39(9), 814–817 (2013). https://doi.org/10.1134/S1063785013090186

10. Kozlov, V.V., Grek, G.R., Dovgal, A.V., *et al.*: Stability of Subsonic Jet Flows. Journal of Flow Control, Measurement & Visualization 01(03), 94–101 (2013). https://doi.org/10.4236/jfcmv.2013.13012

11. Kozlov, V.V., Grek, G.R., Korobeinichev, O.P., *et al.*: Combustion of hydrogen in round and plane microjets in transverse acoustic field at small Reynolds numbers as compared to propane combustion in the same conditions (Part I). International Journal of Hydrogen Energy 41(44), 20231–20239 (2016). https://doi.org/10.1016/j.ijhydene.2016.07.276

12. Oh, J., Heo, P., Yoon, Y.: Acoustic Excitation Effect on NOx Reduction and Flame Stability in a Lifted Non-Premixed Turbulent Hydrogen Jet with Coaxial Air. International Journal of Hydrogen Energy 34(18), 7851–7861 (2009). https://doi.org/10.1016/j.ijhydene.2009.07.050

13. Reumschüssel, J.M., Li, Y., Nedden, P.M., *et al.*: Experimental jet control with Bayesian optimization and persistent data topology. Physics of Fluids 36(9), 095164 (2024). https://doi.org/10.1063/5.0217519

14. Nedden, P.M., Nissen, T., Reumschüssel, J.M., *et al.*: Surrogate-based modeling for investigating and controlling the response of an axisymmetric jet to harmonic forcing. AIAA SCITECH 2024 Forum. https://doi.org/10.2514/6.2024-0462

15. Firsov, A., Savelkin, K.V., Yarantsev, D.A., *et al.*: Plasma-enhanced mixing and flameholding in supersonic flow. Philos. Trans. R. Soc. A 373(2048), 20140337 (2015). https://doi.org/10.1098/rsta.2014.0337

16. Dolgov, E.V., Kolosov, N.S., Firsov, A.A.: The study of the discharge influence on mixing of gaseous fuel jet with the supersonic air flow. Computer Research and Modeling 11(5), 849–860 (2019). https://doi.org/10.20537/2076-7633-2019-11-5-849-860

17. Volkov, L.S., Firsov, A.A.: Modeling the influence of repetitively pulsed heating on the formation of perturbations at the boundary of a transverse jet in a supersonic crossflow. Computer Research and Modeling 15(4), 845–860 (2023). https://doi.org/10.20537/2076-7633-2023-15-4-845-860

18. Yarantsev, D.A., Moralev, I.A.: Forcing microjet instabilities by corona discharge. Vestnik Obedinennogo Instituta Vysokih Temperatur 12, 4–8 (2023). (in Russian) https://doi.org/10.33849/2023401

19. Shevchenko, A.K., Yakovenko, S.N.: Simulation of Instability Development in a Plane Submerged Jet at Low Reynolds Numbers. Siberian Journal of Physics 13(4), 35–45 (2018). https://doi.org/10.25205/2541-9447-2018-13-4-35-45

20. Eri, Q., Ding, W., Kong, D., *et al.*: Numerical Investigation of Jet Control Using Two Pulsed Jets under Different Amplitudes. Energies 15(2), 640 (2022). https://doi.org/10.3390/en15020640

21. Firsov, A.A.: 2D Simulation of Micro-Jet Excitation by Heat Source. Supercomputing Frontiers and Innovations 10(4), 12–20 (2023). https://doi.org/10.14529/jsfi230402

22. Aksenov, A.A.: FlowVision: Industrial computational fluid dynamics. Computer Research and Modeling 9(1), 5–20 (2017). https://doi.org/10.20537/2076-7633-2017-9-5-20

23. Aksenov, A.A., Zhluktov, S.V., Kashirin, V.S., *et al.*: Three-dimensional Numerical Model of Kerosene Evaporation in Gas Turbine Combustors. Supercomputing Frontiers and Innovations 10(4), 27–45 (2024). https://doi.org/10.14529/jsfi230404

24. Moralev, I., Bityurin V., Firsov A., *et al.*: Localized micro-discharges group dielectric barrier discharge vortex generators: Disturbances source for active transition control. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 234(1) (2020). https://doi.org/10.1177/0954410018796988

25. Moreau, E., Audier, P., Benard, N.: Ionic wind produced by positive and negative corona discharges in air. Journal of Electrostatics 93, 85–96 (2018). https://doi.org/10.1016/j.elstat.2018.03.009

DOI: 10.14529/jsfi250103

Numerical Simulation of the Growth of Localized Disturbances in a Supersonic Boundary Layer over a Plate with Longitudinal Slots

Aleksey A. Yatskikh[1], ***Vladimir I. Lysenko***[1], ***Boris V. Smorodsky***[1], ***Leonid V. Afanasev***[1]

This paper presents the results of numerical simulations of the growth of small amplitude localized disturbances in the boundary layer on a flat plate and on a plate with rectangular longitudinal slots at Mach number 2. The simulations were performed with the FlowVision software under flow conditions corresponding to those in the T-325 wind tunnel of the Khristianovich Institute of Theoretical and Applied Mechanics SB RAS. Cases with slots no deeper than the boundary layer thickness were considered. The localized disturbances in the boundary layer were generated by a time-pulsed, spatially localized heat supply from the plate surface. For a smooth plate, the growth of the disturbances was compared with the results of the linear stability theory. Data agreement was achieved for the grid resolution used. For the cases of the smooth surface and the plate with longitudinal slots, the evolution of disturbances in the boundary layer was analyzed in physical and wave space. In case of the smooth plate, the disturbance introduced into the boundary layer increases monotonically downstream. However, in the presence of longitudinal slots, the growth of the disturbance growth is different and depends on the depth of the slots. It has been shown that longitudinal slots can affect the stability of the supersonic boundary layer. A frequency-wavenumber analysis of the disturbance evolution revealed that longitudinal slots shift the range of the most unstable waves towards a region of higher frequencies compared to the smooth plate. This effect becomes more pronounced with increasing depth of the slots.

Keywords: numerical simulation, localized disturbances, supersonic boundary layer,laminar-turbulent transition control, surface microrelief, rectangular slots.

Introduction

The laminar-turbulent transition of high-speed boundary layers leads to a significant increase in viscous friction and heat fluxes, which can lead to serious limitations in the performance of thermal protection systems for high-speed vehicles. At low levels of disturbance, turbulence in the flow is caused by the occurrence and growth of various unstable waves in the boundary layer. These waves interact with each other and can cause the flow to become turbulent [2, 8].

One possible method for controlling the boundary layer laminar-turbulent transition is through the microprofiling of the surface of a streamlined model (porous coatings, slots, grooves, etc). The advantage of microprofiling is the minimal effect on the main flow, while it is possible to influence the stability of the boundary layer. This approach is successfully applied at low subsonic flow velocities to control the laminar-turbulent transition [3, 6, 7, 13, 14].

At high flow velocities, this approach is also used to control the flow in boundary layers. Recently, special attention has been paid to using spatially periodic slots on the surface of streamlined bodies to control the position of the laminar-turbulent transition. This method has proven effective in stabilizing disturbances of the second (acoustic) mode of instability. A review of these studies can be found in [11].

In [5], the influence of small grooves on the surface on the laminar-turbulent transition of a supersonic boundary layer of a swept wing was studied using numerical simulation and linear stability theory. In these studies, the grooves on the surface were located parallel to the leading

[1]Khristianovich Institute of Theoretical and Applied Mechanics SB RAS, Novosibirsk, Russian Federation

edge of the wing. It was shown that such a modification of the surface affects the growth of cross-flow instability modes typical for three-dimensional boundary layers.

At low supersonic flow velocities in two-dimensional boundary layers, the main role in laminar-turbulent transition is played by the growth of the first mode of disturbances (Tollmien–Schlichting waves). The paper [18] considers the possibility of controlling the stability of a supersonic boundary layer on a flat plate using indentations on the surface located across the incident flow. Both calculations using linear stability theory and numerical simulation use only two-dimensional equations. While calculations using linear theory succeeded in stabilizing disturbances of the first modes, more accurate calculations with Navier–Stokes equations (taking into account recirculation flow inside slots and alternating expansion/compression waves induced at groove edges, which were ignored in calculations with linear theory) failed to stabilize the first mode. It should be noted that only transverse slots were considered in the work (with an orientation angle of 90 relative to the incident flow), while different orientations of the slots are of interest.

In experiments in the low-noise wind tunnel T-325 at ITAM SB RAS, described in [11], a stabilization of the development of disturbances in the boundary layer of a plate with longitudinal grooves (slots) at Mach number $M = 2$ was discovered. In the experiments, the effect of longitudinal slots periodically arranged in the direction transverse to the flow was considered. In this case, the depth of the slots was comparable to the thickness of the boundary layer, and the period in the transverse direction was 5–8 times smaller than the characteristic wavelength of the most unstable disturbances. It was found that in the presence of depressions, the natural disturbances of the boundary layer grow significantly slower than in case of a smooth plate.

A well-developed method for studying the evolution of disturbances in the boundary layer is needed to study in detail the influence of longitudinal slots on the laminar-turbulent transition in supersonic boundary layers. Most information about the stability of the boundary layer can be obtained by analyzing the development of disturbances with a wide range of frequencies and wavenumber spectra. In this context, the development of a controlled wave packet has been considered, which has been widely used in the study of laminar-turbulent transitions at supersonic speeds through both numerical simulations [4, 10, 12, 15] and experiments [9, 16, 17].

This paper presents the formulation and results of numerical modeling of the evolution of spatially and temporally localized small amplitude disturbances in the boundary layer of a smooth plate and a plate with longitudinal rectangular slots at Mach number $M = 2$. The first part of the paper describes the formulation of the problem. The second part verifies the results of the non-stationary perturbation evolution for the smooth plate – comparisons of the computational results with those obtained using linear stability theory are given. The third part presents numerical results. The spatio-temporal structure and integral characteristics of the disturbance growth are analyzed and a frequency-wavenumber analysis is performed. Finally, the main conclusions of this work are presented.

1. Simulation Set-Up

The calculations are similar to the experiments [11] carried out in the low-noise supersonic wind tunnel T-325 of the ITAM SB RAS: Mach number $M = 2$, unit Reynolds number of the flow $Re_1 = U_\infty/\nu_\infty = 6 \cdot 10^6\ m^{-1}$. The flow is described within the framework of a continuum model for a compressible, viscous, and heat-conducting gas (air). Calculations were carried out in a three-dimensional formulation using the FlowVision software package. The numerical

implementation in this study utilizes the FlowVision's second-order accurate reconstruction method for convective fluxes at cell interfaces, based on a monotonic upwind scheme. Temporal integration employs an explicit scheme for convective-diffusive terms, while the pressure Poisson equation is solved implicitly [1].

Figure 1 schematically shows the calculations set-up. The length of the computational domain along the flow direction (x-axis) is 130 mm, the width along the z-axis is 20.4 mm. The lower boundary of the computational domain corresponds to the plate with the slots. From the leading edge, which corresponds to the input boundary of the computational domain, to $x = 53$ mm ($Re_x = Re_1 \cdot x = 318000$) there is a flat section of the plate. Downstream there are slots that are aligned along the incident flow and arranged periodically in the transverse direction across the entire width of the computational domain. The width of the slots is 0.6 mm, and the period of their arrangement along z is 1.2 mm. Three cases are studied in this paper: a smooth plate, a plate with slots of $h = 0.18$ and 0.5 mm depth ($Re_h = Re_1 \cdot h = 1080$ and 3000). The depth of the slots is comparable to the thickness of the boundary layer. When constructing the computational domain, it was taken into account that the compression wave from the leading edge and the rarefaction waves from the slots fall on the output boundaries of the computational domain and are not reflected into the boundary layer. Figure 1 shows the pressure distribution in the xy-plane.

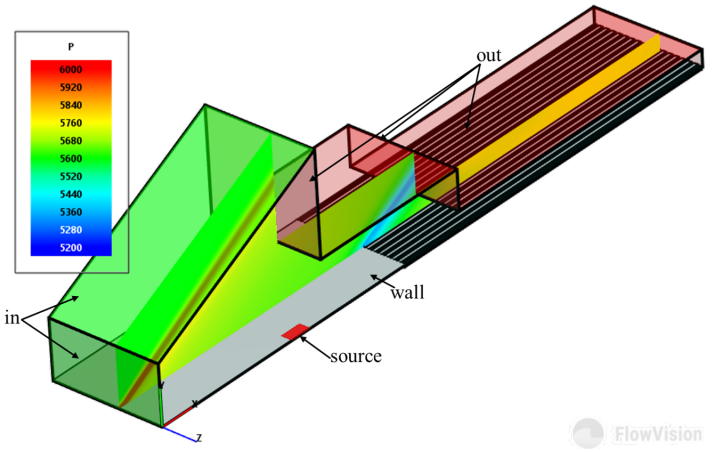


Figure 1. Computational domain

The problem was solved in a symmetric formulation relative to the xy-plane at $z = 0$. When analyzing the results, the numerical data were supplemented by mirroring along z relative to $z = 0$.

The following parameters of the incoming flow were defined at the input boundaries of the computational domain: Velocity along x 501 m/s, flow temperature 156 K, pressure 5590 Pa. These flow parameters correspond to the Mach number $M = 2$ and the unit Reynolds number $Re_1 = 6 \cdot 10^6\ m^{-1}$. The following conditions were set at the output boundaries: Temperature gradient zero, for the velocities and pressure at the boundary, values equal to the value at the center of the boundary cell are specified. This boundary condition is known as the supersonic outlet and is built into FlowVision. The symmetry condition was set on the side boundaries (in the yz-plane). The no-slip boundary condition and the zero heat flux condition were set on the slotted plate. To introduce controlled disturbances in the boundary layer, a source region was

selected on the smooth section of the plate; the disturbances were generated by a non-zero heat flux in the source region for a limited time.

The structure of the computational grid on which the main studies were carried out is shown in Fig. 2. The grid was refined along y in the plate region in order to achieve a high resolution of the flow in the boundary layer. In the boundary layer region, the cell size was 0.01 mm, and there were at least 60 cells per boundary layer thickness in the region of the main flow analysis. Along the flow, the main grid step was 0.5 mm. Preliminary calculations showed that it was additionally necessary to refine the grid along the x-coordinate in the region of the start of the slots. Along the z-coordinate, the cell size was set to 0.05 mm, so that there were 24 cells per slot period. It should be noted that for the conditions of modeling in the boundary layer on a smooth plate according to linear stability theory, the wavelength of the most unstable disturbances is 6–12 mm. Studies on the convergence of the steady-state solution on the grid showed that the grid used gives an accurate mean flow. Ultimately, the number of cells was about 20 million and varied depending on the depth of the slots.

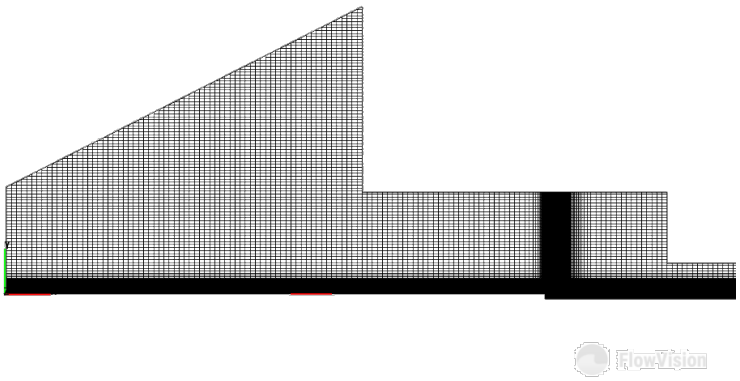


Figure 2. Computational grid

The calculation was carried out in two stages: at the first stage the steady flow was calculated, at the second stage the non-stationary calculation with the introduction of disturbances into the boundary layer. Artificial disturbances were generated into the boundary layer by changing the boundary conditions in a localized area on the plate surface (source in Fig. 1). On the plate surface, the condition for the heat flux q_w was defined at time zero for time t_s in the source region:

$$q_w(x,z,t) = q_0 \cdot \sin(\pi\frac{t}{t_s}) \cdot \cos^3(\frac{\pi}{2} \cdot \frac{x-x_s}{r_s}) \cdot \cos^3(\frac{\pi}{2} \cdot \frac{z-z_s}{r_s}), \quad (1)$$

where $x_s = 30$ mm ($Re_{x_s} = 180000$) and $z_s = 0$ are the coordinates of the center of the source, $r_s = 2$ mm is the size of the source, $t_s = 25\ \mu s$ is the duration of the impact in the boundary layer, $q_0 = 5000\ W/m^2$ is the amplitude. This condition was set in the region $(x_s \pm r_s, z_s \pm r_s)$ for the time t_s. After the time t_s, the condition for the heat flux zero was restored. The location of the source and the operating time were chosen so that they could be implemented in the experiment. The amplitude of the impact was chosen so that the evolution of the excited disturbances in the boundary layer in the investigation area was linear, but the amplitude of the disturbances was clearly above the level of the numerical pulsations.

During the calculations, the instantaneous flow parameters in the zy-sections were recorded with a step in the longitudinal direction of 5 mm in the range $x = 40$–120 mm ($Re_x = 240000$–720000). The recording was done every 500 ns, and the total time of "flight" of the disturbance from the source did not exceed 500 μs.

The calculations were carried out using the FlowVision software package on the ITAM SB RAS ("Mechanics" Shared-Use Centre) cluster. Two computing nodes were used. Each node was equipped with two 64-core AMD EPYC 7763 processors and 512 GB RAM. The computation of the second stage (non-stationary problem) for each case took approximately one week.

In order to analyze the development of local disturbances by sources, pulsations of the longitudinal component of the mass flow are considered, which are normalized to the value in the oncoming flow:

$$m'(x, y, z, t) = \frac{(\rho U(x, y, z, t) - \rho U(x, y, z, 0))}{\rho U_\infty} \cdot 100\%, \tag{2}$$

where ρU_∞ is the mass flow in the oncoming flow.

In order to determine the influence of slots on the growth of a localized disturbance in the boundary layer in physical space, the evolution of the dispersion of mass flow pulsations was calculated using the data obtained, calculated in the yz planes and normalized to the value in the initial section ($x_0 = 40$ mm):

$$S_m(x) = \sqrt{\frac{\sum_{y,z,t} m'(x, y, z, t)^2}{\sum_{y,z,t} m'(x_0, y, z, t)^2}}. \tag{3}$$

The article deals with the downstream development of small amplitude disturbances at Reynolds numbers corresponding to the early stage of the laminar-turbulent transition. The early stages of the transition are described by linear stability theory. Therefore, the use of the wave approach in the data analysis is necessary to determine the influence of slots on the stability of the boundary layer. To determine the frequency-wavenumber characteristics of the disturbance evolution, the Fourier transform was carried out in time and space:

$$m_{f\beta} = \frac{\sqrt{2}}{T \cdot Z_0} \sum_{z,t} m'(z, t) \cdot \exp(i2\pi f t - i\beta z) \Delta t \Delta z, \tag{4}$$

their $T = 500$ μs, $Z_0 = 1$ mm are normalization coefficients.

The amplitudes and phases were determined for waves with different frequencies and wavenumbers. The amplitude was defined as the modulus of the Fourier transform and the phase as an argument. To determine the dispersion relation, the longitudinal wavenumber α_r was also estimated:

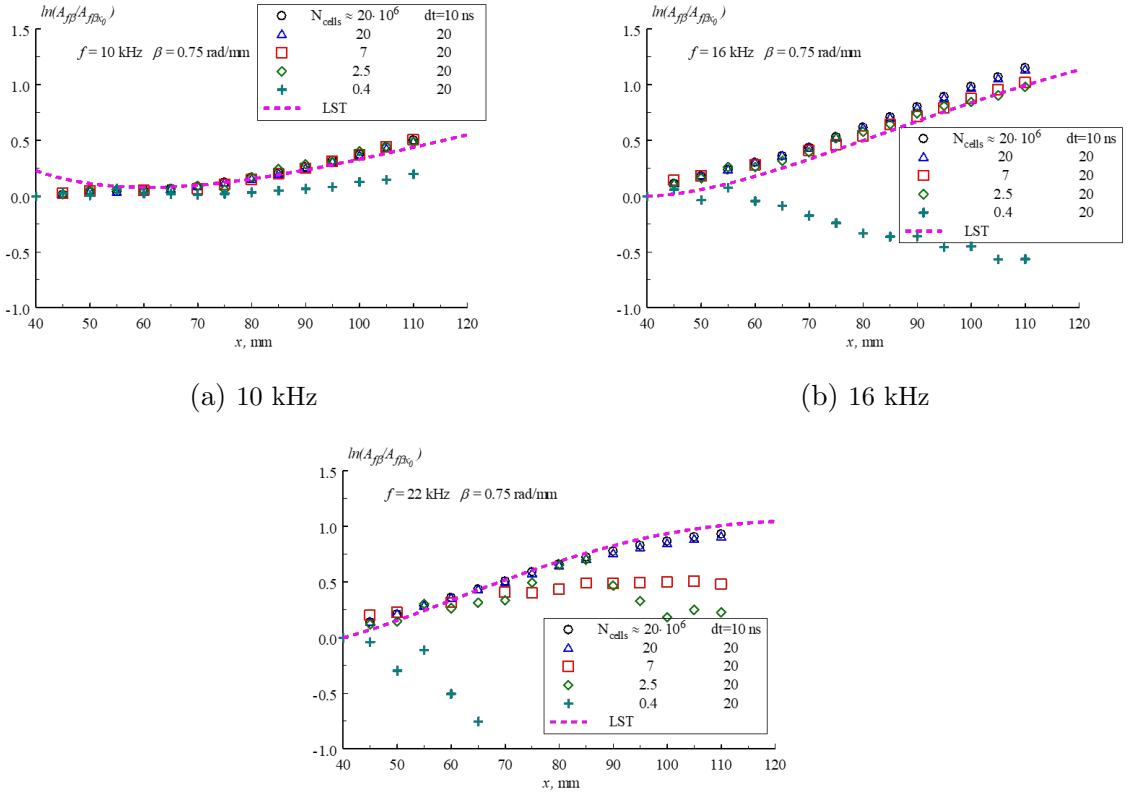
$$A_{f\beta} = |m_{f\beta}|,\ \phi = \arg(m_{f\beta}),\ \alpha_r = \frac{d\phi}{dx}. \tag{5}$$

2. Verification of Calculations

The numerical simulation of disturbance evolution in the boundary layer requires a careful approach. The studies performed with different computational settings have shown that both the computational grid and the time step can have a significant impact on the growth of the disturbances. Moreover, for a correct modeling of the perturbation evolution in the early stage

of the transition, it is necessary that the amplitude of the introduced perturbations is small and their evolution is consistent with the linear stability theory.

In this work, the disturbance growth is compared with the results of the linear stability theory in order to verify and optimize the calculation method. For a laminar two-dimensional supersonic Blasius boundary layer, the calculations of the disturbance growth are performed within the framework of linear stability theory (LST). From the numerical simulation data for the case of the smooth plate, the disturbance amplitudes are determined for different values of the longitudinal coordinate. Waves with a wavenumber corresponding to the most growing waves are considered. Figure 3 shows the results for the frequencies $f = 10$, 16 and 22 kHz and the wavenumber of $\beta = 0.75$ rad/mm. The calculations were carried out with different grid resolutions, whereby the grid structure remained approximately the same (Fig. 2).



(a) 10 kHz

(b) 16 kHz

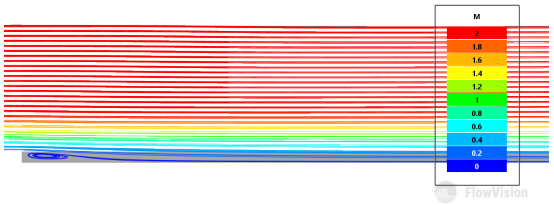
(c) 22 kHz

Figure 3. Comparison of the results of the numerical simulation of disturbance growth with the results of linear stability theory

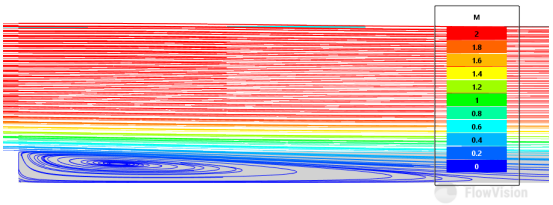
Comparison of the numerical simulation results with the linear stability theory shows that the grid resolution plays a significant role, especially for high-frequency disturbances. The observed discrepancy in disturbance growth rates at high frequencies for coarse grids results from an insufficient number of cells per wavelength. The selected time step of 20 ns gives a reliable result, and its reduction does not change the result. For a flat plate, numerical simulation in the above-described formulation gives a result close to the calculations according to the linear theory. This indicates the correctness of the problem statement and the reliability of the results.

3. Influence of Longitudinal Slots on the Development of Disturbances in the Boundary Layer

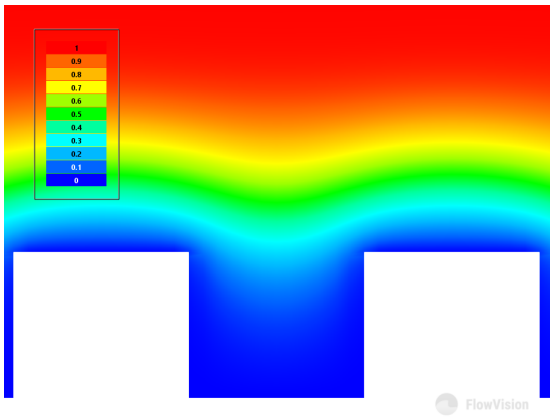
Figure 4 illustrates the features of the mean flow in the boundary layer over a plate with longitudinal slots. Figures 4a and 4b display streamlines in the xy-plane of the slot near its initiation point for slot depths of $h = 0.18$ and 0.5 mm, respectively. The color of the lines corresponds to the flow Mach number. Figure 4c presents the streamwise velocity field, normalized by the freestream velocity, in the yz-plane at $x = 80$ mm for the case of slots depth $h = 0.5$ mm.



(a) Streamlines in the slot. $h = 0.18$ mm



(b) Streamlines in the slot. $h = 0.5$ mm



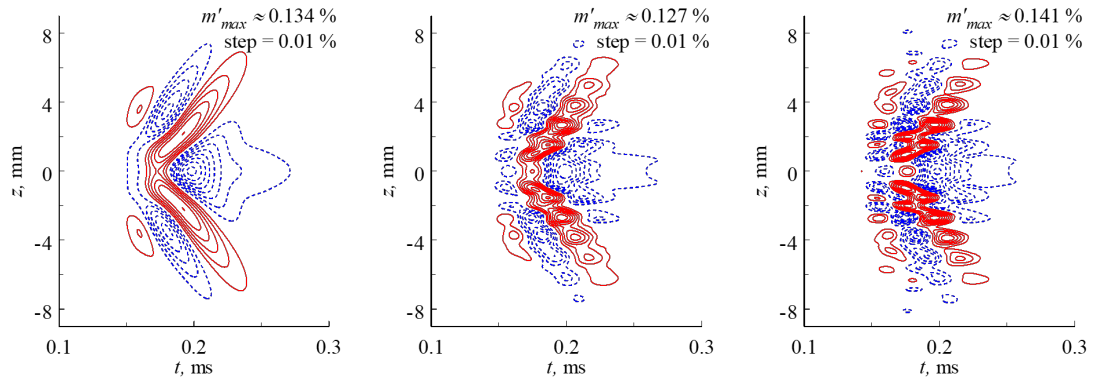
(c) Normalized streamwise velocity field. $x = 80$ mm, $h = 0.5$ mm

Figure 4. Mean flow in the boundary layer over a plate with longitudinal slots

Near the leading edge of the slots, within the recesses, a local flow separation is observed. The streamwise extent of this separation grows with increasing slot depth but does not exceed $15 \cdot h$. Further downstream, the flow inside the slots remains attached. The flow velocity within the slots is subsonic. Transverse modulation of the boundary layer flow is evident, while the outer flow remains uniform. It can also be seen that the slot depth and width are comparable to the boundary layer thickness at this location.

Figure 5 shows the isolines of the mass flow pulsations in the zt-plane at $x = 80$ mm for the cases of a smooth plate and a plate with slots of depth 0.18 and 0.5 mm. The data are shown for the values of the y-coordinate at which the pulsation amplitude is maximum. Note that in the boundary layer above the slots, the position of the maximum disturbance level shifts closer to the plate compared to the case of the smooth surface.

In the region under consideration, the disturbance from the source has an amplitude of about 0.1%. In the presence of slots on the plate surface, the disturbance is modulated in the transverse direction. The spatial extent and structure of the disturbances generally do not change



(a) Smooth plate (b) $h = 0.18$ mm (c) $h = 0.5$ mm

Figure 5. Isolines of mass flow pulsations in the zt-plane in the boundary layer region with maximum disturbance amplitude at $x = 80$ mm

significantly. The propagation speed of the disturbances is the same in case of the smooth surface and slots.

Figure 6 shows the growth of the disturbances downstream. The dependencies of the value of S_m calculated with formula (3) are shown for the cases of the smooth surface and the plate with slots of different depths. In Fig. 6, the coordinate of the slot beginnings on the plate is marked by a vertical black dashed line.

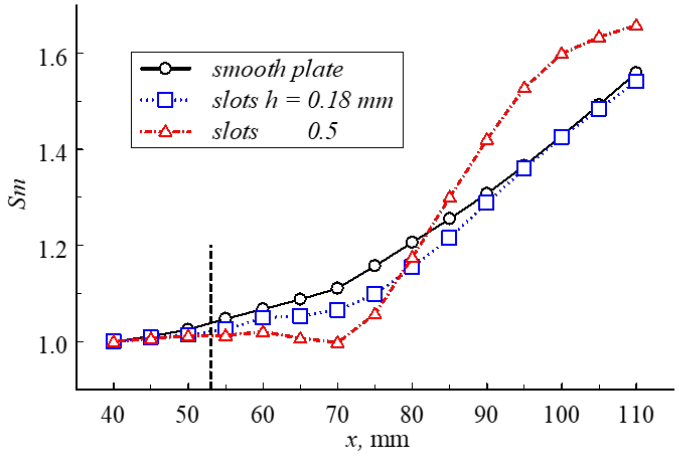
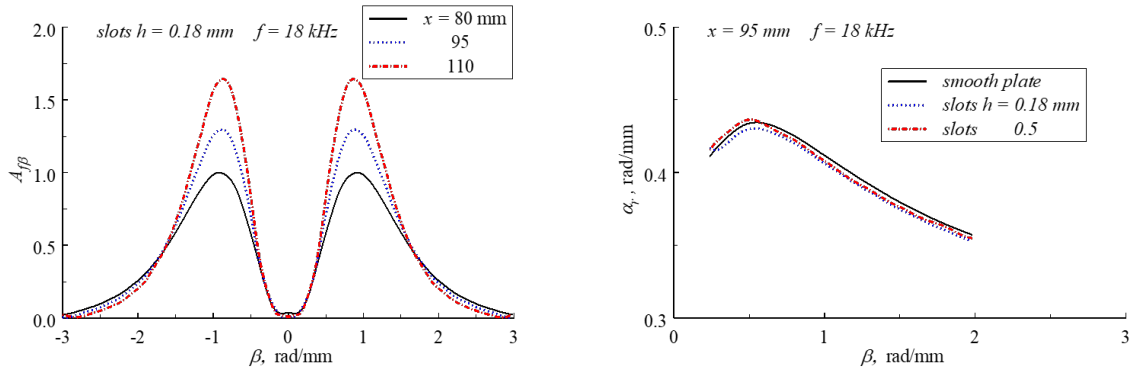


Figure 6. Growing disturbance downstream

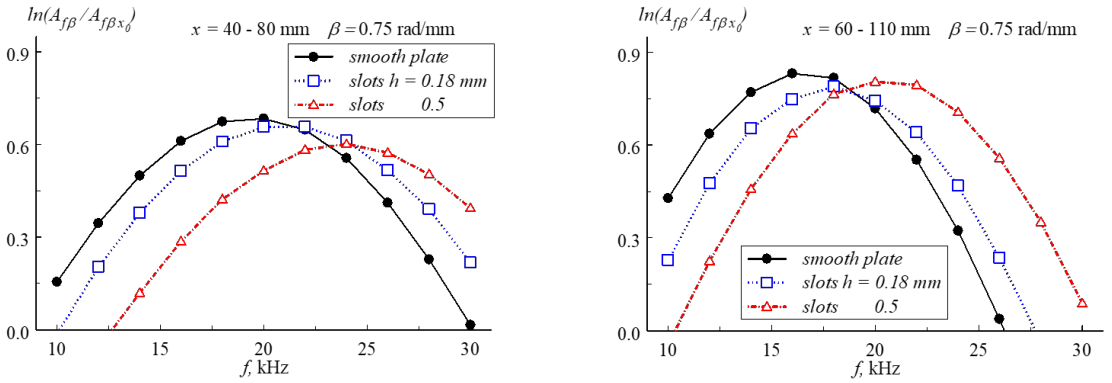
On the smooth plate, the disturbance level increases monotonically downstream. Slots lead to a stabilization of disturbance growth, especially in the area of the start of the slot. In case of 0.18 mm deep slots in the far zone ($x > 90$ mm), the disturbance growth is similar to that of the smooth plate. With increasing slot depth, the disturbance growth increases significantly in the far zone from the start of the slot.

More detailed information on the influence of longitudinal slots on the plate on the development of disturbances at the boundary is provided by frequency-wavenumber analysis. Figure 7a



(a) Wave spectra for $f = 18$ kHz (b) Dispersion relation

Figure 7. Amplitude-wave properties of the disturbance evolution



(a) $x = 40$–80 mm (b) $x = 60$–110 mm

Figure 8. Wave growth with $\beta = 0.75$ rad/mm in the region of the beginning of the slots and in the flow above the slots

shows the wave spectra of a localized disturbance for the frequency $f = 18$ kHz for the case of a flow over slots with a depth of 0.18 mm at different values of the coordinate x. Figure 7b shows the dispersion relation for the same frequency obtained for the cases of the smooth surface and the plate with slots.

The wave spectra obtained have a shape typical of a linear wave packet. The fastest growing waves have wavenumbers in the range of 0.6–1.2 rad/mm. This corresponds to wavelengths of 5–11 mm, which is a multiple of the slot period in the configurations considered. Similar spectra were obtained for deeper slots and for the case of a smooth surface. Figure 7b shows that the slots on the surface have only a minor influence on the dispersion relation. This result can be used in calculations according to the linear stability theory for cases of a plate with slots on the surface.

Figure 8 shows the effects of slots on the stability of the boundary layer. Distributions of wave growth with a wavenumber of $\beta = 0.75$ rad/mm are shown for different frequencies. The relative growth of the disturbances is analyzed in two areas: in the area of the beginning of the slots $x = 40$–80 mm, in the boundary layer above the slots $x = 60$–110 mm.

For both the smooth plate and the slotted plate, the value of the dimensional frequency of the most unstable boundary layer waves decreases downstream. This agrees with the results of the linear theory. From the numerical simulation data, the presence of slots on the plate surface leads to a change in the stability properties of the boundary layer. Initially, a clear shift in the frequency range of the unstable waves is observed. With the increasing slot depth, the frequency of the most strongly growing disturbances increases. For the slot configurations considered, the frequency of the fastest growing waves increases by more than 30% compared to the case of the smooth plate.

In the area of the beginning of the depressions (Fig. 8a), a lower growth of disturbances in the boundary layer can be observed on the plate with slots than in the case of the smooth plate. As the depth of the slots increases, the growth coefficient of the disturbances decreases. Downstream, in the flow above the slots (Fig. 8b), the difference in the growth of the disturbances in the boundary layer on the smooth surface and on the plate with slots decreases. However, there remains a clear shift in the frequency range of the unstable waves.

The data presented in Fig. 8 allow us to formulate the following explanation for the results on the growth of disturbances in physical space (Fig. 6). In the area of the beginning of the slots on the model, the growth of the disturbances is significantly lower compared to the case of the smooth surface (Fig. 8a). Therefore, in Fig. 6, a decrease of the disturbances can be observed in the range of $x = 55$–80 mm. Downstream, the growth of disturbances of the boundary layer on the plate with slots approaches the case of a smooth surface. However, in Fig. 6, for the case of a slot depth of 0.5 mm, a significant increase in disturbances can be observed compared to the smooth model. This can be explained by the fact that the slots lead to a shift of the frequency range of the unstable disturbances into the range of higher frequencies compared to the case of the smooth plate. A more intense growth of the high frequency disturbances can lead to a larger integral increase of the disturbances in the physical space. On this basis, it is possible to draw a conclusion about the influence of the investigated slots on the plate surface on the laminar-turbulent transition on the plate. A significant delay of the laminar-turbulent transition is possible if the spectrum of flow disturbances is low-frequency. However, if high-frequency pulsations predominate in the oncoming flow, early turbulization of the boundary layer is possible.

Conclusion

The influence of longitudinal rectangular slots on the plate surface on the stability of the boundary layer at Mach number 2 is investigated by a numerical simulation of the evolution of small amplitude local disturbances. The calculation formula for the evolution of disturbances in the boundary layer on a smooth plate is verified. The numerical simulation data on the growth of instability waves agree with the results obtained by the linear stability theory. It is shown that the grid resolution in the calculations mainly influences the growth of high-frequency disturbances.

The analysis of the evolution of a localized disturbance in the boundary layer of the smooth plate showed a monotonic growth downstream. In the area where the slots begin, the disturbance growth is lower than on the smooth surface. In the flow above the slots, it was found that the growth of the disturbance depends on the depth of the slots. At a slot depth of $h = 0.18$ mm, the disturbance growth is close to the value of the smooth surface, while it is significantly higher at $h = 0.5$ mm.

A frequency-wavenumber analysis of the disturbance development in the boundary layer of the smooth plate and the plate with slots is carried out. In the flow above the slots, the wave spectrum of the disturbances has the character of a linear wave packet. The dispersion relation is practically the same both for the case of the smooth surface and in the presence of slots on the plate. It is found that the most unstable waves shift to higher frequencies in the presence of slots.

The results of the numerical simulation confirm the possibility discovered earlier in experiments [11] of influencing the stability of the supersonic boundary layer of a plate by means of shallow longitudinal slots.

Acknowledgements

The research was financially supported by the Russian Science Foundation (Project No. 23-79-10167, https://rscf.ru/project/23-79-10167/) using the equipment of the "Mechanics" Shared-Use Center, ITAM SB RAS. Numerical simulation was performed using the FlowVision software (https://flowvision.ru/).

References

1. FlowVision 3.14.01: Users guide. `https://flowvision.ru/webhelp/fvru_31401/index.html?numerical_methods.htm` (2024), accessed: 2025-04-01

2. Bippes, H.: Basic experiments on transition in three-dimensional boundary layers dominated by crossflow instability. Progress in Aerospace Sciences 35(4), 363–412 (May 1999). `https://doi.org/10.1016/s0376-0421(99)00002-0`

3. Boiko, A.V., Kozlov, V.V., Syzrantsev, V.V., Shcherbakov, V.A.: Riblet control of the laminar-turbulent transition in a stationary vortex on an oblique airfoil. Journal of Applied Mechanics and Technical Physics 37(1), 69–79 (Jan 1996). `https://doi.org/10.1007/bf02369404`

4. Chuvakhov, P.V., Egorov, I.V.: Numerical simulation of disturbance evolution in the supersonic boundary layer over an expansion corner. Fluid Dynamics 56(5), 645–656 (Sep 2021). `https://doi.org/10.1134/s0015462821050025`

5. Fedorov, A., Novikov, A.: Stabilization of crossflow mode by grooves on a supersonic swept wing. Theoretical and Computational Fluid Dynamics 37(2), 261–268 (Apr 2023). `https://doi.org/10.1007/s00162-023-00658-x`

6. Grek, G.R., Kozlov, V.V., Titarenko, S.V.: An experimental study of the influence of riblets on transition. Journal of Fluid Mechanics 315, 31–49 (May 1996). `https://doi.org/10.1017/s0022112096002315`

7. Grek, G.R., Kozlov, V.V., Titarenko, S.V., Klingmann, B.G.B.: The influence of riblets on a boundary layer with embedded streamwise vortices. Physics of Fluids 7(10), 2504–2506 (Oct 1995). https://doi.org/10.1063/1.868694

8. Kachanov, Y.S.: Physical mechanisms of laminar-boundary-layer transition. Annual Review of Fluid Mechanics 26(1), 411–482 (Jan 1994). https://doi.org/10.1146/annurev.fl.26.010194.002211

9. Kosinov, A.D., Maslov, A.A., Shevelkov, S.G.: Experiments on the stability of supersonic laminar boundary layers. Journal of Fluid Mechanics 219(1), 621 (Oct 1990). https://doi.org/10.1017/s0022112090003111

10. Kudryavtsev, A.N., Khotyanovsky, D.V.: Direct numerical simulation of transition to turbulence in a supersonic boundary layer. Thermophysics and Aeromechanics 22(5), 559–568 (Sep 2015). https://doi.org/10.1134/s0869864315050042

11. Lysenko, V.I., Smorodsky, B.V., Kosinov, A.D., Yatskikh, A.A.: Effect of the slot depth on supersonic boundary-layer stabilization. Thermophysics and Aeromechanics 31(1), 71–78 (Jul 2024). https://doi.org/10.1134/s0869864324010086

12. Mayer, C.S.J., Laible, A.C., Fasel, H.F.: Numerical Investigation of Wave Packets in a Mach 3.5 Cone Boundary Layer. AIAA Journal 49(1), 67–86 (Jan 2011). https://doi.org/10.2514/1.j050038

13. Ustinov, M., Ivanov, A.: Cross-flow dominated transition control by surface micro-relief. In: AIP Conference Proceedings. vol. 2027, p. 020013 (2018). https://doi.org/10.1063/1.5065091

14. Ustinov, M.V.: Control of laminar-turbulent transition on a swept wing using surface microreliefs. Fluid Dynamics 53(6), 762–773 (Nov 2018). https://doi.org/10.1134/s001546281806023x

15. Yatskikh, A.A., Afanasev, L.V.: Numerical simulation of the evolution of localized disturbances generated by two synchronous separated sources in a supersonic boundary layer. Thermophysics and Aeromechanics 29(6), 875–885 (Mar 2023). https://doi.org/10.1134/s0869864322060075

16. Yatskikh, A.A., Ermolaev, Y.G., Kosinov, A.D., Semionov, N.V.: Evolution of wave packets in supersonic flat-plate boundary layer. Thermophysics and Aeromechanics 22(1), 17–27 (Jan 2015). https://doi.org/10.1134/s0869864314010023

17. Yatskikh, A., Yermolaev, Y., Kosinov, A., *et al.*: Evolution of localized artificial disturbance in 2D and 3D supersonic boundary layers. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering 234(1), 115–123 (Jul 2018). https://doi.org/10.1177/0954410018787120

18. Zhao, R., Liu, X., Wen, C., Wang, X.: Broadband design of acoustic metasurfaces for the stabilization of a Mach 4 boundary layer flow. Advances in Aerodynamics 4(1) (Apr 2022). https://doi.org/10.1186/s42774-022-00106-0

DOI: 10.14529/jsfi250104

Direct Numerical Simulation and Rank Analysis of Two-Dimensional Kolmogorov-type Vortex Flows*

Mikhail A. Guzev[1], ***Alexey N. Doludenko***[2], ***Alexey D. Ermakov***[3], ***Anna O. Posudnevskaya***[3,4], ***Svetlana V. Fortova***[3]

The article is devoted to direct numerical modeling of viscous weakly compressible Kolmogorov-type flows in a square calculation cell. Several different conditions are observed. One of them is dominated by a large vortex with a well-defined average profile. In another state, strong chaotic large-scale fluctuations prevail. In the third state, laminar flow is observed. The nature of the realized state depends on the coefficient of kinematic viscosity of the liquid, the amplitude of the external pumping force, and the bottom friction coefficient. At constant values of the kinematic viscosity and the wave vector, a small value of the friction coefficient leads to the appearance of the first state. As the bottom friction coefficient increases, there is a transition from a flow with one large vortex to a laminar flow through a series of states with several unstable vortices, which we call chaotic flow. A rank analysis of the values of vorticity, energy, and pressure, as well as the frequency of their occurrence, is proposed. It is shown that for chaotic, vortex, laminar and transitional regimes of fluid motion, the inflection point in the rank frequency distributions of the above fields is a universal characteristic for classifying various types of flow.

Keywords: turbulence, numerical simulation, Kolmogorov-type flows, rank analysis.

Introduction

Two-dimensional models for describing vortex flows are widely applied in atmospheric, oceanic, and astrophysical research [19]. The applicability conditions for these models are met for processes, the horizontal scales of which are much larger than the vertical one, and it can be assumed that the main flow movement occurs in two horizontal directions. In 1959, A.N. Kolmogorov proposed to study the simplest model, which is the two-dimensional motion of a viscous fluid prompted by the action of a periodic (along one of the coordinates) field of an external force (pumping) [2]. The first theoretical works [4, 28] devoted to two-dimensional vortex structures reveal a fundamental difference in their behavior from three-dimensional ones. In the spatial case, it is known that there is a direct cascade of energy [23], which, due to nonlinear interaction, transfers energy from the integral scale, on which energy is pumped, to smaller scales up to the dissipative one, determined by viscosity, on which kinetic energy is converted into heat. In the planar case, the situation is the opposite: a reverse energy cascade occurs, in which energy is transferred from small scales to larger ones. Quantitatively, this is manifested in the different behavior of the energy spectrum and the influence of internal dynamic characteristics on it. Thus, in the works [29–32] the formation of sharp vorticity gradients in two-dimensional hydrodynamic turbulence and their influence on turbulent spectra due to flow anisotropy are considered. The energy accumulation on the scale of the system size leads to the emergence of intense large-scale motion, including large vortices [26]. The tendency towards the formation of

*The article is recommended for publication by the Program Committee of the All-Russian Scientific Conference "Computational Experiment in Aeroacoustics and Aerodynamics 2024".

[1]Institute for Applied Mathematics, Far Eastern Branch, Russian Academy of Sciences, Vladivostok, Russian Federation
[2]Joint Institute for High Temperatures, Russian Academy of Sciences, Moscow, Russian Federation
[3]Institute for Computer Aided Design, Russian Academy of Sciences, Moscow, Russian Federation
[4]L.D. Landau Institute for Theoretical Physics, Russian Academy of Sciences, Chernogolovka, Russian Federation

large vortices has been indicated in the research devoted to two-dimensional turbulence, both experimental [47] and numerical [6, 9, 45, 46]. Large coherent vortices were observed in numerical simulation [10, 11, 42, 50] based on the solution of the two-dimensional Navier–Stokes equation with no-slip boundary condition. A similar coherent vortex structure was created in laboratory experiments in a square container [17, 49, 51]. Large vortices were obtained by numerical simulation of static pumping with various types of large-scale dissipation, as reported in [7, 48]. A whole series of computational works has been devoted to studying the properties of a coherent vortex [14, 25, 27, 33]. The first attempt to establish the profile of the average velocity of a coherent vortex was made in [33], where periodic boundary conditions and short-term time-correlated pumping were introduced. The authors show the appearance of a stable vortex dipole. In [25, 27], in a similar formulation, the average velocity profile of a coherent vortex was numerically found, which exhibits isotropy properties at a distance of the order of the vortex radius. Note that, in addition to Kolmogorov pumping, the presence of bottom friction has a significant effect on the formation of such flows, which is introduced into the system of equations under study by adding a term with a coefficient called the bottom friction coefficient [5].

In [14], Kolmogorov-type flow regimes in a square cell that occur at various values of the friction coefficient are numerically investigated. Three types of flow are most clearly classified: laminar, chaotic, and vortex flows. Transitional regimes arise between them, which are difficult to attribute to one type of flow [43], since they change their characteristics over time and are formed through a sequence of bifurcations during the change from a laminar flow to a chaotic flow regime, as well as during the transition from a chaotic to a vortex flow. In [13], the conditions for the existence of a vortex regime were identified and it was shown that it is observed at a sufficiently small value of the friction coefficient. The remaining flows occur when the coefficient is increased. However, this coefficient cannot be directly measured during a physical experiment in which the above flow regimes are observed. The measured characteristic for a two-dimensional flow during a physical experiment is the velocity field. Therefore, a natural problem arises to formulate an algorithm for processing the velocity field, which can be used to answer the question of what state (laminar, chaotic, vortex, or transitional) the hydrodynamic system is in. This will allow us to determine the value of the friction coefficient corresponding to a numerical experiment with the same parameters as in a physical experiment. In [21], for the first time, a method was proposed to classify structure on the analysis of rank distributions [22, 39] for the vorticity field and the frequency of occurrence of various values of this parameter. In [21], a differential characteristic of the vorticity distribution was revealed for various flow regimes of a viscous, slightly compressible fluid that occurs in a square region under the influence of a constantly acting force. This characteristic is determined by the coordinates of the inflection point for the rank distribution of the frequency of occurrence of vorticity. At the same time, numerical modeling has shown that there is no inflection point for the laminar regime; in the case of chaotic motion, it is formed, and when switching to the vortex flow regime, it shifts to the region of high ranks. Thus, the appearance of an inflection point can be used to analyze the types of flows. Since earlier [13, 14] flows were studied at various values of the pumping force and the bottom friction coefficient and a rank analysis was performed only for the vorticity field [21], it is necessary to perform a rank analysis of flow regimes for other characteristics of the model: energy and pressure, as well as to investigate the behavior of the corresponding inflection points of the rank curves. This paper presents a solution to this problem.

Let us briefly describe the content of the work. Section 1 presents a system of equations for the Kolmogorov-type flow and the results of a numerical experiment to identify the types of flows depending on the bottom friction coefficient. In Section 2, the general idea of constructing rank distributions for analyzing system properties is formulated. Sections 3, 4 and 5 are devoted to the analysis of the rank distributions of their values and the corresponding occurrence rates for vorticity, energy, and pressure. The appearance of an inflection point for the rank distributions of the frequency of occurrence of the studied characteristics is shown when the flow regimes change. The article ends with Discussion and Conclusion. For steady-state laminar flow, Appendix provides the construction of an analytical solution and the corresponding rank distribution, and a comparison with the results of numerical simulation is performed.

1. Model Relations and Calculation Results

We study the two-dimensional motion of a viscous weakly compressible fluid in a square cell Ω, satisfying the system of Navier–Stokes equations. The corresponding equations of continuity (1), momentum variation (2), (3), and weak compressibility (4) are presented below:

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \bar{V}) = 0, \tag{1}$$

$$\frac{\partial \rho u}{\partial t} + \nabla(\rho u \bar{V}) = -\frac{\partial p}{\partial x} + \rho G \sin ky + \mu \Delta u - \zeta u, \tag{2}$$

$$\frac{\partial \rho v}{\partial t} + \nabla(\rho v \bar{V}) = -\frac{\partial p}{\partial y} - \rho G \sin kx + \mu \Delta v - \zeta v, \tag{3}$$

$$\mathrm{d}p = c^2 \rho_0 \frac{\mathrm{d}\rho}{\rho}. \tag{4}$$

Here ρ is the density of the liquid ($\rho_0 = 1000\,\mathrm{kg/m^3}$ is the initial density); $\bar{V} = (u, v)$ is the velocity vector, the components of which are equal, respectively, to u and v; $\mu = 0.1\,\mathrm{Pa \cdot s}$ is the dynamic viscosity of the liquid; p is pressure; ζ is the coefficient characterizing the presence of bottom friction; $G = 0.15\,\mathrm{m/s^2}$ is the amplitude of the external force; $k = 5\,\mathrm{m^{-1}}$ is the spatial frequency of the external force; c is the speed of the disturbance propagation. The right-hand sides of equations (2) and (3) contain the terms $\rho G \sin ky$ and $-\rho G \sin kx$, which model the action of the pumping force. The components $-\zeta u$ and $-\zeta v$ characterize the bottom friction force. For velocity, the no-slip condition is required at the boundary of the calculation cell: $\bar{V}|_{\partial\Omega} = 0$. The boundary condition for pressure is $\nabla_n p|_{\partial\Omega} = 0$, here $\partial\Omega$ is the outer boundary of the computational cell Ω, $\bar{n}$ is a vector perpendicular to the boundary surface.

The dimensionless parameter characterizing the formed flow regime is the Reynolds number $Re = \rho_0 V_{max} L/\mu$, here $L = 2\pi\,\mathrm{m}$ is the size of the computational area, V_{max} is the maximum modulus of the flow velocity. The end time of the calculation was selected after the system reached a statistical stable state.

The initial conditions are undisturbed pressure and velocity fields: $P_0 = 10^5\,\mathrm{Pa}$, $u = v = 0\,\mathrm{m/s}$.

In equation (4) the speed of the disturbance propagation is $c = 100$ m/s. The choice of such a speed value allows us to achieve a compromise between the time step and the stability of the calculations. For the McCormack scheme, the following relationship between the time step, the space step, and other quantities has been empirically established [1], ensuring the stability of the scheme:

$$\Delta t \leq \frac{\sigma \Delta x}{\left(V_{max} + \sqrt{2}c\right)\left(1 + 2/Re_{min}\right)}, \tag{5}$$

here σ is the safety factor, $\Delta x \approx 1.2 \cdot 10^{-2}$ m is the spatial step, $V_{max} \sim c/10$. Let us consider the case of a coarse computational grid, as the most favorable in terms of computational time. For laminar flow $Re = 900$ and we assume the minimum $Re_{min} \sim 200$, and we choose the coefficient σ equal to 1. Then from formula (5) it follows that $\Delta t \leq 8 \cdot 10^{-5}$ s. If $\sigma = 0.5$, then $\Delta t \leq 4 \cdot 10^{-5}$ s. For the calculations presented in this paper, a time step of $\Delta t = 2.5 \cdot 10^{-5}$ s was used.

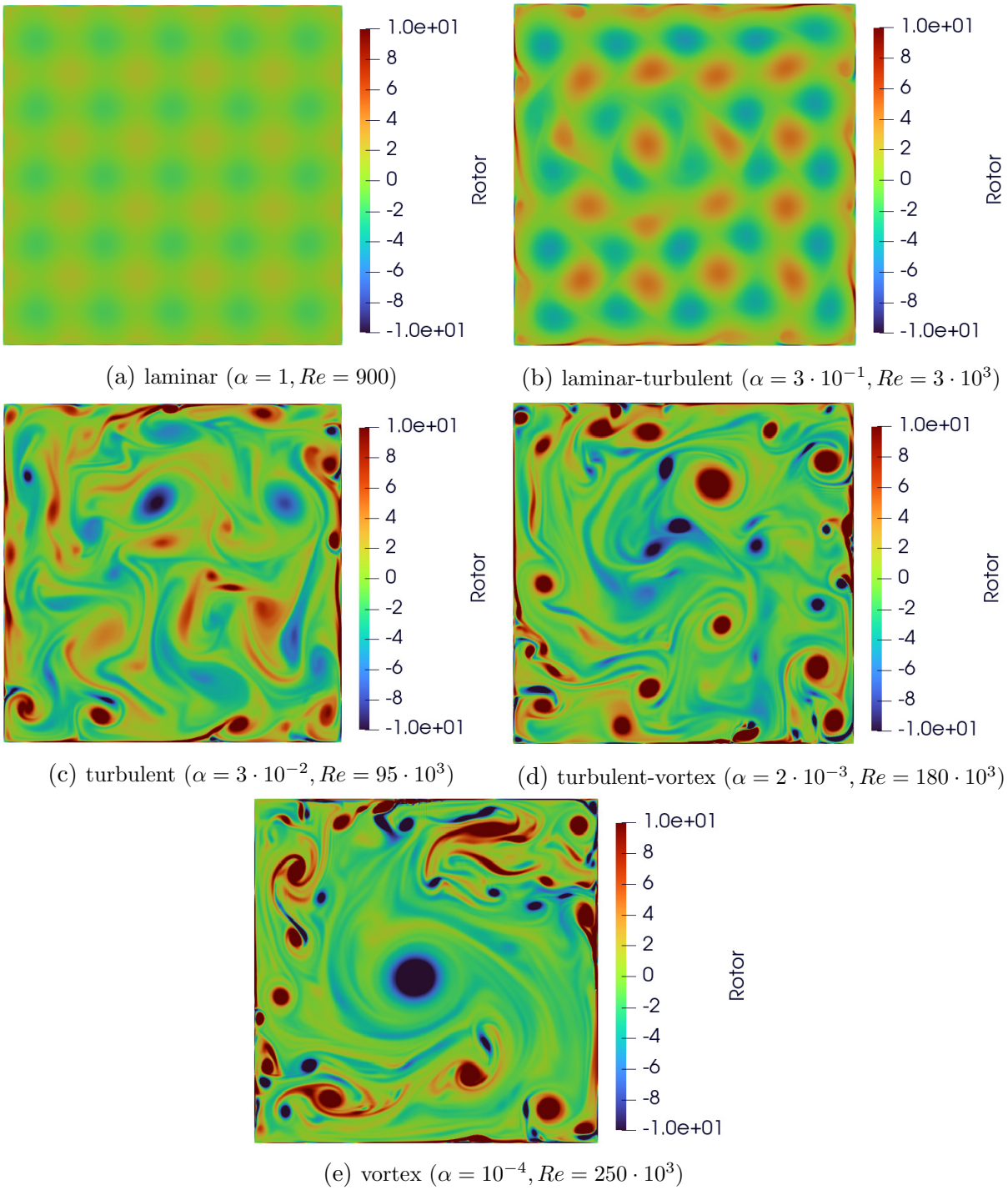
The numerical solution of the Navier–Stokes equations is based on the artificial compressibility method [1]. In this case, the hyperbolic part of the equations is solved by the explicit McCormack method [37], whereas the parabolic part is solved by the standard finite difference method. The McCormack scheme has a second order of accuracy in space and time.

The McCormack scheme is used for the numerical solution of hyperbolic equations, since it has good dissipative and dispersion properties. In addition, it enables to study the behavior of quantities with large gradients, including discontinuities. To understand how suitable this numerical tool is for modeling turbulence, a test was performed, presented in [12, 15]. Overall, the McCormack scheme showed similar results to those that can be obtained using algorithms based on the CABARET scheme and the OpenFOAM finite volume method.

The numerical code is parallelized using the OpenMP library, designed for computers with shared memory. The calculations used the number of cores up to 384.

In [43], three main flow regimes were obtained: laminar, in which the fluid motion pattern retains its initial shape and does not change over time; chaotic (or turbulent), characterized by the presence of randomly moving vortices of various sizes and lifetimes; and vortex, in which a single large coherent vortex appears, it occupies almost the entire computational area and exists for a long time. In the course of a numerical experiment, the dependence of the flow type on the value of the bottom friction coefficient $\alpha = \zeta/\rho_0$ was analyzed. In order to identify it, numerical experiments were performed for various values of α and fixed values of G, μ. The dimensions of the parameters α, G, μ are equal to s^{-1}, $\mathrm{m/s}^2$, Pa $\cdot$ s. It should also be emphasized that we did not set out the task of obtaining results that could be used as the basis for describing physical experiments on turbulence. Discussions with experimental scientists led us to the conclusion that at this stage of the numerical experiment it is necessary to develop a mathematical model that could qualitatively identify new effects of vortex motion. Therefore, the values of the simulation parameters, in particular, regarding viscosity, were selected based on the principle of "saving" computing resources. The experiments were carried out in the range for α from 10^{-4} to 1 and allowed us to identify the presence of transitional flow regimes occurring between the main regimes [43].

Figure 1 shows the vorticity fields of the forming flow depending on the value of α, all other simulation parameters are fixed. The value of the bottom friction coefficient $\alpha = 1$ corresponds to the stationary distribution of the velocity field determined by the pumping force. We call this laminar regime (Fig. 1a). When the coefficient α is in the range from 10^{-1} to $3 \cdot 10^{-1}$, a flow is formed (Fig. 1b) which is transitional from laminar to chaotic (Fig. 1c). This type of flow is characterized by the appearance of a system of vortices with positive and negative vorticity (Fig. 1b). Their position and shape have small deviations from the stationary distribution of the vorticity field. As the bottom friction coefficient decreases, the amplitude of vortex oscillations around their stationary position increases, as well as their size and shape change. The criterion



(a) laminar ($\alpha = 1, Re = 900$)

(b) laminar-turbulent ($\alpha = 3 \cdot 10^{-1}, Re = 3 \cdot 10^{3}$)

(c) turbulent ($\alpha = 3 \cdot 10^{-2}, Re = 95 \cdot 10^{3}$)

(d) turbulent-vortex ($\alpha = 2 \cdot 10^{-3}, Re = 180 \cdot 10^{3}$)

(e) vortex ($\alpha = 10^{-4}, Re = 250 \cdot 10^{3}$)

Figure 1. Vorticity fields with fixed parameters $G = 0.15$, $\mu = 0.1$, and various α for different types of flow

for the transition to a chaotic regime (Fig. 1c) is the separation of vortices from their initial location and their chaotic movement around the cell. The chaotic flow in the figure corresponds to the value of the bottom friction coefficient in the range from $\alpha = 10^{-2}$ to $\alpha = 10^{-3}$. A decrease in the bottom friction coefficient to $\alpha = 10^{-4}$ leads to the appearance of a vortex regime (Fig. 1e), which is characterized by the formation of one large structure occupying the entire computational cell. Also, at the value of $\alpha = 2 \cdot 10^{-3}$, a regime transitional between the

chaotic and vortex ones is observed (Fig. 1d), which is characterized by the alternate existence of a chaotic flow and a coherent vortex.

2. A Method for Analyzing Flows according to Rank Distributions

As it is known, frequency-rank distributions have been used for a long time in various fields of human activity in describing quantitative characteristics of observed phenomena. The fact of their application was discovered in the first half of the twentieth century and is associated with works that have become classical in the field of demography [3], scientometrics [35], biology [18], seismology [20], linguistics [52], etc. It should be noted that, for frequency-rank distributions, foreign researchers [41] use the term "power-law distributions". In [8], the authors give a large number of examples from physics, earth sciences, biology, ecology, paleontology, computer and information sciences, and engineering and social sciences and show that they manifest a power-law distribution with the corresponding parameters. In the modern Russian literature devoted to the problem of rank distributions, we point out the paper [44], in which, as applied to ecology, modern surveys on rank distributions are discussed, and also the paper [36], which contains a rank analysis of technical systems.

Among the rank distributions, a special place is occupied by distributions described by the laws of Zipf, Pareto, Lotka, Mandelbrot, and others. Rank distributions involve the rank r (number) of an object and the frequency E of occurrence of the considered characteristics of the object: the rank is set and the frequency of occurrence is assigned to it. In particular, if, for some rather large text, one compiles a list of all words that occur in the text and then rank these words in decreasing order of the frequency E of their occurrence in the text, then, according to the Zipf's law [44], the multiplication of the rank r of a word and the frequency of occurrence E is constant: $rE = const$. Modifications of the Zipf's law were proposed for other rank distributions; however, there are sufficiently many phenomena for which the description of their characteristics is inconsistent with the above laws.

This stimulated researchers both to search for dependencies approximating empirical data more accurately and to formulate general models leading to rank distributions. In linguistics, when analyzing texts, Mandelbrot's model representations are known [38], in which he explains the Zipf's law using the concept of value of the optimal word code under the assumption that the text consists of words, separated by spaces, and is generated by some random process. A comparison of this approach with others is presented in the survey [34]. During the same decade, A.N. Kolmogorov, one of the founders of the probability theory, outlined in his works a way to revise the probability theory [24], from the viewpoint of the algorithmic approach. According to his approach, the notion of randomness was defined by him as the maximally complicated one. If one explains every random event in a deterministic way, then the algorithm for its occurrence will be very complex, and the decoding of this algorithm will require a very long code. The more complex the description of information, the longer is the required decryption algorithm, and this, according to Kolmogorov, is close to randomness.

A.N. Kolmogorov's concept obtained a constructive implementation in the works of V.P. Maslov, which enables us to study the behavior of observable characteristics without dividing them into deterministic and random ones. His approach differs essentially from that of Zipf, as well as from that of all other researchers who have studied this problem. Namely, he

used another representation in which the number of repeating values of a quantity is regarded as the observed quantity. Let us explain the general approach of V.P. Maslov [39, 40].

Let there be repeating values among the values x_i:

$$x_1 = x_2 = \ldots = x_{N_1}, x_{N_1+1} = x_{N_1+2} = \ldots = x_{N_2}, \ldots, x_{N_{k-2}+1} = x_{N_{k-2}+2} = \ldots = x_{N_{k-1}}$$
$$x_{N_{k-1}+1} = x_{N_{k-1}+2} = \ldots = x_{N_k}, \sum_{i=1}^{k} N_i = N,$$

i.e., the family of numbers $N_1, N_2, \ldots, N_k$ specifies how many times the values $x_{N_1}, x_{N_2}, \ldots, x_{N_k}$ are repeated. Let n_i, $i = 1, \ldots, p$, denote the number of x_i with the same frequency of their occurrence E_i,

$$N_1 = \ldots = N_{n_1} \equiv E_1, N_{n_1+1} = \ldots = N_{n_2} \equiv E_2, \ldots, N_{n_{p-2}+1} = \ldots = N_{n_{p-1}} \equiv E_{p-1},$$

$$N_{n_{p-1}+1} = \ldots = N_{n_p} \equiv E_p,$$

and the values E_i are ordered, i.e., $0 \leq E_i \leq E_{i+1}$. According to the terminology adopted in linguistics, the frequencies of occurrence E_i form a dictionary [40], and n_i is the number of words with the same frequency of occurrence. In this consideration, the frequency E_i acts as an observable value and the number of values n_i with this frequency characterizes the number of realized values E_i. With fairly general assumptions, V.P. Maslov proposed a formula relating the rank r_s of the s-th word to frequency:

$$r_s = \sum_{i=1}^{s} \frac{1}{\exp(\beta E_i + \sigma) - 1}. \tag{6}$$

Here, the parameters β and σ are given by the normalization conditions (the exact formulation of this assertion in the form of a theorem is given in [39]). Thus, formula (6) implements an algorithm of deterministic calculation of the cumulative probability. In this consideration, the frequency E_i acts as a random variable, and the number n_i of words with this frequency characterizes the number of occurrences of this random variable. This distinguishes Maslov's consideration from the approach adopted in linguistic statistics. The idea of using a new kinematic set of variables in the study of the behavior of the observed characteristics of the system was earlier constructively implemented in the work of V.A. Fock on the secondary quantization method [16]. This method was used to construct a wave function when describing a system of quantum particles. According to Fock's idea, it is convenient to analyze the equation for this function not in the configuration space, but in the space of occupation numbers or, in the terminology of quantum mechanics, in the Fock space. In other words, to describe the system, one should select a set of variables to which Fock has moved using the canonical transformation.

In fact V.P. Maslov proposed a representation in which frequency is considered as an observable, and the number of values of the initial variable with this frequency characterizes the number of realized frequency values. With this approach, the Maslov frequency is similar to the Fock's canonically conjugate variable (energy), and the number of realized frequency values corresponds to the number of occupation.

In [21] we constructed an algorithm for classifying Kolmogorov-type flows using the knowledge about their velocity field. The main difficulty in analyzing the characteristics of a chosen dynamical system was the formation of information that is suitable for creating a corresponding algorithmization procedure. We used the theoretical concepts of V.P. Maslov that suggest an

interpretation of the information data as a semiotic system in which the values of the vorticity of the velocity field are regarded as signs [21]. The rank analysis was performed first for specific values of vorticity, which enables us to obtain its distribution. Then, the rank curves were constructed for the frequency of the vorticity recurrence, which correspond to the flow regime as a whole. To obtain the functional dependence of rank on frequency based on formula (6), we used the parameterization for the frequency of occurrence proposed by V.P. Maslov. It was shown that it is possible to identify different fluid motion regimes for Kolmogorov-type flows by comparing them with the graphs of the rank distributions of the vorticity.

Since it was assumed that, in the process of evolution of the system, some forms of order are created in it, it follows that, from the point of view of semiotics, the patterns in the distribution of different signs should occur. Therefore, we will consider the values of energy and pressure as an additional set of signs for which we construct rank distributions of their values and frequencies of their occurrence. It is shown that the rank curves of the vorticity, energy, and pressure fields and their frequencies are divided depending on the type of flow. This behavior can be used to identify and analyze different flow patterns. When conducting a rank analysis, the curves obtained make it possible to establish an unambiguous relationship between behavior a graph of the rank curve and the corresponding type of flow.

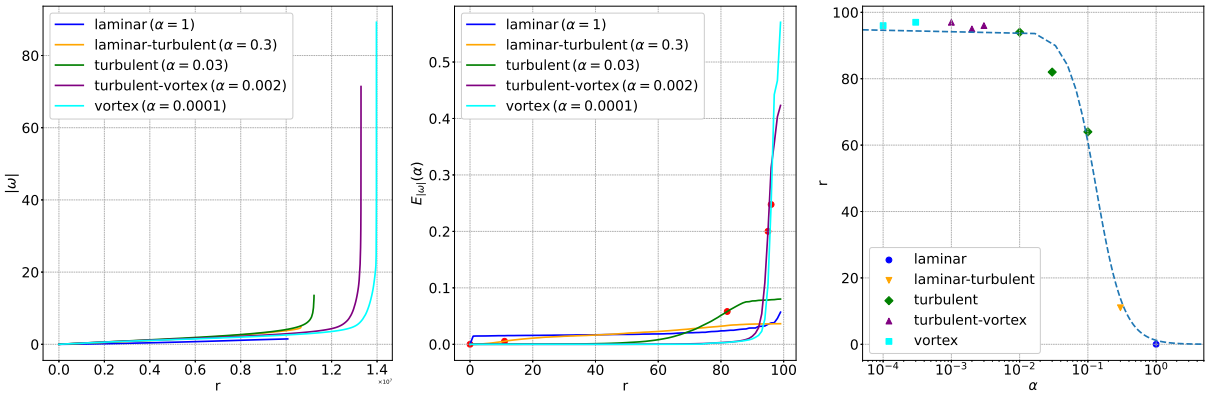
The construction of a rank distribution for a parameter X of an arbitrary physical system can be performed in two ways. The value of X is considered for 100 time steps starting from 700th iteration, which corresponds to reaching a statistical stationary state. For each moment in time, the value of X is calculated in each cell of the computational domain. After that, the values of X obtained for all moments in time are combined into one data array. It is sorted in ascending order of the value of X, and each value is assigned a serial number (rank). The rank is understood as the ordinal number in the ascending (or descending) sequence of values of X.

When using the second method, we move in the description of the system from the variable X to the frequency E of occurrence of this variable. The values of E are sorted from lower to higher, and for the resulting array of values, a distribution is constructed for E depending on the rank r. The implementation of the first approach does not lead to any difficulties. In the algorithmic implementation of the second approach, we identify the minimum $X_{\min}$ and maximum values of $X_{\max}$ of the variable X in the computational domain. Setting $\Delta = (X_{\max} - X_{\min})/N$, where N is the number of intervals selected a priori (in what follows, $N = 100$), we subdivide the segment $[X_{\min}, X_{\max}]$ into intervals of length Δ. In this case, the calculated values of X_k fall within one of these intervals. Calculating the number of values of X_k that fall within one of the intervals, we obtain the frequency E_k of occurrence of the parameter X. As a result, we have an array of length N containing the frequency of occurrence E_k. Finally, this array is sorted: the largest value of the frequency of occurrence E_k is assigned the largest ordinal number (rank) of r, and the smallest value of E_k is assigned the smallest rank of r. Note that traditionally, when considering chaotic flows, a single-point probability density function (PDF) is used. PDF vorticity distributions were constructed for laminar, turbulent, and vortex flow regimes. For the same regimes, rank dependencies of the vorticity frequency of occurrence were constructed [21]. Both in case of rank analysis and in case of constructing a PDF distribution, the same values of the vorticity frequency of occurrence are used. The difference between these approaches lies in different data processing: when performing rank analysis, the frequency of occurrence is distributed depending on the rank, and in case of PDF analysis, the frequency of occurrence is distributed depending on the value of vorticity.

3. Rank Analysis of Vorticity

In accordance with the idea proposed above, we will perform the construction of rank curves for the vorticity value using the rank analysis methodology described in Section 2. Figure 2a shows the vorticity distribution for different bottom friction coefficients. Note the characteristic behavior of the plotted graphs. The curves corresponding to different flow regimes are arranged in a certain sequence. This is clearly seen in the graph for negative vorticity (Fig. 2a). The graph corresponding to the laminar regime can be approximated by a linear function, for which the bottom friction coefficient α is 1. In Appendix, an analytical solution has been built for this regime. As α decreases to 10^{-1}, the shape of the rank distribution graph changes, and the curvature of the graph increases in the area of higher ranks. This corresponds to a regime change from laminar to chaotic. Various forms of turbulent flows are observed in the range of the bottom friction coefficient from 10^{-1} to 10^{-3}. When α changes from 10^{-3} to 10^{-4}, a large coherent vortex formation regime is realized. Separately, it is worth noting that due to the averaging of 100 output over time, the total number of ranks with this approach is about 10^7.

Figure 2b shows the vorticity frequency of occurrence E graphs for various flow regimes. The laminar and vortex regimes correspond to convex functions with different asymptotic behavior. Note that the rank distribution graphs for the turbulent regime have a characteristic S-shape. Analysis of the rank distribution curves for the vorticity frequency of occurrence reveals the presence of an inflection point on the graph. The red dots in Fig. 2b indicate the inflection points.



(a) Distribution of vorticity

(b) Distribution of the vorticity frequency of occurrence

(c) Dependence of the inflection point rank on α

Figure 2. Rank analysis of vorticity for different type of flows

Thus, considering the entire range of the bottom friction coefficient α, the following properties of the constructed distributions can be distinguished. First, when the flow regimes change, the inflection point appears in the graphs of the corresponding rank distributions. Secondly, for each type of flow, this inflection point has its own rank. The dependence of the inflection point rank on α on a logarithmic scale is shown in Fig. 2c. The gray triangles in this graph correspond to the vortex regime, which is characterized by high inflection point ranks and low values of the bottom friction coefficient (the range of α is from 10^{-3} to 10^{-4}). For the laminar regime, the inflection point has small ranks, which correspond to the orange triangles on the graph (the range of α is from 1 to 10^{-1}). For this regime, an analytical solution is constructed in Appendix, for which the frequency distributions of vorticity are indicated. Comparison of

them with numerical results shows a satisfactory correspondence between them. The turbulent type of flow occurring in the range of α from 10^{-1} to 10^{-3} has a wide range of rank values for the inflection point, ranging from large to small. The dependence of the rank of the inflection point on the friction coefficient α can be approximated using the relation:

$$r = \frac{A}{1 + B(\alpha/\alpha_0)^\gamma}. \tag{7}$$

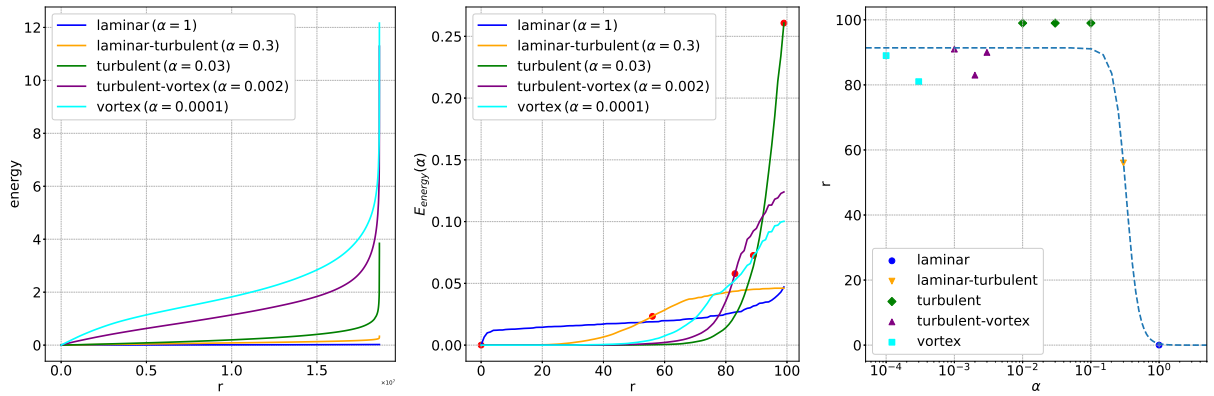
Here α_0 is the dimensional coefficient equal to $1\,\mathrm{s}^{-1}$; A, B and γ are the fitting parameters corresponding to the values 95, 78 and 2.1 for this graph.

4. Energy Rank Analysis

Let us consider another parameter of the flow, energy, and perform a rank analysis for it. Figure 3a shows the energy distribution for different bottom friction coefficients. Note the characteristic behavior of the plotted graphs. The curves corresponding to different flow regimes are arranged in a certain sequence. This is clearly seen in Fig. 3a. The graph corresponding to the laminar regime can be approximated by a linear function, for which the bottom friction coefficient is $\alpha = 1$. As α decreases to 10^{-1}, the shape of the rank distribution graph changes, the curvature of the curve changes in the area of high ranks, and the energy value increases. This corresponds to a regime change from laminar to chaotic. In the range of changes in the bottom friction coefficient from 10^{-1} to 10^{-3}, various forms of turbulent flows and a further increase in energy in the range of higher grades are observed. When α changes from 10^{-3} to 10^{-4}, a large coherent vortex regime is realized. The energy reaches its maximum value for this regime. The localization of the energy on the macroscopic scale of pumping corresponds to the formation of the Kraichnan reverse energy cascade [28] for this system. It is worth noting separately that due to averaging over 100 issues over time, the total number of ranks with this approach is approximately $1.8 \cdot 10^7$. There are more of them than in the vorticity analysis, since only negative vorticity was used for consideration.

The results of the rank analysis of the energy frequency of occurrence for different flow regimes are shown in Fig. 3b. The convex functions of the rank distribution with different asymptotic behaviors correspond to the laminar and vortex regimes. Note that the rank distribution graphs for the turbulent regime have a characteristic S-shape. Analysis of the rank distribution curves of the energy frequency of occurrence, as in case of vorticity, reveals the presence of an inflection point on the graph. The red dots in Fig. 3b indicate the inflection points. In the region of high ranks, localization of curves corresponding to laminar, chaotic, and vortex regimes is noticeable. In particular, for laminar regime, the frequency of occurrence of energy depends almost linearly on the rank and its value does not exceed 0.02. An increase in the frequency of energy occurrence to 0.09 for the ranks in the range from 80 to 100 corresponds to the turbulent regime. For the vortex regime, there is a sharp increase in the frequency of occurrence to 0.14, starting from a rank equal to 95.

Considering the entire range of the bottom friction coefficient α, we can again identify two characteristic properties of rank distributions. Firstly, when the flow regimes change, an inflection point appears on the corresponding graphs. Secondly, for each type of flow, the inflection point has its own rank and its dependence on α is shown in Fig. 3c. For vortex and turbulent regime, the position of the inflection point is localized in the region of high ranks. In the course of transition from a chaotic regime to a laminar one, corresponding to the range of the



(a) Distribution of energy

(b) Distribution of the energy frequency of occurrence

(c) Dependence of the inflection point rank on α

Figure 3. Rank analysis of energy for different type of flows

bottom friction coefficient from 10^{-1} to 1, there is a sharp change in the rank of the inflection point to the minimum values. The graph of the dependence of the inflection point rank on the bottom friction coefficient can also be approximated by dependence (7) with coefficients $A \sim 91$, $B \sim 200$, $\gamma \sim 4.8$.

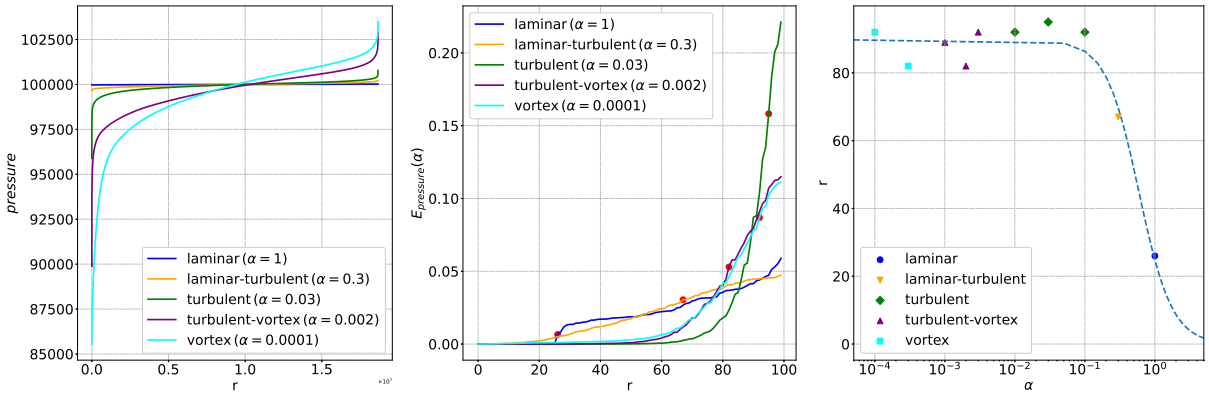
5. Pressure Rank Analysis

Figure 4a shows the pressure distribution for various coefficients α. Note that the graph in Fig. 4a has partial symmetry with respect to the horizontal straight line corresponding to the initial pressure distribution. For the laminar regime, localization of the rank curves relative to this distribution is observed. The transition to turbulent and vortex regime leads to a change in density and, as a result, a deviation (negative for small ranks and positive for large ranks) of the pressure value. The localization of curves corresponding to laminar, chaotic, and vortex regime is noticeable. The total number of ranks, similar to the previous paragraph, is about $1.8 \cdot 10^7$.

For the laminar regime, localization of the rank curves relative to this is observed. For laminar regime, its value does not exceed 0.5 percent of the initial distribution. In turbulent regime, there corresponds an increase in pressure deviations of up to 5 percent. For the vortex regime, there is a sharp increase in pressure deviations of up to 15 percent. Let us now consider the results of a rank analysis of the pressure frequency of occurrence for various flow regimes (Fig. 4b). Convex functions of rank distributions with different asymptotic behavior correspond to the laminar and vortex regimes. For the turbulent regime, a characteristic S-shaped pattern is realized for the rank distribution graphs. Analysis of the rank distribution curves for pressure frequency, as in case of vorticity and energy, reveals the presence of an inflection point on the graph.

Thus, considering the entire range of changes in the value of the friction coefficient α, we observe the appearance of an inflection point on the graphs of the corresponding rank distributions, while for each type of flow the inflection point has its own rank. The red dots in Fig. 4b indicate the inflection points. The dependence of the inflection point rank on α is shown in Fig. 4c. For vortex and turbulent regimes, the value of this rank is localized in the region of its high values. When switching from a chaotic regime to a laminar one, corresponding to the

range of the bottom friction coefficient from 10^{-1} to 1, there is a sharp change in the rank of the inflection point to the minimum values.



(a) Distribution of pressure

(b) Distribution of the pressure frequency of occurrence

(c) Dependence of the inflection point rank on α

Figure 4. Rank analysis of pressure for different type of flows

The graph of the dependence of the inflection point rank on the bottom friction coefficient can also be approximated by the dependence (7) with coefficients $A \sim 90$, $B \sim 2.6$, $\gamma \sim 1.8$.

Discussion

Earlier, the problem of analyzing the rank distributions for vorticity, energy, and pressure was formulated. The calculation results obtained in this paper demonstrate that the constructed rank curves for these quantities have similar behavior. It consists in the fact that for laminar flows, the behavior of the corresponding quantities, depending on the rank, is approximated by a linear function. When the flow regimes change, the graphs become significantly nonlinear in the area of high ranks. The analysis of the rank distribution of the frequency of occurrence for vorticity, energy, and pressure showed that when the flow regime changes, an inflection point appears for the graphs of all the indicated of the frequency of occurrence. Thus, the appearance of an inflection point can be used to analyze the types of fluid motion. The dependence of the inflection point rank on the bottom friction coefficient, which is responsible for changing the flow regime in the numerical experiment, makes it possible to establish a relationship between the internal parameter, the bottom friction coefficient and the velocity field observed in the physical experiment.

Conclusion

The problem of two-dimensional flow of a viscous fluid in a square cell under the influence of an external force (Kolmogorov-type flow) and the presence of bottom friction is considered. The analysis of the emerging flow types during the transition from the laminar flow type to the vortex flow occurs through a series of bifurcations, including the turbulent flow type. The use of rank distributions in analyzing the obtained flows allows us to look at vortex hydrodynamic processes from a new perspective and supplement the characteristics of flow regimes with such parameters as the inflection point of the frequency of vorticity, energy and pressure. This method makes it possible to determine how the rank curves for vorticity, energy, and pressure fields and their

frequencies are divided according to the flow type, which can be further used for identifying and analyzing various flow regimes. When performing a rank analysis, the obtained curves allow us to establish an unambiguous relationship between the behavior of the graph of the rank curve and the corresponding type of flow. Thus, the proposed approach makes it possible to classify flow regimes according to a given velocity field.

In our work, we consider the general idea of rank analysis for a system with a spatiotemporal internal structure, that is, endowed with some forms of order. It has been shown numerically that for the Kolmogorov problem, laminar, turbulent, and vortex flow regimes appear for each fixed value of the driving force, depending on the values of the bottom friction coefficient. In case of each of the structures, the rank distributions of the observed values were constructed and a comparison was made to the selected flow regime, which in a sense is characteristic of this system distribution.

Appendix

The laminar (stationary) solution in the linear approximation satisfies the equations

$$-\frac{\partial p}{\partial x}+\rho G\sin ky+\mu\Delta u-\zeta u=0,\quad -\frac{\partial p}{\partial y}-\rho G\sin kx+\mu\Delta v-\zeta v=0. \tag{8}$$

The vorticity is equal to

$$\omega=\frac{\partial v}{\partial x}-\frac{\partial u}{\partial y}.$$

Assuming $\rho = const$, we obtain from (8) the following equation for vorticity:

$$-\mu\Delta\omega+\zeta\omega+\rho Gk[\cos kx+\cos ky]=0.$$

The solution for ω is given by the formula

$$\omega=\frac{\rho Gk}{\zeta+k^2\mu}[\cos kx+\cos ky].$$

The graph (Fig. 5) shows the frequency distributions of negative vorticity, obtained numerically (blue line) and analytically (red dotted line).

Acknowledgements

The work was carried out with the financial support of the Ministry of Science and Higher Education of the Russian Federation (State Assignment 075-00459-25-00 (Guzev), State Assignment 075-00269-25-00 (Doludenko), State Assignment 124022400174-3 (Ermakov, Posudnevskaya, Fortova). The simulation was carried out using computing facilities of the L.D. Landau Institute for Theoretical Physics of the Russian Academy of Sciences.

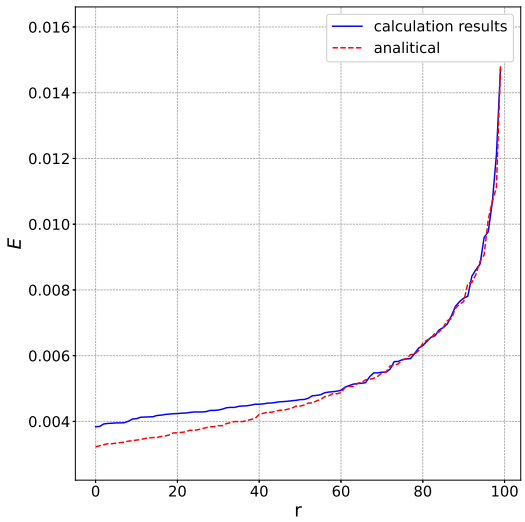


Figure 5. Rank distribution for negative vorticity for the laminar flow regime, constructed analytically and numerically

References

1. Anderson, D.A., Tannehill, J.C., Pletcher, R.H.: Computational Fluid Mechanics and Heat Transfer. Taylor and Francis (2016). https://doi.org/10.1201/b12884

2. Arnold, V.I., Meshalkin, L.D.: A. N. Kolmogorov's seminar on selected issues of analysis (1958-1959). Russ. Math. Surv. 15, 247–250 (1960)

3. Auerbach, F.: Das Gesetz der Bevölkerungskonzentration. Petermanns Geographische Mitteilungen 59, 74–76 (1913)

4. Batchelor, G.K.: Computation of the Energy Spectrum in Homogeneous Two-Dimensional Turbulence. Phys. Fluids 12, 233–239 (1969). https://doi.org/10.1063/1.1692443

5. Bondarenko, N.F., Gak, M.Z., Dolzhansky, F.V.: Laboratory and theoretical models of plane periodic flow. Izvestiia, Fizika Atmosfery i Okeana 15, 1017–1026 (1979)

6. Borue, V.: Inverse energy cascade in stationary two-dimensional homogeneous turbulence. Phys. Rev. Lett. 72, 1475 (1994). https://doi.org/10.1103/PhysRevLett.72.1475

7. Chertkov, M., Connaughton, C., Kolokolov, I., Lebedev, V.: Dynamics of Energy Condensation in Two-Dimensional Turbulence. Phys. Rev. Lett. 99 (2007). https://doi.org/10.1103/PhysRevLett.99.084501

8. Clauset, A., Shalizi, C.R., Newman, M.E.J.: Power-Law Distributions in Empirical Data. SIAM Review 51(4), 661–703 (2009). https://doi.org/10.1137/070710111

9. Clercx, H.J.H.: A Spectral Solver for the Navier-Stokes Equations in the Velocity-Vorticity Formulation for Flows with Two Nonperiodic Directions. J. Comp. Phys. 137, 186–211 (1997). https://doi.org/10.1006/jcph.1997.5799

10. Clercx, H.J.H., Maassen, S.R., van Heijst, G.J.F.: Spontaneous spin-up during the decay of 2D turbulence in a square container with rigid boundaries. Phys. Rev. Lett. 80, 5129 (1998). https://doi.org/10.1103/PhysRevLett.80.5129

11. Clercx, H.J.H., Nielsen, A.H., Torres, D.J., Coutsias, E.A.: Two-dimensional turbulence in square and circular domains with no-slip walls. Eur. J. Mech. B-Fluids 20, 557–576 (2001). https://doi.org/10.1016/S0997-7546(01)01130-X

12. Denisenko, V.V., Doludenko, A.N., Fortova, S.V., *et al.*: Numerical modelling of the Kolmogorov flow in a viscous media, forced by the periodic on space static force. Computer Research and Modeling 14(4), 741–753 (2022). https://doi.org/10.20537/2076-7633-2022-14-4-741-753

13. Doludenko, A.N., Fortova, S.V., Kolokolov, I.V., Lebedev, V.V.: Coherent vortex in a spatially restricted two-dimensional turbulent flow in absence of bottom friction. Physics of Fluids 33, 1–6 (2021). https://doi.org/10.1063/5.0038863

14. Doludenko, A.N., Fortova, S.V., Kolokolov, I.V., Lebedev, V.V.: Coherent vortex versus chaotic state in two-dimension turbulence. Annals of Physics 447 (2022). https://doi.org/10.1016/j.aop.2022.169072

15. Doludenko, A.N., Kulikov, Y.M., Savel'ev, A.S.: Chaotic flow under the influence of volumetric force. Computer Research and Modeling 16(4), 883–912 (2024). https://doi.org/10.20537/2076-7633-2024-16-4-883-912

16. Fock, V.A.: Konfigurationsraum und zweite Quantelung. Zeitschrift für Physik 75, 622–647 (1932). https://doi.org/10.1007/BF01344458

17. Francois, N., Xia, Y., Punzmann, H., *et al.*: Three-Dimensional Fluid Motion in Faraday Waves: Creation of Vorticity and Generation of Two-Dimensional Turbulence. Phys. Rev. X 4 (2014). https://doi.org/10.1103/PhysRevX.4.021021

18. Gleason, H.A.: The Significance of Raunkiaer's Law of Frequency. Ecology 10(4), 406–408 (1929). https://doi.org/10.2307/1931149

19. Golitsyn, G.S.: A. N. Kolmogorov's 1934 paper is the basis for explaining the statistics of natural phenomena of the macrocosm. Physics-Uspekhi 67, 80–90 (2024). https://doi.org/2023.05.039355

20. Gutenberg, B., Richter, C.F.: Frequency of earthquakes in California. Bulletin of the Seismological Society of America 34(4), 185–188 (1944). https://doi.org/10.1785/BSSA0340040185

21. Guzev, M.A., Fortova, S.V., Doludenko, A.N., *et al.*: Maslov Rank Distributions for the Analysis of Two-Dimensional and Quasi-Two-Dimensional Turbulent Flows. Russian Journal of Mathematical Physics 31, 438–449 (2024). https://doi.org/10.1134/S1061920824030075

22. Guzev, M.A., Nikitina, E.Y., Chernysh, E.V.: V. P. Maslov's Approach to the Analysis of Rank Distributions. Russian Journal of Mathematical Physics 28, 56–65 (2021). https://doi.org/10.1134/S1061920821010064

23. Kolmogorov, A.N.: Doklady Akademii Nauk SSSR 30, 299–303 (1941)

24. Kolmogorov, A.N.: Three approaches to the definition of the concept "quantity of information". Probl. Peredachi Inf. 1, 3–11 (1965)

25. Kolokolov, I.V., Lebedev, V.V.: Structure of coherent vortices generated by the inverse cascade of two-dimensional turbulence in a finite box. Phys. Rev. E 93 (2016). https://doi.org/10.1103/PhysRevE.93.033104

26. Kolokolov, I.V., Lebedev, V.V.: Large-scale flow in two-dimensional turbulence at static pumping. JETP Lett. 106, 659–661 (2017). https://doi.org/10.1134/S0021364017220027

27. Kolokolov, I.V., Lebedev, V.V.: Coherent vortex in two-dimensional turbulence: Interplay of viscosity and bottom friction. Phys. Rev. E 102 (2020). https://doi.org/10.1103/PhysRevE.102.023108

28. Kraichnan, R.H.: Inertial Ranges in Two-Dimensional Turbulence. Phys. Fluids 10, 1417–1423 (1967). https://doi.org/10.1063/1.1762301

29. Kudryavtsev, A.N., Kuznetsov, E.A., Sereshchenko, E.V.: Statistical properties of freely decaying two-dimensional hydrodynamic turbulence. JETP Letters 96, 699–705 (2013). https://doi.org/10.1134/S0021364012230105

30. Kuznetsov, E.A., Naulin, V., Nielsen, A.H., Rasmussen, J.J.: Effects of sharp vorticity gradients in two-dimensional hydrodynamic turbulence. Physics of Fluids 19(10), 10 (2007). https://doi.org/10.1063/1.2793150

31. Kuznetsov, E.A., Naulin, V., Nielsen, A.H., Rasmussen, J.J.: Sharp vorticity gradients in two-dimensional turbulence and the energy spectrum. Theoretical and Computational Fluid Dynamics 24, 253–258 (2010). https://doi.org/10.1007/s00162-009-0135-4

32. Kuznetsov, E.A., Sereshchenko, E.V.: Anisotropic characteristics of the kraichnan direct cascade in two-dimensional hydrodynamic turbulence. JETP Letters 102(11), 760–765 (2015). https://doi.org/10.1134/S0021364015230083

33. Laurie, J., Boffetta, G., Falkovich, G., *et al.*: Universal Profile of the Vortex Condensate in Two-Dimensional Turbulence. Phys. Rev. Lett. 113 (2014). https://doi.org/10.1103/PhysRevLett.113.254503

34. Lees, R.B.: Logique, langage et théorie de l'information by Léo Apostel, Benoit Mandelbrot, Albert Morf. Language 35(2) (1959). https://doi.org/10.2307/410536

35. Lotka, A.J.: The Frequency Distribution of Scientific Productivity. Journal of the Washington Academy of Sciences 16(12), 317–323 (1926)

36. Lutsenko, D.V.: Combinatorial Theory of Rank Dynamics. Kaliningrad: Tekhnotsenoz (2018)

37. MacCormack, R.W.: A Numerical Method for Solving the Equations of Compressible Viscous Flow. AIAA Journal 20, 1275–1281 (1982). https://doi.org/10.2514/6.1981-110

38. Mandelbrot, B.: Statistical Macro-Linguistics. Il Nuovo Cimento (1955-1965) 13, 518–520 (1959). https://doi.org/10.1007/bf02724682

39. Maslov, V.P.: On one general theorem of set theory leading to the Gibbs, Bose-Einstein, Pareto distribution and the Mandelbrot–Zipf law for the stock market. Mat. notes. 78, 807–813 (2005). https://doi.org/10.1007/s11006-005-0186-9

40. Maslov, V.P.: Quantum Linguistic Statistics. Russian Journal of Mathematical Physics 13, 315–325 (2006). https://doi.org/10.1134/S1061920806030071

41. Mitzenmacher, M.: A Brief History of Generative Models for Power Law and Lognormal Distributions. Internet Mathematics 1(2), 226–251 (2004). https://doi.org/10.1080/15427951.2004.10129088

42. Molenaar, D., Clercx, H.J.H., van Heijst, G.J.F.: Angular momentum of forced 2D turbulence in a square no-slip domain. Physica D 196, 329–340 (2004). https://doi.org/10.1016/j.physd.2004.06.001

43. Posudnevskaya, A.O., Fortova, S.V., Doludenko, A.N., *et al.*: Numerical Study of Transient Regimes of Kolmogorov Flow in a Square Cell. Computational Mathematics and Mathematical Physics 64, 2102–2110 (2024). https://doi.org/10.1134/S0965542524701033

44. Puzachenko, Y.G.: Rank distribution in ecology and nonextensive statistical mechanism. Archives of Zoological Museum of Lomonosov Moscow State University 54, 42–71 (2016)

45. Smith, L.M., Yakhot, V.: Bose condensation and small-scale structure generation in a random force driven 2D turbulence. Phys. Rev. Lett. 171, 352 (1993). https://doi.org/10.1103/PhysRevLett.71.352

46. Smith, L.M., Yakhot, V.: Finite-size effects in forced two-dimensional turbulence. J. Fluid Mech. 274, 115–138 (1994). https://doi.org/10.1017/S0022112094002065

47. Sommeria, J.: Experimental study of the two-dimensional inverse energy cascade in a square box. J. Fluid Mech 170, 139–168 (1986). https://doi.org/10.1017/S0022112086000836

48. Tsang, Y.K.: Nonuniversal velocity probability densities in two-dimensional turbulence: The effect of large-scale dissipation. Phys. Fluids 22 (2010). https://doi.org/10.1063/1.3504377

49. Tsang, Y.K., R.Young, W.: Forced-dissipative two-dimensional turbulence: A scaling regime controlled by drag. Phys. Rev. E 79 (2009). https://doi.org/10.1103/PhysRevE.79.045308

50. Xia, H., Punzmann, H., Falkovich, G., Shats, M.G.: Turbulence-Condensate Interaction in Two Dimensions. Phys. Rev. Lett. 101 (2008). https://doi.org/10.1103/PhysRevLett.101.194504

51. Xia, H., Shats, M., Falkovich, G.: Spectrally condensed turbulence in thin layers. Phys. Fluids 21 (2009). https://doi.org/10.1063/1.3275861

52. Zipf, G.K.: Human behavior and the principle of least effort. Addison-Wesley Press (1949)

DOI: 10.14529/jsfi250105

Application-Specific Parallel Linear Solver for Nonlinear Harmonics Method with Implicit Time Integration

Alexey P. Duben[1], *Andrey V. Gorobets*[1]

The present paper covers specific parallel implementation details of the nonlinear harmonics (NLH) method within an implicit time integration framework. The NLH method plays an important role in industrial turbomachinery applications as it accounts for unsteady effects in modelling of compressors and turbines on a base of low-cost stationary approaches: the flow is modelled using the Reynolds-Averaged Navier–Stokes approach, the mixing plane method is used for the rotor-stator interface, and only one periodic sector of a blade passage per row is considered. The main focus is on the adaptation of the linear solver used in the Newtonian process of the implicit scheme. The goal of this work is to significantly reduce memory consumption and improve performance. This goal is achieved by using a specialized block sparse matrix storage format, adapted linear solver preconditioners with approximate inverse diagonal blocks, and a combination of single- and double-precision real number formats.

Keywords: turbomachinery, rotor-stator interaction, non-linear harmonics method, unstructured mesh, parallel CFD, supercomputer.

Introduction

The nonlinear harmonics (NLH) method, first proposed in [1] and later applied to realistic turbomachines [2, 3], accounts for unsteady effects in simulations of compressors and turbines using low-cost stationary approaches. The flow is modeled on a base of the Reynolds-Averaged Navier–Stokes (RANS) approach. Only one periodic sector of blade passage per row is considered, which is crucial due to the typically large number of blades in each row, from tens to hundreds. In particular, the NLH is incorporated [4, 5] into the commercial solver Cadence Fidelity Fine Turbo, well known as one of the most efficient simulation tools for turbomachinery.

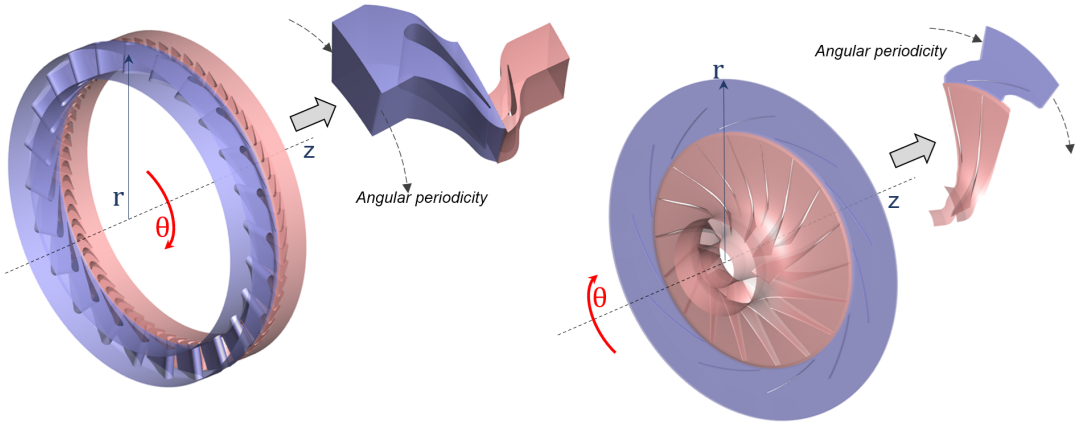


Figure 1. Angular periodicity

The mixing plane (MP) method [6, 7] is used for rotor-stator interfaces, which assumes uniformity of the flow in the circumferential direction at the interface. The NLH method allows capturing the unsteady interaction between adjacent rows by transmitting the perturbations related to the blade passing frequency through the mixing-plane interface.

[1]Keldysh Institute of Applied Mathematics, RAS, Moscow, Russian Federation

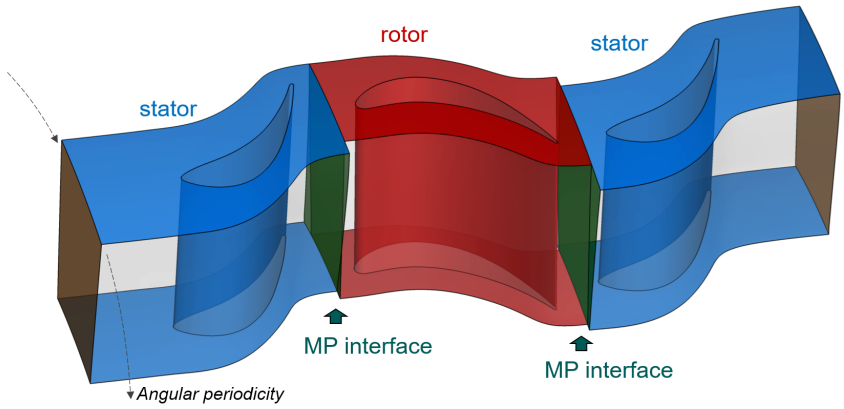


Figure 2. MP rotor-stator interfaces

The NLH technology was successfully implemented [8, 9] within a higher-accuracy numerical algorithm for unstructured meshes in the NOISEtte [10] code. The code has multilevel heterogeneous parallelization [11] by means of MPI, OpenMP and OpenCL parallel standards.

The NLH method needs to solve for each harmonic a supplementary system of transport equations for complex harmonic amplitudes. Thus, each harmonic adds twice as many variables per cell (real and imaginary parts) as in the baseline stationary RANS simulation. What is worse, since the time integration is implicit, the block size of the Jacobian matrix increases from 5 to 10, which increases the storage size and computation cost of the solution by about 4 times. Therefore, the present study is focused on reduction of this undesirable memory consumption.

The rest of the paper is organized as follows. Section 1 briefly describes the NLH technology. Section 2 is devoted to the adaptation of the linear solver by means of a customized sparse block matrix format. The results of memory consumption reduction and performance improvement are presented in Section 3. Finally, the conclusions are summarized.

1. Non-Linear Harmonics Method

1.1. Mathematical Model

The NLH is based on solving the Reynolds-Averaged Navier–Stokes (RANS) equations for simulating a turbulent compressible viscous flow:

$$\frac{\partial \mathbf{Q}}{\partial t} + \nabla \cdot \mathcal{F}^{\mathrm{C}}(\mathbf{Q}) - \nabla \cdot \mathcal{F}^{\mathrm{D}}(\mathbf{Q}) = 0, \tag{1}$$

where $\mathbf{Q} = (\rho,\ \mathbf{m},\ E)^{\intercal}$ is the vector of averaged conservative variables, ρ is the density, $\mathbf{m} = \rho\mathbf{u}$, where $\mathbf{u} = \{u, v, w\}$ is the velocity vector, $E = \rho(e + \mathbf{u}^2/2)$ is the total energy, e is the specific internal energy. The convective and diffusive fluxes read:

$$\mathcal{F}^{\mathrm{C}}(\mathbf{Q}) = \begin{pmatrix} \mathbf{m} \\ \mathbf{u} \otimes \mathbf{m} + p\mathbf{I} \\ (E + p)\mathbf{u} \end{pmatrix}, \quad \mathcal{F}^{\mathrm{D}}(\mathbf{Q}) = \begin{pmatrix} 0 \\ \boldsymbol{\sigma} \\ \boldsymbol{\sigma} \cdot \mathbf{u} - \mathbf{q} \end{pmatrix}, \tag{2}$$

where $\mathbf{I}$ is the unit tensor, $\boldsymbol{\sigma}(\mathbf{u}) = \{\sigma_{jk}\}(\mathbf{u}) = \mu_{\mathrm{eff}}(\nabla_j u_k + \nabla_k u_j - \delta_{jk}\mathrm{div}\mathbf{u})$ is the stress tensor, δ_{jk} is the Kronecker symbol, $\mathbf{q} = -\frac{\gamma\mu}{\mathrm{Pr}}\nabla e$ is the heat flux vector. $\gamma = c_P/c_V$ is the specific heat ratio, c_P and c_V are the specific heat capacities at constant pressure and constant volume, respectively. $\mathrm{Pr} = c_P\mu/\kappa$ is the Prandtl number, κ is the thermal conductivity coefficient. The

perfect gas equation of state is assumed: $p = (\gamma - 1)\rho e$. $\mu_{\text{eff}} = \mu + \mu_t$, μ is the dynamic viscosity, $\mu_t = \rho\nu_t$, ν_t is the turbulent viscosity defined by a particular turbulence model.

The conservative variables $\mathbf{Q}$ within the NLH method are split into the time-averaged part $\overline{\mathbf{Q}}(x)$ and periodic pulsations $\mathbf{Q}'(t,x)$:

$$\mathbf{Q}(t,x) = \overline{\mathbf{Q}}(x) + \mathbf{Q}'(t,x). \tag{3}$$

The pulsation part $\mathbf{Q}'(t,x)$ is represented as a sum of complex harmonic components:

$$\mathbf{Q}'(t,x) = \frac{1}{2}\sum_{k=1}^{\infty}\left[\widetilde{\mathbf{Q}}_k e^{ik\omega_0 t} + \widetilde{\mathbf{Q}}_{-k} e^{-ik\omega_0 t}\right], \tag{4}$$

where i is the imaginary unit, $\widetilde{\mathbf{Q}}_k = \widetilde{\mathbf{Q}}_{a,k} + i\widetilde{\mathbf{Q}}_{b,k}$, $\widetilde{\mathbf{Q}}_{a,k} = \Re\left(\widetilde{\mathbf{Q}}_k\right)$ and $\widetilde{\mathbf{Q}}_{b,k} = \Im\left(\widetilde{\mathbf{Q}}_k\right)$ are the real and imaginary parts of the amplitude for k-th harmonic, respectively, $\widetilde{\mathbf{Q}}_{-k} = \widetilde{\mathbf{Q}}_{a,k} - i\widetilde{\mathbf{Q}}_{b,k}$ is the complex-conjugate for $\widetilde{\mathbf{Q}}_k$. The number of harmonics N_h defines how accurate the approximation of $\mathbf{Q}'(t,x)$ in (4) is, the more harmonics, the more accurate it is, but at a higher computational cost. The harmonic indices k are defined by the blade passing frequencies (BPF) Ω_{BPF} of adjacent rows: $k = n \cdot k_{\text{BPF}}$, $n = 1, 2, ..., N_h$. k_{BPF}, in turn, is defined as $k_{\text{BPF}} = \Omega_{\text{BPF}}/\Omega_{\text{rot}}$, where Ω_{rot} is the rotation frequency ($k_{\text{BPF}} \in \mathbb{N}$ by definition). Thus, every domain (either rotor or stator) has two sets of harmonic indices k, which are proportional to k_{BPF} from adjacent domains.

According to (3), the RANS system (1) is decomposed into the system for averaged variables and the set of systems for amplitudes of each harmonic. The system for averaged variables $\overline{\mathbf{Q}}$ reads:

$$\frac{\partial \overline{\mathbf{Q}}}{\partial t} + \nabla \cdot \overline{\mathcal{F}}^{C}\left(\overline{\mathbf{Q}}, \mathbf{Q}'\right) - \nabla \cdot \overline{\mathcal{F}}^{\text{D}}(\overline{\mathbf{Q}}, \mathbf{Q}') = 0, \tag{5}$$

$\overline{\mathcal{F}}^{C}\left(\overline{\mathbf{Q}}, \mathbf{Q}'\right) = \mathcal{F}^{\text{C}}(\overline{\mathbf{Q}}) + \mathcal{F}^{\text{C}}_{\text{NLH}}(\mathbf{Q}')$, $\overline{\mathcal{F}}^{\text{D}}(\overline{\mathbf{Q}}, \mathbf{Q}') = \mathcal{F}^{\text{D}}(\overline{\mathbf{Q}}) + \mathcal{F}^{\text{D}}_{\text{NLH}}(\mathbf{Q}')$. Nonlinear deterministic stress contributions of harmonic amplitudes are of the form $\mathcal{F}^{\text{C}}_{\text{NLH}}(\mathbf{Q}') = (0, \overline{\mathbf{m}' \otimes \mathbf{u}'}, \overline{(\mathbf{E}+\mathbf{p})'\mathbf{u}'})^{\intercal}$ and $\mathcal{F}^{\text{D}}_{\text{NLH}}(\mathbf{Q}') = (0, 0, 0, 0, \overline{\boldsymbol{\sigma}' \cdot \mathbf{u}'})^{\intercal}$, where $\boldsymbol{\sigma}' = \boldsymbol{\sigma}(\mathbf{u}')$. The system (5) is distinguished from the system (1) by the presence of the nonlinear terms $\mathcal{F}^{\text{C}}_{\text{NLH}}$ and $\mathcal{F}^{\text{D}}_{\text{NLH}}$, which are defined by $\mathbf{Q}'$.

For brevity, the index k of harmonic amplitude for ω_k frequency will be omitted from now on when referring to harmonic amplitude vectors $\widetilde{\mathbf{Q}}_k$. Thus, the system for a harmonic amplitude vector $\widetilde{\mathbf{Q}}$ is

$$\frac{\partial \widetilde{\mathbf{Q}}}{\partial t} + \nabla \cdot \widetilde{\mathcal{F}}^{C}\left(\widetilde{\mathbf{Q}}, \overline{\mathbf{Q}}\right) + i\omega\widetilde{\mathbf{Q}} - \nabla \cdot \widetilde{\mathcal{F}}^{\text{D}}(\widetilde{\mathbf{Q}}, \overline{\mathbf{Q}}) = 0, \tag{6}$$

$$\widetilde{\mathcal{F}}^{\text{C}} = \begin{pmatrix} \widetilde{\mathbf{m}} \\ \widetilde{\mathbf{u}} \otimes \overline{\mathbf{m}} + \overline{\mathbf{u}} \otimes \widetilde{\mathbf{m}} + \widetilde{p}\mathbf{I} \\ \left(\widetilde{E} + \widetilde{p}\right)\overline{\mathbf{u}} + (\overline{E} + \overline{p})\widetilde{\mathbf{u}} \end{pmatrix}, \quad \widetilde{\mathcal{F}}^{\text{D}} = \begin{pmatrix} 0 \\ \widetilde{\boldsymbol{\sigma}} \\ \widetilde{\boldsymbol{\sigma}} \cdot \overline{\mathbf{u}} + \overline{\boldsymbol{\sigma}} \cdot \widetilde{\mathbf{u}} - \widetilde{\mathbf{q}} \end{pmatrix}, \tag{7}$$

where $\widetilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma}(\widetilde{\mathbf{u}})$, $\widetilde{\mathbf{q}} = -\frac{\gamma\mu}{\text{Pr}}\nabla\widetilde{e}$. $\overline{\mu}_{\text{eff}} = \overline{\mu} + \overline{\mu}_t$, $\overline{\mu}_t$ is defined by a turbulence model closing the equations for averaged variables.

Linear systems of equations for harmonic amplitudes (6)–(7) are independent of each other due to orthogonality of harmonics. The equations for averaged variables and harmonic amplitudes are closed by several relations (see [8] for more details), including the following ones: $\overline{f\,g} = \overline{f}\,\overline{g} + \overline{f'g'}$; $(fg)' = f'\,\overline{g} + \overline{f}\,g'$; $\overline{f'g'} = 0.5\sum_{k=1}^{N_h}\left[\Re(\widetilde{f}_k)\Re(\widetilde{g}_k) + \Im(\widetilde{f}_k)\Im(\widetilde{g}_k)\right]$.

1.2. Discretization

The convective terms for harmonic amplitudes $\nabla \cdot \widetilde{\mathcal{F}}^{\mathrm{C}}$ in (6) are discretized in space on an unstructured mesh using the edge-based reconstruction (EBR) [12] (for smooth solutions) and shock-capturing EBR-TVD or EBR-WENO [13] schemes with the scheme [8] based on the Roe [14] scheme for solving the Riemann problem. According to it, the numerical flux $\mathbf{F}_{ij}$ at the surface associated with the edge ij, which connects nodes i and j, is defined as

$$\mathbf{F}_{ij} = \frac{1}{2}\left[\mathcal{F}(\widetilde{\mathbf{Q}}_{ij}, \overline{\mathbf{Q}}_i) + \mathcal{F}(\widetilde{\mathbf{Q}}_{ji}, \overline{\mathbf{Q}}_j)\right] \cdot \mathbf{n}_{ij} - \frac{1}{2}\mathbf{S}_{ij}|\mathbf{\Lambda}_{ij}|\mathbf{S}_{ij}^{-1}\left[\widetilde{\mathbf{Q}}_{ji} - \widetilde{\mathbf{Q}}_{ij}\right], \tag{8}$$

where $\widetilde{\mathbf{Q}}_{ij}$ and $\widetilde{\mathbf{Q}}_{ji}$ are the pre-decay values of conservative harmonic amplitudes, calculated using a EBR-based scheme, at the nodes i and j, respectively. $\mathbf{n}_{ij} = \mathbf{s}_{ij}/|\mathbf{s}_{ij}|$ is the unit vector aligned with $\mathbf{s}_{ij} = -\mathbf{s}_{ji}$ – the surface associated with edge ij. Considering moving of the control volume with the speed $\mathbf{V} = \mathbf{\Omega}_{\mathrm{rot}} \times \mathbf{r}$ ($\mathbf{\Omega}_{\mathrm{rot}}$ is the rotation vector, $\mathbf{r} = (x, y, z)$ is the radius vector of the node), the flux function $\mathcal{F}(\widetilde{\mathbf{Q}}_{ij}, \overline{\mathbf{Q}}_i)$ at the node i can be written as

$$\mathcal{F}(\widetilde{\mathbf{Q}}_{ij}, \overline{\mathbf{Q}}_i) = \begin{pmatrix} \widetilde{\mathbf{m}}_{ij} \\ \widetilde{\mathbf{u}}_{ij} \otimes \overline{\mathbf{m}}_i + \overline{\mathbf{u}}_i \otimes \widetilde{\mathbf{m}}_{ij} + \widetilde{p}_{ij}\mathbf{I} \\ \left(\widetilde{E}_{ij} + \widetilde{p}_{ij}\right)\overline{\mathbf{u}}_i + (\overline{E}_i + \overline{p}_i)\widetilde{\mathbf{u}}_{ij} \end{pmatrix} - \widetilde{\mathbf{Q}}_{ij} \cdot \mathbf{V}. \tag{9}$$

The $\mathcal{F}(\widetilde{\mathbf{Q}}_{ji}, \overline{\mathbf{Q}}_j)$ has the form similar to (9). $\mathbf{\Lambda}_{ij}$ is the diagonal matrix of the eigenvalues of $A_{ij} = d(\mathcal{F} \cdot \mathbf{n}_{ij})/d\widetilde{\mathbf{Q}}$ [14]. $\mathbf{S}_{ij}$ is the matrix of the corresponding eigenvectors. We should emphasize that both $\mathbf{\Lambda}_{ij}$ and $\mathbf{S}_{ij}$ depend only on the averaged variables $\overline{\mathbf{Q}}_i$ and $\overline{\mathbf{Q}}_j$. For the viscous terms $\nabla \cdot \widetilde{\mathcal{F}}^{\mathrm{D}}$ in (6), the method of averaged element splitting (AES) [15] is used.

Since averaged variables and harmonic amplitudes do not depend on time, a pseudo-time iterative process is used to obtain a stationary solution. To do so, the implicit time integration backward differentiation formula (BDF) with a quasi-Newton linearization is used. As already mentioned above, the system (6) differs from (1) by the presence of $\mathcal{F}^{\mathrm{C}}_{\mathrm{NLH}}$ and $\mathcal{F}^{\mathrm{D}}_{\mathrm{NLH}}$ terms. We consider them as source terms independent of $\overline{\mathbf{Q}}$, so (6) is solved using the same numerical scheme as for usual RANS without considering nonstationary perturbations. Also, it enables to perform pseudo-time advancing for the system of averaged variables (5) and harmonic amplitudes (6) separately. At each time step, first a sub-step for averaged variables is computed with harmonic amplitudes frozen, then a sub-step for harmonic amplitudes is carried out with updated averaged variables.

A semidiscrete approximation of (6) for a harmonic amplitude can be formulated as

$$\frac{d\widetilde{\mathbf{Q}}}{dt} - \widetilde{\mathbf{\Phi}}(\widetilde{\mathbf{Q}}, \overline{\mathbf{Q}}, \omega) = 0,$$

where $\widetilde{\mathbf{\Phi}} = -\nabla \cdot \widetilde{\mathcal{F}}^{C}\left(\widetilde{\mathbf{Q}}, \overline{\mathbf{Q}}\right) - i\omega\widetilde{\mathbf{Q}} + \nabla \cdot \widetilde{\mathcal{F}}^{\mathrm{D}}(\widetilde{\mathbf{Q}}, \overline{\mathbf{Q}})$. The implicit scheme reads

$$\left(\frac{\mathbf{I}}{\tau^n} + \widetilde{J}\right)(\widetilde{\mathbf{Q}}^{n+1} - \widetilde{\mathbf{Q}}^n) = \widetilde{\mathbf{\Phi}}\left(\widetilde{\mathbf{Q}}^n, \overline{\mathbf{Q}}^n, \omega\right), \tag{10}$$

where τ^n is the timestep at the n-th pseudo-time step, $\widetilde{J} \approx d\widetilde{\mathbf{\Phi}}/d\widetilde{\mathbf{Q}}$ – an approximated flux Jacobian. We should emphasize that the Jacobian matrix $\widetilde{J}$ for both convective $\widetilde{\mathcal{F}}^{C}$ and diffusive $\widetilde{\mathcal{F}}^{D}$ fluxes depends only on averaged variables $\overline{\mathbf{Q}}$, while a contribution from the source term $i\omega\widetilde{\mathbf{Q}}$ depends on ω only.

The parallel preconditioned BiCGStab solver [16] is applied for the Jacobi linear system. Gauss–Seidel method-based parallel preconditioners [17] for block sparse matrices are used.

Each system of equations for harmonic amplitudes includes the coupled between each other real and imaginary parts. Thus, the blocks of the sparse block matrices for harmonic amplitudes are 10×10, while the system for averaged variables has blocks 5×5. This means that the NLH method requires 4 times more memory to store the matrix and 4 times more computational cost of the sparse matrix-vector product. Fortunately, only one matrix is stored for all the harmonics, since the harmonics are solved one by one.

The boundary conditions for harmonic amplitudes, inlet, outlet, and solid surfaces, are implemented in the same way as for the averaged variables.

The mixing plane (MP) method [7] is used for the rotor-stator interface. The boundary conditions for harmonic amplitudes at the rotor-stator interface are implemented similarly to [2] basing on the MP functionality. For the sector periodicity on the mixing-plane interface, the NLH method uses generalized periodic conditions with phase shift for the closure of harmonic amplitudes with different number of blades in adjacent rows, as in [18]. Further details on the numerical method can be found in [8].

2. Linear Solver Adaptation

The NOISEtte code has an in-house heterogeneous parallel linear solver that uses the preconditioned BiCGStab [16] iterative method with several preconditioners based on the Gauss–Seidel method [17]. It supports multilevel MPI+OpenMP+OpenCL parallelization for running on numerous CPUs and GPUs. The solver is a multi-system solver for systems with block sparse matrices that share the same portrait. The matrices are stored in a block CSR (Compressed Sparse Row) format, with dense blocks of coefficients placed linearly in memory in a row-by-row order. Obviously, sharing one portrait across multiple matrices saves memory for storing portraits. In addition, the group solution for several systems at once allows to reduce the network latency overhead in a distributed parallel mode. MPI messages in the common iterative process are grouped, including halo update operations in sparse matrix-vector products (SpMV) and reduction exchanges in dot products, which reduces the number of messages by several times. As soon as the solution of one of the systems satisfies the residual criterion, it is excluded from the solution process by setting the corresponding convergence flag. Typically, in a basic stationary RANS simulation, there is one system with 5×5 blocks for the 5 main variables (density, 3 velocity vector components, pressure) and from 1 to 4 additional systems for the turbulent model variables (with block sizes 1×1 or 2×2), depending on a particular RANS model.

In the NOISEtte code, a mixed-precision approach is used, in which the Jacobian matrix is stored in single-precision floating-point format, while the mesh functions and discrete operators coefficients are represented in double-precision format (see [11] for details). The linear solver thus operates mainly in single-precision format: vectors and matrices are in single-precision, while BiCGStab scalar coefficients are in double-precision, as well as some intermediate values, such as sum accumulators in dot products, etc. For stability reasons, inversion of diagonal blocks can also be performed in double precision, the result of which is then converted to single precision.

This baseline solver can be directly used for the NLH method as well, but with a matrix block size of 10×10, which is very wasteful. In order to reduce memory consumption and computational cost, the solver has been upgraded with a customized matrix storage format and dedicated matrix-vector product functions. The preconditioner has also been significantly redesigned for

the new matrix format, including the matrix diagonal blocks inversion function and inverted diagonal blocks storage.

2.1. Customized Matrix Format for Harmonics Matrices

The matrix portrait corresponds to nodal adjacency via mesh edges. In the matrices for harmonics, each 10×10 matrix block corresponds to 5 real and 5 imaginary parts of complex variables in nodes. If the variables in blocks are ordered accordingly, first real parts, then imaginary, then the diagonal and off-diagonal 10×10 blocks have the following structure, respectively:

$$\mathbf{A}_{ii} = \begin{pmatrix} \mathcal{A}_{ii} & -\omega v_i \mathcal{I} \\ \omega v_i \mathcal{I} & \mathcal{A}_{ii} \end{pmatrix}, \quad \mathbf{A}_{ij} = \begin{pmatrix} \mathcal{A}_{ij} & 0 \\ 0 & \mathcal{A}_{ij} \end{pmatrix}, \quad i \neq j, \tag{11}$$

where $\mathcal{A}_{ii}$ are dense 5×5 blocks, ω is the frequency to which these harmonic amplitudes correspond (ω_k, actually), v_i is the volume of i-th cell, and $\mathcal{I}$ is the 5×5 identity matrix.

This specific matrix structure allows to easily reduce memory consumption for matrix storage by about 4 times. Instead of 100 values, 26 values are stored: 25 values of the 5×5 block $\mathcal{A}_{ii}$ and one more value ωv_i.

Furthermore, the matrices for harmonics depend only on averaged variables except diagonal coefficients, which contain contributions defined by the source terms $i\omega\widetilde{\mathbf{Q}}$. Thus, once the Jacobian matrix is filled for the averaged variables, it can be reused for the harmonics with only the diagonal elements of diagonal blocks being updated. To implement this memory-saving strategy, apart from minor updates in the matrix format, all the necessary operations involving products with matrix blocks must be provided for this customized block representation. In case of the BiCGStab solver, only the SpMV operation needs to be updated, and the preconditioner may also require changes.

2.2. Preconditioner with Block-Diagonal Inversion

Situation with the preconditioner based on the block Jacobi method, $\mathbf{x}_i^{m+1} = {\mathbf{A}_{ii}}^{-1}\left(\mathbf{b}_i - \sum_{j\neq i}\mathbf{A}_{ij}\mathbf{x}_j^m\right)$, or the block Gauss–Seidel method, $\mathbf{x}_i^{m+1} = {\mathbf{A}_{ii}}^{-1}\left(\mathbf{b}_i - \sum_{j=1}^{i-1}\mathbf{A}_{ij}\mathbf{x}_j^{m+1} - \sum_{j=i+1}^{n}\mathbf{A}_{ij}\mathbf{x}_j^m\right)$, needs more attention, because inverted diagonal blocks are required to obtain solution on the next iteration, $m+1$, of the inner iterative process. Inverted diagonal blocks do not necessarily follow the original block structure, and storage for full 10×10 blocks is required (and about 8 times more arithmetic operations for the inversion compared to 5×5 blocks)

The matrices for harmonics could be treated as block matrices of 5×5 blocks with a portrait 4 times larger. In this case, only one 5×5 diagonal block, $\mathcal{A}_{ii}$, needs to be inverted and stored for each 10×10 block $\mathbf{A}_{ii}$. This requires less computational effort to invert the diagonal blocks and 4 times less memory to store it. On the other hand, the Gauss-Seidel block method would require a double pass for each 10×10 block row then (since it is considered as two 5×5 block rows), which is inefficient enough to overcome all the profit. So we approximate the inverted 10×10 blocks in such a way that only the 5×5 diagonal blocks are stored, while still dealing with the 10×10 block rows in the Jacobi and Gauss–Seidel methods, respectively: $\mathbf{x}_i^{m+1} = {\mathbf{A}_{ii}}^{-1}\left(\mathbf{b}_i - \sum_{j\neq i}\mathbf{A}_{ij}\mathbf{x}_j^m - \mathbf{\Omega}\mathbf{x}_i^m\right)$,

$\mathbf{x}_i^{m+1} = \tilde{\mathbf{A}}_{ii}^{-1}\left(\mathbf{b}_i - \sum_{j=1}^{i-1}\mathbf{A}_{ij}\mathbf{x}_j^{m+1} - \sum_{j=i+1}^{n}\mathbf{A}_{ij}\mathbf{x}_j^{m} - \mathbf{\Omega}\mathbf{x}_i^{m}\right)$, where

$$\tilde{\mathbf{A}}_{ii}^{-1} = \begin{pmatrix} {\mathcal{A}_{ii}}^{-1} & 0 \\ 0 & {\mathcal{A}_{ii}}^{-1} \end{pmatrix}, \quad \mathbf{\Omega} = \begin{pmatrix} 0 & -\omega v_i \mathcal{I} \\ \omega v_i \mathcal{I} & 0 \end{pmatrix}.$$

Using this approximate inversion, complemented with the minor modification by including the extra term $\mathbf{\Omega}\mathbf{x}_i^m$, allows saving a lot of computational effort and memory on dealing with inverted diagonal blocks. However, it is necessary that such an approximation does not spoil the solver convergence process (otherwise, the solver may need many more iterations, which will overcome all the gains). To verify this, a direct comparison is made with the original variant with full storage of 10×10 inverted blocks, results are demonstrated in the following section.

3. Performance Analysis

3.1. Cases Description

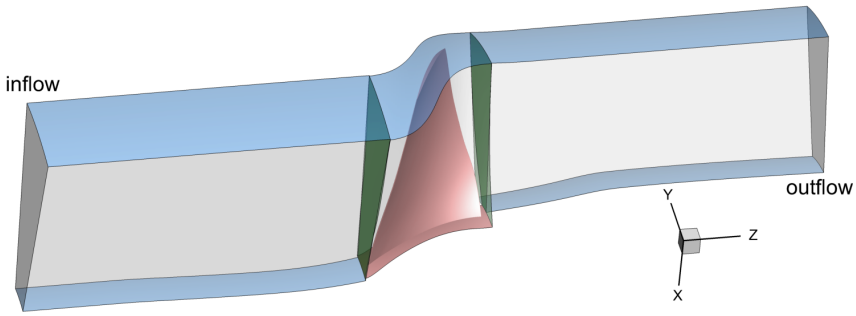
Two model cases are used to demonstrate the performance of the modified implicit solver. Simulations for both test cases were performed using the Menter SST [21] turbulence model. The hexahedral meshes for the test cases are built using the TurboR&D.Mesher software [22].

The first case is an isolated rotor with 22-blade impeller, NASA Rotor-67 [19], which is widely used for validation in turbomachinery applications. Although there is no stator, the computational domain is divided into three parts, as shown in Fig. 3, so that the static subdomains conduct the perturbations generated by the rotating impeller, which are captured by the NLH method. The previously obtained head-capacity characteristics of the Rotor-67 are presented in [9]. Here, the maximum efficiency regime of the rotor is considered for the performance tests. The hexahedral computational mesh contains 3.58 million nodes (single-blade sector with periodic boundary conditions). The same number of periodicity sectors is applied to all three subdomains.

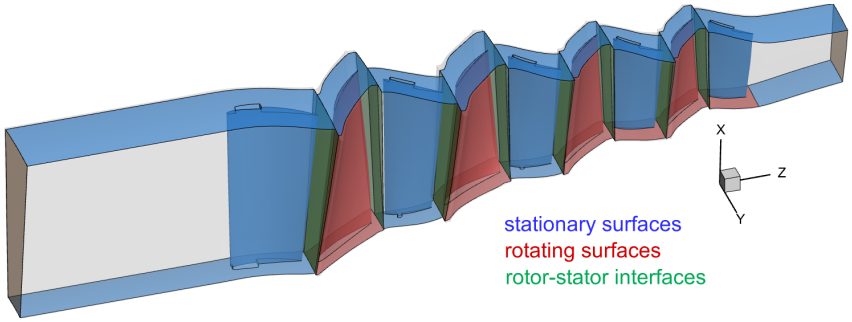
The second case is more computationally intensive, it is part of a model axial compressor of a gas turbine engine with a scaled number of blades (see Fig. 3, and further details can be found in [20]). The configuration considered consists of an inlet guide vane and four stages of rotor and stator, which in total represents 8 rotor-stator interfaces. The mesh for this configuration contains 11.9 million nodes. As for the Rotor-67, head-capacity characteristics can be found in [9]. The highest efficiency regime is considered here for performance tests.

3.2. Demonstration of Performance Improvement

Reduced memory consumption and increased performance are demonstrated in comparison with the previous NLH method implementation [9] on the Rotor-67 model test case. The hexahedral mesh contains about 3.6 million nodes. The total memory consumption depending on the number of harmonics is shown in Fig. 4. It can be seen that the additional memory consumption for storing the matrices for harmonics has become negligible. The performance ratio depends on the number of harmonics, since it is the solver for harmonic systems that has been accelerated. For instance, in the typical case of using three harmonics, a speedup of 1.6 times has been achieved.

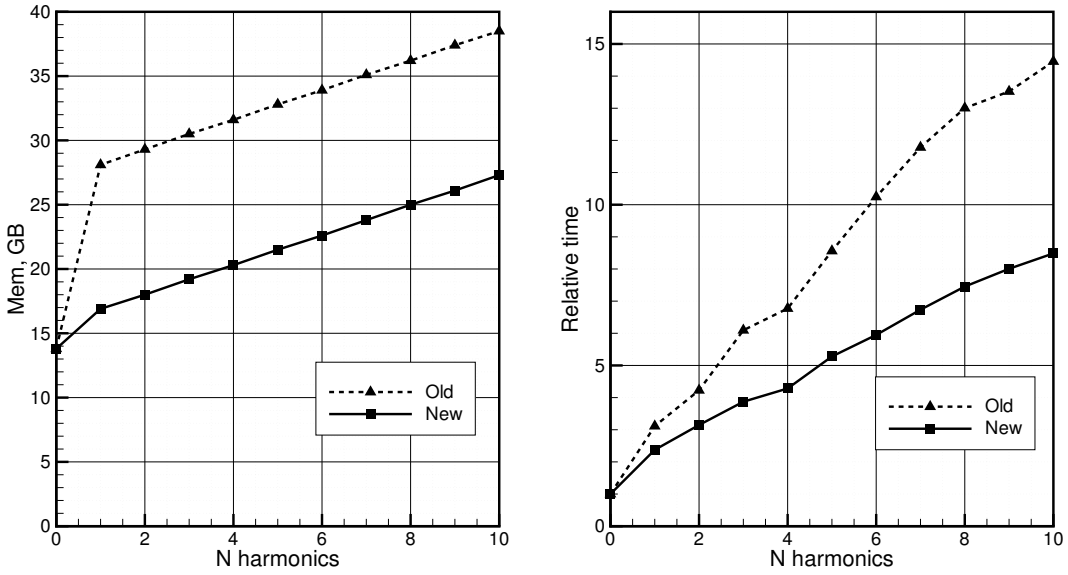


(a) Rotor-67 model rotor



(b) Axial multi-stage model compressor

Figure 3. Computational domains of the cases considered



(a) Memory consumption

(b) Computation time

Figure 4. Memory consumption and computation time relative to the no-harmonics simulation for the Rotor-67 test case as a function of the number of harmonics, comparing the previous implementation [9] (Old) and the improved one (New)

3.3. Performance of Modified Preconditioners

Now let us separately consider the acceleration of preconditioners of the linear iterative solver. A comparison of the baseline preconditioners with full storage of 10×10 inverted blocks

and the modified preconditioners, which use reduced 5×5 inverted blocks, is performed on example of the Rotor-67 test case with 3 harmonics.

To maximize the possible negative effect on the BiCGStab solver convergence from the diagonal inversion approximation, a single iteration of the Jacobi method with zero initial guess is used, which corresponds to the block-diagonal preconditioner. The solver tolerance is set several orders of magnitude smaller than what is typically used in practice to more accurately measure the difference in the number of iterations, with the iteration counts averaged over many time steps. The difference appeared to be negligible, within 2%. Then, Gauss–Seidel (GS) and symmetric GS (SGS) preconditioners were tested, and no notable difference in the resulting number of BiCGStab iterations was observed neither.

Finally, the SGS preconditioner with the approximate diagonal blocks inversion was tested in practical conditions. Comparative testing showed that diagonal block inversion is accelerated by 2.8 times, and the total time for solving linear systems for harmonics has become 1.17 times shorter. The preconditioner solution stage, which takes most of the solver time, is now 1.15 times faster. The overall speedup achieved for the simulation due to the preconditioner upgrade is about 8%. Memory consumption for storing the inverted diagonal of harmonic systems is reduced by 4 times, which corresponds to 6.5% reduction of the total memory used.

3.4. Parallel Performance

Since the computation time was reduced and the data exchanges remained the same, the parallel speedup could have notably degraded. To make sure this did not happen, parallel speedup measurements were performed on a cluster using more than 600 cores. The cluster nodes are equipped with two 16-core Intel Xeon CPUs (8 DDR4-3200 memory channels each). The tests for the Rotor-67 and model multi-stage axial compressor are performed using 3 harmonics. The speedup plots obtained are shown in Fig. 5. No notable degradation of parallel efficiency has been observed compared to the previous version [9], which is about 1.6 times slower. For instance, on 16 nodes, the parallel efficiency (acceleration at 16 nodes relative to one node, divided by the number of nodes and multiplied by 100%) of the multi-stage compressor simulation was 82% and now is 83%, the difference seems to be just within the measurement accuracy. Parallel efficiency of 77% has been obtained on the Rotor-67 case when having about 6 thousand nodes per core.

Single-node parallel performance is demonstrated on the Rotor-67 case using 3 harmonics. The speedup relative to sequential execution is presented in Tab. 1 for various parallel execution modes with different ratios of MPI processes and OpenMP threads. On a single 16-core CPU, the OpenMP speedup is about 10 times. Using simultaneous multithreading with 2 threads per core gives about 6% speedup. This can be explained by mitigating the effect of memory latency on cache misses, since when one thread gets stuck on a memory transaction, another can proceed. It should also be noted that the SGS predonditioner is parallelized by blocks: the matrix is sliced among threads, and each thread applies the preconditioner to its diagonal block, at the interface the unknowns are taken from the previous iteration, as in the Jacobi method. Therefore, the solver convergence may degrade as the number of threads increases. To evaluate this effect, the average number of solver iterations was measured when setting the solver tolerance several orders of magnitude lower than is typically used in practice. The solver with 32 threads needs on average about 12% more iterations than with 1 thread (however, the impact on the overall convergence of the solution to the nonlinear problem is less significant).

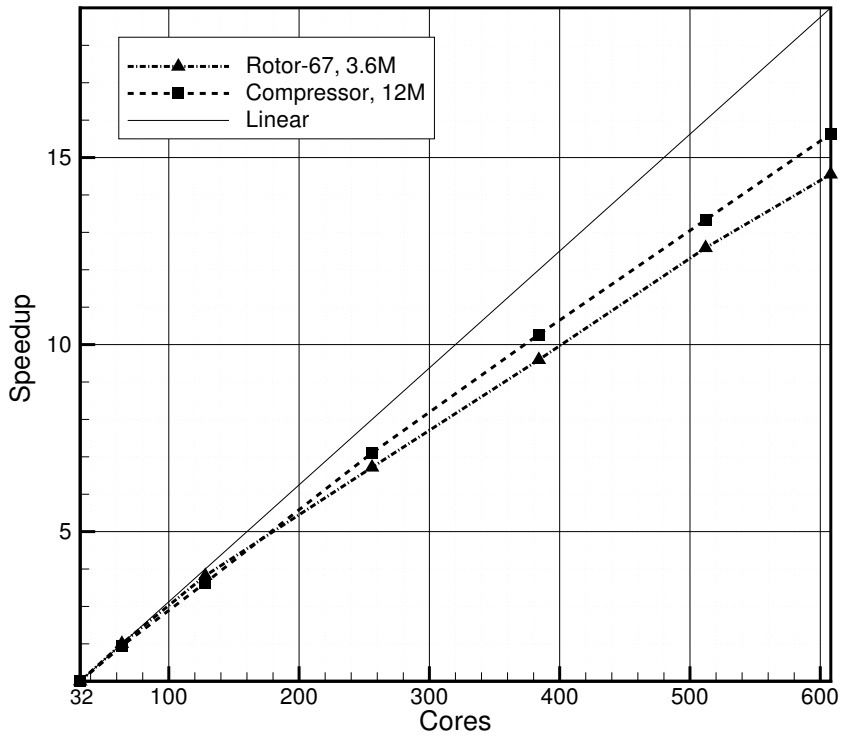


Figure 5. Parallel speedup on a cluster system for the Rotor-67 and model multi-stage axial compressor using 3 harmonics

On two CPUs running one MPI process per CPU (to avoid NUMA factor), the speedup is about 22 times. Considering that the numerical algorithm is rather memory-bound (low arithmetic intensity per unit of memory traffic, especially for the sparse linear solver) and the number of memory channels is 16, this acceleration can be viewed as high enough.

Table 1. Single-node performance on the Rotor-67 case using 3 harmonics

Cores	CPUs	MPI	OpenMP	Time, s	Speedup
1	1	1	1	53.3	1.0
16	1	1	16	5.37	9.9
16	1	1	32	5.08	10.5
32	2	1	64	3.36	15.9
32	2	2	32	2.42	22
32	2	32	1	2.32	23
32	2	32	2	2.23	23.9

Conclusion

The NLH method reproduces unsteady effects in turbomachinery applications simulated using low-cost RANS approaches. It allows for the modeling of only one periodic sector of a single blade passage per row in compressors and turbines and the transmission of non-stationary perturbations through a mixing-plane interface. An important problem with this method is the memory consumption, which grows with the number of harmonics. In case of an implicit

scheme, the Jacobian sparse block matrix for complex variables has a block size twice as large and, accordingly, needs four times more memory. Reducing memory consumption is especially critical for using the NLH method on GPUs, which have very limited memory size. In the present work, this reduction has been achieved by using a specialized block sparse matrix storage format and approximate inverse diagonal blocks in Gauss–Seidel based preconditioners. Compared to our previous NLH implementation [9], the proposed modifications have reduced the memory footprint of the Jacobian matrix for harmonics by a factor of 4, resulting in an overall reduction in memory consumption by about 1.5 times, as well as a speedup of about 1.5×. At the same time, parallel performance in shared and distributed memory was maintained at a fairly high level, showing about 80% parallel efficiency when running on hundreds of CPU cores with a payload per core of about 6 thousand mesh nodes.

The achieved reduction in memory consumption allows us to begin using GPUs in simulations with the NLH method. Our further work will be aimed at porting the NLH computational algorithm, including new solver kernels, to GPUs. This should not be very difficult, since the computational algorithm of the NLH method implemented in the NOISEtte framework is fully compatible with the stream processing paradigm and can rely on the existing GPU computing infrastructure of the code. It is also planned to incorporate the NLH method for use with the full approximation multigrid method [23], which is expected to provide a significant convergence speedup.

Acknowledgements

Our simulations have been carried out using the equipment of the Shared Resource Center of KIAM RAS (http://ckp.kiam.ru) and the shared research facilities of HPC computing resources at Lomonosov Moscow State University [24]. The authors thankfully acknowledge these institutions.

References

1. He, L., Ning, W.: Efficient approach for analysis of unsteady viscous flows in turbomachines. AIAA J. 36(11), 2005–2012 (1998). https://doi.org/10.2514/2.328

2. Chen, T., Vasanthakumar, P., He, L.: Analysis of unsteady blade row interaction using nonlinear harmonic approach. J. Propuls. Power 17(3), 651–658 (2001). https://doi.org/10.2514/2.5792

3. He, L., Chen, T., Wells, R.G., *et al.*: Analysis of Rotor-Rotor and Stator-Stator Interferences in Multi-Stage Turbomachines. J. Turbomach. Trans. ASME 124(4), 564–571 (2002). https://doi.org/10.1115/1.1508382

4. Vilmin, S., Lorrain, E., Hirsch, C., *et al.*: Unsteady flow modeling across the rotor/stator interface using the non-linear harmonic method. In: Proceedings of the ASME Turbo Expo 2006: Power for Land, Sea, and Air. Volume 6: Turbomachinery, Parts A and B. pp. 1227–1237. ASME (2006). https://doi.org/10.1115/GT2006-90210

5. Vilmin, S., Lorrain, E., Tartinville, B., *et al.*: The Nonlinear Harmonic Method: From single stage to multi-row effects. Int. J. Comput. Fluid Dyn. 27(2), 88–99 (2013). https://doi.org/10.1080/10618562.2012.752074

6. Burgos, M.A., Contreras, J., Corral, R.: Efficient Edge-Based Rotor/Stator Interaction Method. AIAA J. 49(1), 19–31 (2011). https://doi.org/10.2514/1.44512

7. Duben, A., Gorobets, A., Soukov, S., *et al.*: Supercomputer Simulations of Turbomachinery Problems with Higher Accuracy on Unstructured Meshes. In: Voevodin, V., Sobolev, S., Yakobovskiy, M., Shagaliev, R. (eds.) RuSCDays 2022. LNCS, vol. 13708, pp. 356–367. Springer (2022). https://doi.org/10.1007/978-3-031-22941-1

8. Duben, A.P., Zagitov, R.A., Shuvaev, N.V.: Nonlinear harmonics method for supercomputer simulations of fluid dynamics in turbomachines with higher accuracy on unstructured meshes. Lobachevskii J. Math. 45(7), 3007–3016 (2024). https://doi.org/10.1134/S1995080224603916

9. Duben, A.P., Zagitov, R.A., Shuvaev, N.V., Marakueva, O.V.: Towards an Adaptation of the Nonlinear Harmonics Method Realized in an Unstructured Flow Solver for Simulation of Turbomachinery Problems on Supercomputers. In: Voevodin, V., Antonov, A., Nikitenko, D. (eds.) Supercomputing. RuSCDays 2024. LNCS, vol. 15406, pp. 253–266. Springer, Cham (2025). https://doi.org/10.1007/978-3-031-78459-0_19

10. Abalakin, I.V., Bakhvalov, P.A., Bobkov, V.G., *et al.*: NOISEtte CFD&CAA Supercomputer Code for Research and Applications. Supercomputing Frontiers and Innovations 11(2), 78–101 (2024). https://doi.org/10.14529/jsfi240206

11. Gorobets, A., Bakhvalov, P.: Heterogeneous CPU+GPU parallelization for high-accuracy scale-resolving simulations of compressible turbulent flows on hybrid supercomputers. Computer Physics Communications 271, 108231 (2022). https://doi.org/10.1016/j.cpc.2021.108231

12. Bakhvalov, P.A., Abalakin, I.V., Kozubskaya, T.K.: Edge-based reconstruction schemes for unstructured tetrahedral meshes. Int. J. Numer. Methods Fluids 81(6), 331–356 (2016). https://doi.org/10.1002/fld.4187

13. Bakhvalov, P.A., Kozubskaya, T.K.: EBR-WENO scheme for solving gas dynamics problems with discontinuities on unstructured meshes. Comput. Fluids 157, 312–324 (2017). https://doi.org/10.1016/j.compfluid.2017.09.004

14. Roe, L.: Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes. J. Comput. Phys. 43, 357–372 (1981).

15. Bakhvalov, P., Surnachev, M.: Method of averaged element splittings for diffusion terms discretization in vertex-centered framework. J. Comput. Phys. 450, 110819 (2022). https://doi.org/10.1016/j.jcp.2021.110819

16. van der Vorst, H.: Bi-CGSTAB: A fast and smoothly converging variant of bi-CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. 13, 631–644 (1992). https://doi.org/10.1137/0913035

17. Magomedov, A.R., Gorobets, A.V.: Heterogeneous Implementation of Preconditioners Based on GaussSeidel Method for Sparse Block Matrices. Computational Mathematics and Modeling 33, 438–442 (2022). https://doi.org/10.1007/s10598-023-09585-2

18. Gerolymos, G.A., Michon, G.J., Neubauer, J.: Analysis and Application of Chorochronic Periodicity in Turbomachinery Rotor/Stator Interaction Computations. J. Propuls. Power 18(6), 1139–1152 (2002). https://doi.org/doi:10.2514/2.6065

19. Strazisar, A.J., Wood, J.R., Hathaway, M.D., Suder, K.L.: Laser Anemometer Measurements in a Transonic Axial-Flow Fan Rotor. NASA TP-2879, 1989.

20. Voroshnin, D.V., Marakueva, O.V., Muraveiko, A.S.: Modeling Unsteady Phenomena in an Axial Compressor. Math. Models Comput. Simul. 12, 413–421 (2020). https://doi.org/10.1134/S2070048220030187

21. Menter, F.R., Kuntz, M., Langtry, R.: Ten Years of Industrial Experience with the SST Turbulence Model. In: Proceedings of the 4th International Symposium on Turbulence, Heat and Mass Transfer, Begell House Inc., West Redding, pp. 625–632 (2003). https://doi.org/10.2514/3.12149

22. Zagitov, R.A., Salnikov, S.D., Shuvaev, N.V.: Automatic Block-Structured Grid Generation in Turbo Machine Blade Passages by TurboR&D.Mesher Software. Math. Models Comput. Simul. 16, 112–122 (2024). https://doi.org/10.1134/S2070048224010125

23. Gorobets, A.V., Soukov, S.A., Magomedov A.R.: Heterogeneous Parallel Implementation of a Multigrid Method with Full Approximation in the Noisette Code. Math. Models Comput. Simul. 16, 609–619 (2024). https://doi.org/10.1134/S2070048224700261

24. Voevodin, V., Antonov, A., Nikitenko, D., *et al.*: Supercomputer Lomonosov-2: Large scale, deep monitoring and fine analytics for the user community. Supercomput. Front. Innov. 6(2), 4–11 (2019). https://doi.org/10.14529/jsfi190201

DOI: 10.14529/jsfi250106

Modelling and Supercomputer Simulation of Hinged Rotor

Ilya V. Abalakin[1], *Vladimir G. Bobkov*[1], *Tatiana K. Kozubskaya*[1], *Aleksey V. Lipatov*[2]

The paper presents a computational technology of numerical simulation of turbulent flow over a hinged rotor on high-performance heterogeneous computer systems. A key part of the technology is the developed mathematical model describing the complex motions of triple-hinged rigid blades of a helicopter under the action of external and aerodynamic forces and its implementation using an original unstructured mesh-deformation algorithm. The mesh-deformation method exploits an auxiliary web-structured mesh with its elastic compression-expansion controlled by low-cost quasi-one-dimensional strand-based algorithms. The mechanics model is verified by solving the pendulum problems. To demonstrate the correctness of the developed techniques, the problems on taper stabilization and blade motion under cyclic control for model helicopter rotors are considered. All the presented computations are carried out using the code NOISEtte for solving aerodynamics and aeroacoustics problem. The code implements higher-accuracy methods of computational fluid dynamics on unstructured mixed-element meshes and operates with a high efficiency on modern supercomputers with arbitrary architectures including CPU cores and GPU accelerators.

Keywords: helicopter, rotor kinematics, cyclic control, flapping motion, taper stabilization, computational fluid dynamics, turbulent flows, unstructured mesh, mesh deformation, higher-accuracy method, CPU+GPU.

Introduction

Numerical simulations of turbulent flows generated by real helicopter rotors are the problems that require large computing resources and therefore modern high-performance supercomputers to solve them. The complexity of simulating non-stationary turbulent flows in such problems is aggravated by the need to correctly take into account the non-trivial mechanics of the hinged rotor blades. Nowadays efficient implementation of high-fidelity computational fluid dynamics for hinged rotors combined with proper blade mechanics presents a challenge to the global community involved in numerical simulation of helicopter problems. That is why among a huge number of publications devoted to numerical investigation of helicopter rotors and estimation of their aerodynamic and acoustic properties (see, for instance, [4, 17, 23–25]) only a few of them consider real hinged rotors in coupling with blade mechanics.

In most of papers on hinged rotors, the authors use the computational techniques offered by commercial CFD codes or even combinations of them. Thus, the CFD codes are often combined with MBDyn [1] and CAMRAD [16] packages, to provide the blade mechanics. The paper [14] studies the hinged rotor problem with the help of FINFLO solver for rotary wing applications.

The team of Prof. George Barakos from University of Glasgow U.K., one of the strongest university teams involved in helicopter research in Europe, develops their own software. Let us highlight the work [20] where they consider both blade cyclic control and flapping motions prescribed by the given laws. To achieve the target thrust coefficient, the special trimming method is used.

Another important phenomenon of hinged rotor is aeroelasticity of helicopter blades. Among the publications considering this feature there are, for instance, the papers [6] and [21]. However it appears difficult to combine it with well-developed CFD and blade mechanics.

[1]Keldysh Institute of Applied Mathematics, RAS, Moscow, Russian Federation
[2]Bauman Moscow State Technical University, Moscow, Russian Federation

In general, it can be said that to date no commonly accepted CFD platform has been developed for the numerical solution of the hinged rotor problem. In this situation, new efficient methods adapted to the rapidly growing performance of modern supercomputers are still in great demand.

A goal of this paper is to present a computational technology to simulate turbulent flows generated by helicopter rotors with three-hinged rigid blades in different flight modes. The developed methods are implemented within the code NOISEtte [5] for solving aerodynamics and aeroacoustics problems on unstructured mixed-element meshes. The proposed technology combines a turbulent flow model (among those from RANS to scale-resolving hybrid RANS-LES methods implemented in NOISEtte) with a novel mathematical description of blade mechanics and original numerical techniques to maintain the blade motions on a moving unstructured mesh.

The paper is organized as follows. Section 1 describes a coupled simulation of turbulent airflow and articulated blade dynamics in helicopter rotors, solved with mesh deformation across multiple reference frames. Section 2 presents mathematical models and numerical methods involved in the framework of turbulent flow simulation. Section 3 describes the models governing the blade mechanics and the corresponding numerical methods. Section 4 is devoted to the mesh deformation technique. Section 5 contains the numerical results. The Conclusion summarizes the results of the work.

1. General Scheme of Numerical Simulation of Hinged Rotor Problems

Numerical simulation of a helicopter rotor with articulated blades involves modeling the motion of each blade as its key component. The spatial position of the blade depends on the acting inertial forces caused by its rotation and externally given cyclic control, gravity and aerodynamic forces arising from the motion (i.e., rotation and forward motion) of it in the air. Accounting for aerodynamic forces requires the simulation of the flow near the helicopter rotor, and the aerodynamic forces themselves depend on the position of the blade.

Thus, the computation of the flow near a helicopter rotor and the motion of each rotor blade must be carried out jointly at a discrete time step within the selected model for describing the fluid and the model for the blade motion.

In helicopter applications, a liquid medium near rotors is viscous compressible turbulent flow of air. As a model for describing it, the Reynolds-averaged Navier–Stokes system of equations (RANS) closed by different turbulence models is widely used. It should be noted that, depending of the problem, the Euler or Navier–Stokes equations can be also applicable. The best description of turbulent flow is currently provided by scale-resolving hybrid RANS-LES (Large Eddy Simulation) models, however the implementation of these models requires high computational costs.

The mathematical model that determines the movement of the articulated blades of a helicopter rotor presents a system of nonlinear ordinary differential equations written in the form of the law of conservation of angular momentum.

The general algorithm for the joint solution of the gasdynamics equations and the equations of rotor blade motion at each discrete time step can be written as the following stages:

1. Determination of the pressure field from the simulation of flow over the helicopter rotor.

2. Calculation of the aerodynamic forces acting on the blade, basing on the pressure distribution on its surface.
3. Calculation of the blade displacement and determination of its new spatial position at a new time step.
4. Deformation of the computational mesh for making it consistent to the new blade position and preparation of initial information needed for computations at the next time step.
5. Returning to point 1.

The above algorithm involves three main processes: simulation of turbulent flow around the helicopter rotor, modeling of blade movements, and deformation of the computational mesh.

To simulate the turbulent flow around hinged rotor, to model the motion of its blades and to implement the corresponding mesh movement, it is convenient to consider the following different frames of reference (FR):

– *Helicopter frame of reference (HFR).* The origin of the FR coincides with the center of the rotor hub. The Oz axis is co-directed with the axis of rotation of the rotor shaft, the Ox axis is co-directed with the line of the zero-azimuth position of the blade downstream, the Oy axis complements the FR to the right-hand system. This FR can be also considered as absolute.
– *Rotating, non-inertial, frame of reference (RFR).* The origin of the FR coincides with the center of the rotor hub. The Oz axis is co-directed with the axis of rotation of the rotor shaft, the Ox axis is directed parallel to the longitudinal axis of the blade at zero angles of flapping and lagging, the Oy axis complements the FR to the right-hand system.
– *Blade, non-inertial, frame of reference (BFR).* Its description is given in detail in the section devoted to modeling the blade motion.

2. Mathematical Models and Numerical Methods for Turbulent Flow Simulation

2.1. Governing Equations

Let us model the viscous compressible gas flow by the RANS system of equations with Spalart–Allmaras (SA) turbulence model in the form of the conservation laws of mass, momentum and total energy. Consider these equations in an integral form suitable for implementing an arbitrary Lagrangian-Eulerian approach (ALE) for constructing numerical finite-volume schemes on moving meshes and/or in a non-inertial RFR. It is important to note here that in all the FR under consideration we use the flow velocity components determined in the absolute HFR. Below we designate the velocity in the HFR as "absolute velocity".

When considering the RANS system in the RFR, the rotation of the axes of this FR occurs around the rotor shaft axis with a time-independent angular velocity vector $\boldsymbol{\omega}$ with magnitude equal to the rotation speed of the helicopter rotor. In contrast to the HFR case, the absolute velocity in the RFR is projected to the rotating axes which results in the appearance of an additional term responsible for the velocity turn in the governing gasdynamic equations. With this description, the streamlined rotor remains motionless, and the direction of the upstream flow changes dependently to the azimuthal angle $\psi = |\boldsymbol{\omega}|t$. Note that at the discretized level the linear tangential velocity $\boldsymbol{V}_\omega = \boldsymbol{\omega} \times \boldsymbol{r}$ of the blade rotation, where $\boldsymbol{r}$ is the radius-vector, can be interpreted as a stationary velocity of mesh movement in the HFR.

To write the system of Navier–Stokes equations in the form of conservation laws, we introduce the vector of conservative variables

$$\boldsymbol{Q} = (\rho,\ \rho\boldsymbol{u},\ E,\ \rho\tilde{\nu})^{\intercal},$$

where $\boldsymbol{u} = (u_1, u_2, u_3)$ is the velocity vector in HFR, ρ is the density, $E = \rho\boldsymbol{u}^2/2 + \rho\varepsilon$ is the total energy, ε is the specific internal energy, $p = \rho\varepsilon\,(\gamma - 1)$ is the pressure defined by the ideal perfect gas state of equation, the constant γ is the specific ratio, $\tilde{\nu}$ is turbulent eddy viscosity.

Let $C(t)$ be the computational cell on a given mesh with volume $|C(t)|$ and $\overline{\boldsymbol{Q}}$ be the integral average of $\overline{\boldsymbol{Q}}(t)$ over this cell. Then

$$\frac{d}{dt}\int\limits_{C_i(t)} \boldsymbol{Q}dV = \frac{d\overline{\boldsymbol{Q}}_i\,|C_i(t)|}{dt}, \qquad |C_i(t)| = \int\limits_{C_i(t)} dV, \qquad \overline{\boldsymbol{Q}}_i = \frac{1}{|C_i(t)|}\int\limits_{C_i(t)} \boldsymbol{Q}dV$$

and the RANS system implementing ALE approach can be written as shown in [10] as

$$\begin{aligned}\frac{d\overline{\boldsymbol{Q}}_i\,|C_i(t)|}{dt} \;&+\; \int\limits_{\partial C_i(t)} \mathcal{F}^C\left(\boldsymbol{Q}\right)\cdot\boldsymbol{n}dS - \int\limits_{\partial C_i(t)} \boldsymbol{Q}\left(\boldsymbol{V}_c\cdot\boldsymbol{n}\right)dS \\ &=\; \int\limits_{C_i(t)} \mathcal{F}^D\left(\boldsymbol{Q},\nabla\boldsymbol{Q}\right)dV + \int\limits_{C_i(t)} \boldsymbol{S}\left(\boldsymbol{Q},\nabla\boldsymbol{Q}\right)dV.\end{aligned} \tag{1}$$

Here $\partial C_i(t)$ is the boundary of cell, $\boldsymbol{n}$ is the unit external normal to the boundary $\partial C_i(t)$, $\boldsymbol{V}_c$ is the velocity of the moving-cell boundary. In general, when using the RFR, the boundary-cell velocity is the vector sum of the mesh deformation rate $\boldsymbol{V}_d$ and the linear tangential rotation velocity $\boldsymbol{V}_\omega$.

System (1) includes composite vectors $\mathcal{F}^C$ and $\mathcal{F}^D$, each component of which F_i^C and F_i^D in coordinate direction x_i $(i = 1, 2, 3)$ represents the convective transport and diffusion flux vectors, respectively.

The components of convective transport flux vector are given as functions of the physical variables ρ, $\boldsymbol{u}$, p:

$$F_i^C\left(\boldsymbol{Q}\right) = (\rho u_i, \rho\boldsymbol{u}u_i + p\boldsymbol{e}_i, (E+p)u_i, \rho\tilde{\nu}u_i)^{\intercal},$$

where $\boldsymbol{e}_i = (\delta_{i1}, \delta_{i2}, \delta_{i3})^{\intercal}$ is the row-vector of the identity matrix and δ_{ij} is the Kronecker symbol. The diffusion flux vector is defined as a vector-function of physical variables and their gradients as

$$F_i^D\left(\boldsymbol{Q},\nabla\boldsymbol{Q}\right) = (0, \tau_{i1}, \tau_{i2}, \tau_{i3}, \tau_{ij}u_j + q_i, k_d(\nabla\rho\tilde{\nu})_i)^{\intercal},$$

where the components of the viscous stress tensor τ_{ij}, the heat flux vector q_i, and the diffusion coefficient k_d are defined according to the RANS system with the closing SA model (see, for example, [3]).

Vector $S(\boldsymbol{Q},\nabla\boldsymbol{Q})$ is the source term describing the influence of the external forces that are not related to the transport of conservative variables $\boldsymbol{Q}$

$$\boldsymbol{S}\left(\boldsymbol{Q},\nabla\boldsymbol{Q}\right) = (0, -\rho(\boldsymbol{\omega}\times\boldsymbol{u}), 0, S_{SA})^{\intercal},$$

where S_{SA} is the source term in the SA equation describing turbulence generation and turbulence destruction (a complete description can be found, for example, in [3]), the term $-\rho(\boldsymbol{\omega} \times \boldsymbol{u})$ in the momentum equation determines the velocity vector turn to the angle equal to $-|\boldsymbol{\omega}|t$ when using the RFR.

Note that when using a movable (deformable) computational mesh with the mesh velocity depending on time and spatial coordinates, it is necessary to recompute the new positions of all the nodes and rebuild the final (control) volumes at each time step. When using the RFR the mesh velocity does not depend on time so the geometry of the mesh elements does not change, and the mesh is transported as a rigid body together with the rotor.

2.2. Higher-Accuracy Quasi-One-Dimensional Reconstruction-Based Methods

To solve system (1) we use the CFD/CAA NOISEtte code [5] which exploits vertex-centered algorithms on unstructured mixed-element meshes. According to the NOISEtte framework for the discretization of the convective terms, the family of EBR (Edge-Based Reconstruction) finite-volume schemes is used [2, 7, 8]. A higher accuracy of EBR schemes is achieved thanks to the edge-based quasi-one-dimensional reconstruction of variables involved in the flux calculation according to one or another Riemann solver. For the problems considered in the paper, as a Riemann solver, we use the Roe method. For the discretization of the diffusion terms, the standard P1-Galerkin method and the method of averaged element splittings (AES) are used [9]. The time integration is implemented using the backward time differentiation formulas of the first and second orders. To solve the nonlinear algebraic systems, the Newton method is used. The corresponding linear systems are solved using the biconjugate gradient stabilized method.

2.3. Parallel Implementation

All the above mentioned models, methods and algorithms have effective and robust parallel implementation within the CFD/CAA NOISEtte code (see [5, 12]). The NOISEtte code is written in C++ using MPI, OpenMP and OpenCL frameworks for parallel implementation. It consists of a core computational library and connectable functional modules that are linked to console applications for preprocessing, running simulations, and postprocessing results. The code is designed to provide the maximum portability and can be used on Windows and Linux, on a wide range of computing systems from workstations to hybrid supercomputers. It has been tested on various computing architectures, including multicore CPUs (Intel, AMD, IBM, ARM, Elbrus), multi-core accelerators (Intel Xeon Phi KNC, KNL), GPUs (Intel, AMD, NVIDIA), systems on a chip combining central and graphics processors.

A parallel implementation of the computational algorithm is based on hierarchical multilevel model (see [12, 13]). At the top level, the MPI standard is used to couple multiple nodes of a cluster system. Then, the second-level mesh partitioning is used to further distribute the workload among computing devices of hybrid cluster nodes, such as CPUs and GPUs. MPI parallelization uses asynchronous, non-blocking exchanges and allows communications to be hidden behind computations to improve parallel efficiency, which is especially important for GPU computing. To reduce the data transfer overhead, multi-threaded message processing is used as well.

3. Mathematical Models and Numerical Methods for Rotor Mechanics

3.1. Description of Three-Hinged Blade and Its Kinematics

The hinged attachment of the blade implies the attachment of it to the hub by means of the following three hinges: 1) a flap hinge (FH) allowing the blade to move in the vertical plane (flapping motion); 2) a lag hinge (LH) allowing the blade to move in the horizontal plane (lagger motion); 3) and an pitch hinge (PH) allowing the blade to rotate around its longitudinal axis (pitching motion).

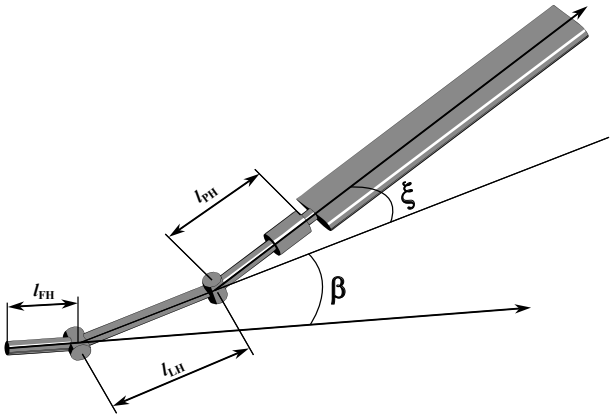


Figure 1. Schematic representation of an articulated blade

The sequence of hinges is specified as FH→LH→PH, with the distances between them defined as: l_{RH} is the distance from the hub center to the LH, l_{LH} is the distance from the FH to the LH and l_{PH} is the distance from the LH to the PH (see Fig. 1).

The modeling of the articulated blade motion is carried out under the following assumptions:

- The blade is considered an absolutely rigid body, meaning it does not deform under load.
- The mass of the blade is concentrated along its longitudinal axis to simplify the analysis of its motion.
- The moments created by the control system lever relative to the FH and LH are assumed to be zero, i.e., no external torques are applied at these hinges.
- The dimensions of the hinges are small compared to the blade which allows them to be treated as point connections.
- Only the blade has mass, the hinges and hub are considered massless.

The kinematics of the blade is determined by the following angular values:

- Flapping angle β is the angle between the plane of rotation of the rotor and the longitudinal axis of the blade.
- Lagging angle ξ is the angle between the plane passing through the azimuthal position of the blade (i.e., the plane perpendicular to the rotor's plane of rotation at a specific azimuth) and the longitudinal axis of the blade.
- Pitching angle φ is the angle between the chord of the blade section at $0.7R$ (where R is the radius of the rotor) and the plane of rotation of the rotor.

3.2. Blade Dynamics

The problem of determining the spatial position of the blade depending on the aerodynamic and inertial forces acting on it includes solving the equations for finding the angles β, ξ, φ under the following assumptions:

- The helicopter moves in forward flight mode, that is, in a straight line and at a constant velocity (a particular case is the hover mode).
- The blade control, or the change in its pitch angle φ, is determined by an externally specified control law

$$\varphi(\psi) = \varphi_0 + \varphi_1 \cos(\psi) + \varphi_2 \sin(\psi) + k_\beta \beta + k_\xi \xi, \tag{2}$$

where ψ is the azimuth angle, φ_0 is the collective pitch, φ_1 and φ_2 are the control parameters, k_β is the pitch-flap coupling, k_ξ is the pitch-lag coupling. Note that the influence of aerodynamic forces on the pitch angle occurs implicitly through the flapping angle and lagging angle.

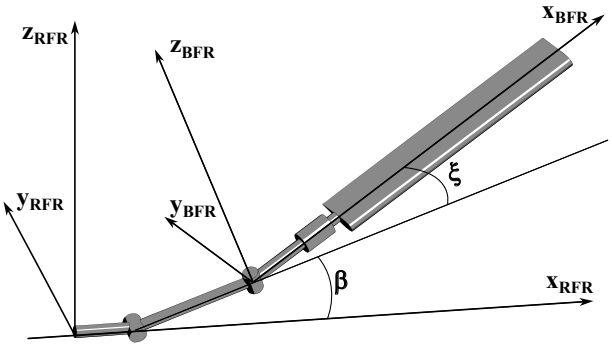


Figure 2. The helicopter frame of reference and the blade frame of reference

The blade movement is considered in the BFR which is defined as follows: the FR origin coincides with the center of LH, the Oz axis is co-directed with the LH axis, the Ox axis is directed along the blade, and the Oy axis complements the FR to the right-hand system (see Fig. 2).

3.3. Governing Equations

Let us denote the current distance from the start of the blade by r and introduce vector $\boldsymbol{g}$ which is the vector of gravity acceleration defined in the BFR. Let us also determine the values of blade mass per unit length as ρ and the aerodynamic force per unit length as $\boldsymbol{q_a}$

$$\rho(r) = \frac{dm(r)}{dr}, \quad \boldsymbol{q}_a(r) = \frac{d\boldsymbol{F}_a(r)}{dr}.$$

Then the equations for finding the angles β, ξ can be written in the form of the following laws of conservation of angular momentum [19].

The equation for flapping hinge is

$$\ddot{\beta}\int_0^R((l_{PH}+r)\cos\xi+l_{LH})^2\rho\,dr= \int_0^R(A_zq_{az}+B_zg_z\rho)\,dr+\int_0^R(C_z\omega^2+D_z\omega\dot{\xi}+E_z\dot{\beta}\dot{\xi})\rho\,dr. \tag{3}$$

The equation for lagging hinge is

$$\ddot{\xi}\int_0^R(l_{PH}+r)^2\rho\,dr= \int_0^R(A_yq_{ay}+B_yg_y\rho)\,dr+\int_0^R(C_y\omega^2+D_y\omega\dot{\beta}+E_y\dot{\beta}^2)\rho\,dr. \tag{4}$$

Here the coefficients A_z, A_y, B_z, B_y, …, E_y are the nonlinear functions depending on the variables β, ξ.

The terms in the right-hand side of these equations are the moments of forces relative to FH and LH. They can be divided into moments of external and inertial forces. The factors in front of $\ddot{\beta}$ and $\ddot{\xi}$ are the moments of inertia of the considered mass-geometric model of the blade relative to FH and LH

$$J_{FH}=\int_0^R((l_{PH}+r)\cos\xi+l_{LH})^2\rho\,dr,$$

$$J_{LH}=\int_0^R(l_{PH}+r)^2\rho\,dr.$$

The laws of conservation of angular momentum (3) and (4) can be written in the vector form as

$$\boldsymbol{J}\frac{d\boldsymbol{\omega}_{rel}}{dt}=\boldsymbol{M}_{EXT}+\boldsymbol{M}_I, \tag{5}$$

where $\boldsymbol{J}$ is the tensor of the blade momentum defined as the momentum J_{PH} and J_{LH} referred to PH and LH

$$\boldsymbol{J}=\begin{pmatrix}J_{FH}&0\\0&J_{LH}\end{pmatrix}.$$

Here $\boldsymbol{\omega}_{rel}=(\dot{\beta},\dot{\xi})^\intercal$ is the vector of angular velocities of relative rotation, $\boldsymbol{M}_{EXT}=\left(M_{EXT/FH},M_{EXT/LH}\right)^\intercal$ is the vector of external forces moments, $\boldsymbol{M}_I=\left(M_{I/FH},M_{I/LH}\right)^\intercal$ is the vector of inertial forces moments.

The solution of system (5) describes the flapping and lagging motions of the blade assuming that the cyclic pitch and the collective pitch of the rotor are given analytically.

The aerodynamic force vector $\mathbf{F}_a$ is obtained from the solution of RANS equations. Note that it is defined in the absolute HFR and before using in the system of equations (5) it must be projected onto the BFR as

$$\boldsymbol{F}_a^{BFR}=\boldsymbol{A}_{HFR}^{BFR}\boldsymbol{F}_a^{HFR},$$

where $\boldsymbol{A}_{HFR}^{BFR}$ is the transfer matrix.

By solving the system (5), we find the values of the angles and angular velocities β, $\dot{\beta}$, ξ, $\dot{\xi}$ at the next time step, and thereby determine the position of the blade relative to the hinges and the mesh velocity at the next time step.

3.4. Numerical Methods

In general, a system of ordinary differential equations (ODE) can be written in the form

$$\begin{aligned} \ddot{\beta} &= f_\beta(\beta, \dot{\beta}, \xi, \dot{\xi}), \\ \ddot{\xi} &= f_\xi(\beta, \dot{\beta}, \xi, \dot{\xi}). \end{aligned} \tag{6}$$

Let us introduce the new vector $\boldsymbol{Y}$ of unknown components defined as $y_1 = \beta$, $y_2 = \xi$, $y_3 = \dot{\beta} = \dot{y}_1$ and $y_4 = \dot{\xi} = \dot{y}_2$ and denote the right-hand side vector $\boldsymbol{F}$ as

$$\boldsymbol{Y} = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}, \quad \boldsymbol{F} = \begin{pmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{pmatrix} = \begin{pmatrix} y_3 \\ y_4 \\ f_\beta(y_1, y_3, y_2, y_4) \\ f_\xi(y_1, y_3, y_2, y_4) \end{pmatrix}. \tag{7}$$

Then the system (6) can be identically transformed to the first-order autonomous ODE system:

$$\dot{\boldsymbol{Y}} = \boldsymbol{F}(\boldsymbol{Y}). \tag{8}$$

The system (7)–(8) is solved by a multi-step Taylor Series Method. This method is sometimes called the linearized Runge–Kutta method or the Runge–Kutta method with minimal storage [15, 18]. Let $\boldsymbol{Y}^n = \boldsymbol{Y}(t^n)$, then, according to this method, the unknown vector at the next time step $\boldsymbol{Y}^{n+1} = \boldsymbol{Y}(t^n + \Delta t)$ is defined as

$$\begin{aligned} &\boldsymbol{Y}^{(0)} = \boldsymbol{Y}^n, \\ &\boldsymbol{Y}^{(k)} = \boldsymbol{Y}^n + \alpha_k \Delta t \boldsymbol{F}(t^n, \boldsymbol{Y}^{(k-1)}), \quad k = 1, \ldots, m, \\ &\boldsymbol{Y}^{n+1} = \boldsymbol{Y}^{(m)}, \end{aligned}$$

where the coefficients α_k are

$$\alpha_k = \frac{1}{m - k + 1}.$$

This multi-step method for a linear autonomous system has the m-th order of accuracy, for a nonlinear system only the second. Note that for small deviation angles β and ξ, the system (7)–(8) can be considered as linear and therefore solved with an order of accuracy equal to the number of integration steps m.

4. Elastic Mesh for Cyclic Pitch Control and Flapping Motions

Previously in [11] we introduced a computational method for simulating the motion and deformation of a body in a fluid or gas dynamic environment, focusing on maintaining the quality of the computational mesh during the body movement. The method is particularly suitable for the problems involving the motions controlled externally and/or by the aerodynamic forces such as the cyclic pitch control and flapping motions of helicopter rotor blades. Due to its ability to

handle small displacements and rotations while maintaining mesh quality, it can be effectively applied to simulate the flow around hinged rotors. An articulated blade typically undergoes cyclic pitch, flapping, and lead-lag motions. These movements can be described using the Euler angles and small displacements, which align well with the method assumption of small linear and angular motions.

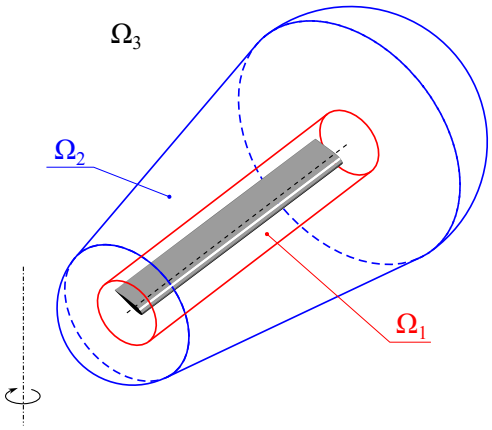


Figure 3. Deformation zones near the rotor blade

Following the method, a computational domain is divided into three subdomains. The first domain Ω_1 contains the rotor blade and its immediate vicinity. The mesh nodes in this region move with the blade, ensuring accurate resolution of the boundary layer and thereby calculation of aerodynamic forces. Domain Ω_3 includes the nodes far from the blade that remain either motionless or rotating with the rotor depending on the chosen FR. Domain Ω_2 is the deformation region where the mesh nodes move to accommodate the blade motion while preserving the mesh topology and quality. The deformation in domain Ω_2 relies on constructing an auxiliary web-structured "strand mesh" which connects the moving and stationary regions via radial strands. An inner surface ($\partial\Omega_1$) is defined around the blade, while the outer surface ($\partial\Omega_2$) is constructed concentrically (see Fig. 3). Elastic radial strands connecting these surfaces allow the mesh to deform smoothly as the blade moves.

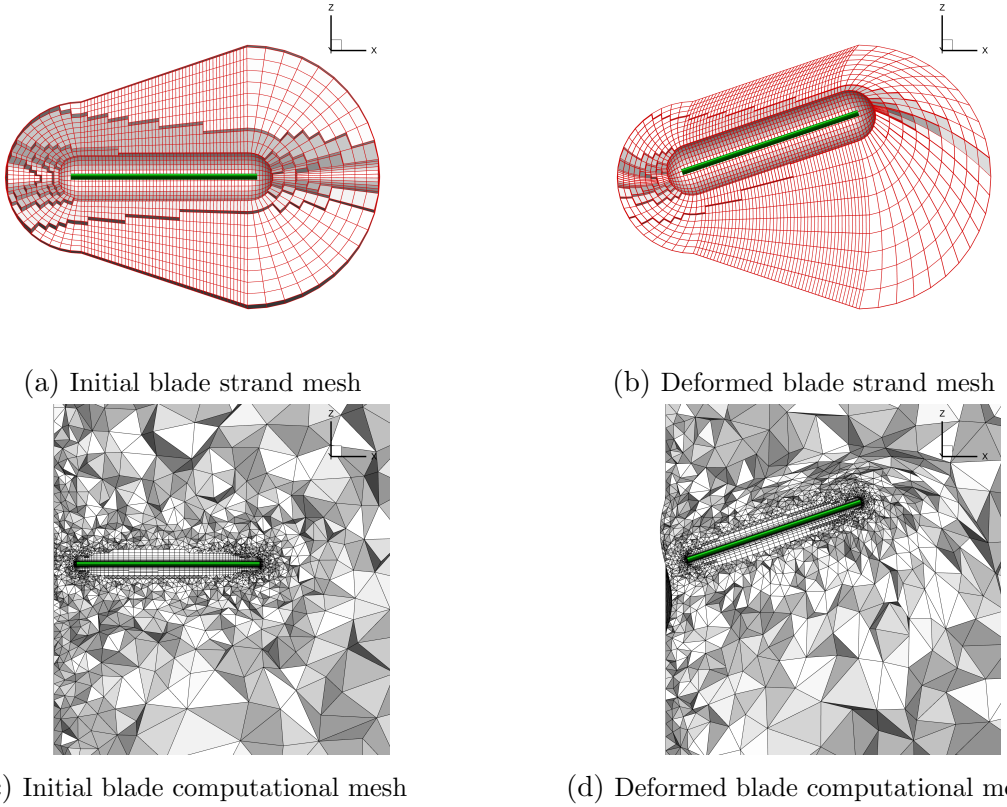
The compression-stretching of the auxiliary mesh is controlled by one-dimensional laws acting along each strand and providing smooth changes in the size of mesh cells. Note that this approach requires very little additional computing resources. Moreover, the web structure of strand mesh simplifies the interpolation of flow variables from it to the main unstructured mesh.

The implemented algorithm [11] allows to move nodes in the deformation region according with the blade displacements while maintaining the initial unstructured mesh topology and preventing its tangling. As for the hinged rotor problems, the auxiliary mesh is automatically constructed at the beginning of the computation if the minimum and maximum hinge angles can be defined (see Fig. 4). Then it is used within the whole simulation.

5. Numerical Results

5.1. Pendulum

In the pendulum test, the algorithm implementing the motion of a pendulum under the influence of gravity on a deformable unstructured mesh is verified. In this problem, the combined operation of the implemented rotor blade mechanics and the mesh deformation algorithm are



(a) Initial blade strand mesh

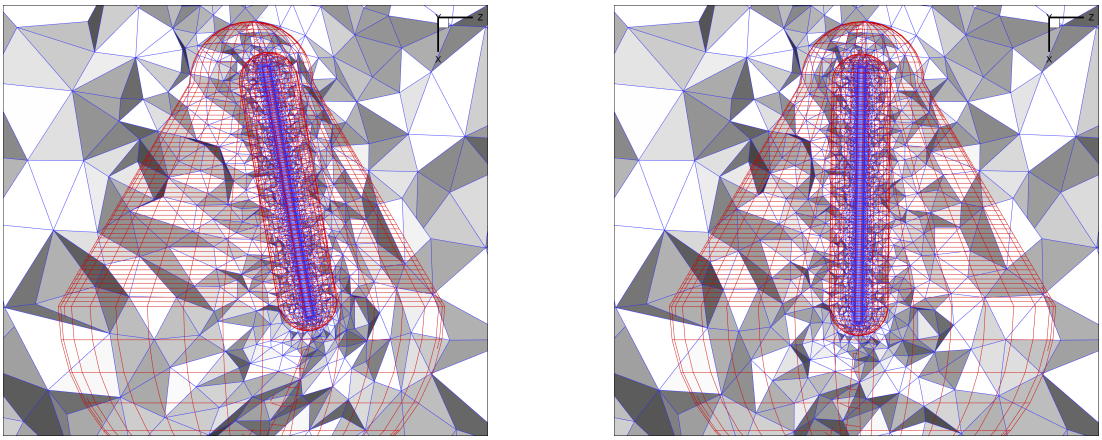
(b) Deformed blade strand mesh

(c) Initial blade computational mesh

(d) Deformed blade computational mesh

Figure 4. Strand mesh deformation and corresponding computational mesh deformation

verified without taking into account the impact of aerodynamic forces. The pendulum represents a rigid rod 3 meters long with no hinges. An auxiliary strand mesh is built large enough to cover a region of possible rod deflection of angles up to 20° (see Fig. 5).



(a) $\beta = \pi/18$

(b) $\beta = 0$

Figure 5. Computational (blue) and strand (red) meshes for pendulum cases

It is assumed that the pendulum is described by set of N different point units with mass m_i located at a distance $r_i,\ i = 1, \ldots,\ N$ from the suspension point. The motion of such pendulum is described by the equation

$$\ddot{\beta} + g\frac{\sum_{i=1}^{N} m_i r_i}{\sum_{i=1}^{N} m_i r_i^2} \sin\beta = 0.$$

If the angle β is assumed small enough and the linearized equation becomes valid, the solution can be given by the formula

$$\beta(t) = \beta(0) \cos\left(\sqrt{g \frac{\sum_{i=1}^{N} m_i r_i}{\sum_{i=1}^{N} m_i r_i^2}}\, t\right).$$

In the first case, the radial velocity $\omega = 0$, the azimuthal blade displacement $\psi = 0$, the zero hinges displacements $l_{VH} = l_{HH} = l_{PH} = 0$, the initial hinges angular velocities $\dot{\beta}(0) = \dot{\xi}(0) = 0$ and the initial pendulum position is set by the deflection angle $\beta(0) = \dfrac{\pi}{18}$, $\xi(0) = 0$.

The rest case parameters are the following: the mass distribution: $dm_i\left(\dfrac{r_i}{R}\right) = \begin{bmatrix} 0.25 & 1 \\ 1 \text{ kg} & 1 \text{ kg} \end{bmatrix}$, the pendulum length $R = 3$ m, the gravity acceleration $\boldsymbol{g} = (9.8, 0, 0)$ m/s^2.

For the given parameters, the solution for a small flapping angle is:

$$\beta(t) = \frac{\pi}{18} \cos\left(\sqrt{\frac{9.8 \cdot 5}{17}}\, t\right). \tag{9}$$

Figure 6 shows a comparison of the numerical solution with the analytical one (9) for the linearized equation. It can be seen that the solutions coincide in amplitude, but there is a little phase discrepancy between the numerical solution and the exact solution of linear equation. The magnitude of this error decreases with decreasing the initial angle of deflection, otherwise, with the diminishing influence of nonlinearity.

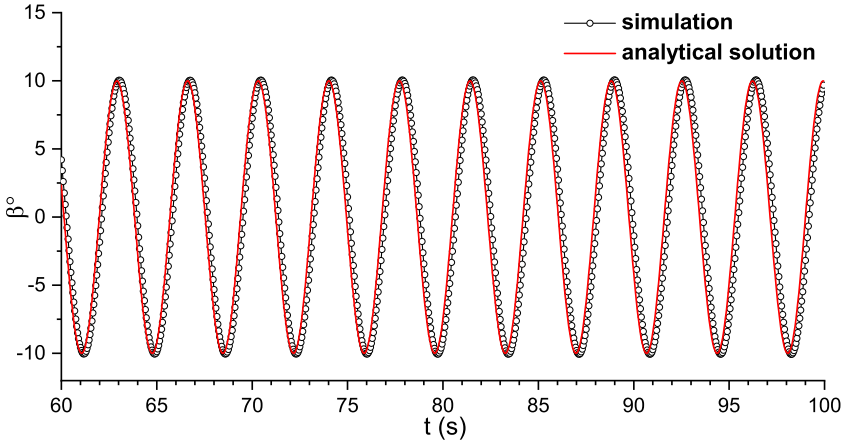


Figure 6. Evolution of deflection angle for pendulum

5.2. Rod Rotation Cases

In the following two cases, the algorithm implementing the motion of a rotating rigid body under the influence of inertial forces on a deformable unstructured mesh is verified. The configuration setup is the following: the radial velocity $\omega = 1$ rad/s, the azimuthal blade displacement $\psi = 0$, the hinges displacements $l_{HH} = 10$ m, $l_{VH} = l_{PH} = 0$, the hinges angular velocities: $\beta(0) = 0.02\pi$, $\dot{\beta}(0) = 0.01\pi$ rad/s, $\xi(0) = \dot{\xi}(0) = 0$ and no gravity acceleration $\boldsymbol{g} = (0, 0, 0)$. The blade length is 3 m and the mass distribution of single point unit is $dm_i\left(\dfrac{r_i}{R}\right) = \begin{bmatrix} 0.5 \\ 2 \text{ kg} \end{bmatrix}$.

Analytically, for small angles, the motion in terms of the evolution of the angle β under the action of centrifugal forces due to the rotation is described by the equation:

$$\ddot{\beta} + (1+\epsilon)\,\omega_e^2 \beta = 0,$$

where $\epsilon = 1 + S_{HH} \cdot l_{HH}/J_{HH}$, $S_{HH} = \sum dm_i(l_{VH} + (l_{PH} + r_i)\cos\xi)$ is the center of mass of the rod along the Oy_{BCS} axis, and $J_{HH} = \sum dm_i(l_{VH} + (l_{PH} + r_i)\cos\xi)^2$ is the moment of inertia of the rod relative to the Oy_{BCS} axis. This linear equation has the following solution:

$$\beta\,(t) = \beta_0 \cos\left(\omega_e t\sqrt{1+\epsilon}\right) + \frac{\dot{\beta}_0}{\omega_e\sqrt{1+\epsilon}} \sin\left(\omega_e t\sqrt{1+\epsilon}\right)$$

or, for the given configuration:

$$\beta\,(t) = 0.02\pi \cos\left(t\sqrt{3}\right) + \frac{0.01\pi}{\sqrt{3}} \sin(t\sqrt{3}).$$

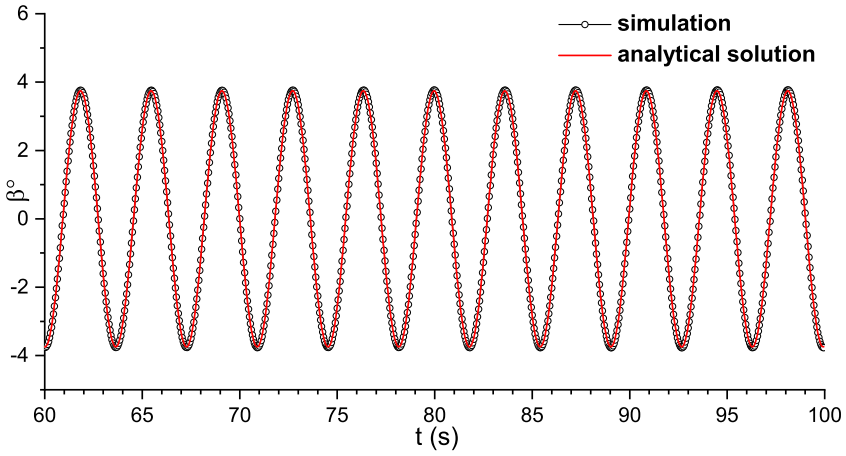


Figure 7. Evolution of rotating rod flapping angle

Figure 7 demonstrates a comparison of the numerical solution with the analytical one of the linear equation. It can be seen that the numerical solution practically coincides with the exact solution.

Now let us consider the test on the motion of the rod in terms of the angle ξ determining its lead-lag movement. The test configuration is similar to the previous one except the hinges angular velocities given as $\beta\,(0) = \dot{\beta}\,(0) = 0$, $\xi\,(0) = 0.02\pi$, $\dot{\xi}\,(0) = 0.01\pi$ rad/s.

For small angles, the lead-lag motion is analytically described by the following equation:

$$\ddot{\xi} + \nu^2\omega_e^2\xi = 0,$$

where $\nu^2 = S_{VH}(l_{HH} + l_{VH})/J_{VH}$, $S_{VH} = \sum dm_i\,(l_{PH} + r_i)$ is the center of mass of the rod along the Oz_{BCS} axis, and $J_{VH} = \sum dm_i\,(l_{PH} + r_i)^2$ is the moment of inertia of the rod relative to the Oz_{BCS} axis. This linear equation has the following solution:

$$\xi\,(t) = \xi(0)\cos\left(\nu\omega_e t\right) + \frac{\dot{\xi}(0)}{\nu\omega_e} \sin\left(\nu\omega_e t\right)$$

or, for the given configuration:

$$\xi\,(t) = 0.02\pi \cos\left(t\sqrt{1.6}\right) + \frac{0.01\pi}{\sqrt{1.6}} \sin\left(t\sqrt{1.6}\right).$$

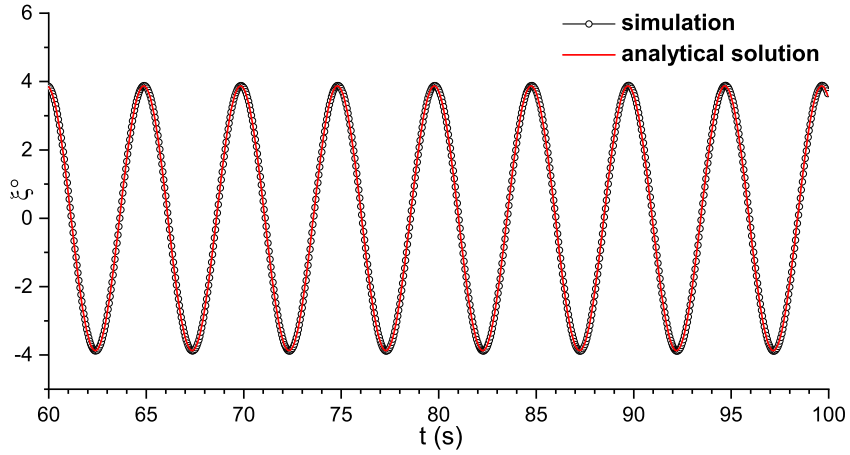


Figure 8. Evolution of rotating rod lead-lag angle

In Fig. 8, a comparison between the numerical solution and the analytical one of the linear equation is shown. As in case of the flapping angle, the numerical and exact solutions are practically indistinguishable.

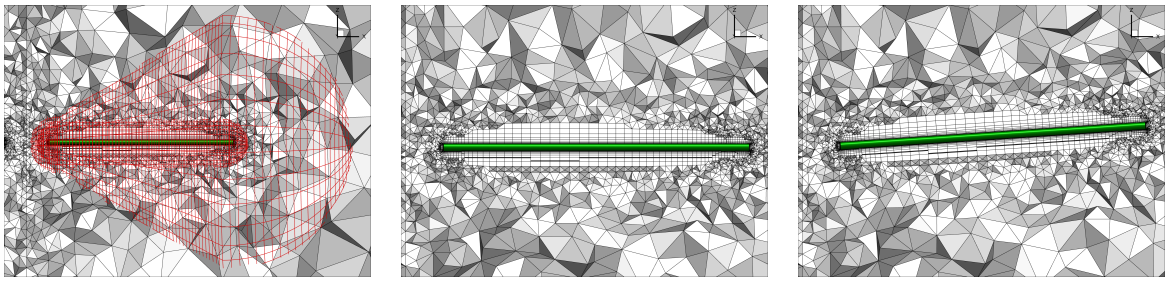
5.3. Model Hinged Rotor in Hover

In this case, the algorithm implementing the flapping motion of an articulated blade and the work of flap hinge under the influence of aerodynamic forces on a deformable unstructured mesh is verified. During the rotation of the hinged rotor, the FH allows the blades to freely perform flapping motions. Over time, the flapping angle, the same for all the blades, should stabilize, forming a taper of the rotor.

In the test case under consideration, the flow around a hinged rotor with two hinged blades, rectangular in plan and made on the basis of NACA0012 airfoil without twist, is simulated. The rotor radius is 1.142 meters, and the blade chord is 0.1903 meters with the aspect ratio equal to 6. The pitch angle is fixed at 8°. The horizontal hinges are located at a distance of 0.05 meters from the rotor axis, which allows the blades to perform flapping movements. Each blade has mass of 1 kg, the center of mass is located in the middle of the blade mid-span. The rotational speed is 650 RPM, corresponding to the tip speed of 77.7 m/s and the tip Mach number of 0.23. The environmental parameters are chosen to ensure a Reynolds number of 10^4, reducing the requirements for boundary layer cell height to satisfy the condition $y^+ < 1$. The Reynolds number is determined using a viscosity coefficient of 0.0413 Pa·s. The flow simulation is based on solving the RANS equations with the SA turbulence model.

At the pre-processing stage, the deformation zone covered with the strand mesh is automatically built for each blade (see Fig. 9, left). The inner cylindrical part contains the blade surface with the surrounding boundary layer elements (hexahedrons and triangular prisms).

The computational domain represents a cylindrical region with the diameter and height of 60 meters and the axis coinciding with the rotor axis. An unstructured mesh is constructed on the blade surface, and the prismatic boundary layer consisting of hexahedrons and triangular prisms is then built starting from the surface, transitioning isotropically to tetrahedrons with increasing the distance from the surface. The near wall cell size is chosen to ensure the condition $y^+ < 1$. The computational mesh in whole contains 2.8×10^5 nodes and 8.6×10^5 volume elements (see Fig. 9, center, right).



(a) Initial computational and strand meshes

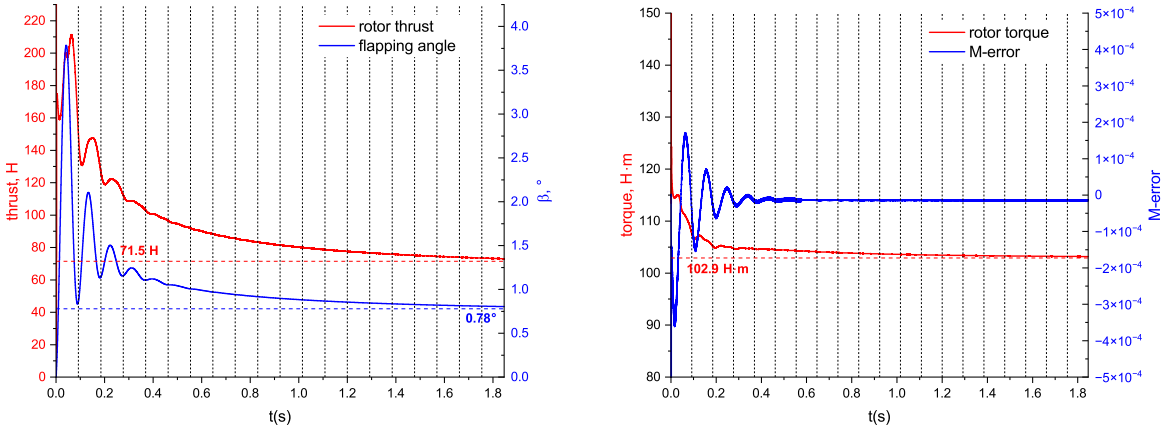
(b) Computational mesh, flapping angle 0

(c) Computational mesh, flapping angle 3.7°

Figure 9. The mesh pattern near the blade: the strand mesh and the unstructured computational mesh deformations

During the computation, at each time step, the aerodynamic forces acting on the blade are calculated based on the pressure distribution on the blade surface. These forces are fed to the input of the algorithm of the hinged rotor mechanics. The angular velocities of the blade motion are determined basing on the current force distribution along the blade span with considering the presence of hinges. Then, on the base of the location on the current time step and the angular velocities, the new position of the blade is calculated.

Within the computation after 20 revolutions, the thrust, torque, and coning angle of the rotor stabilize and reach the steady-state values (see Fig. 10, where vertical lines indicate the beginning of each rotor revolution).



(a) Thrust and flapping angle

(b) Moment and relative error

Figure 10. Two-bladed rotor characteristics evolution

To assess the reliability of the numerical result, one can check the fulfillment of the equilibrium condition at specific moments in time from the perspective of d'Alembert's principle. In the considered problem formulation, from the standpoint of this principle, the following forces act on the blade: the gravitational force, the aerodynamic force, and the inertial force, which consists of the terms arising from the rotation with the speed ω and the acceleration $\ddot{\beta}$. The equilibrium condition in the hinge is written as:

$$-M_\omega - M_g + M_{AD} - M_{\ddot{\beta}} = 0,$$

where $M_{\ddot{\beta}} = J \cdot \ddot{\beta}$ and J is the moment of inertia of the blade relative to the hinge.

The values of the moments of inertial forces and gravitational force M_ω, M_g and $M_{\ddot{\beta}}$ can be determined from the blade configuration and the data on its spatial position. To determine the moment of aerodynamic forces relative to the hinge, it is necessary to evaluate it in the BFR and integrate it with respect to the corresponding level arm length (for details see [19]).

These equilibrium equations are checked for each simulation time step. The measure of the discrepancy is estimated as

$$M_{err} = \frac{M_{AD} - M_\omega - M_g - M_{\ddot{\beta}}}{|M_{AD}|}.$$

The result is presented in Fig. 10b. The maximum relative error M_{err} in the equilibrium equation is less than 4×10^{-4}. This error corresponds to a change in the blade mass in the current configuration of approximately 0.1%, from which it can be concluded that the equilibrium is maintained and the algorithm is implemented correctly.

5.4. Model Hinged Rotor in Forward Flight

In this test case, the computation exploits the whole computational method including the mesh deformation algorithm, and the algorithms implemented the hinged rotor blade mechanics and turbulent flow simulation. The case represents the simulation of the flow around a model hinged rotor in forward flight. In such flight regime, the cyclic collective pitch control intended to reduce the aerodynamic force and moment oscillations is required. Thus, the test problem is considered in a setup with the cyclic pitch control and free flapping motions of the blades in the horizontal hinge.

Let us consider the model hinged rotor of four articulated blades. Each blade is based on NACA23012 airfoil with linear twist. The rotor radius is 2.442 meters, and the blade chord is 0.16 meters. The horizontal hinges are located on the rotor axis and allow the blades to perform flapping motions. The distribution of elementary masses along the blade span is given in Tab. 1.

Table 1. Four-bladed rotor mass distribution

r/R	0.0335	0.0570	0.0887	0.1673	0.2137	0.2636	0.3219	0.3721
m, kg	0.9111	0.8146	0.2585	0.1831	0.1345	0.1427	0.1333	0.1311
r/R	0.4291	0.5431	0.6572	0.7712	0.8292	0.8683	0.9116	0.9541
m, kg	0.2507	0.3066	0.2866	0.1937	0.1050	0.0876	0.0890	0.1392

The rotational speed of 840 RPM is constant with the tip velocity of 215 m/s corresponding to the tip Mach number of 0.63. The environmental parameters are as follows: the air density is 1.2051 kg/m^3, the pressure is 101325 Pa. The Reynolds number calculated based on the blade chord and the tip speed is 2.7×10^6. The flow simulation is carried out by numerically solving the RANS equations with the closing SA turbulence model.

The simulation of turbulent flow around the helicopter rotor in forward flight takes into account the complex curvilinear motion of the articulated blades. In this setup, the rotor angle of attack is zero, the upstream flow velocity is 60 m/s, and the blade pitch angle at the section $r/R = 0.7$ is $\varphi_0 = 12.6^\circ$ (or 11.1° with the pitch-flap coupling, see the details below). The cyclic pitch control is governed by the following law for the blade azimuthal position ψ:

$$\varphi(\psi) = \varphi_0 + a_1 \sin\psi + b_1 \cos\psi + k\beta, \tag{10}$$

where the coefficients have the following values: $\varphi_0 = 12.6°$, $a_1 = -4.5°$, $b_1 = 1°$, $k = -0.5$.

The computational mesh is constructed in such a way that the initial blade pitch angle corresponds to the law (10) at the initial time moment with the flapping angle $\beta = 5°$. Thus, the initial blade pitch angle in the mesh is set to $\varphi(\psi = 0, \beta = 5°) = 11.1°$. Note that this step is not mandatory as the blade is positioned correctly through the mesh deformation at the initial stage of the computation; however, such mesh prescription improves the robustness and expands the limitations of the mesh deformation algorithm.

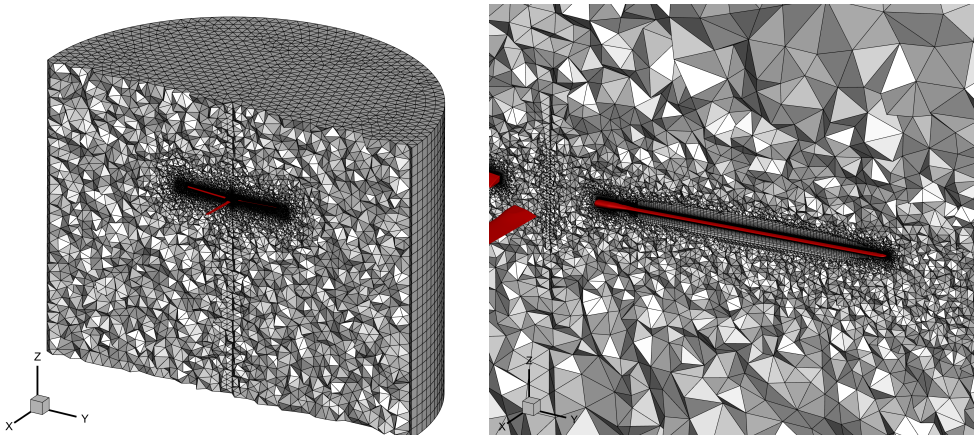


Figure 11. Computational mesh for the model four-bladed hinged rotor

As in the previous case, the unstructured mesh is constructed on the blade surface, and a prismatic boundary layer consisting of hexahedrons and triangular prisms is built starting from the surface and transitioning isotropically to tetrahedrons with increasing the distance from the surface (see Fig. 11). The nearwall cell size is chosen to ensure the condition $y^+ < 1$. The constructed computational mesh contains 4.8×10^6 nodes and 9×10^6 volume elements.

To reach a quasi-steady state, eight full rotor revolutions are needed. As practice shows, in the presence of an upstream flow (under the oblique flow conditions), it is enough to reach a periodicity of the aerodynamic characteristics. The evolution of thrust and torque confirms this (see Fig. 12).

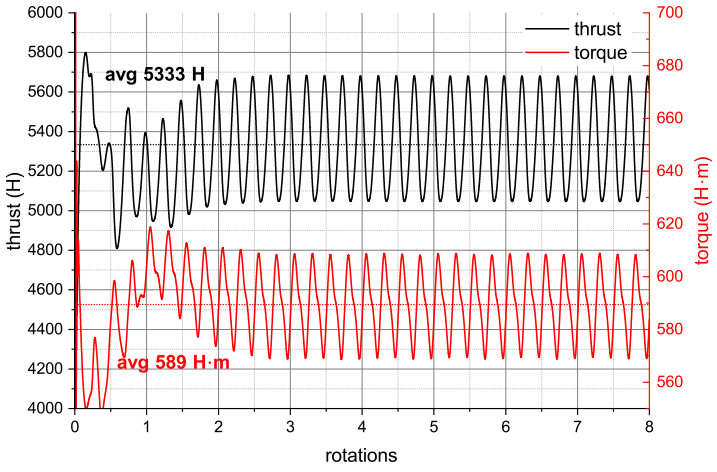
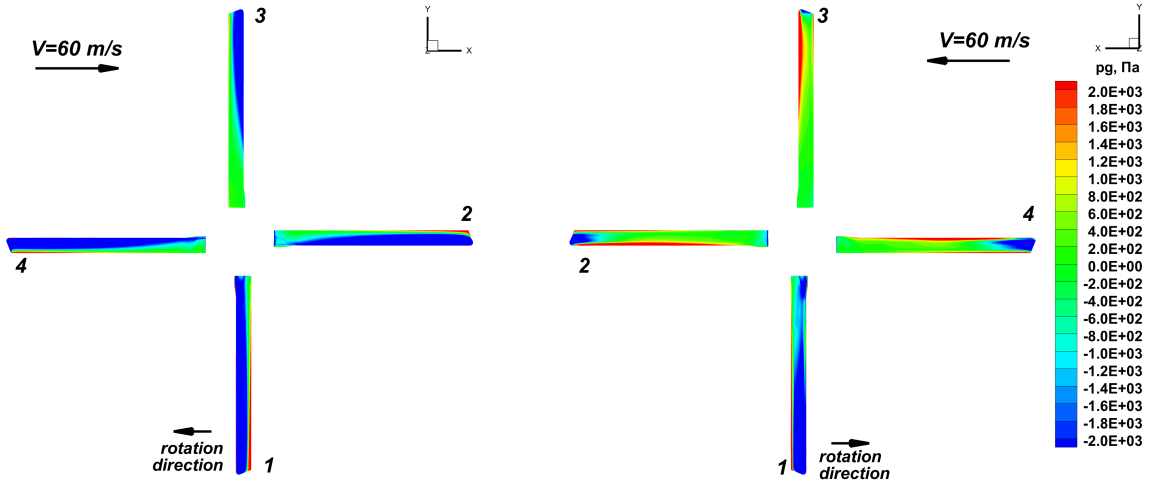


Figure 12. Evolution of four-bladed rotor thrust and torque

The overall flow pattern obtained by the simulation meets the expectations: the turbulent wake and tip vortices are carried downstream; the largest regions of negative gauge pressure are observed on the surface of the advancing blade while the smallest ones are on the blade moving downstream (Fig. 13).



(a) Rotor upper surface (b) Rotor lower surface

Figure 13. Gauge pressure distribution

Figure 13 shows the distribution of gauge pressure on the upper and lower surfaces of the rotor. As expected, the region of low pressure is the largest on the surface of the advancing blade (1) making the greatest contribution to rotor thrust at this azimuth. The last is well seen in the blade and rotor thrust charts (Fig. 14). The thrust maxima correspond to the azimuth of 82° and the multiples of them in 90° increment. Similarly, the thrust minima occur at the azimuth of 34° and the multiples thereof with a step of 90°.

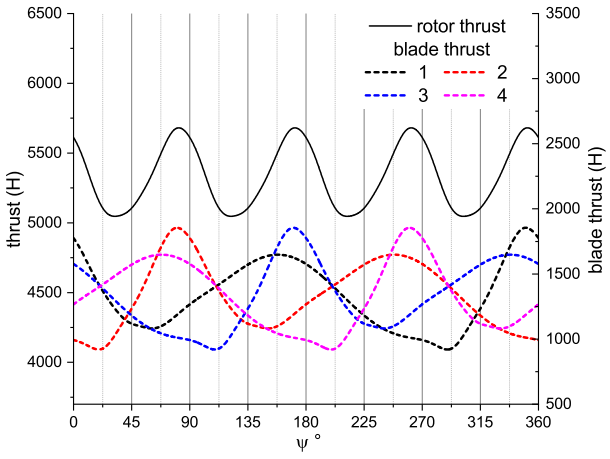
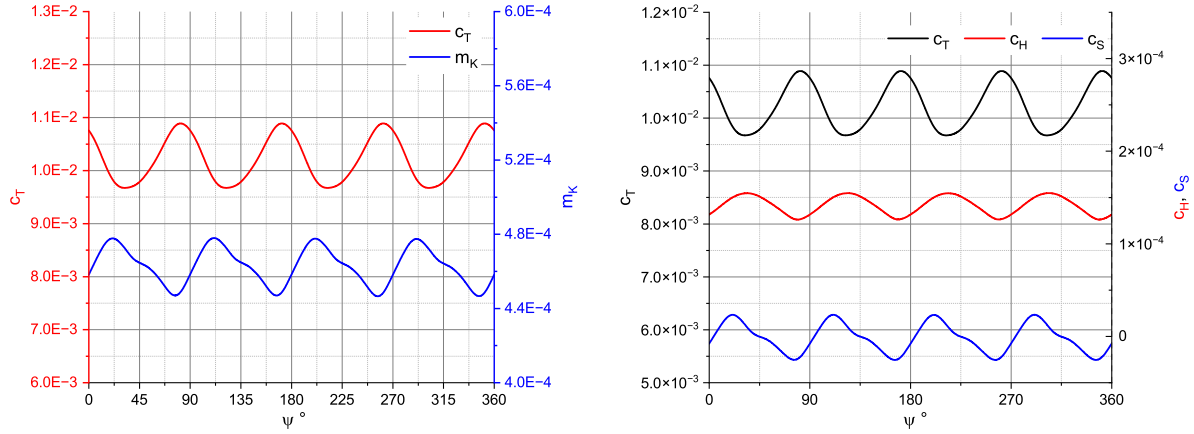


Figure 14. Distribution of rotor and blades thrust over one rotor revolution

Note that the deviation from the average thrust and torque values (and, consequently, their coefficients, see Fig. 15a) does not exceed 4.2% and 2.1%, respectively. Additionally, the influence of cyclic control leads to a significant reduction of the amplitude of fluctuations in the transverse and lateral forces, as expected (see Fig. 15b, where c_T is the thrust force coefficient, c_H is the horizontal force coefficient and c_S is the side force coefficient).



(a) The thrust and torque coefficients

(b) The coefficients of aerodynamic forces

Figure 15. Distribution of rotor aerodynamic and force coefficients over one rotor revolution

Conclusion

This paper presents a way of numerically simulating the turbulent flow near a helicopter hinged rotor, i.e., the rotor consisting of articulated rigid blades. As practice shows, accounting for hinged-blade motions within its rotation is extremely needed for designing new helicopters since it may strongly contribute to their aerodynamic and acoustic properties. At the same time, from a computational point of view, taking these movements into account significantly complicates the numerical simulation. The latter is built basing on an efficient coupling of high-fidelity computational gas dynamics and mechanics of hinged blades, and requires special computational techniques to implement it. In particular, robust economic solutions are needed to provide the proper mobility of a hinge blade under the action of inertial forces, gravity and external cyclic control. The resulting complete computational algorithm is one way or another highly resource-intensive, so the use of high-performance supercomputers is in great demand for solving such problems.

In the future, the authors intend to develop this method, to continue its verification and validation, and to extend it to the case of interacting helicopter rotors both for co-axial and spaced apart formulations.

Acknowledgements

The research is carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University [22], and Shared Resource Center of KIAM RAS `http://ckp.kiam.ru`.

References

1. MBDyn - Multi-Body Dynamics. `https://www.mbdyn.org`

2. Abalakin, I., Bakhvalov, P., Kozubskaya, T.: Edge-based reconstruction schemes for unstructured tetrahedral meshes. International Journal for Numerical Methods in Fluids 81(6), 331–356 (2016). https://doi.org/10.1002/fld.4187

3. Abalakin, I.A., Bobkov, V.G., Kozubskaya, T.K.: Numerical study of fuselage impact on acoustic characteristics of a helicopter rotor. Supercomputing Frontiers and Innovations 9(4), 100–113 (Dec 2022). https://doi.org/10.14529/jsfi220409

4. Abalakin, I.V., Bobkov, V.G., Kozubskaya, T.K., *et al.*: Numerical simulation of flow around rigid rotor in forward flight. Fluid Dynamics 55(4), 534–544 (Jul 2020). https://doi.org/10.1134/S0015462820040011

5. Abalakin, I.V., Bakhvalov, P.A., Bobkov, V.G., *et al.*: Noisette CFD&CAA supercomputer code for research and applications. Supercomputing Frontiers and Innovations 11(2), 78–101 (aug 2024). https://doi.org/10.14529/jsfi240206

6. Altmikus, A.R.M., Wagner, S., Beaumier, P., Servera, G.: A comparison: Weak versus strong modular coupling for trimmed aeroelastic rotor simulations. In: AHS International, 58th Annual Forum Proceedings. vol. 1, pp. 697–710 (2002), https://api.semanticscholar.org/CorpusID:125998085

7. Bakhvalov, P.A., Kozubskaya, T.K.: EBR-WENO scheme for solving gas dynamics problems with discontinuities on unstructured meshes. Computers and Fluids 157, 312–324 (2017). https://doi.org/10.1016/j.compfluid.2017.09.004

8. Bakhvalov, P.A., Kozubskaya, T.K., Rodionov, P.V.: EBR schemes with curvilinear reconstructions for hybrid meshes. Computers and Fluids 239(105352) (2022). https://doi.org/10.1016/j.compfluid.2022.105352

9. Bakhvalov, P.A., Surnachev, M.D.: Method of averaged element splittings for diffusion terms discretization in vertex-centered framework. Journal of Computational Physics 450(110819), 110819 (2022). https://doi.org/10.1016/j.jcp.2021.110819

10. Bakhvalov, P.A., Vershkov, V.A.: Edge-based schemes on moving hybrid meshes in the NOISEtte code. Keldysh Inst. Appl. Math. Preprint 127 (2018). https://doi.org/10.20948/prepr-2018-127

11. Bobkov, V.G., Vershkov, V.A., Kozubskaya, T.K., Tsvetkova, V.O.: Deformation technique of unstructured mesh deformation to find the aerodynamic characteristics of bodies at small displacements. Mathematical Models and Computer Simulations 13(6), 986–1001 (Nov 2021). https://doi.org/10.1134/s2070048221060028

12. Gorobets, A., Bakhvalov, P.: Heterogeneous CPU+GPU parallelization for high-accuracy scale-resolving simulations of compressible turbulent flows on hybrid supercomputers. Computer Physics Communications 271, 108231 (Dec 2022). https://doi.org/10.1016/j.cpc.2021.108231

13. Gorobets, A.V., Duben, A.P.: Technology for supercomputer simulation of turbulent flows in the good new days of exascale computing. Supercomputing Frontiers and Innovations 8(4), 4–10 (Dec 2021). https://doi.org/10.14529/jsfi210401

14. Ilkko, J., Hoffren, J., Siikonen, T.: Simulation of a helicopter rotor flow. Journal of Structural Mechanics 43(3), 186–205 (2011). https://doi.org/10.1016/j.jcp.2021.110819

15. Jameson, A.: Numerical solution of the Euler equations for compressible inviscid fluids. In: Angard, F., *et al.* (eds.) Numerical methods for the Euler equations of Fluid Dynamics, pp. 199–245. SIAM (1985)

16. Johnson, W.: Rotorcraft aerodynamics models for a comprehensive analysis. In: Proceedings of the 54th Annual Forum of the American Helicopter Society. vol. 2, pp. 1184–1206. Curran Associates, American Helicopter Society International, Washington, D.C. (May 1998)

17. Kopiev, V.F., Zaytsev, M.Yu., Vorontsov, V.I., *et al.*: Helicopter noise in hover: Computational modelling and experimental validation. Acoustical Physics 63(6), 686–698 (Nov 2017). https://doi.org/10.1134/s1063771017060070

18. Lallemand, M.H.: Dissipative properties of Runge-Kutta schemes with upwind spatial approximation for the Euler equations, Research Report RR-1173, INRIA (1990), https://inria.hal.science/inria-00075385

19. Lipatov, A.V., Abalakin, I.V.: The hinged rotor mechanics. Keldysh Institute Preprints (8), 1–24 (2025), https://library.keldysh.ru/preprint.asp?id=2025-8

20. Steijl, R., Barakos, G., Badcock, K.: A framework for CFD analysis of helicopter rotors in hover and forward flight. International Journal for Numerical Methods in Fluids 51(8), 819–847 (Jan 2006). https://doi.org/10.1002/fld.1086

21. van der Ven, H., Boelens, O.J.: A framework for aeroealistic simulations of trimmed rotor systems in forward fight. In: European Rotorcraft Forum, Marseille, September 1416 (2004)

22. Voevodin, Vl., Antonov, A., Nikitenko, D., *et al.*: Supercomputer Lomonosov-2: Large scale, deep monitoring and fine analytics for the user community. Supercomput. Front. Innov. 6(2), 4–11 (2019). https://doi.org/10.14529/jsfi190201

23. Vorontsov, V.I., Faranosov, G.A., Karabasov, S.A., Zaitsev, M.Yu.: Comparison of the noise directivity pattern of the main rotor of a helicopter for flight and hover modes. Acoustical Physics 66(3), 303–312 (may 2020). https://doi.org/10.1134/s1063771020030082

24. Xu, L., Weng, Pf.: Rotor wake capture improvement based on high-order spatially accurate schemes and chimera grids. Applied Mathematics and Mechanics 32(12), 1565–1576 (Dec 2011). https://doi.org/10.1007/s10483-011-1523-6

25. Zhang, Z., Qian, Y.: Unsteady RANS/DES analysis of flow around helicopter rotor blades at forword flight conditions. Modern Physics Letters B 32(12n13), 1840015 (2018). https://doi.org/10.1142/S0217984918400158

DOI: 10.14529/jsfi250107

Methodology for Scale-Resolving Simulation of Unsteady Effects in Turbomachines

Alexey P. Duben[1], ***Viacheslav A. Sapozhnikov***[1]

Methodology for studying effects associated with periodic unsteady impact of neighbouring rows in turbomachines is presented. The two-stage procedure of an investigation is as follows: simulation using an approach based on solving the Reynolds-averaged Navier–Stokes equations (RANS) of an entire turbomachine at the first stage and scale-resolving simulation (SRS) of a particular row at the second. The methodology exploits the following methods and technologies, which are implemented in the NOISEtte computational algorithm: the nonlinear harmonics method as a RANS approach to obtain unsteady inflow parameters for SRS; the hybrid Improved Delayed Detached Eddy Simulation approach for SRS of the row under detailed study. SRS considers using the dynamic synthetic turbulence generator in a form of volumetric source terms (VSTG) to reproduce unsteady periodic turbulent perturbations. A dynamic version of the VSTG, the parameters of which depend on the flow upstream the source region, is formulated. Details of the parallel heterogeneous implementation of the dynamic VSTG are discussed. To demonstrate the applicability of the presented methodology, a simulation of non-stationary effects in a cascade of T106 low-pressure turbine blades was performed.

Keywords: turbulent flows, non-linear harmonics method, scale-resolving simulation, hybrid RANS-LES approach, IDDES, CPU+GPU, MPI+OpenMP+OpenCL, synthetic turbulence.

Introduction

Increasing performance of computing resources and development of high-fidelity numerical models, methods, and algorithms for CFD (Computational Fluid Dynamics) promotes further engagement of scale resolving simulations (SRS) for solving turbomachinery problems [21, 29]. In terms of modeling approaches, either direct numerical simulation (DNS) or wall-resolved large-eddy simulation (WRLES) are mainly used due to the complex physical phenomena and flow specifics that significantly affect the turbine or compressor performance (e.g., integral ones such as total pressure losses or efficiency). The most impactful are the following: inflow turbulence parameters, laminar-turbulence transition on the suction side of a blade, the influence of upstream unsteady perturbations caused by wakes from the previous row. Simulations are usually carried out for simplified configurations like a plane-parallel blades cascade. Unsteady RANS (URANS) and hybrid RANS-LES methods are exploited for high Reynolds number flows over more industrial configurations, e.g., simulations of the bypass duct of a turbofan engine [23, 28]. Most investigations of a plane-parallel blades cascade (e.g., see [8, 15, 22]) consider homogeneous inflow conditions by maintaining a specified turbulence intensity level. One of the necessary components accompanying scale-resolving simulation for such investigations is a technique for creating unsteady turbulent pulsations. It allows to replicate adequate flow characteristics upstream a blade under consideration. The impact of unsteady flow features, namely, wakes coming from the previous stage, is less frequently studied. For instance, the T106 low-pressure turbine (LPT) blades cascade was investigated using SRS in [11, 18, 32]. The test case is based on the corresponding experimental data [27]. The incoming wakes produced by the moving bars in the experiment, were simulated either artificially in [18, 32] or directly [11]. In the latter, the flow behind the moving bars was modeled with a sliding interface to transfer the induced perturbations to the domain with the T106 blade. This methodology is practically not applicable to

[1]Keldysh Institute of Applied Mathematics, RAS, Moscow, Russian Federation

industrial configurations, because consideration of an upstream row in SRS increases the cost of the computation by several times, which is prohibitive given the already significant cost for a row. Thus, a feasible technique should engage a technology for creating artificial pulsations.

The most efficient and widespread solution to consider turbulent caracteristics of the incoming flow is to inject artificially generated pulsations of velocity. Among this class of techniques the synthetic turbulence generator (STG), presented in [25], is considered to be quite an impactful comprehensive solution ready for practical applications. Its most advantageous feature is a possibility to obtain adequate turbulent fields at a relatively small distance downstream the source region (also called "relaxation distance"). For instance, the relaxation distance for boundary layer turbulence usually varies between 3 and 5 of its thicknesses, which is relatively small. This feature is essential for turbomachinery because blade rows are mounted close to each other. STG was formulated initially for the form of inlet boundary conditions and adjusted to replicate boundary layer and shear layer turbulence. In [24], a methodology of its injection as a distributed volume source (Volume STG, VSTG) was proposed. The flow in the highly loaded T106C turbine cascade with different turbulent inflow conditions was investigated [8] by authors of the present paper earlier. The VSTG was used there to reproduce an intake with a turbulence level of 3%.

We present a methodology for studying effects associated with periodic impact of wakes from the preceding row in turbomachines using SRS based on the dynamic VSTG. An essential part of it is the nonlinear harmonics (NLH) method, implemented [7, 10] in NOISEtte [2, 13], which is used to obtain unsteady inflow parameters in a preliminary RANS computation. In contrast to the mixing plane technology [5, 9], which assumes circumferential uniformity, NLH allows modeling of unsteady effects that are related to the blade-passing frequency of adjacent rows. Thus, the time-dependent RANS solution can be restored and set in the form of an unsteady boundary condition at the rotor-stator interface upstream the row which is being investigated using SRS. In turn, VSTG is adapted to the capabilities of dynamically varying turbulent flow downstream of the source region. We use the T106 LPT cascade, which is investigated in [11, 18, 32], to demonstrate performance of the technology.

The paper is organized as follows. Section 1 presents short overview of mathematical models and numerical methods implemented in the NOISEtte algorithm that are used for simulations. Section 2 presents an overview of the proposed methodology. Section 3 is dedicated to the dynamic version of VSTG including its parallel implementation and testing on a turbulent mixing layer flow, which is similar to a wake downstream a turbomachine blade. Section 4 presents the simulation results of the flow over T106A low-pressure turbine blades to demonstrate the performance of the technology. Conclusion summarizes the results of the work.

1. NOISEtte: Mathematical Models and Numerical Methods

The numerical algorithm realized in the NOISEtte [2, 13] research code is based on the Navier–Stokes equations for a compressible perfect gas.

We use the latest version of the hybrid non-zonal RANS-LES method IDDES [14] as an SRS approach. Despite the fact that the flow over the T106 blade is actually simulated in LES regime (we use a mesh with proper resolution), the RANS branch is required in the region downstream the inlet boundary and upstream the VSTG source to conduct unsteady RANS solution (see Section 2 for details). We use the recent version of the IDDES based on the Menter $k-\omega$ SST model [17] and Δ_{SLA} [19, 26] dynamic subgrid scale.

NOISEtte is a vertex-centered code on mixed-element unstructured meshes. It is based on the edge-based reconstruction scheme EBR [3], which provides higher accuracy for approximation of convective fluxes. The method of averaged element splittings (AES) [4] is used for discretization of the diffusion terms.

The 2nd order numerical scheme based on the backward differentiation formula is used. The BDF1 was applied for RANS simulations and BDF2 for SRS. To solve nonlinear algebraic systems, a simplified Newton method is used, and the linear systems are solved using the BiCGStab solver [20] with Gauss–Seidel method-based parallel preconditioner [16].

The parallel implementation of NOISEtte is based on MPI, OpenMP and OpenCL frameworks. The heterogeneous parallel algorithm and its implementation are described in detail in [13]. All simulations were carried out on GPUs.

2. Overview of the Methodology

The methodology is as follows (see Fig. 1 for visualization). Let us consider that a multistage turbomachine is being examined, and a particular row, either rotor or stator, is needed to be thoroughly investigated using SRS (a rotating row in Fig. 1). Thus, the study is divided into two sequential stages.

1. RANS+NLH simulation of an entire turbomachine (RANS+NLH stage).
2. SRS simulation of a separate row with unsteady boundary conditions and the dynamic VSTG (SRS stage).

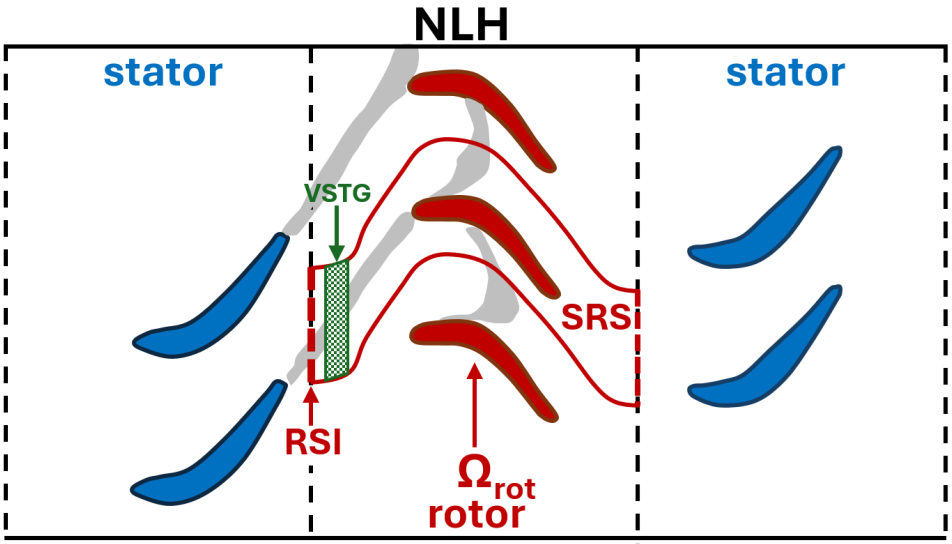


Figure 1. Sketch of the two-stage methodology

The main goal of the first stage is to obtain unsteady inflow characteristics, which reproduce flow dynamics of the adjacent upstream row, mainly, wakes past the corresponding blade. The simulation using NLH assumes only one vane channel per row with periodic boundary conditions in the circumferential direction. As result of the computation, those distributions can be extracted at every rotor-stator interface (RSI): averaged flow distributions of fields; Fourier coefficients from an adjacent row, the perturbations of which are approximated using the flow decomposition. Passing through the RSI, the stationary fields with respect to one stage turn into unsteady fields relative to the adjacent domain, which rotates at a different speed. The corre-

sponding formulas are given in [7]. Here we provide the formulas for the simplified configuration that we consider (see Section 4).

The second stage assumes scale-resolving simulation of the row under consideration (see SRS region in Fig. 1). The VSTG is imposed downstream the inlet boundary and upstream the blade. All the required data for it could be obtained from the solution using RANS+NLH. The target solution for the sponge layer can be extracted too. It is needed to properly dump perturbations downstream the blade till the outlet boundary. Besides the NLH for the first stage, another important part of the methodology is the involvement of Detached Eddy Simulation (DES) for the SRS stage. The reason is that, operating within the RANS regime, it allows to conduct unsteady periodic pulsations from the inlet boundary downstream towards the VSTG region. More specifically, the IDDES approach [14] is the most eligible because it switches to SRS near walls in the presence of resolved turbulence, either WRLES or WMLES (Wall-Modelled LES), depending on the mesh resolution.

Both RANS+NLH and SRS stages assume consideration of one vane channel per row. But, in case of non-multiplicity of the number of blades in adjacent rows, which is a common practice for turbomachines, performing the SRS stage seems complicated. As for the RANS+NLH stage, harmonic amplitudes for a domain are treated using the generalized periodicity BC with phase shift. There are two options to deal with the non-multiplicity problem. The first is to slightly change, namely, rescale, the geometry of a turbomachine, so that the number of blades in the rows became a multiple, that, for instance, was done in [31]. This should not significantly affect the aerodynamics, but allows the proposed technique to be used for a detailed study of unsteady effects, with only one vane channel being resolved by the corresponding computational mesh. The second option is to implement a generalized periodicity BC with phase shift for SRS.

3. Unsteady VSTG

3.1. Formulation

The VSTG [24] involves adding pulsations as source terms for a node, defined by the radius-vector $\mathbf{r} = (x, y, z)$, to the momentum and energy equations by the following form

$$\mathbf{F}(\mathbf{r}) = C_{\mathrm{VSTG}} \rho U_0 \mathbf{u}'(\mathbf{r}) \alpha(\mathbf{r}), \tag{1}$$

where $C_{\mathrm{VSTG}} = 1.1$ is empirical constant, $\rho = \rho(\mathbf{r})$ is density, U_0 is characteristic convective velocity inside the source region, $\mathbf{u}' = \mathbf{u}'(\mathbf{r})$ is the vector of velocity fluctuations computed using the STG [25], $\alpha(\mathbf{r})$ is a weight function constructed so that its integral value along the streamwise direction (denote it by $\boldsymbol{\tau}_{\mathrm{sw}}$) throughout the source region equals to 1. An example of the source region is depicted as "VSTG" in Fig. 1. Also, the transport equation for the turbulence kinetic energy k is extended by the source term too:

$$F_k(\mathbf{r}) = -\rho U_0 \alpha(\mathbf{r}) \omega \max\left(\nu_t^{\mathrm{SRS}} - \nu_t^{\mathrm{SMG}}, 0\right), \tag{2}$$

where $\omega = \omega(\mathbf{r})$ is the specific turbulence dissipation rate, $\nu_t^{\mathrm{SRS}} = \nu_t^{\mathrm{SRS}}(\mathbf{r})$ is actual turbulence viscosity, $\nu_t^{\mathrm{SMG}} = \nu_t^{\mathrm{SMG}}(\mathbf{r})$ is Smagorinsky eddy viscosity.

According to the STG formulation [24, 25], time-dependent velocity pulsations are calculated as

$$\mathbf{u}'(\mathbf{r}, t) = A_{ij} \mathbf{v}'(\mathbf{r}, t), \;\; \mathbf{v}'(\mathbf{r}, t) = 2\sqrt{(3/2)} \sum_{n=1}^{N_m} \sqrt{q^n} [\boldsymbol{\sigma}^n \cos(k^n \mathbf{d}^n \cdot \mathbf{r}' + \phi^n)], \tag{3}$$

where N_m is the number of modes. Vectors $\boldsymbol{\sigma}^n$ and $\mathbf{d}^n$, and scalars ϕ^n and k^n (amplitude of the wave number vector of mode n) are equal for every node inside the VSTG source region and do not depend on both space and time (see [25] for details). $\mathbf{A} = A_{ij}(\mathbf{r})$ is defined by the Reynolds stresses $\mathbf{R} = R_{ij}(\mathbf{r})$ so that $\mathbf{R} = \mathbf{A}^\intercal\mathbf{A}$. $q^n = q^n(\mathbf{r})$ is normalized mode amplitude defined by local turbulence energy spectrum. It depends on the length-scale $l_t = k^{1/2}/\omega$ obtained from the RANS solution, which could vary in time in case of the unsteady inflow. Also, time-dependent flow upstream the VSTG leads to variation of $\mathbf{R}$.

The sketch depicting a fragment of the VSTG zone on some mesh is shown in Fig. 2. The following sets of parameters and are assumed to be defined at the initialization of a computation, and do not change during simulation: they are extracted from the RANS solution obtained at the stage prior the scale-resolving simulation (RANS+NLH stage).

- The nodes $\{i^{\text{VSTG}}\}$ inside the VSTG source region (marked with circles in Fig. 2).
- "Frontal" nodes $\{i_{\text{f}}^{\text{VSTG}}\} \subset \{i^{\text{VSTG}}\}$ are the nodes located at the very edge of the upstream VSTG zone (marked with a star). Each VSTG node is associated with a front node by being within a streamwise-aligned band, the width of which is determined by the average mesh resolution Δ_{VSTG} in the source area. It is considered that all nodes which have the same frontal node inject pulsation with the same turbulent characteristics (R_{ij} and l_t).
- R_{ij} and l_t, in turn, are extracted from the "reference" nodes $\{i_{\text{f}}^{\text{ref}}\}$ (marked with a square), which are the closest nodes to the points obtained by shifting the corresponding frontals on the vector $\mathbf{l}_{\text{us}} = L_{\text{us}}\boldsymbol{\tau}_{\text{sw}}$ upstream (the length L_{us} is usually several Δ_{VSTG}).
- $\boldsymbol{\tau}_{\text{sw}}$, $\boldsymbol{\tau}_{\text{tr}}$, and $\boldsymbol{\tau}_{\text{sp}}$ are the are orthogonal unit vectors, in the along-streamwise, transverse and spanwise directions, respectively. $\boldsymbol{\tau}_{\text{sw}}$ is obtained from the solution in the reference nodes, $\boldsymbol{\tau}_{\text{sp}}$ corresponds to the periodicity direction (e.g., circumferential for a rotating machine).

The set of spatial modes and their parameters within the VSTG source region is defined at the initialization too. It depends on the minimum mesh step inside the source region and the maximum turbulent length scale. These parameters do not change over time, so the instantaneous RANS solution with NLH harmonics can be taken to extract them. The averaged solution is not suitable because it does not contain wakes past the blades from the previous step: they are reproduced by a set of NLH harmonics.

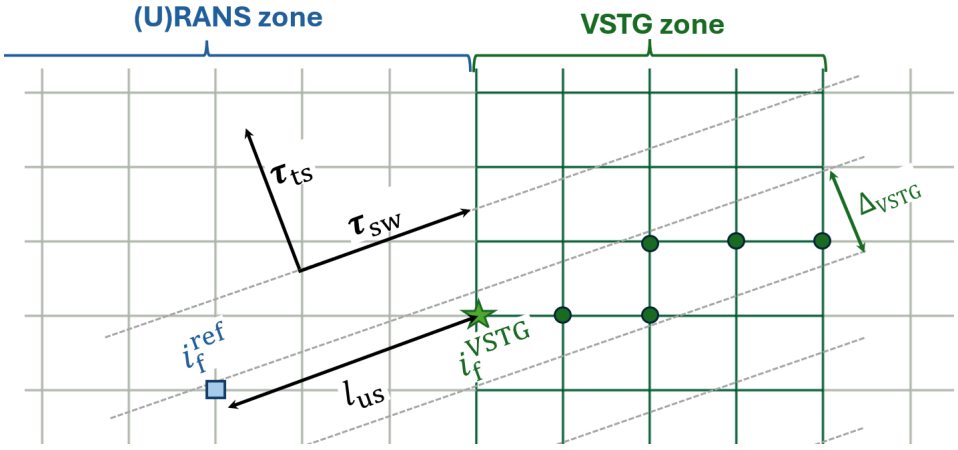


Figure 2. Fragment of a mesh with the VSTG zone

We define the pseudo-position vector $\mathbf{r}'$ in (3) for the node with radius-vector $\mathbf{r}$ as

$$\mathbf{r}' = \{\xi', \eta', \zeta'\}, \xi' = \frac{2\pi\,(\mathbf{r}\cdot\boldsymbol{\tau}_{\rm sw} - U_0 t)}{k^n \max\{l_e(\mathbf{r})\}}, \eta' = \mathbf{r}\cdot\boldsymbol{\tau}_{\rm tr}, \zeta' = \mathbf{r}\cdot\boldsymbol{\tau}_{\rm sp}. \tag{4}$$

$\max\{l_e(\mathbf{r})\}$ in (4) is a function of l_t and the distance to the wall. The formulation (4) is similar to the original formulation of $\mathbf{r}'$ as shown in formula (5) of [25]. It is transformed into this form when the $\tau_{\rm sw}$ direction is aligned with the positive direction of the OX axis.

The following parameters are updated during the simulation (they are taken from reference nodes $\{i_{\rm f}^{\rm ref}\}$) when a non-stationary periodic flow upstream the source region is considered: R_{ij} and l_t.

Note that Shorstov [23] slightly modified the formulation (4) to provide adequate velocity scales inside the source region: spatially variable distribution of convective velocity was assumed instead of a single velocity scale U_0. This option was implemented, but was not used for the simulations presented in this paper, because it did not affect the turbulence generated by the VSTG.

3.2. Parallel Implementation

The dynamic VSTG was implemented in both CPU and GPU versions of the finite-volume computational algorithm for unstructured meshes, implemented in the parallel heterogeneous code NOISEtte [2, 13]. The kernels realizing all computationally intensive procedures are incorporated in the GPU version.

All data required for VSTG operation is initialized before time stepping. Note that reference nodes, which contain values needed to calculate R_{ij} and l_t, could not belong to the same MPI domain as a VSTG node, in which artificial pulsations are generated. Thus, updating of these defining parameters requires global MPI synchronization of all parallel processes. However, it takes a negligible amount of time because the number of the reference nodes is very small.

The number of VSTG nodes is orders of magnitude smaller (usually it is about several percent) than the number of mesh nodes. But they are distributed among MPI processes in a highly unequal way, which of course leads to load imbalance. Moreover, the computational load for a single VSTG node is quite large due to the cycle on harmonics (see equation (3)), which can number in hundreds. As practice has shown, calculations implementing the formulas (1)–(3) take on average about several percent of the time spent on processing the entire step.

The implementation of the dynamic version of VSTG revealed a similar problem to the one mentioned in [12]. Initially, updating VSTG node parameters, namely, R_{ij} and l_t, was implemented entirely on the CPU. It led to the situation that this procedure occupied a significant part of the step (tens of percent). The most computationally intensive procedure that slows down the computation is the recalculation of normalized mode amplitudes $\{q^n\}_{n=1}^{N_m}$ for each VSTG node: its load is comparable to the calculation of source terms according to the formulas (1)–(3). The static VSTG requires this cycle to be run only once per computation during initialization, which is done entirely on the CPU. Therefore the recalculation of normalized mode amplitudes was ported to the GPU too. As a result, only operations related to MPI exchanges remained on the CPU, and the costs of dynamic VSTG operations do not exceed a few percent of the total time step. This value can be reduced if the VSTG node parameters are not updated at every time iteration.

3.3. Testing on a Mixing Layer

The LaRC Turbulence Modeling Resource (TMR) 2D Mixing Layer case [1] is used for validation of the VSTG in the static mode. Schematics of the flow configuration and simulation set-up is presented in Fig. 3. It is characterized by the following parameters: ambient pressure $P_\infty = 101325$ Pa, temperature $T_\infty = 293$ K, the freestream velocities are $U_1 = 41.54$ m/s and $U_2 = 22.4$ m/s, the Reynolds number based on U_1 and $L = 1$ m is $\mathrm{Re} = 2.76 \cdot 10^6$. The results are evaluated using the experimental data [6].

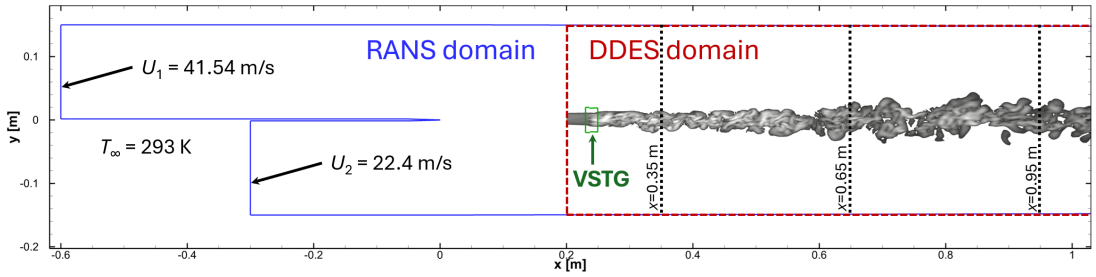
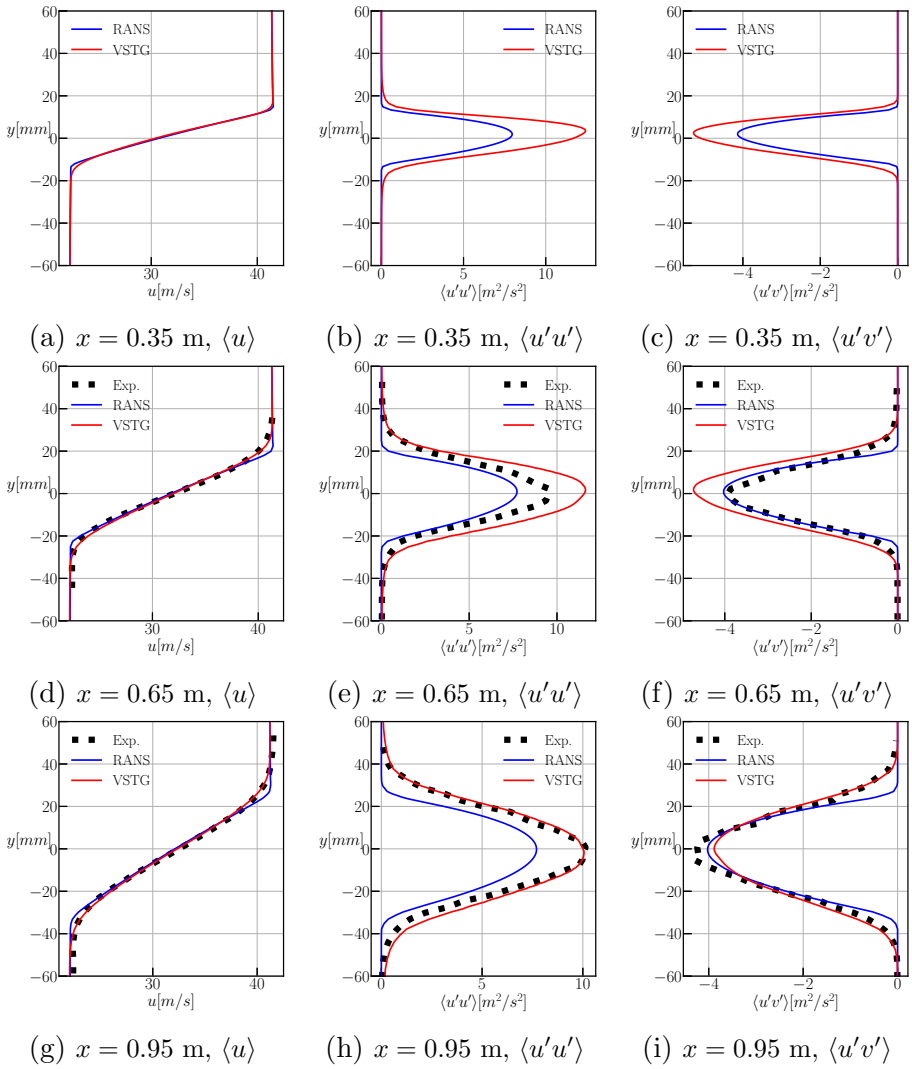


Figure 3. Mixing layer: schematics of the flow configuration and simulation set-up

The simulations were carried out in two stages. 2D SST RANS solution was obtained at the first stage using a low-Reynolds mesh (wall-normal mesh step $\Delta^+_{y,1} < 1$) taken from [1] The computational domain (see RANS domain in Fig. 3) fits into a rectangle $-0.6 \le x/L \le 1.2$ and $-0.15 \le y/L \le 0.15\}$, the trailing edge is located at $(0, 0)$. The freestream velocities were set at the inlet: the upper one is at $x/L = -0.6$, the lower one – at $x/L = -0.3$. Ambient pressure was prescribed at the outlet. The walls of the plate were treated as adiabatic no-slip. At the second stage, the DDES [14] simulation was carried out using the 3D mesh containing 7.61 M nodes (220 nodes in the spanwise direction, along OZ axis). The profiles obtained using 2D SST RANS were imposed at the inlet, which is located at $x/L = 0.2$ (see DDES domain in Fig. 3). The outlet boundary was extended till $x/L = 2$. The streamwise mesh step starts from $\Delta_x/L = 10^{-3}$ till $\Delta_x/L = 4 \cdot 10^{-3}$ at $x/L = 1.1$, and coarsens towards the outlet boundary. The transverse mesh step starts from $\Delta_y/L = 1.5 \cdot 10^{-4}$ till $\Delta_y/L = 6 \cdot 10^{-4}$ at $x/L = 1.1$ along the line $y = 0$, and coarsens towards the outlet boundaries. The spanwise mesh step is $\Delta_z/L = 5 \cdot 10^{-4}$. The VSTG was imposed inside the region $0.23 \le x/L \le 0.25$, the reference nodes are located $l_{\mathrm{us}}/L = 0.01$ upstream.

An instantaneous snapshot of the vorticity magnitude from the SRS simulation is shown in Fig. 3 (see visualisation within the DDES domain). The flow in the mixing layer becomes plausibly turbulent rather quickly: the length of the relaxation period does not exceed its several thicknesses. Averaged streamwise velocity profiles and Reynolds stresses are presented in Fig. 4. Note that the $x/L = 0.35$ profile corresponds to a location approximately $6\delta_{\omega 0}$ downstream the VSTG zone ($\delta_{\omega 0}$ is the vorticity thickness at $x/L = 0.2$). It can be seen that the SRS distributions are close to both the experimantal data and the RANS solution. The shape of the graphs is captured properly, while peak values are slightly overestimated at $x/L = 0.65$.



(a) $x = 0.35$ m, $\langle u \rangle$ (b) $x = 0.35$ m, $\langle u'u' \rangle$ (c) $x = 0.35$ m, $\langle u'v' \rangle$

(d) $x = 0.65$ m, $\langle u \rangle$ (e) $x = 0.65$ m, $\langle u'u' \rangle$ (f) $x = 0.65$ m, $\langle u'v' \rangle$

(g) $x = 0.95$ m, $\langle u \rangle$ (h) $x = 0.95$ m, $\langle u'u' \rangle$ (i) $x = 0.95$ m, $\langle u'v' \rangle$

Figure 4. Mixing layer: profiles of averaged streamwise velocity and Reynolds stresses

4. Unsteady Flow over the T106 LPT Cascade

4.1. Computational Setup

The test case corresponds to an experimental LPT linear rig [27] with the following geometrical details: inlet flow angle 45.5°, axial chord $C_x = 0.86C$ (C is chord), pitch $P = 0.799C$. The T106 LPT cascade was investigated in [11, 18, 32], there is reference data for both steady (without turbulent income) and unsteady regimes. The following parameters define the flow over the T106 cascade: isentropic exit Mach number $\mathrm{Ma}_{2\mathrm{nd}} = 0.4$, Reynolds number based on C and exit velocity u_{out} (hereinafter also denoted as u_{ref}) is $9.8 \cdot 10^4$. The Reynolds number based on the inlet velocity u_{in} is $5.1 \cdot 10^4$. The blade aspect ratio is 1.76, thus the flow is considered to be nearly two-dimensional at mid-span. Schematics of the flow configuration and simulation set-up is visualized in Fig. 5. The inlet boundary is located at $x/C = -0.35$, the outlet – at $x/C = 2.05$. The point $(x, y) = (0, 0)$ corresponds to the leading edge of the blade.

The effect of an upstream blade row is simulated by a moving cylinder wake generator (moving bars) with a cylinder diameter $d = 0.02C$. The bars located at C_x upstream the leading edge of the T106 blade move with tangential velocity $u_{\mathrm{bar}} = 0.41u_{\mathrm{in}}$ along the OY axis down-

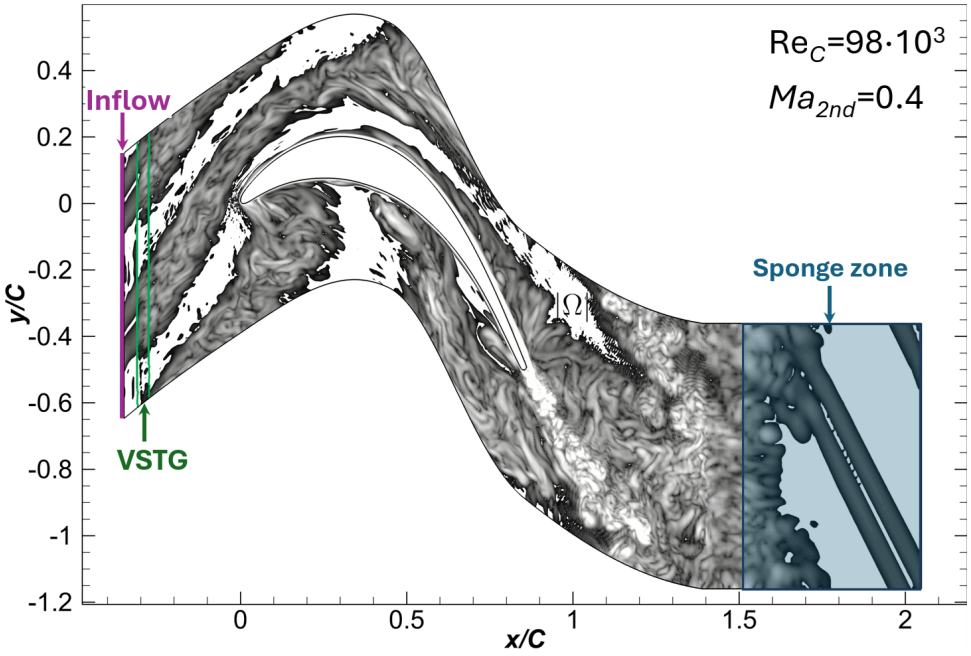


Figure 5. T106: schematics of the flow configuration and simulation set-up

ward y. It is considered that blades do not affect the flow over bars, so we simulate the flow over bars separately. The computational domain for the bars (see Fig. 6 top) is aligned with the coordinate system for the blade. It is two times smaller along OY due to the fact that there are two cylinders per pitch P and ranges from $-1.72C$ to $1.032C$ along the OX axis. RANS computation was performed in the coordinate system associated with the moving cylinders.

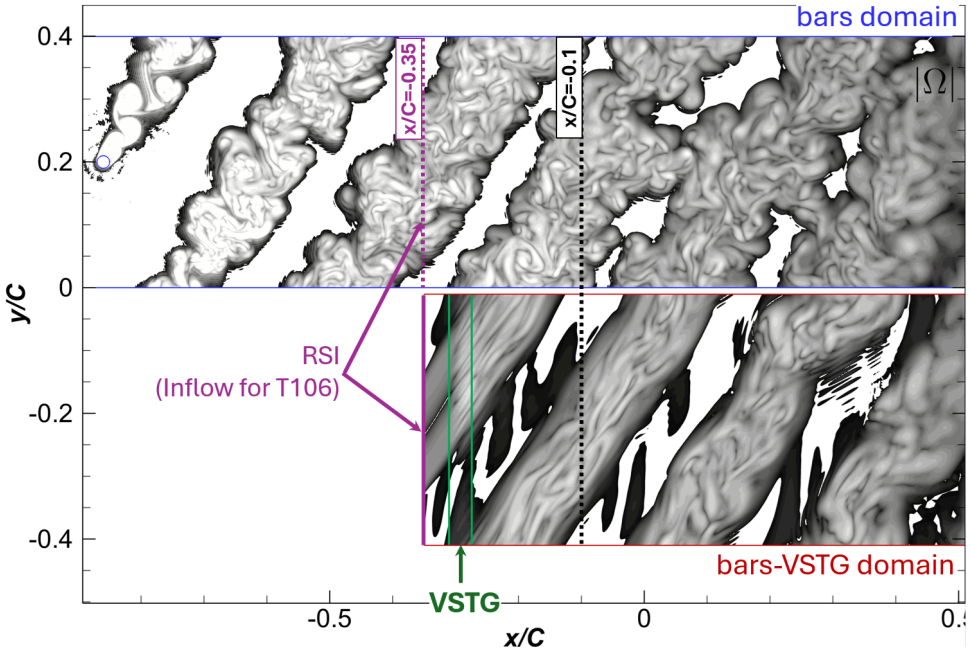


Figure 6. Moving bars: schematics of the flow configuration and simulation set-up (a fragment of the bars domain at the top and of the bars-VSTG domain at the bottom

The following boundary conditions were applied in simulations: specified total pressure P_t, total temperature T_t, flow direction and turbulence characteristics (k and ω) at the inflow

boundaries; static pressure p at the outlet; periodicity along y and z directions. To generate data for providing the unsteady boundary conditions for the SRS computation, the RANS solution of the bars configuration at cross section $x/C = -0.35$ was decomposed by N_h spatial harmonics with base wave number $2\pi/P$ (first harmonic). It was done for P_t, T_t, k and ω, but not for flow directions due to their negligible variation. The fundamental frequency of relative periodic flicker (actually, the blade passing frequency) of these spatial perturbations from bars is equal to $\Omega_{\text{bar}} = 2\pi P/u_{\text{bar}}$. Having Fourier coefficients $\{\widetilde{Q}^a_j + i\widetilde{Q}^b_j\}_{j=0}^{N_h}$ (i is the imaginary unit) for the variable Q, its instantaneous value $Q(y,t)$ (it depends on the spatial coordinate y and time t) at the inlet boundary is calculated by the formula

$$Q(t,y) = \overline{Q} + \sum_{j=1}^{N_h} \left[\Big(\widetilde{Q}^a_j \cos(sj) + \widetilde{Q}^b_j \sin(sj) \Big) \cos(\tau j) + \Big(\widetilde{Q}^b_j \cos(sj) - \widetilde{Q}^a_j \sin(sj) \Big) \sin(\tau j) \right], \quad (5)$$

where $s = 2\pi y/P$ and $\tau = \Omega_{\text{bar}} t$, $\overline{Q} = Q_0$ is the averaged value of Q. The formula (5) is an adaptation for the simplified configuration under study of the generalized formula to transform the spatial harmonics stationary with respect to the rotor into unsteady time harmonics with respect to the stator (see [7]).

The results of unsteady effects on the T106 blades were obtained in the following order:

1. 2D SST RANS simulation of a cylinder (Bars2D mesh, see Tab. 1).
2. Extraction of profiles for SRS, their decomposition into N_h spatial harmonics.
3. 3D SST IDDES simulations with unsteady inflow BC (T106 mesh).

Also, several auxiliary scale-resolving computations were performed, as follows.

- A computation for the bars configuration (3D bars domain) was carried out to obtain the Reynolds stresses in the wake downstream the cylinder (the Bars3D mesh was used, see Tab. 1) that were used as a reference for evaluation of results.
- To evaluate the capability of reproducing turbulence in the cylinder wake with VSTG based on the parameters given by the RANS solution, computations on two meshes, Wake3Dc and Wake3Df, were performed in the domain, the input boundary of which was the same as for the T106 (at $x/C = -0.35$). A fragment of the computational domain is shown in Fig. 6 bottom (see bars-VSTG domain).

Parameters of the unstructured meshes used for computations are shown in Tab. 1. All 3D meshes were generated by extrusion of a 2D mesh (e.g., Bars3D is based on Bars2D) along the OZ axis with a constant step (denoted as Δ_z). 2D quad-dominant meshes are isotropic except for the near-wall region, where a proper number of layers with anisotropic elements were inserted to meet the $\Delta^+_{\text{wn},1} < 1$ condition. Isotropic resolution with the corresponding maximum mesh step $\Delta^{xy}_{\max}$ was maintained in the area specified as "$\Delta^{xy}_{\max}$ area" in Tab. 1. Also, the following designations are used in Tab. 1: N_n and $N_{n,\text{2D}}$ are the total numbers of nodes in the 3D and 2D meshes, respectively; N_z is the number of cells in the spanwise direction.

IDDES was used for all scale-resolving computations. The RANS regime of IDDES was forced from the input boundary down to the VSTG region. Otherwise, the RANS solution was damped fairly quickly by operating in LES mode. The VSTG zone $-0.31 \le x/C \le -0.275$ is fixed for all computations where it was used, the reference nodes are located $0.03C$ upstream the source area. Downstream the VSTG zone the IDDES operates in the LES regime, including the near-wall regions. Distributions of mesh steps (in wall-law variables) along the T106 blade surface from the unsteady simulation are presented in Fig. 7. It is evident from this that the mesh has sufficient resolution for WRLES. The upwind EBR3 scheme was used for RANS simulations,

Table 1. Parameters of meshes for T106 LPT related computations

Mesh	N_n	$N_{n,2D}$	N_z	Δ_z/C	$\Delta^{xy}_{\max}/C$	$\Delta^{xy}_{\max}$ area
Bars2D	0.282M	–	–	–	$1.5 \cdot 10^{-3}$	$-1.05 \le x/C \le 0.35$
Bars3D	14.4M	0.282M	50	$1.5 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$	$-1.05 \le x/C \le 0.35$
Wake3Dc	3.68M	0.072M	50	$1.5 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$0.35 \le x/C \le 0.35$
Wake3Df	6.76M	0.133M	50	$1.5 \cdot 10^{-3}$	$1.5 \cdot 10^{-3}$	$0.35 \le x/C \le 0.35$
T106	18M	0.175M	100	$1.5 \cdot 10^{-3}$	$3 \cdot 10^{-3}$	$0.35 \le x/C \le 1.5$

while the lower-dissipation EBR4 scheme blended with EBR3 (with the coefficient 0.01) was exploited for SRS. The global CourantFriedrichsLewy (CFL) condition, $\mathrm{CFL}_{\max} = 40$, was set for all scale-resolving computations. The local CFL values did not exceed 1 in the disturbed flow areas, which is needed to resolve well all relevant turbulent time scales. The time step $\Delta t u_{\text{out}}/C$ slightly varied dynamically according to the global CFL condition, the average value was $1.23 \cdot 10^{-3}$. To accumulate the average flow statistics, we used averaging in time and in space along the homogeneous direction. The transient period of the simulations started from a preliminary 2D RANS solution is $8C/u_{\text{out}}$. After the transient period, the time integration interval of data accumulation is $30C/u_{\text{out}}$, which is more than sufficient to obtain converged flows parameters.

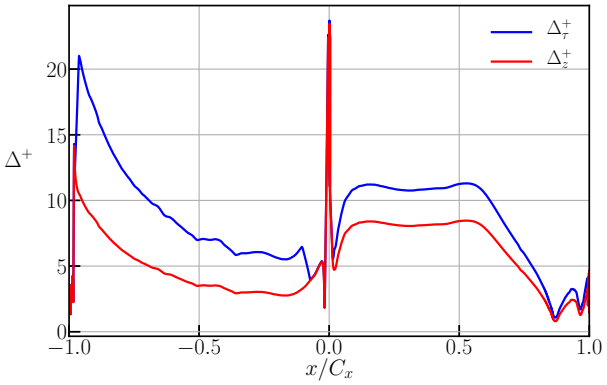


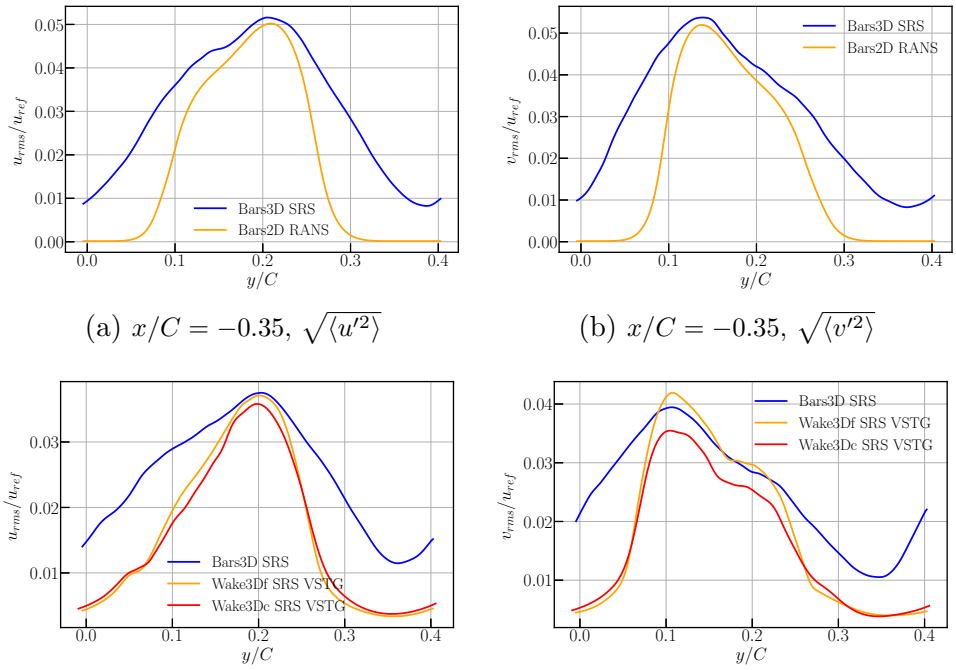
Figure 7. T106: distributions of wall-tangent Δ^+_τ and spanwise Δ^+_z mesh steps at the pressure ($x/C_x < 0$) and suction ($x/C_x \ge 0$) sides of the blade

All scale-resolving simulations were performed on NVIDIA V100 GPUs (900 GB/s). One timestep for simulation of the T106 blades cascade (unsteady regime) on 4 GPUs in the K60-GPU cluster (1 node with 2 Intel Xeon Gold 6142 CPUs and 4 GPUs) takes approximately 0.9 s of wall clock time. Consequently, it takes about 2 hours to simulate the period $10C/u_{\text{out}}$. Note that all VSTG nodes (their number is about 1.5% of the total number of nodes) belong to the first MPI domain (of four), so all source-related operations are performed by only one GPU (for four). The number of STG modes is 167.

4.2. Results

The results of the computations preceding SRS of the T106 blades cascade are presented in Figures 8 and 9: profiles of the Reynolds stresses are shown in Fig. 8, P_t and k distributions are presented in Fig. 9 (here $p_{\text{ref}} = \rho_{\text{ref}} u^2_{\text{ref}}$, where ρ_{ref} is the density at the outlet of the blades

cascade). The latter contains approximations based on spatial Fourier decomposition depending on the number of harmonics N_h too. The cross section $x/C = -0.35$ corresponds to the inlet boundary for the T106 domain (and for the bars-VSTG domain too), $x/C = -0.1$ is a section slightly upstream the leading edge of the blade. It is clearly seen that the Reynolds stresses obtained using RANS differ from the SRS results, which we consider as a reference. The shape of the graphs and peak values in Fig. 8 correlate well, whereas the main discrepancy lies in the region outside the center of the wake: turbulence levels are much higher there. Underestimation is related to limitations of the SST RANS model to predict flow characteristics past bluff bodies such as a round cylinder: presence of coherent structures and their downstream evolution is not considered correctly. However, shear layers characteristics downstream streamlined bodies (e.g., aerodynamic profiles or blades of rotating machines) are usually captured more accurately by Boussinesq-type turbulence models. Comparing the results obtained on the meshes Wake3Dc and Wake3Df, we conclude that the resolution of the Wake3Dc mesh is sufficient to provide adequate turbulence generated by the VSTG, the parameters of which are set based on the RANS solution. Note that the Wake3Dc mesh has the same resolution as the T106 one.



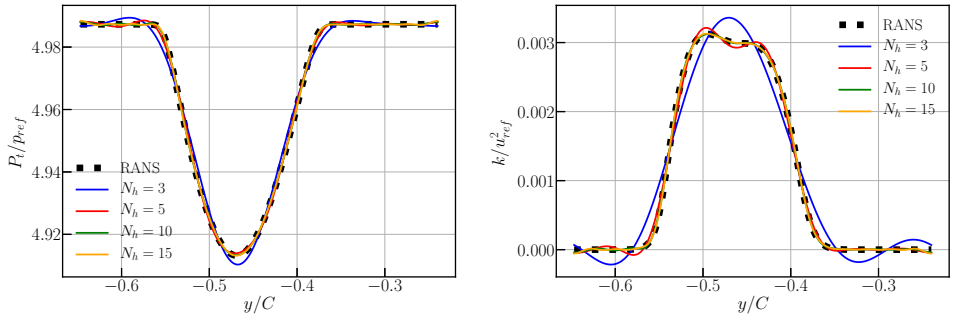
(a) $x/C = -0.35$, $\sqrt{\langle u'^2 \rangle}$ (b) $x/C = -0.35$, $\sqrt{\langle v'^2 \rangle}$

(c) $x/C = -0.1$, $\sqrt{\langle u'^2 \rangle}$ (d) $x/C = -0.1$, $\sqrt{\langle v'^2 \rangle}$

Figure 8. Bar results: profiles of normal Reynolds stresses (rms values)

Evaluating the graphs in Fig. 9, it can be seen that actual distributions of fields are approximated quite accurate by spatial harmonics starting from $N_h = 5$. After this value the difference between restored functions lies in the region outside peaks. We applied $N_h = 15$ to set unsteady BC for the T106 computation. Note that there are deviations of the approximated functions with values less than zero for turbulence kinetic energy, that is not appropriate. We limited the k values reconstructed through the sum of harmonics to zero from below to solve this problem.

Distributions of vorticity magnitude $|\Omega|$ at three consecutive time instants with a step $0.615C/u_{\text{out}}$ are shown in Fig. 10. It can be seen that artificial pulsations within the VSTG quickly transform into plausible resolved turbulence at a distance of about 2-3 thicknesses of



(a) total pressure (b) turbulence kinetic energy

Figure 9. Bar results: profiles of variables – actual and reconstructed by the corresponding number of harmonics N_h

the source zone downstream. These wakes interfere with both pressure and suction sides of the blade. The boundary layer at the suction side states predominantly laminar except for a very small region located in close proximity to the trailing edge downstream the reattachment after a bubble-type LT transition (it is almost indistinguishable from the visualizations shown in Fig. 10). On the pressure side, near-wall turbulence appears after interference with the tails of periodic wakes.

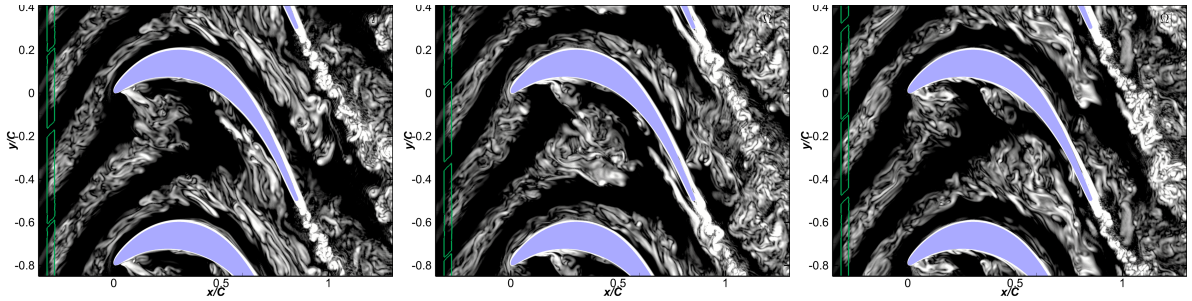
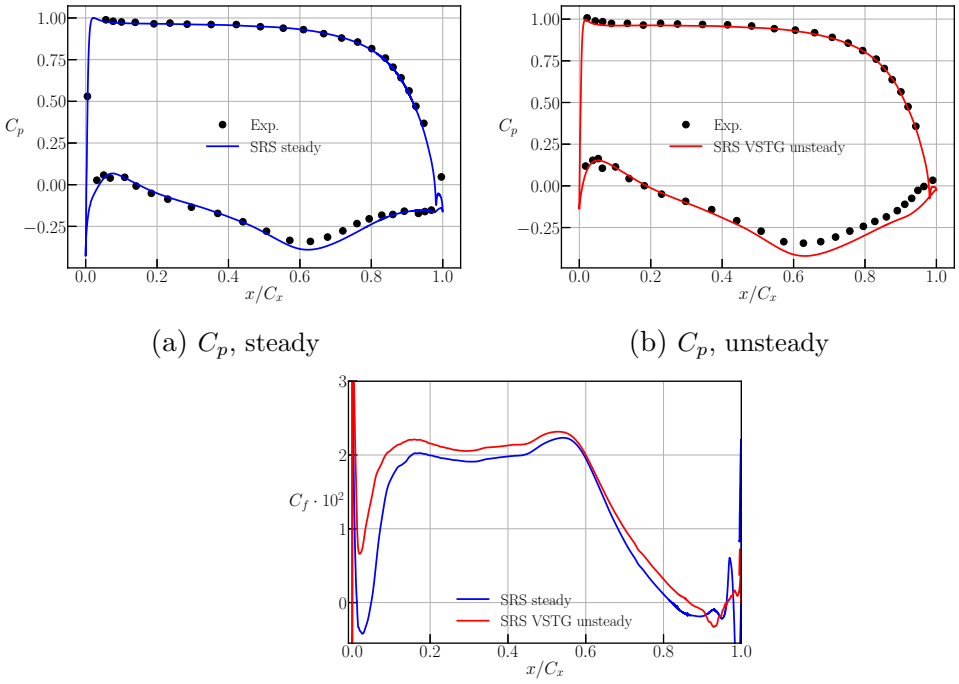


Figure 10. T106: distributions of vorticity magnitude at three consecutive time instants with a step $0.615C/u_{\text{out}}$

The effect of unsteady periodic perturbations is clearly manifested in the distributions of pressure C_p coefficient and friction C_f coefficient (for the suction side only) presented in Fig. 11. First of all, we note good agreement with the experimental data for both regimes, with zero turbulence level at the inlet (denoted as "SRS steady") and with unsteady perturbations ("SRS VSTG unsteady"). The presence of the so-called bubble laminar turbulent (LT) transition on the suction side of the blade characterizes the first regime. There is also a small laminar bubble near the leading edge (see the C_f distribution of SRS steady in Fig. 11 c). As for the unsteady regime, the wakes interfere with the blade, causing the LT transition zone to move closer to the trailing edge and the recirculation zone to become noticeably smaller. In this case, the separation near the leading edge disappears completely. The LES simulation of the regime with unsteady perturbations from [11] predicted the absence of a bubble near the trailing edge, i.e., the transition becomes a bypass rather than of a bubble type. As a consequence, their results are in better agreement with the experiment in the region $x/C_x > 0.6$. But the flow over the bars was reproduced more "realistic" in [11]: pulsations were not simulated artificially. Therefore, we attribute the lower accuracy of our results to the limited accuracy of the RANS solution on



(a) C_p, steady

(b) C_p, unsteady

(c) C_f, both steady and unsteady

Figure 11. T106 results: averaged distributions of pressure (C_p) and friction (C_f) coefficients

the basis of which the pulsations are generated and, in general, to the difficulty in reproducing turbulence with the presence of specific coherent structures using the generator we apply. This is despite the computational meshes used, which are close in resolution (our mesh to the mesh from [11]). However, in turbomachines, the blade wakes are closer to the mixing layers, for the artificial reproduction of which the STG is better adapted.

Conclusion

The methodology for studying the effects associated with periodic unsteady impact from the preceding row in turbomachines using scale-resolving simulation is presented. It considers the two-stage procedure of an investigation: RANS simulation of an entire turbomachine at the first stage; SRS of the row under consideration at the second. The methodology is based on the following methods and technologies, which are implemented in the NOISEtte computational algorithm.

- Nonlinear harmonics method which is used to obtain unsteady inflow parameters for SRS in a preliminary RANS computation.
- Hybrid RANS-LES IDDES approach for scale-resolving simulation of the row under detailed study.
- Dynamic VSTG to reproduce unsteady periodic turbulent perturbations within the SRS stage.

The VSTG is adapted to the capabilities of dynamically varying turbulent flow downstream of the source region. All computationally intensive calculations are conducted using the GPU version of the NOISEtte code following its heterogeneous MPI+OpenMP+OpenCL parallelization model. All possible operations of scale-resolving computations are efficiently performed on

graphic accelerators. The primary issue is attributed to the fact that, due to the locality of the source region and the substantial computational load associated with its operation, a load imbalance arises in the context of MPI parallelism (only one GPU of multiple GPUs could be loaded with VSTG-related operations).

The advantage of the methodology is that the most computationally intensive part, namely, scale-resolving simulation, is carried out only for one domain representing a periodic sector of a row, the characteristics which are studied in detail. At the same time, the use of NLH allows to obtain plausible inflow conditions representing the unsteadiness of adjacent rows for realistic turbomachines.

The simulation of unsteady effects at the T106 LPT cascade was performed to demonstrate applicability of the methodology presented. The obtained results correlate well with the experimental data for both simulated regimes, with zero turbulence level at the inlet and with unsteady perturbations. However, it was not possible to exactly reproduce the effect of the periodic wakes on the T10C blades cascade, likely due to the insufficiently accurate reproduction of the flow downstream the cylinders using VSTG. It is therefore evident that the capabilities of the proposed technique are constrained by the capabilities of the synthetic turbulence generator that is utilised. In the context of realistic turbomachines, the blade wakes are in closer proximity to the mixing layers. And the VSTG is better adjusted to the artificial reproduction of them. Consequently, the proposed technology appears to be a potentially viable solution.

Subsequent research will address the potential for reducing the computational cost of operations associated with VSTG. Also, there is a strong interest in the full application of the developed technology, i.e., on a realistic turbomachinery problem (multi-stage turbomachine), where the non-simplified NLH technology is employed at the RANS stage.

Acknowledgements

The research is carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University [30], the equipment of Shared Resource Center of KIAM RAS (http://ckp.kiam.ru). The authors thankfully acknowledge these institutions.

References

1. NASA Turbulence Modeling Resource. 2DML: 2D Mixing Layer Validation Case. `https://turbmodels.larc.nasa.gov/delvilleshear_val.html`

2. Abalakin, I.V., Bakhvalov, P.A., Bobkov, V.G., *et al.*: NOISEtte CFD&CAA Supercomputer Code for Research and Applications. Supercomputing Frontiers and Innovations 11(2), 78–101 (Aug 2024). `https://doi.org/10.14529/jsfi240206`

3. Abalakin I., Bakhvalov P., Kozubskaya, T.: Edge-based reconstruction schemes for unstructured tetrahedral meshes. International Journal for Numerical Methods in Fluids 81, 331–356 (2016). `https://doi.org/10.1002/fld.4187`

4. Bakhvalov, P.A., Surnachev, M.D.: Method of averaged element splittings for diffusion terms discretization in vertex-centered framework. Journal of Computational Physics 450, 110819 (Feb 2022). https://doi.org/10.1016/j.jcp.2021.110819

5. Burgos, M.A., Contreras, J., Corral, R.: Efficient edge-based rotor/stator interaction method. AIAA Journal 49(1), 19–31 (Jan 2011). https://doi.org/10.2514/1.44512

6. Delville, J., Bellin, S., Garem, J.H., Bonnet, J.P.: Analysis of Structures in a Turbulent, Plane Mixing Layer by Use of a Pseudo Flow Visualization Method Based on Hot-Wire Anemometry, pp. 251–256. Springer Berlin Heidelberg (1989). https://doi.org/10.1007/978-3-642-83822-4_38

7. Duben, A.P., Zagitov, R.A., Shuvaev, N.V.: Nonlinear harmonics method for supercomputer simulations of fluid dynamics in turbomachines with higher accuracy on unstructured meshes. Lobachevskii Journal of Mathematics 45(7), 3007–3016 (Jul 2024). https://doi.org/10.1134/s1995080224603916

8. Duben, A., Gorobets, A.: Scale-resolving simulation of a low-pressure turbine on hybrid supercomputers. Computers & Fluids 265, 105984 (Oct 2023). https://doi.org/10.1016/j.compfluid.2023.105984

9. Duben, A., Gorobets, A., Soukov, S., *et al.*: Supercomputer Simulations of Turbomachinery Problems with Higher Accuracy on Unstructured Meshes, pp. 356–367. Springer International Publishing (2022). https://doi.org/10.1007/978-3-031-22941-1_26

10. Duben, A., Zagitov, R., Shuvaev, N., Marakueva, O.: Towards anAdaptation of the Nonlinear Harmonics Method Realized in an Unstructured Flow Solver for Simulation of Turbomachinery Problems on Supercomputers, pp. 253–266. Springer Nature Switzerland (2025). https://doi.org/10.1007/978-3-031-78459-0_19

11. Fiore, M., Gojon, R., Sáez-Mischlich, G., Gressier, J.: LES of the T106 low-pressure turbine: Spectral proper orthogonal decomposition of the flow based on a fluctuating energy norm. Computers & Fluids 252, 105761 (Feb 2023). https://doi.org/10.1016/j.compfluid.2022.105761

12. Gorobets, A.: Adapting a Scientific CFD Code to Industrial Applications on Hybrid Supercomputers. Supercomputing Frontiers and Innovations 9(4), 49–54 (Nov 2022). https://doi.org/10.14529/jsfi220405

13. Gorobets, A., Bakhvalov, P.: Heterogeneous CPU+GPU parallelization for high-accuracy scale-resolving simulations of compressible turbulent flows on hybrid supercomputers. Computer Physics Communications 271, 108231 (Dec 2022). https://doi.org/10.1016/j.cpc.2021.108231

14. Guseva, E.K., Garbaruk, A.V., Strelets, M.K.: Assessment of Delayed DES and Improved Delayed DES Combined with a Shear-Layer-Adapted Subgrid Length-Scale in Separated Flows. Flow, Turbulence and Combustion 98, 481–502 (2017). https://doi.org/10.1007/s10494-016-9769-7

15. Hu, S., Zhou, C., Xia, Z., Chen, S.: Large Eddy Simulation and CDNS Investigation of T106C Low-Pressure Turbine. Journal of Fluids Engineering 140(1) (Oct 2017). https://doi.org/10.1115/1.4037489

16. Magomedov, A.R., Gorobets, A.V.: Heterogeneous Implementation of Preconditioners Based on Gauss-Seidel Method for Sparse Block Matrices. Computational Mathematics and Modeling 33(4), 438–442 (Oct 2022). https://doi.org/10.1007/s10598-023-09585-2

17. Menter, F.R.: Two-equation eddy-viscosity turbulence models for engineering applications. AIAA Journal 32(8), 1598–1605 (Aug 1994). https://doi.org/10.2514/3.12149

18. Michelassi, V., Wissink, J., Rodi, W.: Analysis of DNS and LES of Flow in a Low Pressure Turbine Cascade with Incoming Wakes and Comparison with Experiments. Flow, Turbulence and Combustion 69, 295–329 (2002). https://doi.org/10.1023/a:1027334303200

19. Mockett, C., Fuchs, M., Garbaruk, A., *et al.*: Two non-zonal approaches to accelerate RANS to LES transition of free shear layers in DES. In: Progress in Hybrid RANS-LES Modelling, pp. 187–201. Springer International Publishing (2015). https://doi.org/10.1007/978-3-319-15141-0_15

20. Saad, Y.: Iterative methods for sparse linear systems. Society for Industrial and Applied Mathematics, Philadelphia (2003)

21. Sandberg, R.D., Michelassi, V.: The current state of high-fidelity simulations for main gas path turbomachinery components and their industrial impact. Flow, Turbulence and Combustion 102(4), 797–848 (Feb 2019). https://doi.org/10.1007/s10494-019-00013-3

22. Sandberg, R.D., Michelassi, V., Pichler, R., *et al.*: Compressible Direct Numerical Simulation of Low-Pressure Turbines–Part I: Methodology. Journal of Turbomachinery 137(5) (May 2015). https://doi.org/10.1115/1.4028731

23. Shorstov, V.A.: Possibilities and Restrictions on the Use of the Zonal RANS-IDDES Approach to Calculate Fan Noise. Mathematical Models and Computer Simulations 15(2), 265–276 (Apr 2023). https://doi.org/10.1134/s2070048223020163

24. Shur, M., Strelets, M., Travin, A., *et al.*: Improved Embedded Approaches. Notes on Numerical Fluid Mechanics and Multidisciplinary Design 134, 65–69 (2017). https://doi.org/10.1007/978-3-319-52995-0_3

25. Shur, M.L., Spalart, P.R., Strelets, M.K., Travin, A.K.: Synthetic Turbulence Generators for RANS-LES Interfaces in Zonal Simulations of Aerodynamic and Aeroacoustic Problems. Flow, Turbulence and Combustion 93, 63–92 (2014). https://doi.org/10.1007/s10494-014-9534-8

26. Shur, M.L., Spalart, P.R., Strelets, M.K., Travin, A.K.: An enhanced version of DES with rapid transition from RANS to LES in separated flows. Flow, Turbulence and Combustion 95(4), 709–737 (2015). https://doi.org/10.1007/s10494-015-9618-0

27. Stadtmüller, P.: Investigation of wake-induced transition on the LPturbine cascade T106A-EIZ. Tech. Rep. Fo 136/11, Version 1.1, DFG-Verbundproject (2002)

28. Suzuki, T., Spalart, P.R., Shur, M.L., *et al.*: Unsteady simulations of a fan/outlet-guide-vane system: Broadband-noise computation. AIAA Journal 57(12), 5168–5181 (Dec 2019). https://doi.org/10.2514/1.j058177

29. Tyacke, J., Vadlamani, N.R., Trojak, W., *et al.*: Turbomachinery simulation challenges and the future. Progress in Aerospace Sciences 110, 100554 (Oct 2019). https://doi.org/10.1016/j.paerosci.2019.100554

30. Voevodin, V., Antonov, A., Nikitenko, D., *et al.*: Supercomputer Lomonosov-2: Large scale, deep monitoring and fine analytics for the user community. Supercomput. Front. Innov. 6(2), 4–11 (2019). https://doi.org/10.14529/jsfi190201

31. Voroshnin, D.V., Marakueva, O.V., Muraveiko, A.S.: Modeling unsteady phenomena in an axial compressor. Mathematical Models and Computer Simulations 12(3), 413–421 (May 2020). https://doi.org/10.1134/s2070048220030187

32. Wu, X., Durbin, P.A.: Evidence of longitudinal vortices evolved from distorted wakes in a turbine passage. Journal of Fluid Mechanics 446, 199–228 (Oct 2001). https://doi.org/10.1017/s0022112001005717

DOI: 10.14529/jsfi250108

Numerical Study of Noise Sources Generated by Wing of Supersonic Business Jet in Landing Mode

Tatiana K. Kozubskaya[1], ***Gleb M. Plaksin***[1], ***Ivan L. Sofronov***[2], ***Pavel V. Rodionov***[1]

The paper is devoted to the numerical study of wing noise for the prototype of supersonic business jet in landing mode. Near the wing, the acoustics is simulated using the CFD/CAA methods within the Delayed Detached Eddy Simulation approach. The Ffowcs Williams–Hawkings method is used for calculation of noise radiation in the far field. To localize the near-field acoustic sources, the advanced postprocessing including the numerical beamforming method is applied. The numerical beamforming formulated for monopole- and dipole-type sources allowed for detecting the main sources of the wing noise in the vicinity of leading and trailing edges. Analysis of the sound pressure level calculated for signals recorded on the Ffowcs Williams–Hawkings surface during the CFD simulations generally confirmed these results. Direct comparison of the noise spectra calculated by the Ffowcs Williams–Hawkings and numerical beamforming methods is provided for selected mid-field points. According to the presented noise radiation pattern, the far-field noise generated by the considered wing in landing mode has dominating dipole-type component for frequencies lower than 250 Hz and dominating monopole-type component for frequencies higher than 1 kHz.

Keywords: supercomputer simulation, supersonic business jet, acoustic source, computational fluid dynamics, computational aeroacoustics, numerical beamforming, monopole, dipole, turbulent flow, unstructured mesh.

Introduction

The design of modern civil aircraft, even at the earliest stages, includes not only optimization of aerodynamic characteristics, but also the reduction of total aircraft noisiness. The main impetus to the noise reduction arises from the severe requirements of International Civil Aviation Organization (ICAO) regarding the level of noise generated by civil aircraft near airports. These requirements apply to all types of civil aircraft including supersonic business jets (SSBJ) which are currently being developed in different countries.

Nowadays, numerical simulation is an important design tool widely used in the aerospace industry. While numerical assessment of aerodynamic characteristics has already become a standard component of the design process, numerical assessment of acoustic characteristics for aerospace problems is still in the development and implementation phase due to its high computational cost. Fortunately, the constantly increasing performance and availability of modern supercomputer clusters make aeroacoustic simulations feasible for single high-priority evaluations. In this paper, we present the results of aeroacoustic simulation for prototype of SSBJ airframe in landing mode.

Numerical assessment of acoustic characteristics of an aircraft is primarily based on resource-intensive CFD (Computational Fluid Dynamics) simulations of unsteady turbulent flow. Such simulations, sometimes referred to as eddy-resolving or scale-resolving, require to resolve high-energy turbulent eddies on computational grid and allow modeling the smaller-scale turbulence from inertial range by subgrid models. Note that for reliable results of scale-resolving simulations the characteristic mesh size in the region of unsteady turbulent flow should belong to inertial

[1]Keldysh Institute of Applied Mathematics, RAS, Moscow, Russian Federation
[2]Moscow Institute of Physics and Technology, Dolgoprudny, Moscow Region, Russian Federation

range of turbulence scales. For airframe noise simulations, there are two main approaches currently in use: the first is Detached Eddy Simulation (DES) [6, 7, 19, 25] based on hybridization of Reynolds averaged Navier–Stokes (RANS) method and Large Eddy Simulation (LES), and the second is the lattice Boltzmann method (LBM) [5, 8, 14, 15, 22]. In the present paper, we follow the DES approach to simulate the noise of turbulent flow over the SSBJ wing.

During scale-resolving CFD/CAA (Computational Aeroacoustics) simulations a huge amount of raw numerical data can be extracted from the obtained flow fields. However, to get practically useful information about the target flow, accumulated data should be properly processed by specialized methods and techniques. The great example of such postprocessing methods is the Ffowcs Williams–Hawkings (FWH) method [9] that allows to calculate acoustics at arbitrary far-field points by processing the near-field signals at special discretized surface. Another example of postprocessing methods is conventional beamforming [26, 30]. This method is widely used during wind-tunnel experiments and flight tests to localize and evaluate the noise sources generated by considered geometry or aircraft by processing signals recorded with far-field microphone arrays. In the recent papers [13, 16, 17, 21], we develop a novel method to localize and evaluate the near-field acoustic sources using CFD data – *the numerical beamforming* based on processing signals recorded with an arbitrarily large number of virtual near-field microphones moving with the considered object. One of the important features of our approach is that the formulation of the problem and the method of its solution do not require any a priori assumptions about whether the acoustic source is correlated or uncorrelated. In the present study, we apply the numerical beamforming to detect the noise sources generated by the SSBJ wing in landing mode. It is for the first time that this method is used to analyze acoustics of the full-scale aircraft geometry.

We should note that the numerical beamforming method solves the inverse problem regarding the near-field acoustic signals extracted from CFD simulation. Hence, the numerical beamforming has typical characteristics of inverse problems such as ill-posedness and solution non-uniqueness.

The paper consists of three sections. The first section is devoted to the scale-resolving simulations of turbulent flow near the prototype of SSBJ airframe in landing mode. This section presents the obtained characteristics of acoustic pressure pulsations near the leading and trailing edges of the SSBJ wing, at the far-field points and on the FWH surface. The second section formulates the numerical beamforming method for monopole-, dipole-type sources and describes the computational setup used to localize and evaluate the noise sources generated by the SSBJ wing. In the third section the results of numerical beamforming for the SSBJ wing in landing mode are discussed.

1. Scale-Resolving Simulations of Wing Noise

1.1. Problem Formulation

The full-scale airframe of SSBJ prototype [7] with $10^\circ - 10^\circ$ deflection of droop noses and $10^\circ - 20^\circ - 20^\circ - 10^\circ$ deflection of elevons on each side of the wing (Fig. 1) is placed inside the uniform airflow with the velocity $U_\infty = 68$ m/s, the pressure $P_\infty = 101325$ Pa and the temperature $T_\infty = 288.15$ K at an angle of attack 10°. The length of the considered geometry is 45 m, the wingspan is 20 m. The corresponding Mach number is 0.2, the Reynolds number based on the characteristic length $L = 1$ m is 4.6×10^6.

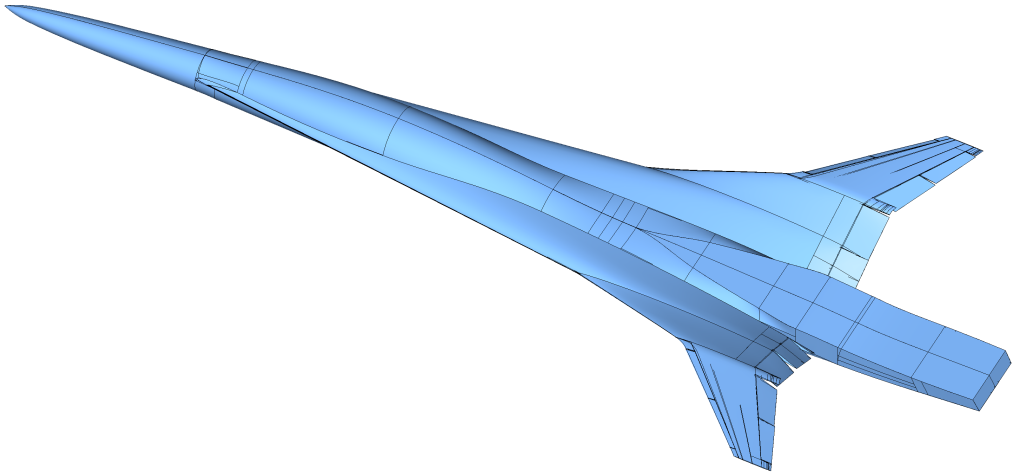


Figure 1. Airframe of SSBJ prototype with high-lift devices in landing configuration [7]

1.2. Computational Setup

To model noise of the SSBJ wing in near field, we perform scale-resolving simulations according to the Delayed Detached Eddy Simulation (DDES) approach [19, 25] with the subgrid scale $\Delta = \tilde{\Delta}_\omega$ [19] and the sigma subgrid-scale model [20] in the LES region and the Spalart–Allmaras (SA) turbulence model [27] in the RANS region. To model the properties of air, we use the calorically perfect gas with the ratio of specific heats $\gamma = 1.4$ and the specific gas constant $R_{\text{sp}} = 287.05$ J/(kg K).

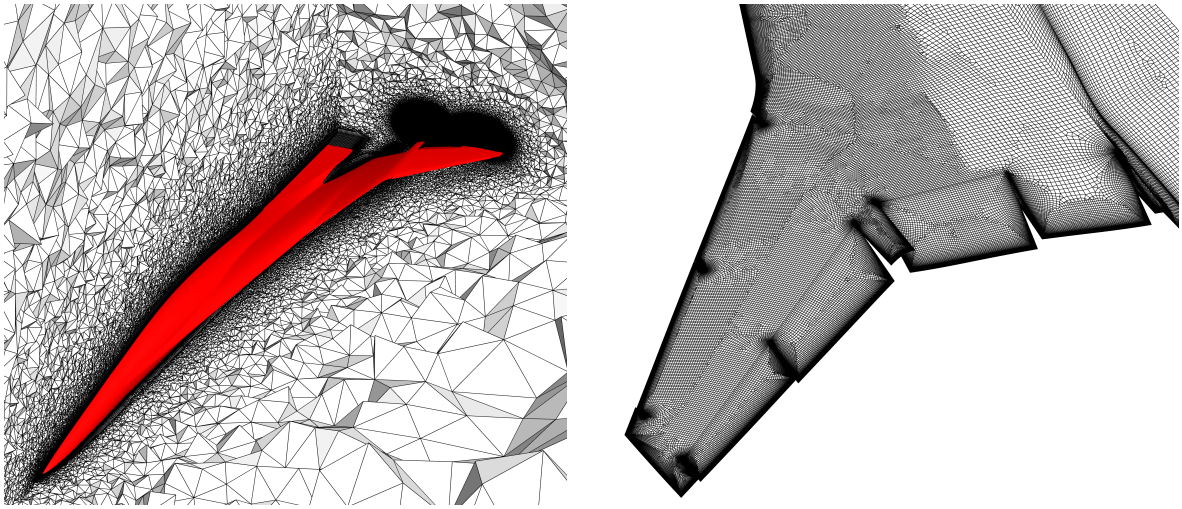
Due to the symmetry of the considered geometry and the problem parameters, we simulate the flow only for half of the airframe. This approach allows us to significantly reduce the computational costs of simulations. However, it eliminates asymmetric features of the flow that could potentially arise if the problem is solved in the entire region. All the acoustic data presented in the paper is calculated for half of the airframe as well. Because the acoustic sources located on different sides of the airframe are spatially separated, we can consider them as uncorrelated. Hence, to obtain sound intensity for the full airframe in far field, one can increase the corresponding intensity for half of the geometry by 3 dB.

The slip boundary conditions are set at the plane of symmetry $y = 0$, the free-stream conditions are used at the outer boundaries. Zero velocity and zero heat flux are specified on the streamlined geometry. To prevent the reflection of acoustic waves from the plane of symmetry in DDES simulations, the sponge layer [10] based on the averaged RANS solution is set in the region $0 \text{ m} \leq z \leq 1.5$ m.

The computational domain is defined by the parallelepiped 2000 m $\times$ 2000 m $\times$ 1000 m with the exclusion of SSBJ airframe interior. The center of the boundary at the plane of symmetry is located 31.5 m away from the SSBJ fore point along the x-axis.

We use the finite-volume vertex-centered EBR5 PL scheme [4] to approximate the convective flux and the method of local element splittings [3] to approximate the diffusive flux. For time integration, we apply the second-order implicit scheme based on the backward differentiation formula (BDF2). To solve the system of nonlinear equations at each time step, we use two Newton iterations. At each iteration, we solve the system of linear equations by the bi-conjugate gradient stabilized (BiCGStab) method [29] with the symmetric Gauss–Seidel (SGS) preconditioner.

Simulations are performed on two meshes denoted as Level A and Level B. Their general structure is shown in Fig. 2, their parameters are summarized in Tab. 1, where h_{fuselage} is the length of mesh edges in tangential directions near the fuselage and the lower surface of the wing, h_{vortices} is the length of mesh edges in the region of stable vortex flow over the wing. Outside the prismatic layers built near the streamlined geometry, the zone of increased mesh resolution over the wing is filled with an isotropic unstructured tetrahedral mesh.



(a) Near-field view (Level A mesh) (b) Airframe surface (Level A mesh)

Figure 2. General mesh structure

Table 1. Mesh parameters

Mesh	N_{nodes}	N_{elements}	$N_{\text{surf.nodes}}$	$N_{\text{surf.elements}}$	h_{fuselage}	h_{vortices}
Level A	21 166 948	46 552 132	337 330	342 475	70 mm	35 mm
Level B	61 601 940	219 587 977	678 233	687 362	70 mm	17.5 mm

We decrease the weight of upwind component of the EBR5 PL scheme according to the approach proposed in [12]. The minimum weight of upwind component in the zone of increased mesh resolution over the wing is set to 0.15. We choose the values of time step that provide relatively small regions of numerical instability. In terms of $\text{CFL}_{\text{vortices}} = \Delta t \times (c_\infty + U_\infty)/h_{\text{vortices}}$, where Δt is the time step and c_∞ is the speed of sound at infinity, we use $\text{CFL}_{\text{vortices}} = 0.083$ on the Level A mesh and $\text{CFL}_{\text{vortices}} = 0.125$ on the Level B mesh. When the flow is reached the steady state and the instantaneous solution is proved to have only small regions of numerical instability, we start to record the near-field acoustic data. The data recording is performed for time interval 60 L/U_∞ or 0.88 s. This interval size allows us to obtain smoothed spectra (averaged for 30 time segments with 0.5 overlapping) at the near-field and far-field points with the minimum resolved frequency 20 Hz.

All the DDES simulations presented in this paper are performed using the NOISEtte code [1] written in C++ and suitable for computations in CPU, GPU (OpenCL) and heterogeneous CPU+GPU modes with combined MPI+OpenMP parallelization. The parallel efficiency of the NOISEtte code and its performance on various supercomputers are presented in [11]. The aforementioned simulations are carried out using NVIDIA Tesla V100 GPUs on the Lomonosov-2

supercomputer [28] installed at Lomonosov Moscow State University. For the DDES simulation on the Level A mesh, 8 GPUs (4 compute nodes each equipped with 2 GPUs) are utilized for 21 hours to accumulate the required data on the time interval 60 L/U_∞. For the DDES simulation on the Level B mesh, 24 GPUs (12 compute nodes) are utilized for 24 hours to achieve the steady flow state after starting from the instantaneous DDES solution obtained on the Level A mesh and for 72 hours to accumulate the required data on the time interval 60 L/U_∞.

1.3. Far-Field Noise Calculation

We use the second-order FWH method [2, 9, 23, 24] to calculate acoustic pressure pulsations in the far field. The corresponding FWH surface used to accumulate the required acoustic data during DDES simulations is located near the boundaries of the zone of increased mesh resolution and contains five end caps (Fig. 3). Note that this surface has a slit on the fuselage side to prevent intersection with the wing surface. The far-field points used to calculate noise radiation pattern belong to the sphere of radius 150 m (Fig. 3).

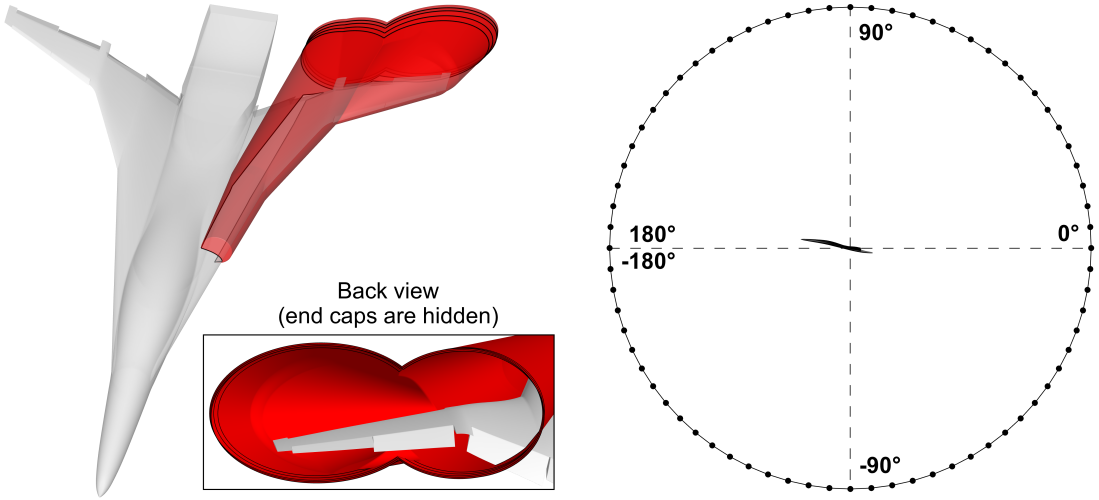


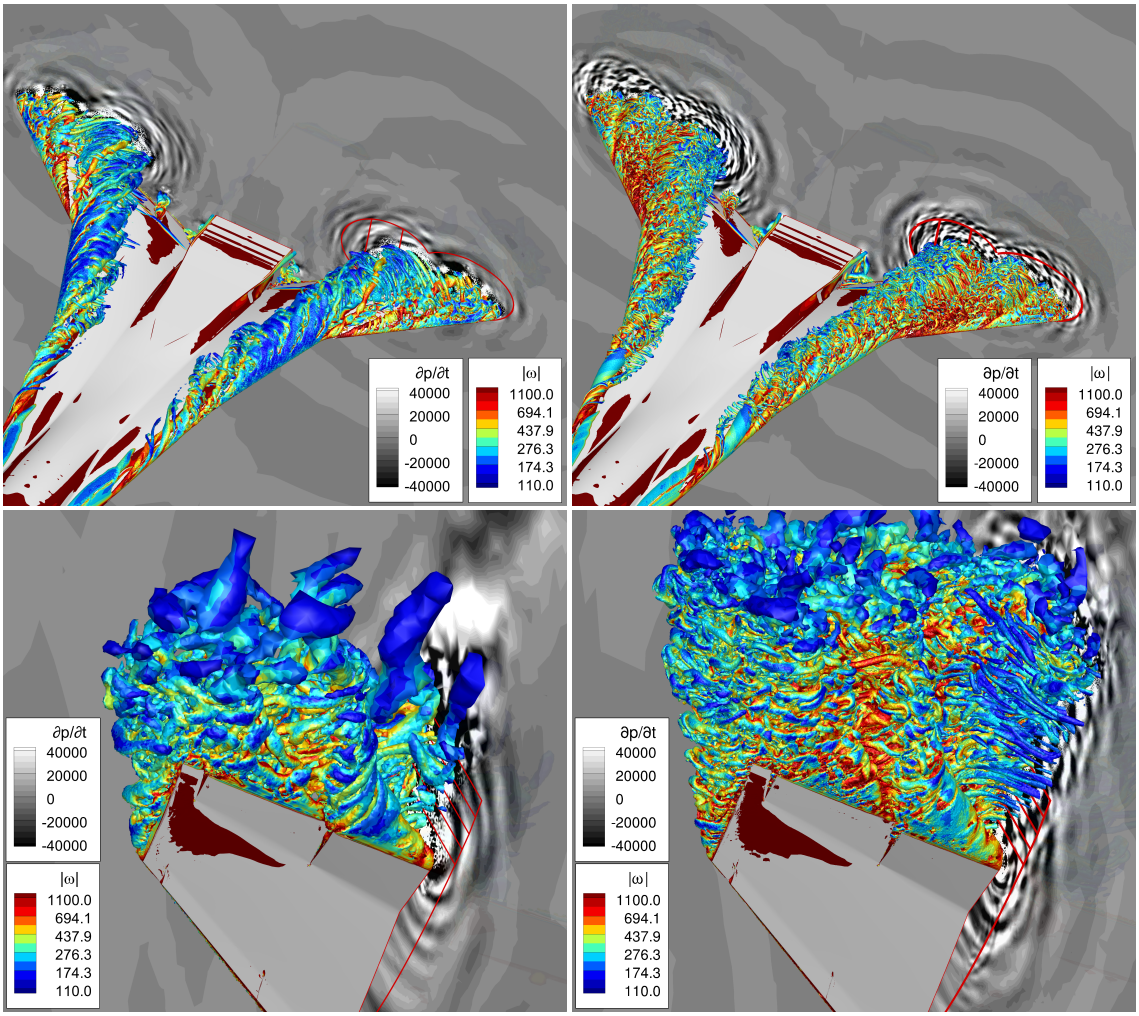
Figure 3. FWH surface and far-field points used to calculate noise radiation pattern

We accumulate the required acoustic data on the two types of meshes formed mainly by quadrilaterals. We use the label Coarse for isotropic meshes with edge length $2h_{\text{vortices}}$ and the label Fine for isotropic meshes with edge length h_{vortices}. We record the data with the sampling frequency $(c_\infty + U_\infty)/(2h_{\text{vortices}})$ on the Coarse FWH meshes and with the sampling frequency $(c_\infty + U_\infty)/h_{\text{vortices}}$ on the Fine FWH meshes.

The described methodology of far-field noise calculation has been tested for the SSBJ wing by comparing acoustic spectra obtained by direct DDES simulations and calculations based on the FWH method at some near-field points [7].

1.4. Numerical Results

In this section, we summarize the main results presented in [7] and supplement them with the new postprocessing data. For detailed information about DDES performance and flow aerodynamics of the SSBJ airframe in landing mode, see [7].



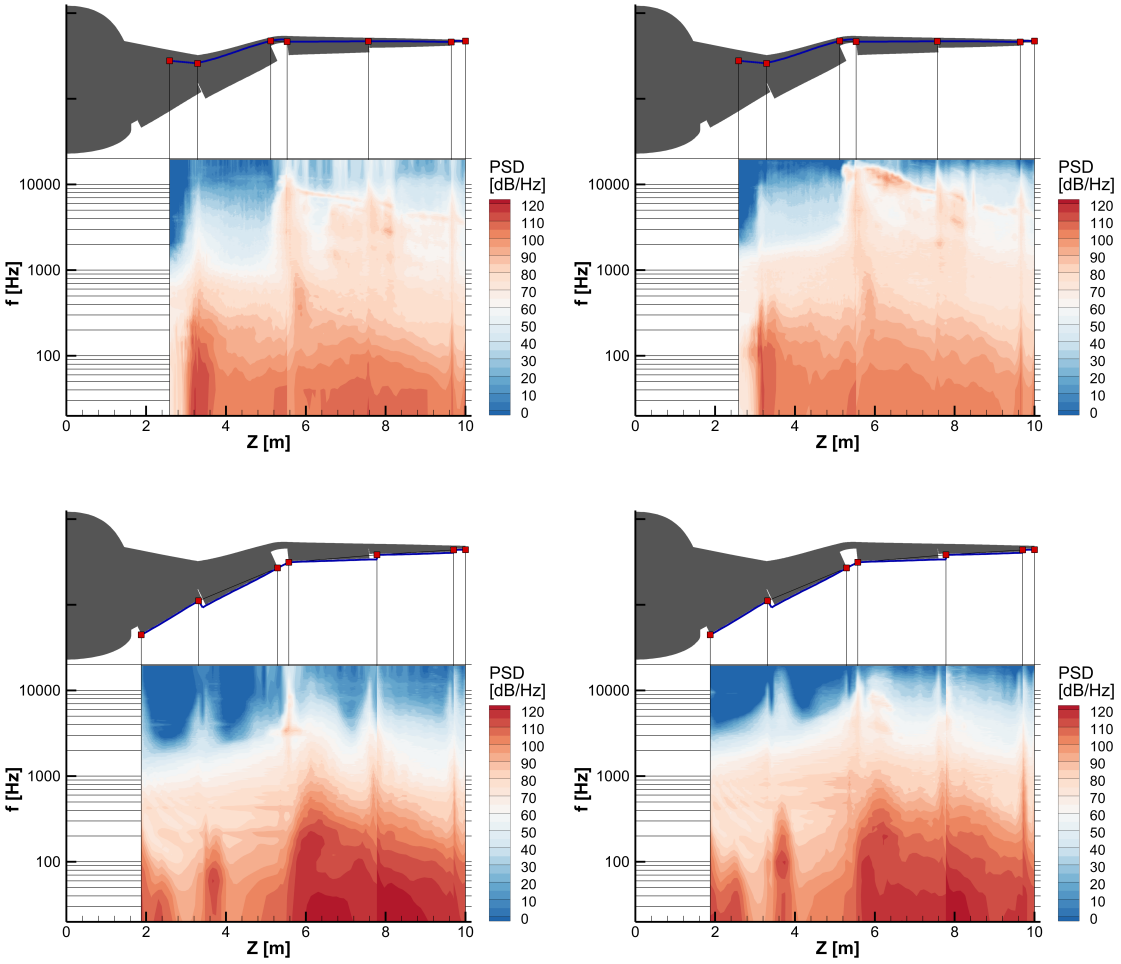
(a) DDES simulation on Level A mesh (b) DDES simulation on Level B mesh

Figure 4. Instantaneous flow fields obtained by DDES simulations (time derivative of pressure and Q-criterion isosurfaces corresponding to the value 5000 $1/s^2$ colored by vorticity magnitude). The red curve denotes location of the FWH surface

The instantaneous flow fields obtained by DDES simulations after reaching the steady state are shown in Fig. 4. For visualization purposes, we duplicate and reflect the resulting flow fields relatively to the plane of symmetry $y = 0$. We see that the turbulent vortical flow over the wing is a source of acoustic pulsations. In the vicinity of the fuselage, the flow is almost stationary and does not contain any significant acoustic sources. As expected, the mesh refinement allows DDES method to reproduce smaller turbulent structures, which leads to the emergence of higher-frequency harmonics in the simulated noise. Outside the zone of increased mesh resolution over the wing acoustic pulsations rapidly dissipate due to increasing size of mesh edges.

During DDES simulations, we record pressure values with the sampling frequency $(c_\infty + U_\infty)/(h_{\text{vortices}}/4)$ at the points of the discretized curves located along the leading and trailing edges of the SSBJ wing. These curves are placed approximately 4 cm below the wing edge, the discretization step is approximately 10 cm. The corresponding power spectral density (PSD) of acoustic pressure pulsations is shown in Fig. 5. The presented spectra are broadband, and, at most points, the noise level decreases with increasing frequency. The similar features of

SSBJ wing noise are obtained in [15, 22]. Near the leading edge, we see the two main PSD peaks: the first is located near the point of sweep angle change at $z = 3.3$ m, the second is located at $z = 5.86$ m in a small distance from the fuselage-side tip of the first droop nose ($z = 5.53$ m). After comparing these results with the flow fields presented in Fig. 4, we can conclude that such PSD peaks are connected with the loss of vortex stability near the wing edge. This observation is confirmed by PSD plots for the trailing edge where the peak at $z = 6.13$ m is associated with the vortex over the third elevon that originates at the elevon fuselage-side tip ($z = 5.57$ m), the peaks at $z = 1.9$ m and $z = 2.5$ m are caused by the tip vortex and wing-root vortex over the first elevon, the peak at $z = 3.68$ m is caused by the tip vortex over the second elevon, the peak at $z = 7.75$ m is associated with the interaction of the stable vortex over the third elevon with the tip of the forth elevon supplemented by tip vortex effects, and the peak at $z = 9.7$ m is associated with the complex vortex flow in a vicinity of the slit between the forth elevon and the wing.



(a) DDES simulation on Level A mesh

(b) DDES simulation on Level B mesh

Figure 5. Power spectral density of acoustic pressure pulsations near the leading and trailing edges

Sound pressure level (SPL) for selected one-third octave bands at the centers of FWH mesh elements is shown in Fig. 6. The main intention of building such plots as well as the plots

presented in Fig. 5 is to demonstrate the possible types of acoustic data that can be directly extracted from DDES solution without involving more complex postprocessing methods such as beamforming or dynamic mode decomposition (DMD). Because valid acoustic spectrum can not be calculated for signal recorded at non-linear flow region, in Fig. 6 one should ignore the presented data at the parts of the FWH surface through which non-linear structures pass. Such parts are located at the side closest to the fuselage and near the end caps (see the corresponding irregular 105–110 dB SPL peaks at 250 Hz band obtained in the DDES simulation on the Level B mesh). The footprint of the acoustic sources near the point of sweep angle change is clearly visible at 250 Hz and 500 Hz bands. The approximate position of the noise sources associated with the vortex originating at the fuselage-side tip of the first droop nose can be seen at all demonstrated frequency bands. The footprints of the acoustic sources associated with the stable vortices over the elevons are localized along the trailing edge.

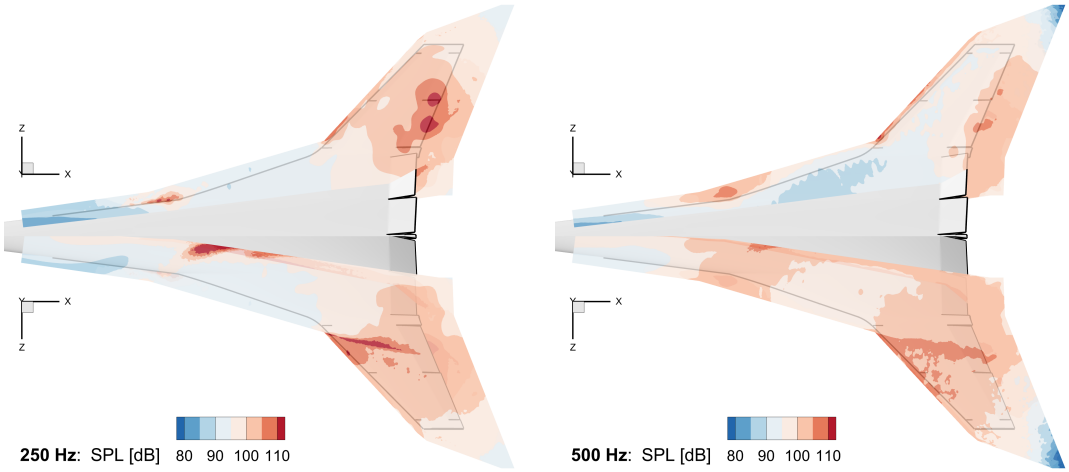
We should note that the noise level at FWH surface is highly dependent on its position and shape. Since we construct the FWH surface according to the approximate shapes of the stable vortices formed over the wing, this surface follows the shape of leading edge, but not the shape of trailing edge (see Fig. 3). Hence, the results presented in Fig. 5 and Fig. 6 are in good agreement for the leading edge, while for the trailing edge the results given in Fig. 5 are more reliable and correct.

Noise radiation pattern for selected one-third octave bands at the far-field points (Fig. 3) is demonstrated in Fig. 7. As expected, resulting radiation pattern depends on meshes used for DDES simulation and FWH calculation. We see that the replacement of Coarse FWH mesh with Fine FWH mesh leads to 2–3 dB SPL difference at 250–500 Hz bands for the data recorded on the Level A mesh and to 1–1.5 dB SPL difference at 250–2000 Hz bands for the data recorded on the Level B mesh. The SPL difference between the results obtained for the Level A and Level B meshes using Fine FWH mesh is 2–3.5 dB at 250 Hz band and 1.5–4 dB at 500 Hz band. At 250 Hz band noise radiation pattern for all DDES and FWH meshes has the dominating dipole-type component, which leads to local SPL maximums approximately at $-100°$ and $80°$ and to local SPL minimums approximately at $-10°$ and $170°$. For the Level A mesh at 0.5–1 kHz bands and for the Level B mesh at 1–2 kHz bands, the obtained radiation patterns have the dominating monopole-type component, which provides SPL maximum between $100°$ and $160°$ and SPL minimum between $-140°$ and $-50°$. For the Level B mesh, the noise radiation pattern at 500 Hz band probably has a mixed structure formed by both monopole-type and dipole-type components.

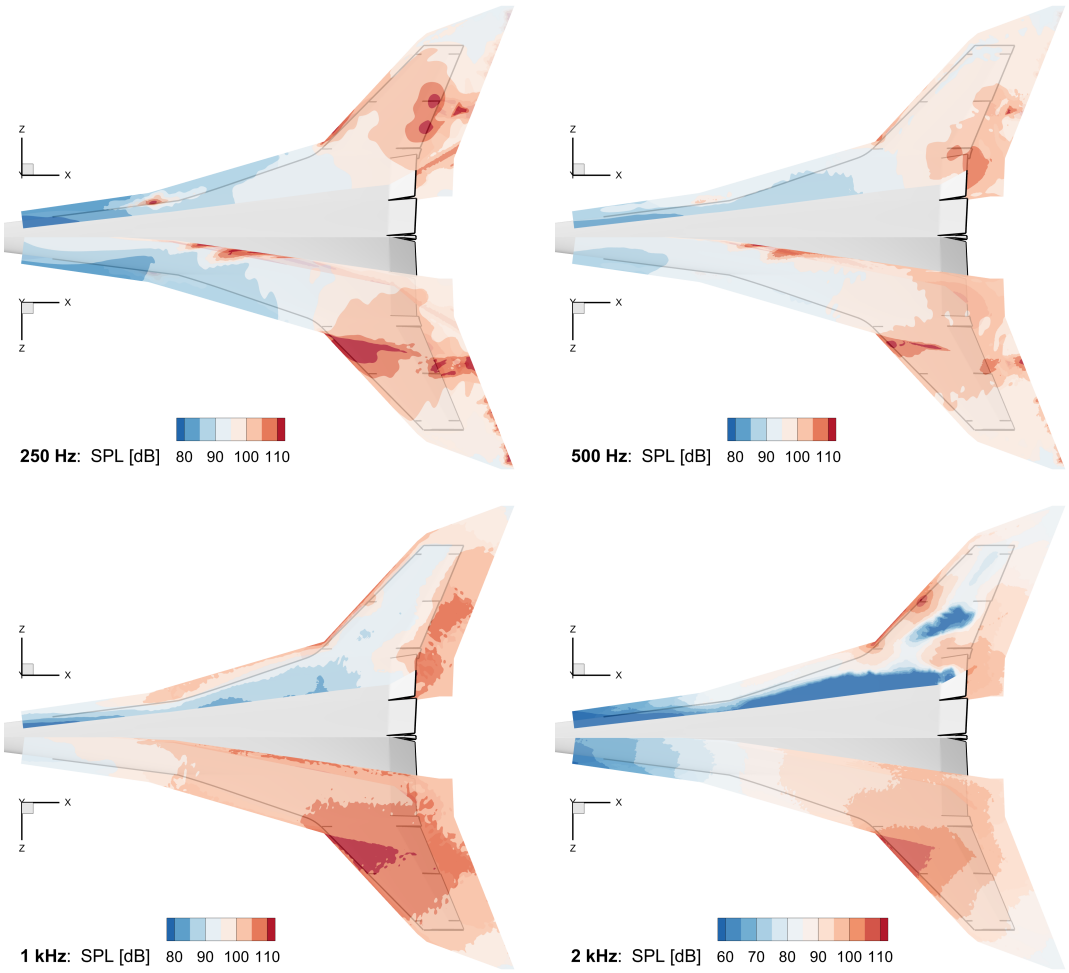
2. Inverse Problem – Numerical Beamforming

2.1. Problem Setup and Geometry

The aim of numerical beamforming is to identify acoustic sources in a certain regions (on a surface or on a line) using space-time data obtained by numerical simulation. For the considered method, the acoustic pressure pulsations recorded at some points of the computational domain are used as input data. Such points can be considered as virtual microphones, so we will refer to them as microphones in this paper. Microphones can be placed on surfaces, lines or in any other arbitrary way, in accordance with the selected source domain. The number of microphones can be chosen arbitrarily. As microphone surface used to identify acoustic sources on the wing of the SSBJ model, one can choose either the FWH surface (Fig. 3), or, in general, an arbitrary set of



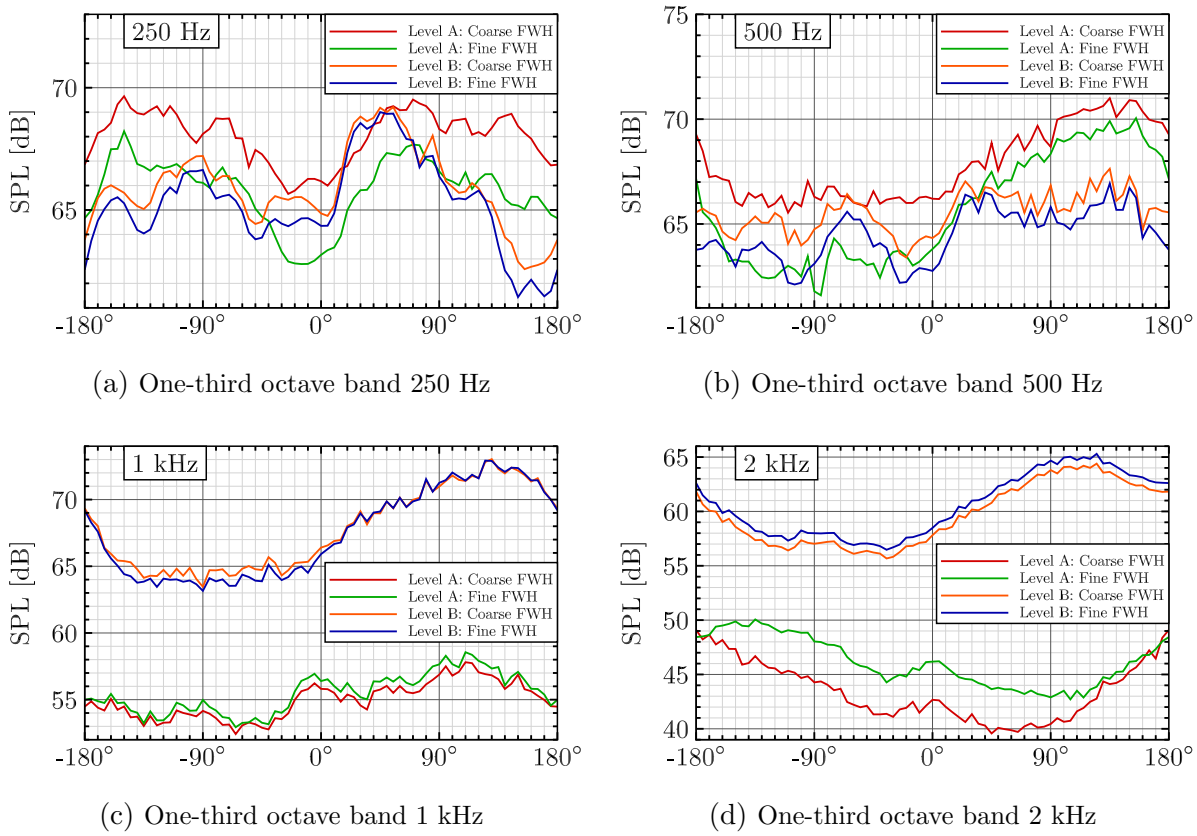
(a) DDES simulation on Level A mesh (Fine FWH mesh)



(b) DDES simulation on Level B mesh (Coarse FWH mesh)

Figure 6. Sound pressure level at FWH surface for one-third octave bands with central frequencies 250 Hz, 500 Hz, 1 kHz and 2 kHz

points in the far field, at which the signal can be calculated using integral formulas according to the Ffowcs Williams–Hawkings acoustic analogy [9].



(a) One-third octave band 250 Hz

(b) One-third octave band 500 Hz

(c) One-third octave band 1 kHz

(d) One-third octave band 2 kHz

Figure 7. Noise radiation pattern ($R = 150$ m) for one-third octave bands with central frequencies 250 Hz, 500 Hz, 1 kHz and 2 kHz

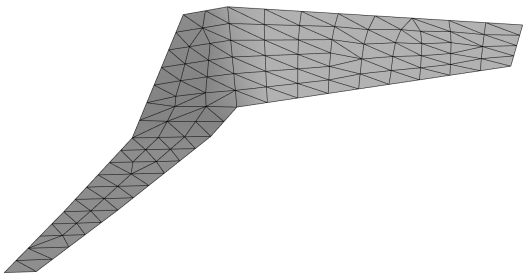
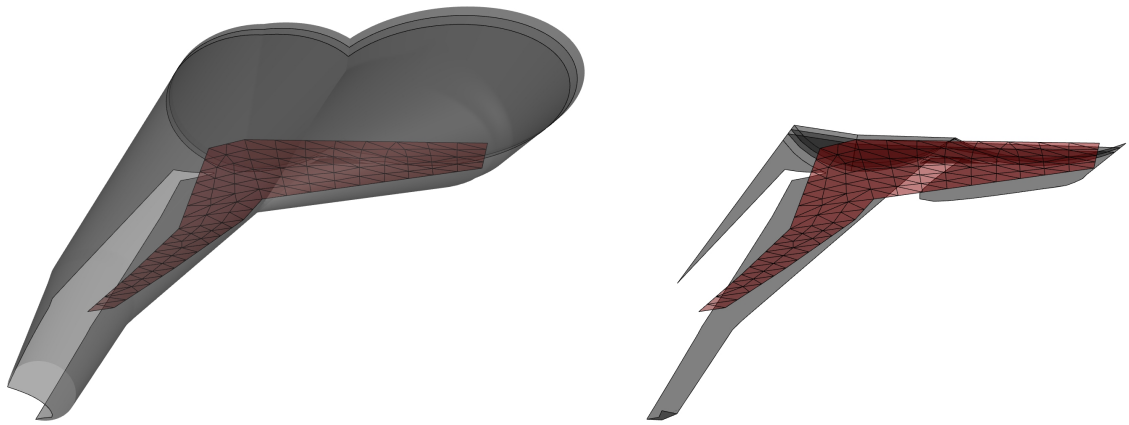


Figure 8. Example of a source grid for identifying an acoustic source on the SSBJ wing

The source domain is defined as a surface, in some sense approximating the wing of the SSBJ model (Fig. 8). As in [21], we consider two microphone configurations: one consists of points located on entire FWH surface (Fig. 9a) and another consists of points located on its lower part only (Fig. 9b). Microphones of the latter configuration record acoustics radiated toward the ground. This configuration is of practical interest and is sufficient for reliable reconstruction of the monopole source function. Nevertheless, there are configurations or types of acoustic sources which require the use of the entire FWH surface as microphone domain in order to ensure the solution uniqueness (see [21]). Thus, the aim of numerical beamforming for the SSBJ wing is to

localize and determine the intensity of the acoustic sources on the surface that approximates this wing using the acoustic signals recorded by microphones located in the spatial configurations described above.



(a) Entire FWH configuration (b) Lower FWH configuration

Figure 9. Configurations of the source and microphone surfaces for the SSBJ wing

2.2. Numerical Method

The problem of acoustic beamforming for identifying a source on a given surface $S \subset \mathbb{R}^3$ defined by the delta function δ_S is formulated as the inverse problem of finding the right-hand side of the Helmholtz equation for the acoustic pressure $P(x)$ in a medium moving with a velocity U:

$$-\frac{1}{c^2}(i\omega + U \cdot \nabla)^2 P + \Delta P = -Q\delta_S - \operatorname{div}(\boldsymbol{F}\delta_S), \tag{1}$$

where $Q(x) \in \mathbb{C}$, $x = \{x_1, x_2, x_3\} \in S$ is the unknown intensity of a monopole-type source, $F_l(x) \in \mathbb{C}$, $l = 1, 2, 3$ is the unknown intensity of a dipole moment in the direction of the Cartesian coordinate x_l, $\mathbf{F} \equiv \{F_1, F_2, F_3\}$, ω is the circular frequency, $c > U$ is the sound speed. To solve this problem, the values of acoustic pressure $P(x)$, $x \in D$, where D is some given microphone surface, are used, which are called *beamforming data* and are accumulated within the CFD/CAA simulation. A possible configuration of the source and microphone surfaces is shown in Fig. 10.

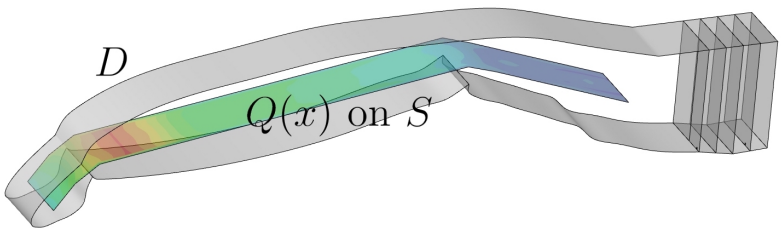


Figure 10. Configuration of the source (S) and microphone (D) surfaces

Monopole-type sources. The dependence of acoustic pressure in Eq. (1) on a monopole-type source given by the function Q is calculated using the surface potential

$$P(x) = \int_S Q(y)\, \mathrm{G}_{\mathbf{M}}(x-y)\, d\sigma,\ x \in D,$$
$$\text{where } \mathrm{G}_{\mathbf{M}}(x) = \frac{1}{4\pi}\frac{e^{-ik'\Delta t_e}}{x'},$$
$$x' = \sqrt{(\mathbf{M}\cdot\boldsymbol{r})^2 + \beta^2|\boldsymbol{r}|^2},\quad \beta^2 = 1 - |\mathbf{M}|^2,\ k' = \frac{\omega}{c\beta^2},$$
$$\Delta t_e = (-(\mathbf{M}\cdot\boldsymbol{r}) + x').$$

or in operator form:

$$P = \mathcal{T}Q.$$

The grids consisting of N and M nodes, respectively, are introduced on the surfaces of the source and microphones to construct a numerical method for solving the beamforming problem, $M \gg N$. The intensity function of a distributed monopole-type source is represented by its values s_n in the grid nodes in the form of a vector $\mathbf{s} = (s_1, s_2, ..., s_N)^\intercal$ using the piecewise linear basis functions $\{\psi_n(y)\}_{n=1}^N$:

$$Q(y) = \sum_{n=1}^{N} s_n \psi_n(y),$$

while the pressure P at the microphone points is written in the form of a vector $\mathbf{d} = (d_1, d_2, ..., d_M)^\intercal$. Thus, the radiation from the source to the microphones is transferred according to

$$\mathbf{d} = \mathcal{T}_a \mathbf{s} \tag{2}$$

where $\mathcal{T}_a$ is a discrete approximation of the operator $\mathcal{T}$, which is obtained according to the finite element approach:

$$(\mathcal{T}_a)_{mn} = \int_S \psi_n(y)\, G_{\mathbf{M}}(x_m - y)\, d\sigma, \tag{3}$$

using Gauss formulas to calculate the integrals.

Given the *beamforming data* $\widetilde{\mathbf{d}}$ at the microphone grid points the condition of minimizing the residual norm $\widetilde{\mathbf{d}} - \mathbf{d}$ is set with the possible addition of Tikhonov regularization, $\gamma \geq 0$, to find the source intensity vector $\mathbf{s}$:

$$\|\widetilde{\mathbf{d}} - \mathcal{T}_a\mathbf{s}\|_2^2 + \gamma\|\mathbf{s}\|_2^2 \to \min_{\mathbf{s}}.$$

The solution to such a problem has the form:

$$\mathbf{s} = (\mathcal{T}_a^*\mathcal{T}_a + \gamma I)^{-1}\mathcal{T}_a^*\widetilde{\mathbf{d}}.$$

Analysis [16] shows that the numerical beamforming algorithm for monopole-type sources has high accuracy and good stability without using regularization if the following conditions are met:

$$\begin{aligned} &M \gg N \\ &\mathrm{dist}_S = a_S\lambda, && 0.8 < a_S < 1.5 \\ &\mathrm{dist}_{SD} = a_{SD}\lambda, && 3 < a_{SD} < 30, \end{aligned} \tag{4}$$

where dist_S is the source function grid step, dist_{SD} is the distance between surfaces S and D, $\lambda = 2\pi/k'$.

Dipole-type sources. For discretization of the problem in case of a dipole-type source, the same approach is used as for monopole-type sources. In this case, the radiation transfer matrix $\mathcal{T}_a$ is formed from three component matrices $\mathcal{T}_a^1,\ \mathcal{T}_a^2,\ \mathcal{T}_a^3$:

$$(\mathcal{T}_a^l)_{mn} = \int_S \psi_n(y)\, \mathrm{G}_{\mathbf{M}}^{(l)}(x_m - y)\, d\sigma, \quad l = 1, 2, 3, \tag{5}$$

with the fundamental solutions $\mathrm{G}_{\mathbf{M}}^{(l)}(x) = \frac{\partial}{\partial x_l}\mathrm{G}_{\mathbf{M}}(x)$.

However, for dipole-type sources in the right-hand side of Eq. (1) the non-uniqueness of beamforming problem solution arises due to the presence of the surface divergence operator Div that vanishes any function of the form Curl($\boldsymbol{a}$), where Curl is the surface curl operator. In [17] some restrictions on the dipole-type source are considered in order to address the non-uniqueness. For example, in case of a priori information that some dipole component is zero, it is excluded. Another useful restriction arises when the direction of the dipole moment, the unit vector $e(y)$, is known in advance at each grid node of the source,

$$e(y_n) = \{\cos(\theta_n)\sin(\varphi_n),\ \cos(\theta_n)\cos(\varphi_n),\ \sin(\theta_n)\ \}, \quad n = 1..N, \tag{6}$$

(for example, $e(y)$ is the normal to the surface). In this case the resulting dipole radiation transfer matrix has the form:

$$\mathcal{T}_a^e = \mathcal{T}_a^1 \mathrm{diag}(\cos\boldsymbol{\theta}.\sin\boldsymbol{\varphi}) + \mathcal{T}_a^2 \mathrm{diag}(\cos\boldsymbol{\theta}.\cos\boldsymbol{\varphi}) + \mathcal{T}_a^3 \mathrm{diag}(\sin\boldsymbol{\theta}).$$

Here $\cos\boldsymbol{\theta} = (\cos\theta_1, \cos\theta_2, \ldots, \cos\theta_N)^\intercal$, $\sin\boldsymbol{\theta} = (\sin\theta_1, \sin\theta_2, \ldots, \sin\theta_N)^\intercal$, $\cos\boldsymbol{\varphi} = (\cos\varphi_1, \cos\varphi_2, \ldots, \cos\varphi_N)^\intercal$, $\sin\boldsymbol{\varphi} = (\sin\varphi_1, \sin\varphi_2, \ldots, \sin\varphi_N)^\intercal$, and the lower dot symbol denotes component-wise multiplication of vectors: $\mathbf{a}.\mathbf{b} \equiv (a_n b_n)_{n=1}^N$.

Note that from Eq. (1) follows the equivalence of considering the problems of finding both the monopole-type source $Q\delta_S$ and the tangential component $\mathrm{Div}(\boldsymbol{F})\,\delta_S$ of the dipole-type source. Therefore, the question of the distribution of the found total intensity between these two types of sources can be solved only using a priori information.

2.3. Parallel Algorithms

The main resource-intensive part of the numerical beamforming algorithm is the calculation of the radiation transfer matrix. Even for relatively small number of nodes of source grid, which is limited on a source surface with fixed area by the condition Eq. (4) and which cannot be increased sufficiently to study the convergence of the method, it is necessary to calculate integral convolutions of piecewise linear basis functions with fundamental solutions of the Helmholtz equation for a moving medium (see Eq. (3), (5)). These integrals are calculated using Gauss formulas in double-precision complex numbers.

In order to speed up the numerical beamforming algorithm, the code implements MPI+OpenMP parallelization of the calculation of the radiation transfer matrix. Parallelization is performed along the matrix rows, since the number of microphones greatly exceeds the number of source grid nodes (each row corresponds to a single microphone, each column corresponds to a single node of source grid). Each process or thread is processing its own block of rows followed by data exchange in case of MPI parallelization. The dependence of speedup on the number of threads for OpenMP mode is showed in Fig. 11.

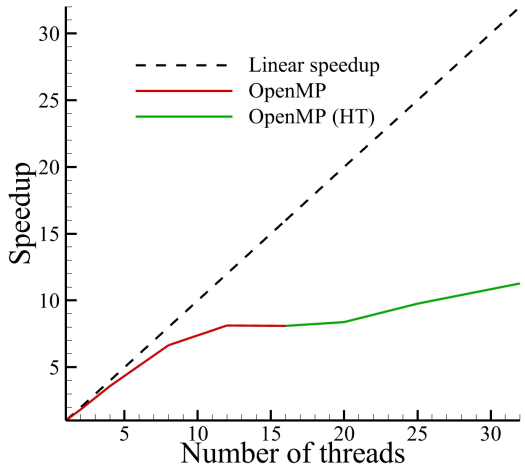


Figure 11. Dependence of speedup on the number of threads for OpenMP mode. The green line shows the speedup when using hyper-threading (HT)

Matrix calculation is also implemented for GPU using OpenCL framework. In GPU version of the algorithm, parallelization is implemented not only for rows, but also for columns. Heterogeneous computations show a significant speedup of the algorithm in comparison with the CPU version. The achieved speedup is mainly helpful for mass beamforming when a large number of frequency bands should be processed.

Even though we currently consider beamforming problems characterized by small dimensions of the radiation transfer matrix, the potential profit of parallel computations is rather high, especially for the industrial problems. In case of a fixed wavelength-matched grid step while the area of the surface can be of any size, the method has no upper limit on the number of microphones or source grid nodes.

2.4. Data Preprocessing

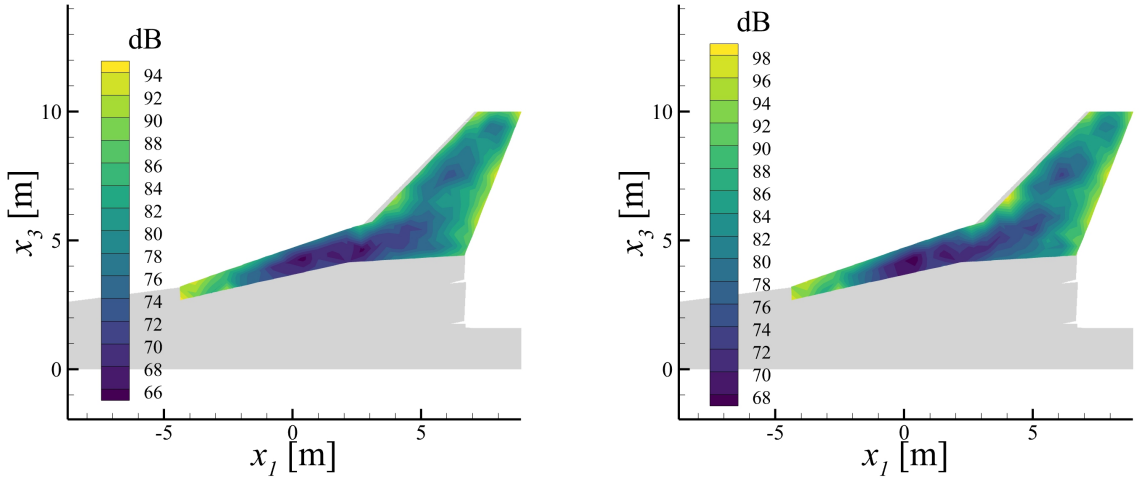
Beamforming based on CFD data processes pressure signals recorded at large number of points located in the region of linear perturbations. The most convenient approach is to use the points on the FWH surface. In many aeroacoustic simulations, the FWH surface is already used to accumulate near-field data required to calculate far-field noise. Hence, it is rather simple to add new variable, namely the pressure, in the list of variables to be recorded. On the other hand, different spatial microphone configurations may be required to refine resolution of acoustic sources obtained by beamforming (see [13]). Using the FWH method, one can easily calculate signals at the points of required microphone grids. In the present paper, microphone grid is located on the FWH surface only.

The signals extracted from CFD simulation usually contain spurious noise. To improve stability of the results obtained by the numerical beamforming method, it is useful to perform signal preprocessing with statistical averaging. Let us describe this algorithm in more detail. In the time domain, the starting point of the signal is randomly selected, and each starting point defines its own *scenario*. In this paper, we use 100 scenarios for signal preprocessing. For each scenario the signal is cut out and divided into several parts of equal length. These parts are averaged, the Fourier transformation is applied to the resulting signal, and the obtained data

corresponding to selected frequency band are used as input for the beamforming algorithm. This "stacking" approach allows to reduce the impact of spurious noise and leads to more stable results of numerical beamforming for neighboring frequency bands.

3. Numerical Beamforming for SSBJ Wing in Landing Mode

First, we apply the numerical beamforming to identify acoustic monopole-type sources on the wing of the SSBJ prototype using a microphone array located on the bottom of the FWH surface. This configuration of microphone grid is also used in [21] and allows to obtain a stable solution while capturing only the noise radiating towards the ground. Figure 12a presents the intensity map for the distributed monopole-type source obtained by numerical beamforming for the one-third octave band with central frequency $f_c = 500$ Hz. For each isolated point of distributed acoustic source, we calculate SPL at distance $\frac{1}{\sqrt{4\pi}}$ m according to [18]. Despite the high relative residual $\delta = 0.89$, the obtained solution for the monopole-type source is consistent with the results of the near-field analysis carried out in Section 1.4. As we see, the main noise sources are located near the wing edges, and the most intense ones are detected at the trailing edge.



(a) Monopole-type source

(b) Dipole-type source oriented along normals to the surface

Figure 12. Numerical beamforming for the SSBJ wing and the one-third octave band with central frequency $f_c = 500$ Hz

For dipole-type sources oriented along normals to the surface, the numerical beamforming method demonstrates similar results (Fig. 12b). The microphone grid for detecting dipole-type sources coincides with the the microphone grid used for the monopole-type sources. The norm of relative residual is $\delta = 0.92$.

To assess the accuracy and reliability of the numerical beamforming, we compare SPL for the signals obtained at some mid-field points by the FWH method (the direct method) with SPL for the signals obtained at these points by the linear transfer of acoustic radiation from distributed sources reconstructed by the numerical beamforming (the inverse method). We use the following four points: $P_1 = (6, -10, 8)$, $P_2 = (7, -10, 8)$, $P_3 = (6, -10, 6)$, $P_4 = (7, -14, 8)$ (Fig. 13).

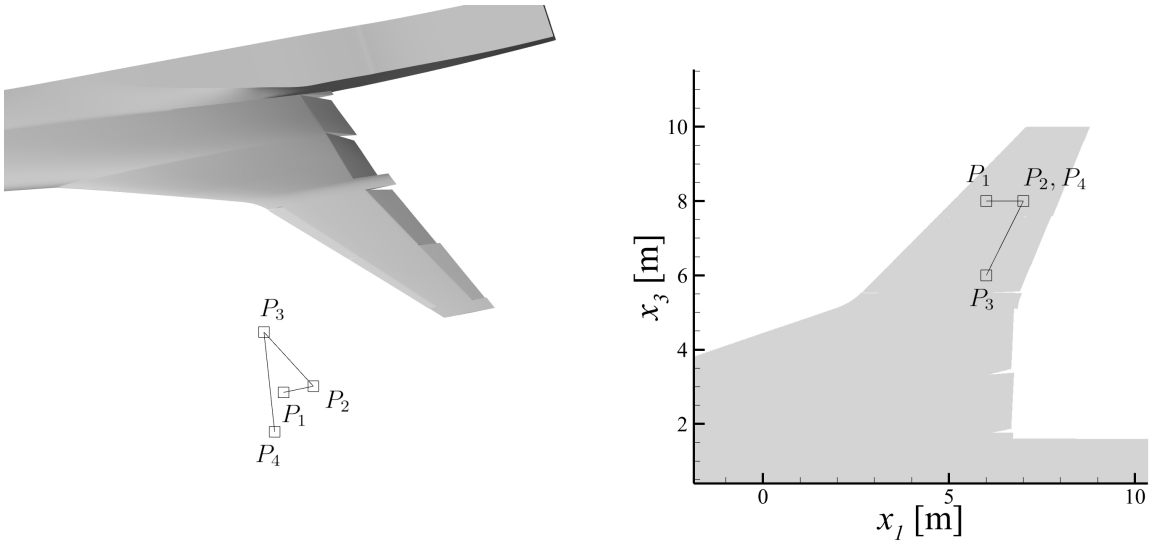
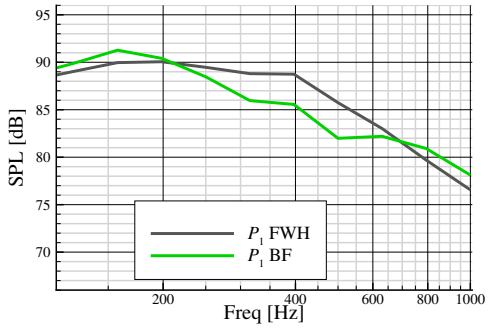


Figure 13. Location of points used to compare spectra obtained by FWH method and radiation transfer from the noise sources detected by numerical beamforming

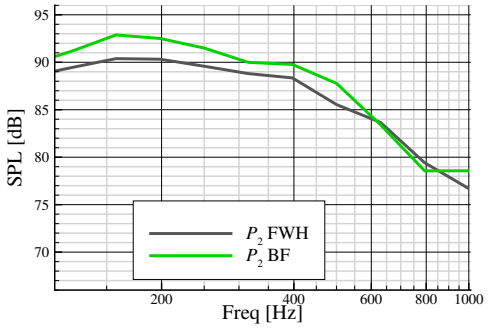
To obtain the characteristics of pressure pulsations generated by detected acoustic sources, we perform mass beamforming by scanning over 1/12-octave bands for the frequency range 100 Hz $< f <$ 1000 Hz. For each 1/12-octave band, we use source grid of the required resolution that meets the *correctness* conditions (Eq. (4)) depending on the wavelength of the signal, and calculate the average source function. Then we compute signals at the points P_1, P_2, P_3, P_4 using the obtained averaged source functions, according to Eq. (2). Finally, we calculate 1/12-octave band SPL spectra and convert them to 1/3-octave band SPL spectra by averaging and integration procedures.

In Fig. 14 and Fig. 15, the comparison of the SPL spectra for one-third octave bands at points P_1, P_2, P_3 and P_4 obtained by the FWH method and by radiation transfer from the noise sources detected by the numerical beamforming are presented. The green line corresponds to the results obtained by beamforming, and the dark gray one corresponds to direct calculations using the FWH method. The spectra obtained for the monopole-type source (Fig. 14) are in better agreement with the corresponding spectra based on FWH calculations (particularly, for point P_4) than the spectra obtained for the dipole-type source oriented along normals to the surface (Fig. 15). It is possible that the assumed source model in the form of the dipole, oriented along normals to the wing surface, can describe the real source localized on the wing worse due to the incorrectness of the assumption itself. We should note that the results for the monopole-type source can be interpreted as results for a dipole-type source with moment oriented tangentially to the surface (see Section 2.2). The dipole sources of this type are generated, for example, by a turbulent boundary layer.

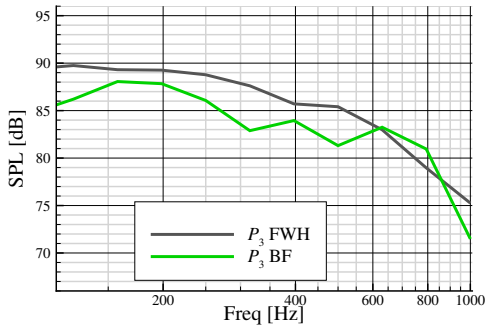
Since the FWH and numerical beamforming methods are based on significantly different mathematical approaches, we do not expect the results obtained by these methods to coincide with each other with high accuracy. Despite the fact that the initial data for both methods are results of CFD computations accumulated on the same surface, when obtaining the values of acoustic pressure at the control points, the FWH method starts directly from these data, while numerical beamforming by its definition has its own model simplifications and assumptions. If we accept the above remark, we can speak about a very satisfactory reproduction of the results of the direct problem by beamforming.



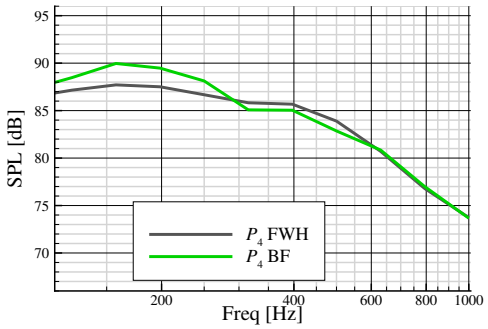
(a) Point P_1



(b) Point P_2

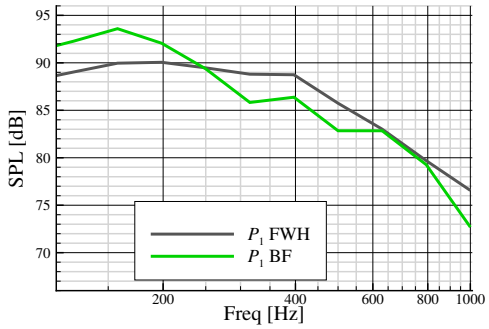


(c) Point P_3

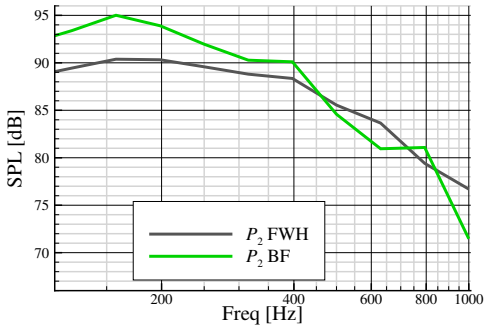


(d) Point P_4

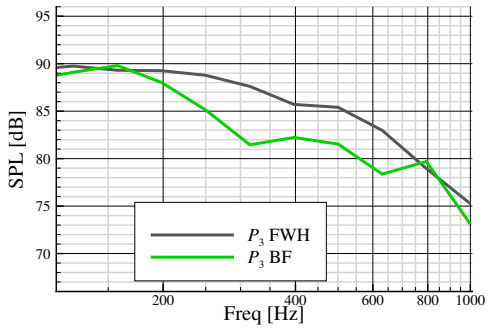
Figure 14. SPL spectra at mid-field points obtained by FWH and numerical beamforming (BF) methods. Monopole-type source



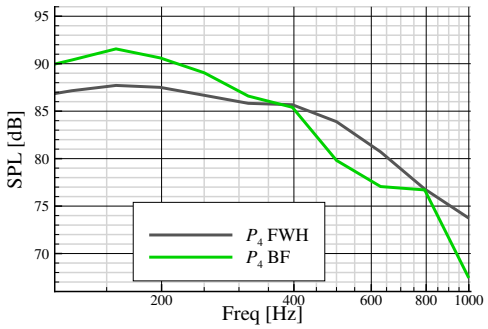
(a) Point P_1



(b) Point P_2



(c) Point P_3



(d) Point P_4

Figure 15. SPL spectra at mid-field points obtained by FWH and numerical beamforming (BF) methods. Dipole-type source with moment oriented along normals to the surface

Conclusions

This paper presents a numerical study of the noise sources generated by the wing of SSBJ prototype in landing mode.

A distinctive feature of this study is a combination of two ways of predicting and analyzing noise sources. The first is based on direct postprocessing of turbulent flow around the SSBJ airframe obtained by resource-inventive scale-resolving CFD/CAA simulation. The second is based on solving the inverse problem of noise source reconstruction using the acoustic data accumulated during CFD/CAA simulation. For the second approach, we apply the numerical beamforming method which assumes the near-field acoustics to be governed by the wave equation with a distributed source term in the right-hand side. It is for the first time that the numerical beamforming is used to analyze acoustic sources of the full-scale aircraft geometry.

An important detail of the study is the verification of the numerical beamforming by comparison SPL spectra for signals at mid-field points obtained by the FWH method and by radiation of acoustic perturbations from the noise sources detected by the numerical beamforming method.

According to the results of numerical beamforming and the SPL distributions obtained on the FWH surface, the main acoustic sources generated by the SSBJ wing in landing mode are located near the leading and tailing edges. PSD analysis of the acoustic pressure pulsations recorded under these edges shows that the loss of vortex stability near the edges of the SSBJ wing leads to the local increases in noise intensity. The analysis of the noise radiation patterns calculated using the FWH method demonstrates that the far-field noise has dominating dipole-type component for the one-third octave bands with central frequencies lower (see [7]) or equal to 250 Hz and dominating monopole-type component for the one-third octave bands with central frequencies equal to 1 kHz or higher. For the rest one-third octave bands, the noise radiation pattern probably has a mixed structure formed by both monopole-type and dipole-type components.

Acknowledgements

The research is carried out using the equipment of the shared research facilities of HPC computing resources at Lomonosov Moscow State University with the additional use of the hybrid supercomputer K60 installed in the Supercomputer Centre of Collective Usage of KIAM RAS.

References

1. Abalakin, I.V., Bakhvalov, P.A., Bobkov, V.G., *et al.*: NOISEtte CFD&CAA Supercomputer Code for Research and Applications. Supercomput. Front. Innov. 11(2), 78–101 (aug 2024). https://doi.org/10.14529/jsfi240206

2. Bakhvalov, P.A., Kozubskaya, T.K., Kornilina, E.D., *et al.*: Technology of predicting acoustic turbulence in the far-field flow. Math. Model. Comput. Simulations 4(3), 363–373 (may 2012). https://doi.org/10.1134/S2070048212030039

3. Bakhvalov, P.A., Surnachev, M.D.: Method of averaged element splittings for diffusion terms discretization in vertex-centered framework. J. Comput. Phys. 450, 110819 (feb 2022). https://doi.org/10.1016/J.JCP.2021.110819

4. Bakhvalov, P., Kozubskaya, T., Rodionov, P.: EBR schemes with curvilinear reconstructions for hybrid meshes. Comput. Fluids 239, 105352 (may 2022). https://doi.org/10.1016/J.COMPFLUID.2022.105352

5. Barad, M.F., Kocheemoolayil, J.G., Kiris, C.C.: Lattice Boltzmann and Navier-Stokes Cartesian CFD Approaches for Airframe Noise Predictions. In: 23rd AIAA Comput. Fluid Dyn. Conf. AIAA AVIATION Forum, American Institute of Aeronautics and Astronautics (jun 2017). https://doi.org/10.2514/6.2017-4404

6. Deck, S., Renard, N.: Towards an enhanced protection of attached boundary layers in hybrid RANS/LES methods. J. Comput. Phys. 400, 108970 (2020). https://doi.org/10.1016/j.jcp.2019.108970

7. Duben, A.P., Kozubskaya, T.K., Rodionov, P.V.: Wing Noise Simulation of Supersonic Business Jet in Landing Configuration. Supercomput. Front. Innov. 11(3), 74–92 (oct 2024). https://doi.org/10.14529/jsfi240305

8. Ferris, R., Sacks, M., Cerizza, D., *et al.*: Aeroacoustic Computations of a Generic Low Boom Concept in Landing Configuration: Part 1 – Aerodynamic Simulations. AIAA Aviat. Aeronaut. Forum Expo. AIAA Aviat. Forum 2021 (2021). https://doi.org/10.2514/6.2021-2195

9. Ffowcs Williams, J.E., Hawkings, D.L.: Sound generation by turbulence and surfaces in arbitrary motion. Philos. Trans. R. Soc. London. Ser. A, Math. Phys. Sci. 264(1151), 321–342 (may 1969). https://doi.org/10.1098/rsta.1969.0031

10. Gorobets, A.V., Duben, A.P., Kozubskaya, T.K., Rodionov, P.V.: Approaches to the Numerical Simulation of the Acoustic Field Generated by a Multi-Element Aircraft Wing in High-Lift Configuration. Math. Model. Comput. Simulations 15(1), 92–108 (feb 2023). https://doi.org/10.1134/S2070048223010088

11. Gorobets, A., Bakhvalov, P.: Heterogeneous CPU+GPU parallelization for high-accuracy scale-resolving simulations of compressible turbulent flows on hybrid supercomputers. Comput. Phys. Commun. 271, 108231 (feb 2022). https://doi.org/10.1016/J.CPC.2021.108231

12. Guseva, E.K., Garbaruk, A.V., Strelets, M.K.: An automatic hybrid numerical scheme for global RANS-LES approaches. J. Phys. Conf. Ser. 929(1), 012099 (nov 2017). https://doi.org/10.1088/1742-6596/929/1/012099

13. Karakulev, A., Kozubskaya, T., Plaksin, G., Sofronov, I.: Ffowcs Williams–Hawkings analogy for near-field acoustic sources analysis. International Journal of Aeroacoustics 21(5-7), 457–475 (2022). https://doi.org/10.1177/1475472X221107367

14. Khorrami, M.R., Fares, E.: Simulation-based airframe noise prediction of a full-scale, full aircraft. 22nd AIAA/CEAS Aeroacoustics Conf. 2016 (2016). https://doi.org/10.2514/6.2016-2706

15. Khorrami, M.R., Shea, P.R., Winski, C.S., *et al.*: Aeroacoustic Computations of a Generic Low Boom Concept in Landing Configuration: Part 3 – Aerodynamic Validation and Noise Source Identification. AIAA Aviat. Aeronaut. Forum Expo. AIAA Aviat. Forum 2021 (2021). https://doi.org/10.2514/6.2021-2197

16. Kozubskaya, T., Plaksin, G., Sofronov, I.: Statement of the beamforming problem and a method of its solution for the localization of an acoustic source based on computational experiment data. Comput. Math. Math. Phys. 61, 1864–1885 (11 2021). https://doi.org/10.1134/S0965542521110129

17. Kozubskaya, T., Plaksin, G., Sofronov, I.: On numerical beamforming for correlated dipole-type sources. Computational Mathematics and Mathematical Physics 63, 2162–2175 (12 2023). https://doi.org/10.1134/S0965542523110131

18. Liu, Y., Quayle, A., Dowling, A., Sijtsma, P.: Beamforming correction for dipole measurement using two-dimensional microphone arrays. The Journal of the Acoustical Society of America 124, 182–191 (07 2008). https://doi.org/10.1121/1.2931950

19. Mockett, C., Fuchs, M., Garbaruk, A., *et al.*: Two Non-zonal Approaches to Accelerate RANS to LES Transition of Free Shear Layers in DES. In: Prog. Hybrid RANS-LES Model., vol. 130, pp. 187–201. Springer Verlag (2015). https://doi.org/10.1007/978-3-319-15141-0_15

20. Nicoud, F., Toda, H.B., Cabrit, O., *et al.*: Using singular values to build a subgrid-scale model for large eddy simulations. Phys. Fluids 23(8), 085106 (aug 2011). https://doi.org/10.1063/1.3623274

21. Plaksin, G.M., Kozubskaya, T.K., Sofronov, I.L.: On numerical beamforming for acoustic source identification basing on supercomputer simulation data. Dokl. Math. 110, 435–441 (2024). https://doi.org/10.1134/S1064562424601550

22. Ribeiro, A.F., Ferris, R., Khorrami, M.R.: Aeroacoustic Computations of a Generic Low Boom Concept in Landing Configuration: Part 2 – Airframe Noise Simulations. AIAA Aviat. Aeronaut. Forum Expo. AIAA Aviat. Forum 2021 (2021). https://doi.org/10.2514/6.2021-2196

23. Shur, M.L., Spalart, P.R., Strelets, M.K.: Noise Prediction for Increasingly Complex Jets. Part I: Methods and Tests. Int. J. Aeroacoustics 4(3), 213–245 (jul 2005). https://doi.org/10.1260/1475472054771376

24. Shur, M.L., Spalart, P.R., Strelets, M.K.: Noise Prediction for Increasingly Complex Jets. Part II: Applications. Int. J. Aeroacoustics 4(3), 247–266 (jul 2005). https://doi.org/10.1260/72054771385

25. Shur, M.L., Spalart, P.R., Strelets, M.K., Travin, A.K.: An Enhanced Version of DES with Rapid Transition from RANS to LES in Separated Flows. Flow, Turbul. Combust. 95(4), 709–737 (jun 2015). https://doi.org/10.1007/S10494-015-9618-0

26. Sijtsma, P.: Acoustic beamforming for the ranking of aircraft noise. Tech. rep., National Aerospace Laboratory NLR (2012), https://api.semanticscholar.org/CorpusID:60149502

27. Spalart, P.R., Allmaras, S.R.: A one-equation turbulence model for aerodynamic flows. In: AIAA Pap. 92-0439 (1992). https://doi.org/10.2514/6.1992-439

28. Voevodin, V.V., Antonov, A.S., Nikitenko, D.A., *et al.*: Supercomputer Lomonosov-2: Large Scale, Deep Monitoring and Fine Analytics for the User Community. Supercomput. Front. Innov. 6(2), 4–11 (jun 2019). https://doi.org/10.14529/jsfi190201

29. van der Vorst, H.A.: Bi-CGSTAB: A Fast and Smoothly Converging Variant of Bi-CG for the Solution of Nonsymmetric Linear Systems. SIAM J. Sci. Stat. Comput. 13(2), 631–644 (1992). https://doi.org/10.1137/0913035

30. Zaytsev, M.Y., Kopiev, V.F., Velichko, S.A., Belyaev, I.V.: Localization and ranking of aircraft noise sources in flight tests and comparison with acoustic measurements of a large-scale wing model. Acoust. Phys. 69(2), 182–192 (2023). https://doi.org/10.1134/S1063771023700562