A SYSTEM FOR ACOUSTIC FIELD MEASUREMENT EMPLOYING CARTESIAN ROBOT

Maciej Szczodrak, Adam Kurowski, Józef Kotus, Andrzej Czyżewski, Bożena Kostek

Gdańsk University of Technology, Faculty of Electronics, Telecommunications and Informatics, Narutowicza 11/12, 80-233 Gdańsk, Poland (✉ ksm@sound.eti.pg.gda.pl, +48 58 347 1114, adakurow@student.pg.gda.pl, jozkotus@pg.gda.pl, andrzej.czyzewski@eti.pg.gda.pl, bozena.kostek@pg.gda)

Abstract

A system setup for measurements of acoustic field, together with the results of 3D visualisations of acoustic energy flow are presented in the paper. Spatial sampling of the field is performed by a Cartesian robot. Automatization of the measurement process is achieved with the use of a specialized control system. The method is based on measuring the sound pressure (scalar) and particle velocity(vector) quantities. The aim of the system is to collect data with a high precision and repeatability. The system is employed for measurements of acoustic energy flow in the proximity of an artificial head in an anechoic chamber. In the measurement setup an algorithm for generation of the probe movement path is included. The algorithm finds the optimum path of the robot movement, taking into account a given 3D object shape present in the measurement space. The results are presented for two cases, first without any obstacle and the other – with an artificial head in the sound field.

Keywords: Cartesian robot, sound intensity, sound field, anechoic chamber.

1. Introduction

At present, technologies of measurement of the power flow in a sound field have reached a matured state [1–3]. The power flow in an acoustic field can be directly measured with the sound intensity technique and the results can be presented in a visualized form.

The sound intensity is a measure of the flow of acoustic energy in a sound field [4, 5]. This is a vector quantity defined as the time average of the flow of sound energy through a unit area in a direction perpendicular to the area.

The intensity in a certain direction is the product of sound pressure (scalar) $p(t)$ and particle velocity (vector) component in that direction $u(t)$. The time-averaged intensity I in a single direction is given by (1):

$$I = \frac{1}{T}\int_T p(t)u(t)dt.$$
(1)

The sound intensity vector in three dimensions is composed of the acoustic intensities in each of three orthogonal directions (x, y, z):

$$\vec{I} = I_x\vec{e}_x + I_y\vec{e}_y + I_z\vec{e}_z.$$
(2)

Therefore, the sound intensity vector is defined and calculated according to (3):

$$\vec{I}(i) = \begin{bmatrix} I_x(i) \\ I_y(i) \\ I_z(i) \end{bmatrix} = \begin{bmatrix} X_p(i)\cdot\overline{X_{ux}(i)} \\ X_p(i)\cdot\overline{X_{uy}(i)} \\ X_p(i)\cdot\overline{X_{uz}(i)} \end{bmatrix},$$
(3)

where: $I_x(i)$ – the sound intensity component for x direction for the i-th spectral component; $X_p(i)$ – the coefficients of complex spectrum for the i-th spectral component of acoustic pressure signal; $\overline{X_{ux}(i)}$, the coefficients of conjugated spectrum for the particle velocity in x direction.

A historical review of evolution of sound intensity measurement is given in Fahy's monograph entitled "Sound Intensity" [6]. The sound intensity measurement has become one of the most interesting techniques employed in the acoustic metrology as well as in methods of solving vibro-acoustic problems, as it simplifies measurement techniques, thus effectively replacing the classical methods. Flow motion parameters – as the acoustic particle velocity – may be experimentally measured using a sound intensity probe, which can be used to collect the data to visualize all the phenomena occurring in the examined acoustic vector fields, in both the near field and three dimensional space. The visualization of acoustic energy flow in real acoustic 3D space fields can explain many of energetic effects (scattering, vortex flow, shielding area, *etc*.) [7].

In the conventional acoustic metrology, analysis of acoustic fields deals only with distribution of pressure levels (scalar variable), however in a real acoustic field both scalar (acoustic pressure) and vector (the acoustic particle velocity) effects are closely related. Only when the acoustic field is described by both potential and kinetic energies, one may understand mechanisms of propagation, diffraction and scattering of acoustic waves by obstacles, presented in a form of energy image.

For measurement of spatial sound level or sound intensity distribution, a grid of points has to be defined. In particular, this means excluding those points, which refer to the sound wave encountering an obstacle, *i.e.* the examined object. Defining the grid manually is often difficult and can lead to errors, especially if the measurement in a certain point has to be repeated due to particular circumstances. Therefore, a dedicated algorithm has been developed by the authors to automatize this task.

The presented work aims to automatize measurements of the acoustic field parameters, *i.e.* its sound level and sound intensity. The measurement results are provided in a form of visualizing the acoustic field around an artificial head, using the sound intensity measurement technique. To measure the sound intensity, the authors applied 3D Acoustic Vector Sensor. A similar methodology was applied by the authors when they used a human head simulator in a free field [8].

The main aim of the work was achieved by two means. Firstly, an algorithm for finding the optimum path of the robot movement, taking into account any shape of a 3D object present in the measurement volume, was constructed. Then, an autonomous robot control software was created to perform measurements according to a specified list of points. The software includes such features as: playing back the excitation signal in a measurement point, automatic detection of the signal and synchronous recording the response of the examined system.

Section 2 describes the Cartesian robot. In Section 3, a detailed description of the measurement system setup is provided. Section 4 presents sample results obtained by the proposed system and Section 5 contains the conclusions.

2. Cartesian robot

The Cartesian robot is a motion system with an operational range of 1.85 m × 2 m × 1.54 m. The robot is installed in an anechoic chamber. The robot mechanics and its accompanying control devices were designed and manufactured according to the detailed specifications prepared by the authors. They included the requirements of a high precision positioning and a small weight. The main and critical issue was to ensure "silent" conditions during measurements, so that the background noise would not impair the quality of collected data. In the design and manufacturing processes following these requirements, the noise level

produced by operation of the robot does not exceed 40 dB, even with simultaneous motion along all three axes. Moreover, when the robot is in a measurement position, it does not emit noise at all. The control part of the robot is compatible with the PXI (PCI eXtensions for Instrumentation) platform that was used for the actual audio measurements. The movement speed is 50 mm/s for X and Y axes and 15 mm/s for Z axis. The positioning accuracy of the robot is 200 µm.

A dedicated programming API for controlling the robot movement was delivered by the robot manufacturer. The movement towards a required 3D spatial position can be done in steps, either separately along each axis X, Y, Z (every time in one axial direction) or simultaneously along all three axes.

The positioning time for each axis was measured. The results obtained for 10 mm and 100 mm movement distances are presented in Table 1. A series of 50 measurements were performed for each case, therefore there are shown the minimum and maximum values.

Table 1. Comparison of the measured robot positioning times for distances of 10 mm and 100 mm.

	t[s], 10 mm		t[s], 100 mm	
	min	max	min	max
x	2.52	5.38	3.05	3.25
y	2.53	5.83	3.07	3.52
z	6.62	21.45	7.97	13.16

The difference between the movement times referring to the distances of 10 mm and 100 mm is not proportional. It is directly related to the characteristic of the robot, whose servo mechanisms consume proportionally more time at the end phase of motion in order to maintain precision of movement, achieving slow and precise approaching. Therefore, large distance movements are more effective as the time cost is lower than for a series of short distance movements.

3. Measurement control system

3.1. Overview of measurement control system

A block diagram of the measurement control system is presented in Fig. 1. The main element is the Control Application running on a PC which controls all the processes automatically. The NI PXI-8532 module is used for controlling the position of the robot arm. Reading the results (the response signal) from the USP (*Ultimate Sound Probe*) probe is executed by the NI PXIe-4492 module. The excitation signal is provided to the loudspeaker by the Maya44 audio converter. The robot with the USP probe and a sound source are located in an anechoic chamber; the remaining system elements are placed in a separate control room.

A general concept of the measurement management system is presented in Fig. 2. Two main stages of the measurement process can be distinguished. The role of the first one is to prepare a list of points ordered according to the optimum movement rules. A numerical model of either a measured object or an obstacle is created and the points of grid, in which the robot arm with the USP probe can be located, are calculated. The details of the algorithm are provided in Subsection 3.2. The second stage is related to automatic collection of the measurement results in the grid of points. The software integrates the robot movement control and data acquisition in given measurement points. It is the result of the authors' conceptual work, written in C++ language. Each measurement point is described by its spatial coordinates (x, y, z), together with the information along which axes the robot is to be moved and whether to record the response signal and/or play back the input signal while the robot is positioned at this point. Without

entering too deeply into the technical details of implementation, the main elements of the measurement cycle are as follows: move the robot arm to a required position, play back the test signal and start recording the response signal, then save the result for further analysis, typically in a computer memory file. The programming libraries, written in C language, enabling to control the basic robot functions (such as: initialization, movement along required axes, turning on/off the motors), capture data from the hardware A/D converters and play back signals, were provided by the robot manufacturer.
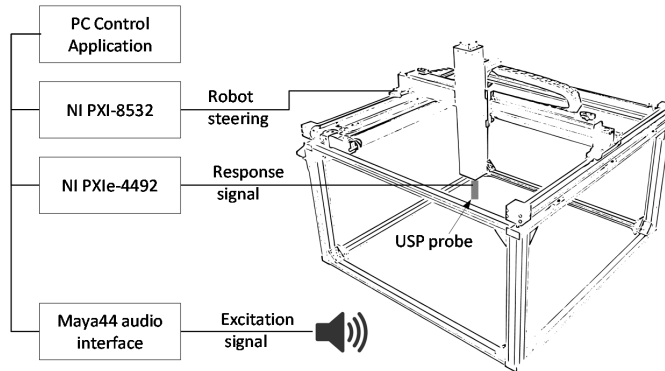
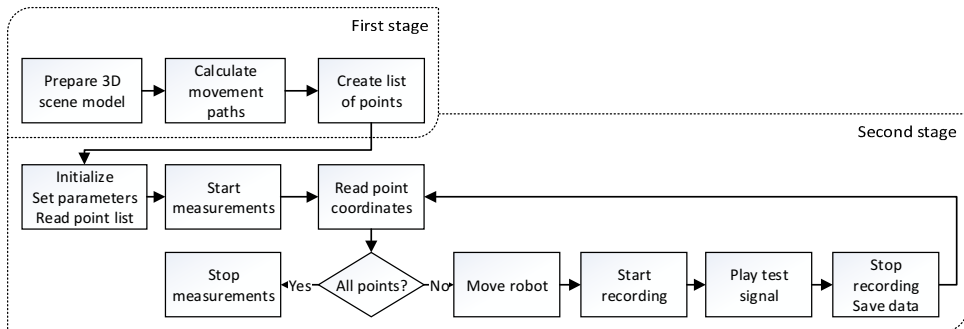Fig. 1. A diagram of the measurement system.

Fig. 2. A diagram of the measurement control application concept.

The software is designed in such a way that the dedicated threads are responsible for each hardware module. The main thread is responsible for the robot iterative operation through a given list of points using PXI-8532. A separate program thread controls both playing back the test sound and reading the result (the response signal) from the USP probe via the NI PXIe-4492 module. Recording from the USP probe requires four PXIe-4492 channels, one for the sound pressure, and three for the spatial components of air particle velocity (x, y, z). Recording the response signal can be triggered by the input signal according to the implemented algorithm, in order to detect the sound activity. The input signal includes a required content preceded by a tone, so that the recorded response in each point is time-synchronized.

The measurement system is designed to work in the following modes:

− Measurement of sound intensity distribution in the steady state conditions, *i.e.* continuous emitting an invariable acoustic signal by the sound source and the measurement being taken by the USP probe during the assigned time period in each point.

- An acoustic event triggers the measurement, *i.e.* a particular sound source activity is treated as an acoustic event that is detected by the measurement system and interpreted as the signal to start the measurement.
- Driving the sound source activity depending on the USP probe position. If the USP probe is positioned at a measurement point by the robot arm, the sound source emits a special signal, *i.e.* chirp. The pulse response can be measured precisely in all points of the grid.

Such a modular design of the measurement management process has a major advantage. The first stage is needed only when a new type of object has to be measured or if the object position or grid parameters are changed.

Moreover, compared to the techniques applied in the authors' earlier research which were based on manual determination of the list of measurement points and continuous recording of the output data during measurement, the presented system provides a remarkable improvement. If, for some reasons, the process is stopped, it can be easily resumed. Moreover, the storage requirements for the output data are optimized, since the response signal is recorded only when the USP probe is located exactly at a grid point, while in the previous study, the recording during measurement was continuous.

3.2. Algorithm of generation of probe movement path

Creation of a grid of measurement points is usually an initial step of preparing the sound intensity measurement. Such a grid is loaded to the program which controls positioning the sound intensity probe of the Cartesian robot. Simple grids can be defined manually, but such an approach becomes inefficient when either the number of measurement points is large or geometry of the obstacles placed in the working area of the Cartesian robot is complicated. Therefore, a script for generating such grids and calculating the motion path for the probe was implemented in the Python programming language. It is capable of calculating a grid of points based upon an approximate geometry of the obstacle objects and establishing a route of passing those points by the sound intensity probe. The shape of obstacles is approximated by geometric primitives which are represented by triangle grids. The obstacles may consist of circles, triangles, rectangles and ellipses. They are the areas, in which the intensity probe must not be positioned. It is possible to visualize both the shapes of defined obstacles and the probe path generated by the script. Examples of paths and shapes created with the use of the presented algorithm are shown in Fig. 3.
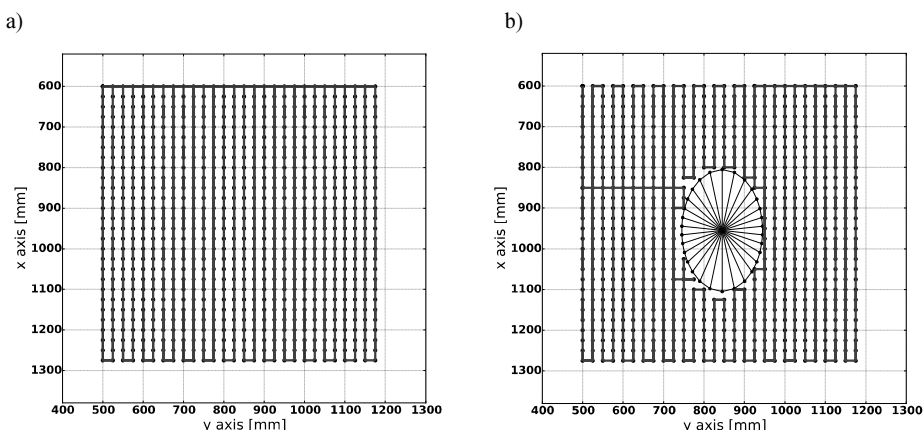

Fig. 3. Examples of the intensity probe motion paths generated with the use of the constructed algorithm.
a) The intensity probe motion path for measurement of the field without a dummy head;
b) the intensity probe motion path for the field around a dummy head placed in the measurement area.

The mesh generation process consists of a few steps which are presented in Fig. 4. In Step 1, a grid of uniformly distributed measurement points is created. In Step 2, the geometric restrictions representing obstacles placed in the working area of the robot are applied to the grid obtained in the previous step. All vertices which are inside any of the triangles of any obstacle mesh are deleted. The same principle applies to the edges of the graph which are adjacent to the deleted vertex. Moreover, all edges of the graph, which are intersecting any edge of any obstacle, are also deleted. After Step 2 there is no possibility of colliding with the restricted area if all the moves are performed along the remaining edges of the graph. Such a procedure eliminates diagonal moves which may lead to problems if one of the robot motor fails to move the arm of the Cartesian robot and the movement of the probe would be performed only along one of the axes. Checking whether a point is inside a triangle of the obstacle mesh is done on a geometric basis. If it is placed inside a triangle, then the angles between three vectors routed from the checked point to the vertices of the triangle sum up to 2π radians. Such a check should be performed for all triangles of the meshes representing the obstacles. The edges of the resulting graph represent all allowed movements of the Cartesian robot arm.
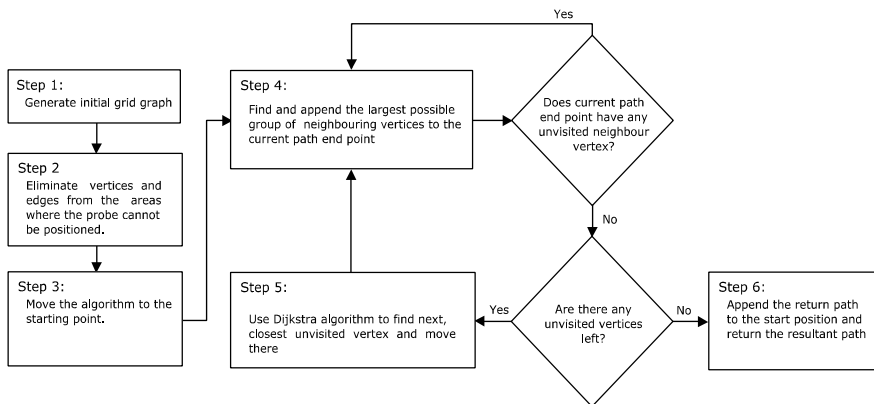

Fig. 4. A diagram of the path generation algorithm.

Step 4 together with the associated condition is a stage of path construction according to the given graph structure. The neighbouring vertices are being connected with each other until there is no possibility for the probe to move to the next unvisited vertex without passing any vertex which was visited earlier. Due to the technical limitations of the Cartesian robot, the most used axis should be the x axis. Therefore, the algorithm tends to perform the majority of movements along the vertical axis. Such a behaviour is obtained by constructing a path from six pre-defined path expansion shape types. These shapes are: straight vertical lines, vertical bends of straight probe trajectories and vertical meanders of trajectory. All those types of the path expansion shapes are shown in Fig. 5. A path formed after many repetitions of Step 4 consists of many expansion shapes connected with each other. Each appended part of the path has to preserve the vertical-traverse nature of the sound intensity probe movement. The second criterion used for selection of the shape is the maximum length of the expansion shape. Construction of a path segment starts with a single vertical line created in either upward or downward direction depending on which direction allows for connecting more vertices. Then, six possible expansion shapes are calculated: three of them for expansion of the path to the right side, three of them − to the left side. An expansion shape with the greatest length is selected and appended to the current path. The whole process is repeated until there is no possibility of the path expansion without passing an earlier visited vertex of the graph. An example of such a path segment construction process is shown in Fig. 6. Due to the fact that expansion shapes

can be appended to the constructed route from both left and right sides and the constructed path may run from the bottom to the top and vice versa, there are three mirror images available for each kind of the expansion shape. This enables to keep the traverse nature of movement regardless of the current path topology. An example of four mirror images of path expansion shape is shown in Fig. 7. Moving into Step 5 is done after checking whether there are any unvisited vertices and finding at least one. The shortest path leading to the closest unvisited vertex according to the current path end is computed by the Dijsktra algorithm, which is a popular algorithm, frequently used in route planning tasks [9, 10]. Only the necessary points from the path created by the Dijkstra algorithm are appended to the designed probe route. These are the points where the direction of the sound intensity probe movement changes. The vertices of that path are marked as the transit points in order to inform the system to not perform the measurement there. The structure of the graph forced by Step 2 makes sure that the route would not cross the area where the obstacles are placed. After reaching an unvisited vertex, the algorithm from Step 3 is launched once again with this vertex as a starting point. The process continues until all vertices from the graph are visited at least once.

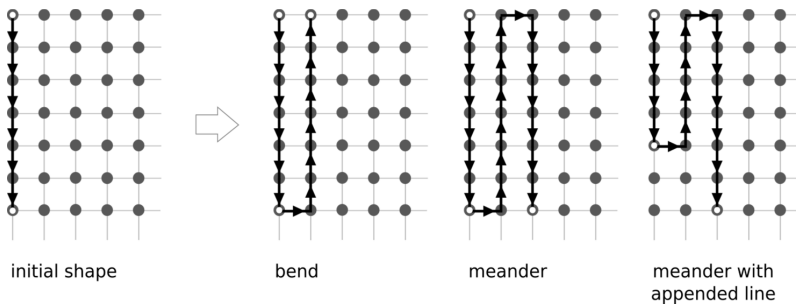initial shape   bend   meander   meander with appended line

Fig. 5. Three types of the measurement probe movement path expansion methods.
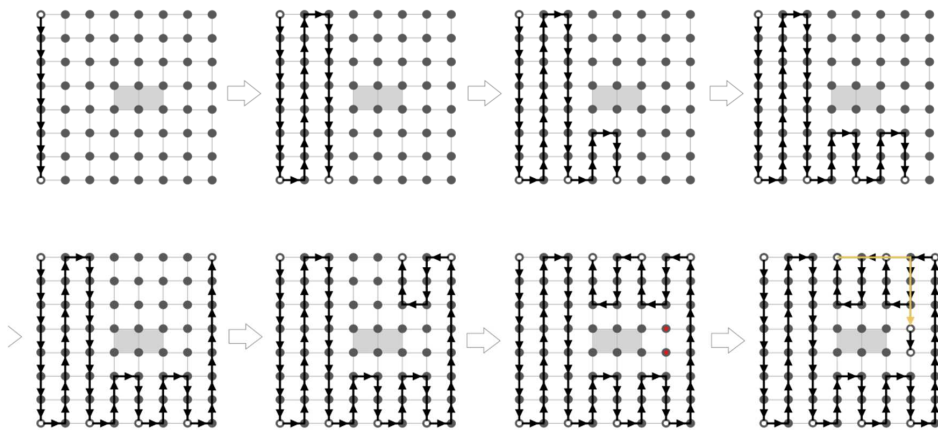

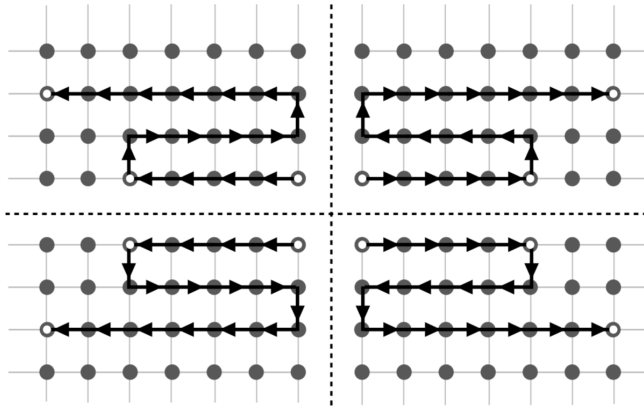Fig. 6. An example of path formation by the path formation process.

Fig. 7. An example of the expansion shape and mirror images associated with it.

The algorithm was tested and compared with the *naive* version of the algorithm which connects neighbouring vertices according to the simple rule based upon priority of each movement direction, *i.e.* first move down, if this is not possible move up, then left, then right. The test scenarios included areas with a rectangular obstacle either without rotation or rotated 30 or 60 degrees clockwise and counter-clockwise. The results of performance measurement are presented in Table 2. The measured parameters were: the number of algorithm iterations which corresponded to Dijkstra algorithm calls, the length of calculated path, the duration of calculation process, the ratio of the numbers of vertical and horizontal movements denoted as *v/r* ratio and the number of path structures called loops. The loops are sequences of four movements which may lead to self-intersection of the path. An example of such a sequence is: up, right, down, left. A large number of such sequences shows that the resultant path is chaotic. The drawback of the *naive* algorithm is a chance that in particular circumstances it can generate a significant number of loop-shaped patterns. This kind of situation occurred for the scenario with a rectangular obstacle rotated 30 degrees clockwise.

Table 2. Comparison of performance of the *naive* and proposed route computation algorithms.

Angle [deg]	Algorithm	Number of iterations	Route length [points]	Computation time [ms]	v/r ratio	Loops
−60	*naive*	8	740	799	18.42	1
	proposed	8	754	814	14.06	2
−30	*naive*	11	746	1060	15.91	2
	proposed	11	754	1075	12.93	2
0	*naive*	3	735	362	16.45	1
	proposed	3	735	369	16.45	1
+30	*naive*	13	755	1245	12.21	10
	proposed	9	753	900	13.17	3
+60	*naive*	8	743	802	17.52	1
	proposed	8	756	808	13.5	2

The proposed algorithm performed better with this scenario and generated three loop structures instead of 10. Comparison of both output routes is shown in Fig. 8. In this case it also generated the route 345 ms faster, with a higher value of *v/r* ratio and a lower number of iterations. For the rest of cases, the proposed algorithm achieved results similar to the *naive* version of the route calculation algorithm.
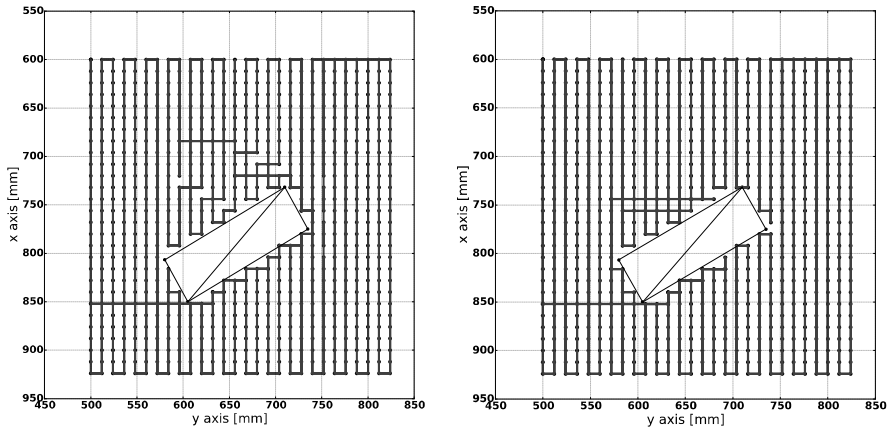
Fig. 8. Comparison of paths created by the *naive* (left) and the proposed (right) versions
of path creation algorithm.

4. Measurement results

The practical usage of the considered measurement control system is presented in a form of an example of sound intensity distribution in the proximity of a human head. The head was simulated by an artificial head and torso made by B&K, model 4128C. The sound intensity value in a grid of points was obtained by the described method of measuring the sound pressure and the air particle velocity.

According to the steps described in Subsection 3.1, there was created a 3D model of torso and − using the algorithm described in Subsection 3.2 − a grid of points defining the route of moving the robot arm with the USP probe .

The measurement setup is presented in Fig. 9. The artificial head was placed in the middle of the robot operational range. The sound source was a loudspeaker situated in such a way that the incident acoustic wave was directed towards the head from approx. 30 degrees.
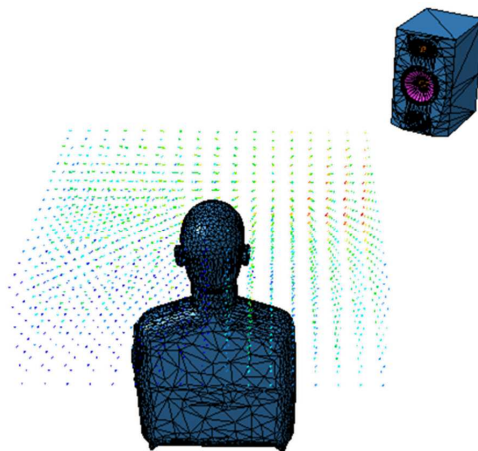

Fig. 9. The setup for measurements of acoustic field distribution near a simulated human head.

The results of measurements of sound intensity for the grid raster of 25 mm, each for two cases, *i.e.* without and with an artificial head, are presented in Figs. 10 and 11. The sound intensity vectors and the sound intensity magnitude are marked in each drawing. The white spaces visible in the graphs represent the locations of the artificial head. Therefore, the measurement data were not collected from within that area.
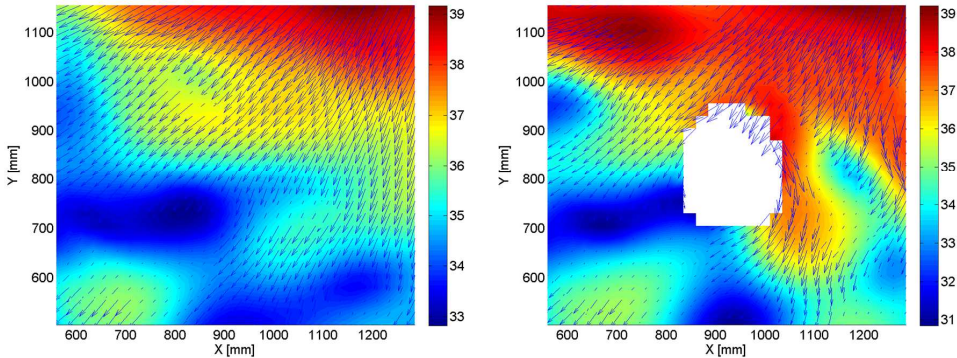
Fig. 10. The *xy*-plane projection of vector field and sound intensity magnitude section at a height equal to the middle of head, for 1 kHz – without (left) and with a head (right).

Fig. 11. The *xy*-plane projection of vector field and sound intensity magnitude section at a height equal to the middle of head, for 4 kHz –without (left) and with a head (right).

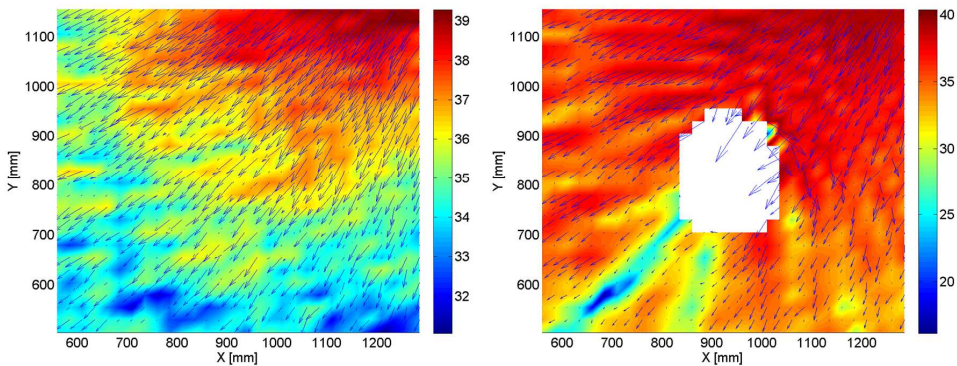The influence of the head on the acoustic field is clearly visible. The magnitude of vectors is lower behind the head, and a specific shadow can be observed. Moreover, the incident wave reflects from the obstacle and interferes, what gives characteristic local maxima and minima of the vector field magnitude. The effect of reflection is particularly visible in front of the head.

As a concluding remark it can be said that – to some extent – the above results may be compared with the results obtained employing the technique called *Particle Image Velocimetry* [11, 12], as it enables to present evolution of the flow field, as well as to thoroughly study aero-acoustical velocity fields. The details of using *the phase-locked particle image velocimetry* along with applying the *Proper Orthogonal Decomposition* method [13] are shown in the papers by Mickiewicz [11, 12] and could be used for such a comparison.

5. Summary

The system for performing automatic measurements of the sound intensity using a Cartesian robot has been presented. The system has a modular design and is versatile. Moreover, compared to our earlier applied techniques which were based on manual determination of the list of measurement points, and on continuous recording of the output data during measurement, the presented system provides a remarkable improvement.

The system can be extended, for example by providing methods for autonomous obstacle detection and avoidance. Then, the step of modelling the scene could be omitted.

Acknowledgements

References

[1] Weyna, S. (2012). Visualization method of acoustic wave propagation based on the sound intensity measurement. *Acoustical Imaging*, 31, Nowicki A., Litniewski J., Kujawska T. (eds.), 243–252.

[2] Weyna, S. (2010). Acoustic Intensity Imaging Methods for in-situ Wave Propagation. *Archives of Acoustics*, 35(2), 265–273.

[3] Weyna, S. (2005). Microflown based identification of vortex shaddind in the space of real acoustic flow fields. *Twelfth International Congress on Sound and Vibration (ICSV)*, Lisbon, Jul. 11–14, 2005.

[4] de Bree, H.E. (2003). The Microflown: an acoustic particle velocity sensor. *Acoust Aust*, 31(3), 91–94.

[5] Jacobsen, F., de Bree, H.E. (2005). A comparison of two different sound intensity measurement principles. *The Journal of the Acoustical Society of America*, 118, 1510–1517.

[6] Fahy, F.J. (1995). *Sound intensity*. E & F.N. Spon.

[7] Weyna, S. (2010). An Acoustics Intensity Based Investigation of the Energy Flow Over the Barriers. *Acta Physica Polonica A. No. 1, Acoustic and Biomedical Engineering*, 118, 172–178.

[8] Kotus, J., Kostek, B. (2015). Measurements and Visualization of Sound Intensity Around the Human Head in Free Field Using Acoustic Vector Sensor. *J. Audio Eng. Soc.*, 63(1/2), 99–109.

[9] Fan, D., Shi, P. (2010). Improvement of Dijkstra's algorithm and its application in route planning, Fuzzy Systems and Knowledge Discovery (FSKD). *Seventh International Conference on, Yantai, Shandong*, 1901–1904.

[10] Sniedovich, M. (2006). Dijkstra's algorithm revisited: the dynamic programming connexion. *Control and Cybernetics*, 35(3), 599–620.

[11] Mickiewicz, W. (2015). Particle Image Velocimetry and Proper Orthogonal Decomposition Applied to Aerodynamic Sound Source Region Visualization in Organ Flue Pipe. *Archives of Acoustics*, 40(4), 475–484.

[12] Mickiewicz, W. (2014). Systematic error of acoustic particle image velocimetry and its correction. *Metrol. Meas. Syst.*, 21(3), 447–460.

[13] Weyna, S., Mickiewicz, W. (2013), Multi-Modal Acoustic Flow Decomposition Examined in a Hard Walled Cylindrical Duct. *Archives of Acoustics*, 39(2), 289–296.