(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 2 648 425 A1

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
09.10.2013 Bulletin 2013/41

(51) Int Cl.:
H04R 27/00 *(2006.01)*

(21) Application number: **12163054.5**

(22) Date of filing: **03.04.2012**

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME

(71) Applicant: **Rinnic/Vaude Beheer BV**
3144 DG Maassluis (NL)

(72) Inventors:
- **Van Duin, Eric**
  3144 DG MAASSLUIS (NL)
- **Kleine, Kevin Ivan**
  3144 DG MAASSLUIS (NL)
- **Van der Voet, Zeger**
  3144 DG MAASSLUIS (NL)

(74) Representative: **De Vries & Metman**
Overschiestraat 180
1062 XK Amsterdam (NL)

(54) Simulating and configuring an acoustic system

(57) The simulated speaker settings are generated such that said speakers produce a desired sound pressure distribution in an acoustical space of a venue. The method includes the steps of: providing an at least partially bounded virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue; determining one or more sound spots by selecting one or more volumes in said virtual space and assigning said one or more sound pressure values to said volumes; determining at least part of said target speaker settings by executing an evolutionary algorithm, comprising:
generating a population of speaker settings, preferably said speaker settings including at least one of: speaker location information, speaker phase delay information and/or speaker amplitude information; and, optimizing said population of speaker settings on the basis of said one or more sound pressure values associated with one or more sound spots.

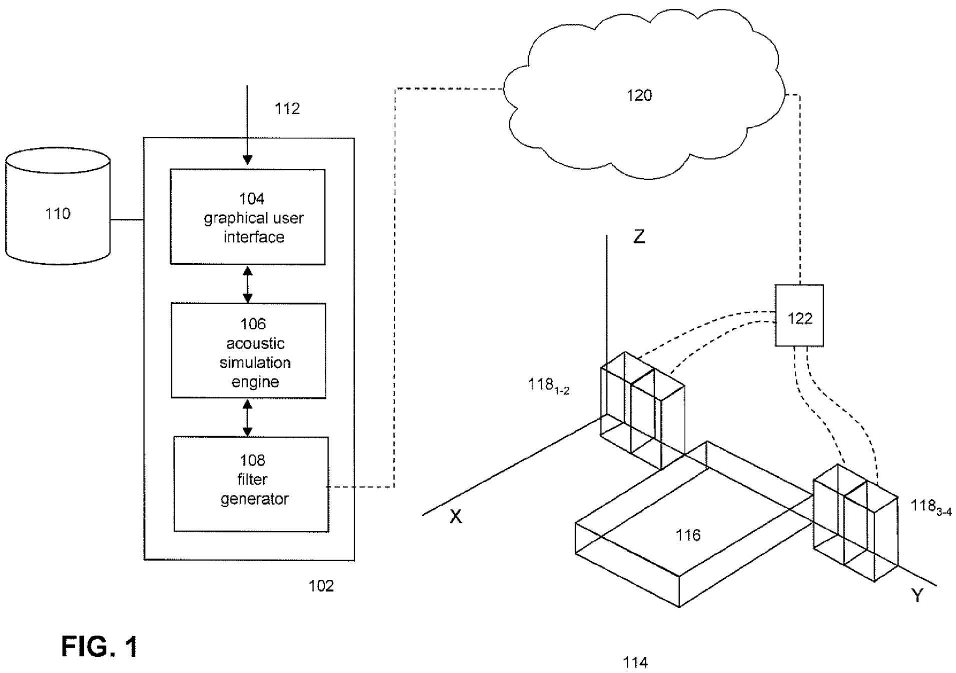

FIG. 1

EP 2 648 425 A1

Description

Field of the invention

[0001] The invention relates to simulating and/or configuring an acoustic system, and, in particular, though not exclusively, to methods for simulating and/or configuring an acoustic system comprising acoustic sources, an electronically configurable acoustic system, an electronically configurable acoustic source for use in such system and a software program product for simulating and/or configuring such electronically configurable acoustic sources.

Background of the invention

[0002] Optimizing the configuration of an acoustic system comprising a plurality of speakers for a predetermined venue, e.g. an outdoor music festival, a concert hall, a shopping mall or simply a living hall, is complex process requiring know-how that is only available to an expert, e.g. a sound engineer. For each situation, a sound engineer needs to look on-site for an optimal speaker configuration wherein multiple speakers need to be positioned and, preferably, configured within the constraints of a particular three-dimensional (3D) space such that a desired sound quality is achieved.

[0003] Sound engineers, who need to find an optimal configuration on the basis of multiple system parameters (e.g. source position and direction, delay between different sources, dimensions and acoustic properties of objects located within that space etc.), may use simulation tools and know-how from earlier, similar situations in order to find a best or at least an acceptable solution for a new situation. Currently available simulation tools however only provide limited functionality in the sense that such they allow simulation and - in some cases - visualization of a particular configuration defined by a sound engineer. Examples of such conventional simulation systems are described in US2009/0144036 and US7206415.

[0004] These conventional simulation tools do not provide and/or suggest a particular optimal speaker configuration in terms of location and parameter settings of the speakers. Moreover, these simulation tools do not provide functionality wherein a user may define and shape the acoustic characteristic of a 3D venue space bound by e.g. walls, a floor and a ceiling and comprising objects with specific acoustic properties. For example, if an outdoor concert is held at a venue close to a residential area, it may be desired "engineer" the 3D sound pressure distribution around the venue such that the sound level at the residential area is kept below a certain level. Similarly, in an indoor concert hall, it may be desired to define silent zones where people can talk to each other without the need to isolate that such area using e.g. sound absorbing or reflecting objects.

[0005] Acoustic simulation systems allowing such functions are currently not available because the problem is simply too complex. Shaping the sound pressure in a 3D space, which is populated with objects, would require precise knowledge of the acoustic properties of those objects in the space as well as precise control of the sound produced by each source. Known acoustic systems and software for configuring such systems are not suitable for providing such control. Hence, there is need in the art for improved methods and systems for simulating and/or configuring an acoustic system.

Summary of the invention

[0006] It is an objective of the invention to reduce or eliminate at least one of the drawbacks known in the prior art. In a first aspect the invention may relate to a computer-implemented acoustic simulation method for generating one or more speaker set-ups for electronically configuring speakers in an acoustical system such that, when configured, said speakers produce a desired sound pressure distribution in an acoustical space of a venue, wherein said method may comprise the steps of: providing an at least a partially bounded virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue; determining one or more sound spots by selection of one or more volumes at one or more sound spot locations in said virtual space and assigning said one or more sound pressure values to said one or more volumes; determining at least part of said speaker set-ups settings by executing an evolutionary algorithm, preferably a differential evolutionary algorithm. In an embodiment said algorithm comprising the steps of: generating a population of speaker set-ups, preferably a speaker set-up including at least one of: speaker location information, speaker phase delay information and/or speaker amplitude information; and, optimizing said population of speaker set-ups on the basis of said one or more sound pressure values associated with said one or more sound spots.

[0007] Hence, method allows simple and intuitive modeling on a venue of arbitrary design and optimization of an acoustical system located in that virtual space of the modeled venue such that speaker set- ups are generated which match a desired 3D sound pressure distribution.

[0008] In an embodiment said algorithm may further comprise: creating a further speaker set-up on the basis one or more speaker set-ups in said population; and, optimizing said population of speaker settings by replacing a speaker set-up in said population with said further speaker set-up on the basis of a fitness function configured for comparing sound pressure values associated with said one or more sound spots with sound pressure values calculated on the basis of said further speaker set-up.

[0009] In another embodiment, said algorithm may comprise: determining one or more speaker envelopes by selecting one or more volumes in said displayed virtual space, a speaker envelope defining a volume for locating one or more configurable speakers; and, generating said population of speaker settings on the basis of said one or more speaker envelopes.

[0010] In yet another embodiment, defining said one or more speaker envelopes further may include: selecting a predetermined number of speakers; associating speaker locations to said predetermined number of speakers on the basis of said speaker envelopes.

[0011] In an embodiment optimizing said population of speaker set-ups may further comprise: checking whether the speaker locations associated with said further speaker setting are at least partly located within said one or more speaker envelopes.

[0012] In an embodiment determining one or more sound spots may include: selecting said one or more volumes in said virtual space using a graphical user interface; or, selecting one or more sound spots from a database comprising a list of stored sound spots.

[0013] In yet another embodiment, determining one or more speaker envelopes may include selecting said one or more volumes in said virtual space using a graphical user interface or selecting one or more speaker envelopes from a database comprising a list of stored sound spots.

[0014] In an embodiment, in case of using a graphical user interface, selecting said one or more sounds spots or one or more speaker envelopes may further include: rendering of a graphical selector, preferably a 3D wire-frame, in said displayed virtual space; selecting a sound spot or speaker envelope location by locating said graphical selector in said virtual space; selecting a sound spot or a speaker envelope volume by graphically modifying the 3D shape of said selector.

[0015] In an embodiment said method may further comprise: selecting at least one speaker set-up from said optimized population of speaker set-up; sending said at least one speaker set-up, preferably said speaker comprising filter coefficients, to at least part of said electronically configurable speakers.

[0016] In one aspect, the invention may relate to an acoustic simulation system for configuring one or more electronically configurable speakers in an acoustical system such that said configurable speakers produce a desired sound pressure distribution in an acoustical space of a venue, said acoustic simulation system being configured for: providing an at least partially bounded virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue; determining one or more sound spots by selecting one or more volumes at one or more sound spot locations in said virtual space and assigning said one or more sound pressure values to said one or more volumes; determining at least part of said speaker set-ups settings by executing an evolutionary algorithm, preferably a differential evolutionary algorithm, said algorithm comprising: generating a population of speaker set-ups, preferably said speaker set-ups including at least one of: speaker location information, speaker phase delay information and/or speaker amplitude information; and, optimizing said population of speaker set-ups on the basis of said one or more sound pressure values associated with said one or more sound spots.

[0017] In a further aspect, the invention may relate to a configurable acoustical source comprising: a digital signal processor configured for receiving at least one optimized speaker set-up from an acoustical simulation system as described above, said digital signal process further being configured to configure at least part of a digital audio filter on the basis of filter coefficients in said speaker set-up; or, said digital signal processor further being configured for executing a computer-implemented acoustic simulation method as described above.

[0018] In another aspect, the invention may relate to graphical user interface for use in an acoustical simulation system as described above, wherein said graphical user interface may be configured for graphically defining boundary conditions for said evolutionary algorithm, wherein said graphical user interface may be configured for: displaying a partially bounded 3D virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue; rendering of a 3D shaped graphical selector, preferably a 3D wire-frame, in said displayed virtual space, said 3D shaped graphical selector being associated with location coordinates in said virtual space and wherein said 3D shaped graphical selector comprising one or more shape manipulation markers wherein a user may interact with said 3D shaped graphical selector such that: a sound spot location is selected by locating said graphical selector at predetermined location coordinates in said virtual space; a sound spot volume is selected by graphically modifying the 3D shape of said selector so that it defines volume of a predetermined size in said virtual space.

[0019] The invention also relates to a computer program product, implemented on computer-readable non-transitory storage medium, wherein the computer program product may be configured for, when run on a computer, executing the method steps as described above. The invention will be further illustrated with reference to the attached drawings, which schematically will show embodiments according to the invention. It will be understood that the invention is not in any way restricted to these specific embodiments.

Brief description of the drawings

[0020]

Fig. 1 depicts an acoustic simulation system according to one embodiment of the invention.
Fig. 2 schematically depicts a flow diagram for defining boundaries and conditions used by the acoustic simulation system for simulating an acoustic system in a venue according to an embodiment of the invention.
Fig. 3 depicts a GUI for building a virtual 3D venue space according to an embodiment of the invention.
Fig. 4 depicts a GUI for building a virtual 3D venue space according to an embodiment of the invention.
Fig. 5 depicts a GUI for building a virtual 3D venue space according to an embodiment of the invention.
Fig. 6 depicts a GUI for building a virtual 3D venue space according to an embodiment of the invention.
Fig. 7 depicts a GUI for building a virtual 3D venue space according to an embodiment of the invention.
Fig. 8 depicts a GUI for building a virtual 3D venue space according to an embodiment of the invention.
Fig. 9 schematically depicts a flow diagram of a computer-implemented acoustic simulation method according one embodiment of the invention.
Fig. 10 depicts a graphical representation of a simulated optimized speaker setting according to an embodiment of the invention.
Fig. 11 depicts a graphical representation of at least part of a simulated optimized speaker setting according to an embodiment of the invention.

Detailed description

[0021] **Fig. 1** schematically depicts an acoustic simulation system **102** according to an embodiment of the invention. In particular, **Fig. 1** depicts an acoustic simulation system comprising a 3D venue editor **104** for graphically building and modeling a virtual 3D model of a venue using a special graphical user interface (GUI), an acoustic simulation engine **106** and a filter generator **108.** The acoustic simulation system is configured to design, build and simulate a virtual 3D model of a real-life venue **114** and to use the simulation results to determine locations for configurable speakers in the real-life venue and to generate filter coefficients for the configurable speakers $\mathbf{118_{1\text{-}4}}$.

[0022] A venue may be defined as a place wherein the acoustical system is installed or going to be installed, e.g. a concert hall, a shopping mall or simply a living hall at home. The venue typically defines a 3D space, hereafter referred to as a (3D) venue space, which is bound or at least partially bound by structures, e.g. walls, floors and/or a ceiling, which may (partially) reflect, diffract and/or absorb sound (sound pressure waves) generated by the speakers. Sound may also be (partially) reflected, diffracted and/or absorbed by objects located in the venue, e.g. a stage **116,** tribunes, people, etc. Hence, the sound experienced by a person in the venue is determined by the acoustical properties of the above-mentioned structures and objects (including people) and the configuration of the acoustic system.

[0023] The sound pressure waves generated by the speakers will cause a complex sound pressure distribution in the 3D venue space, which may be manipulated by either changing the venue structure or by changing the locations and/or setting of the speakers. Preferably, an acoustic system should be able to deliver sound of a particular quality and/or level. For a given venue, the sound pressure distribution depends on many parameters and is therefore too complex to determine. Therefore, typically, only sub-optimal configurations are realized using trail-and-error methods combined with specific know-how and/or software tools.

[0024] In order to alleviate this problem and to allow even a layman to build and configure an acoustic system for a particular venue layout, the simulation system comprises a 3D venue editor. The editor is configured to render in a graphical user interface (GUI), which allows user interaction **112,** in order to graphically define virtual structures and objects in 3D space. Further, the GUI may allow to define predetermined volumes - sound spots and/or listening planes - in the 3D venue space and assigning certain targeted acoustic pressure values to thereto. The GUI of the 3D venue editor may further allow a user to select speaker envelopes, i.e. predetermined volumes wherein configurable speakers are allowed to be located. The sound spots, listening planes and speaker envelopes will be described hereunder in more detail.

[0025] Once a venue, including the sound spots, listening planes and speaker envelopes, is defined and modeled, the venue data may be stored in a database **110** for later use. Different pre-defined venues may be stored in the database allowing a user to select one and, optionally, to customize the venue via the GUI and to assign desired sound pressure levels to predetermined sound spots and listening planes.

[0026] The acoustic simulation engine may use the venue data to execute an algorithm which is configured to search for sound pressure distributions in the virtual venue space which closely match the targeted sound levels defined by sound spots and/or listening planes. A sound pressure distribution may be calculated on the basis of speaker set-up information or - in short a speaker set-up - comprising e.g. speaker locations, and a speaker direction and audio filter coefficients for each speaker as optimization parameters. Preferably, speaker locations are located within predefined speaker envelopes. The audio filter coefficients may be used for configuring an electronic filter in each of the speakers. The audio filter coefficient may define a digital filter comprising a predetermined response in the audio spectrum in terms of phase delays and amplitudes.

[0027] The 3D venue editor may be implemented as a software program comprising a number of GUI functions for modeling structures and objects which are used to graphically design a (partially) bounded virtual 3D venue space (e.g. floor(s), walls, ceiling(s) etc.), and objects within that space (e.g. a stage or tribunes), wherein acoustic properties such as absorption and/or reflection coefficients may be assigned to these elements and objects. The 3D venue editor thus allows the modeling of a virtual 3D venue space, which closely matches the acoustical behavior of the real-life venue space.

[0028] Moreover, the GUI rendered by the 3D venue editor is designed to easily assign predetermined areas, e.g. volumes and/or planes, in the virtual 3D venue space to certain desired sound pressure levels using user- configurable 3D wire frames. The GUI may display the wire frame and allow a user to manipulate it in 3D space using e.g. multi-touch gestures or mouse manipulations. This way, a user may manipulate a wire- framed 3D volume (e.g. a sphere or a cube) in order to define a sound spot in the venue space and to assign a certain desirable sound pressure or sound pressure range to it.

[0029] Hence, the (partially) bounded virtual 3D venue space and the sound spots and/or listening planes in that space define a set of boundaries and conditions for a complex equation wherein sound pressure waves are emitted by configurable speakers into the virtual 3D venue space and wherein interference of these transmitted, diffracted and reflected sound pressure waves forms a complex sound pressure distribution in the 3D venue space. This sound pressure distribution may be calculated for different frequency bands in the acoustic spectrum and for different speaker set- ups. Solutions matching the conditions, which are set by the sound spots and/or listening planes, may be determined and presented as possible optimal solutions. Finding speaker set- ups for a certain desired 3D sound pressure distribution taking into account all frequencies in the audio spectrum is a very complex problem for which no analytical solution exists. As will be described hereunder in more detail, best solutions for the above described problem are determined using an evolutionally algorithm (EA) . The EA algorithm parameters may be configured such that a set of best solutions, i.e. best speaker configurations, may be determined very rapidly.

[0030] Once solutions, i.e. speaker set-ups, are found, the simulation system may present different solutions graphically to the user. Each speaker set-up comprising speakers located at different speaker positions, wherein each speaker has a predetermined audio filter which may be defined in terms of phase delays and amplitudes for different frequency bands in the sound spectrum. Once a particular speaker configuration is selected, the filter generator may generate a set of filter coefficients for each speaker in the speaker configuration. The filter generator may comprise an interface, which is configured to communicate with speakers in the real life venue. The speaker interface may send the filter coefficients in a suitable communication protocol via a (wireless) network **120** or the Internet and a gateway **122** to the speakers.

[0031] A configurable speaker **118_{1-4}** may comprise one or more electro-acoustical transducers **116,** and a signal-processing module **118** for digitally processing an audio signal received from an external amplifier system into a signal, which is suitable for driving the transducers. The electro-acoustical transducers may comprise actuators or drivers of a particular type (e.g. piezoelectric, magnetostrictive and/or electrostatic) for transforming the audio signal into sound waves in the audible frequency range (typically between 20 Hz and 20 KHz). A driver may be configured to reproduce either the full or a part of the audio spectrum, e.g. a subwoofer driver for very low frequencies, woofers for low frequencies, mid-range speakers for the middle frequencies and tweeters for the high audible frequencies. A speaker may be characterized by a speaker directivity function, i.e. a predetermined function determining in which directions a speaker transmits sound pressure waves of a particular frequency.

[0032] The acoustic source may be digitally configurable in that sense that the signal processing module comprise one or more configurable digital audio filters **120** for controlling part of the acoustic properties (in particular the gain and magnitude) of the acoustic source in one or more predetermined frequency bands. The signal-processing module may further comprise a communication module **122** for communication with the acoustic simulation system via a network, e.g. a (W)LAN, WiFi-network, and/or the Internet using a suitable control e.g. Ethernet. For example, in one embodiment, an IP address may be assigned to the signal-processing module so that an acoustic source may be easily accessed and programmed by the acoustic simulation system via the network.

[0033] The acoustic simulation system may be implemented as one or more software programs executed on one or more a (mobile) computing devices, including a PC, a smart phone, an electronic tablet, etc. The simulation system executed on the computing device may send filter coefficients associated with a calculated speaker set-up via a network and/or the Internet to electronically configurable speakers in the real-life venue.

[0034] Alternatively, in another embodiment, the simulation system may be distributed, wherein e.g. a client on a (mobile) computing device may execute the 3D venue editor as an app, which may subsequently send a predetermined venue model of to a server hosting the acoustic simulation engine. On the basis of the boundary and conditions defined in the venue module, the acoustic simulation engine executes the evolutionary algorithm for determining one or more optimized speaker set- ups. Thereafter, the filter coefficients associated with an optimized speaker set- up may be sent to the configurable speakers in the real- life venue.

[0035] In yet another embodiment, several acoustic sources may each receive an individual IP address in order to communicate filter information and/or a venue model (generated by a user using the 3D venue editor) to each other and/or to calculate (fully or partly) the filter information based on the information in the user-generated venue model. In those embodiments, the acoustic sources and simulation system(s) may behave like a distributed computing system.

[0036] **Fig. 2** schematically depicts a flow diagram for defining boundaries and conditions used by the acoustic simulation system for simulating a speaker set- up in a venue according to an embodiment of the invention. In particular, **Fig. 2** depicts a computer- implemented method, which may be executed by the 3D venue editor program as described with reference to **Fig. 1.** When executing the program, a GUI is rendered, which allows a user to easily and intuitively define desired boundaries and conditions of a particular virtual 3D venue space. In a first step **202,** a user may define an at least partially bounded virtual space on a graphical display, which substantially matches at least part of an acoustic space of a (real- life) venue.

[0037] Further, on the basis of a GUI, a user may locate one or more (virtual) objects, e.g. a stage, a wall, people, etc. in said virtual space (step **204) .** A user may further set locations in said space to desired sound pressure values by defining sound spots in said virtual space (step **206) .** In particular, a user may use the GUI to manipulate the location and size of a 3D wire frame defining a volume of a sound spot, which may be associated with at least one sound pressure value or a range of sound pressure values (e.g. a minimum and/or a maximum sound pressure value) . Similarly, a user may set speaker envelopes, i.e. volumes in the virtual space in which speakers may be located (step **208) .** Typically, the speaker envelope is larger than the dimensions of a speaker so that during simulation an optimal speaker location in a speaker envelope may be determined. A speaker envelope may be determined in a similar way as sound spots, i.e. by manipulating a 3D wire frame defining a volume of the speaker envelope. Further, (virtual) speakers may be selected and located in one or more speaker envelopes (step **210) .** The acoustic properties of a virtual speaker may be selected such that they substantial match the acoustic properties of predetermined (real- life) electronically configurable speakers.

[0038] In an embodiment, the user may select speakers from a list of predefined speakers stored in a speaker library. Thereafter, if the sound spots and speaker envelops comprising speakers are properly defined, the simulation system may execute an evolutionary algorithm for optimizing speaker settings (i.e. the location of a speaker in the speaker envelope, and the phase delay and amplitude for frequencies in the acoustic spectrum) of the virtual speakers a on the basis of said sound pressure values in said sound spots and speaker locations in said speaker envelopes (step **212).**

[0039] Hence, simulation system allows simple and intuitive modeling on a venue of arbitrary design. Further, it allows optimization of an acoustical system located in that virtual space of the modeled venue such that speaker set- ups are generated which match a desired 3D sound pressure distribution.

[0040] The process of defining a virtual acoustic space and the objects, sound spots and speaker envelopes located (i.e. steps **202-210**) is illustrated hereunder in more detail with reference to **Fig. 3-8.**

[0041] **Fig. 3** depicts a GUI for building a 3D venue space according to an embodiment of the invention. The GUI may comprise one or more (touch-screen or mouse) buttons **302-306** for selecting a particular graphical tool for building and modeling the 3D venue space. For example, the display may comprise a stage button **302** for activating a stage building function, an object button **304** for activating an object building function and a plane button **306** for activating a plane building function. By selecting these functions, a user may graphically define predetermined planes, objects and stages in a virtual 3D space and assign particular acoustic properties thereto so that the basic bounds of the venue are defined.

[0042] For example, in a first step, a user may select a coordinate system **300** (e.g. conventional Cartesian coordinate system) and the orientation of that coordinate system in the 3D space. Then, a user may activate the plane building function for defining a first plane **308** normal to the z-direction and determine this plane to match the properties of a floor material e.g. concrete.

[0043] The stage building function may allow a user to select a basic stage object **312** from a predetermined number of preconfigured stage objects stored in a database associated with the acoustic simulation system. After having selected a stage object, the displayed object may be dragged to a predetermined position in the 3D venue space. After positioning the stage object, its dimensions may be manipulated. In order to accurately manipulate the size of the 3D object, touch sensitive manipulation markers $\mathbf{314_{1\text{-}2}}$ may be used. The markers may be used to resize the stage object. During manipulation, an object label **316** may be displayed to the user indicating basic dimensions.

[0044] In more general, the plane-, stage- and object building functions may be adapted to create planes and objects (i.e. geometrical objects and/or people) in the 3D venue space and to manipulate the shape, size and angle of the planes and objects by means of a touch-sensitive GUI. In one embodiment, the manipulation of the planes and objects in 3D space may be assisted by manipulation markers $\mathbf{314_{1\text{-}2}}$, e.g. in the form of a predetermined shape (e.g. a triangular shape) which may be manipulated by touch or a mouse. These manipulation markers may be aligned to the axis of the coordinate system and are designed such that the user may easily manipulate a complicated 3D object using e.g. known multi-touch gestures.

[0045] During manipulation, an object label **316** may be displayed to the user indicating the coordinates and dimensions **320** of the plane and, optionally, the material and/or acoustic properties **318** of the object. In one embodiment, when tapping the material indicator in the displayed label, a menu may appear allowing a user to assign certain properties to the plane, e.g. material and/or acoustic properties. When assigning a predetermined material to a plane, the program may link this material to predetermined acoustic characteristics, e.g. reflection and absorption coefficients. A user may repeat the above-described process in order to define predetermined planes and stage objects and other objects in the 3D venue space. For example, planes $\mathbf{310_{1\text{-}6}}$, which are positioned under a predetermined angle from the floor plane may represent a virtual 3D structure, which closely matches a tribune.

[0046] **Fig. 4** depicts a GUI for building a 3D venue space according to another embodiment of the invention. In this embodiment, a user may select the object building function **402** in order to create and located one or more objects **404,406** in the 3D venue space. Upon selecting the object button, a menu structure may appear, allowing a user to select from basic object structures, e.g. a cuboid or spherical 3D frame wire structure, which can be manipulated using manipulation markers **408** into an 3D object of any desired shape and properties. For example, a cuboid object **404** may be selected and stretched to a desired length on the basis of the dimensions displayed in the object label **410.**

[0047] **Fig. 5** depicts a GUI for building a 3D venue space according to yet another embodiment of the invention. In this particular embodiment, a plane **504** associated with part of the tribune may be selected using the plane button **502.** Upon selection, a menu may be displayed comprising - amongst others - a button to activate a listening plane **505,** which is a relatively thin, planar volume parallel to the plane of the tribune. For example, in this particular example, the listening plane may be formed by a planar volume positioned at a predetermined height (e.g., 1.26 meters, the height of a sitting person) above the plane of the tribune. Manipulation markers **506** may modify the height of the listening plane. Further, a sound level button **522** may be used to assign predetermined sound pressure values, e.g. a range comprising a minimum and maximum sound pressures (e.g. 96-100 dB) to the listening plane. The sound pressure value may be values minimal and maximal values associated with one or more predetermined sound frequency bands.

[0048] Further, the influence of the amount of people on the acoustic properties of the tribune structure may be determined using a crowd fill factor **520.** For example, in **Fig. 5,** a concrete tribune structure is for 75% occupied with (sitting) people having an average listening height of 1.26 meter. During the manipulation of the plane, the object parameters, e.g. listening height, area and material, can be view via the object label **508.** The listening plane thus defines a particular condition for the acoustic simulation engine which will look for speaker settings which produce a sound pressure distribution in the venue space which meets the listening plane conditions as close as possible.

[0049] **Fig. 6** depicts a GUI for building a 3D venue space according to a further embodiment of the invention. In this particular embodiment, selecting the silent button **602** may activate a menu for creating sound spots, i.e. predetermined volumes in the 3D venue space associated with a predetermined desired (targeted) maximum sound pressure. If a sound spot defines a volume of relative low sound pressures, such sound spot may be referred to as silent spot). **Different** sound spots may be selected using an "add" button **618.** Upon selection, the GUI may render a 3D wire-framed sphere in the graphical display. The wire-framed sphere may comprise a centre point and axis, which may be parallel to the central coordinate system of the venue.

[0050] One or more wire frame circles may visualize the radius of the spherical volume of the sound spot. A radial manipulation marker, i.e. a triangular-shaped marker **626** pointing toward the centre point of the spherical volume may be used to manipulate the volume of the sound spot. During manipulation, a sound spot label **620** may display the location and the radius of the sound spot. A user may assign a particular maximum sound pressure value to the sound spot **610.** Alternatively and/or in addition, a user may assign further parameters, which may be of importance for the acoustical properties of a sound spot e.g. temperature **604** and humidity **606** of the medium (air) in the sound spot. Multiple sound spots may be defined in order to assign predetermined sound pressure levels to particular volumes in of the venue space. Similar to the listening planes, these sound spots define particular conditions for the acoustic simulation engine, which will look for speaker set-ups which meet these conditions as close as possible.

[0051] As will be described hereunder in more detail, the acoustic simulation engine may look for solutions, which meet the conditions as set by the listening planes and the sound spots. The algorithm will select a group of solutions, i.e. speaker set-ups associated with sound pressure distributions, which match the conditions set by the sound spots/ listening planes as close as possible. In one embodiment, a user may influence the selection of the solutions by the algorithm using a selector, e.g. a slider and a sliding bar **628,630,** in which one sound spot or listening plane may be prioritized over another sound spot or listening plane. Hence, when moving the slider **628** more towards the "silent zones", the algorithm may prioritize solutions in which the silent spots are matched best.

[0052] As will be described hereunder in more detail, the acoustic simulation engine will look for solutions, i.e. speaker set-ups and associated 3D sound pressure distributions, which meet the conditions as set by the listening planes and the sound spots. The algorithm will select a group of solutions, which match the conditions as close as possible. A user may influence the selection of the solutions using a slider and a sliding bar **628,630** in which one or more sound spots and/or listening planes may be more prioritized over other sound spots and/or listening plances. Hence, when moving the slider **628** more towards the "silent zones", the algorithm may prioritize solutions in which the silent spots are matched best.

[0053] **Fig. 7** depicts a GUI for building a 3D venue space according to yet a further embodiment of the invention. In this particular embodiment, a user may define speaker envelopes by selecting the envelope button **702.** A speaker envelope defines a volume in the virtual 3D venue space in which it is desired to position the speakers. This button may open a menu for creating speaker envelopes **704**$_{1-2}$. The menu may comprise first envelope creating button **718** for the creation of an envelope centered in the middle (under) the stage and second envelope creating button **720** for creation of two (symmetrical) envelopes which may be positioned left and right from the stage. After creation of a speaker envelope, the GUI may allow a user to drag the envelope to a desired location. Further, the size of the envelopes may be manipulated on the basis of manipulation markers **708** in a similar way as described above with respect to the sound spots and listening planes. During manipulation, the location **728,** dimensions **726** and angular orientation **724** of a speaker may be displayed in a speaker label **722.** A speaker envelope may comprise one or more speakers up to a predetermined maximum **714.** Further, different types of speakers may be selected. Preferably, a user may select predetermined speakers from a speaker library, wherein each speaker may be associated with a predetermined speaker directivity function.

[0054] **Fig. 8** depicts a GUI for building a 3D venue space according to an embodiment of the invention. This embodiment depicts a more complex speaker envelope configuration comprising a speaker envelope **800** associated with speakers **802**$_{1-3}$**,** e.g. sub-woofers, which are positioned under the stage, speaker envelopes **804**$_{1-2}$ for speakers **806**$_{1-2}$ which are positioned next to the stage and speaker envelopes **808**$_{1-2}$ for speakers **8010**$_{1-2}$ which are positioned above the stage. This figure illustrates that the GUI may allow a user to easy model a real-life venue comprising a complex speaker configuration.

[0055] Hence, **Fig. 3- 8** illustrate that the GUI functions of the 3D venue editor allows a user, a layman in acoustics, to build complex venues and speaker configurations in a simple and intuitive way. Once the 3D model is fully defined, the acoustic simulation engine may execute a simulation algorithm, in particular an evolutionary algorithm, wherein the 3D model defines the boundaries and conditions in which an optimization program may calculate the best solution for the 3D model. During the execution of the 3D venue editor, the algorithm may initialize and define the parameters with are used by the evolutionary algorithm. During the executing of the 3D venue editor, the user may define the number of speakers and assign an initial location to the speakers. The algorithm may use this information in order to reserve memory space for defining a predetermined number of speakers in terms of speaker positions and filter coefficients.

[0056] **Fig. 9** schematically depicts a flow diagram of a computer-implemented acoustic simulation method according one embodiment of the invention. In a first step **902,** a random initial population of virtual speaker set-ups is generated wherein each speaker set-up represents a predetermined set-up of speakers in the virtual space. Each speaker in a virtual speaker set-up is associated with a speaker location and a predetermined audio filter. The audio filter may be defined by a predetermined number of filter coefficients. Thereafter, the population is optimized on the basis of an evolutionary algorithm, preferably a differential evolutionary algorithm (step **904).** The evolutionary algorithm will optimize the population of speaker set-ups by forming one or more new speaker set-ups on the basis of existing speaker set-ups in the population and by executing a cost function (sometimes also referred to as a fitness function), which assigns a penalty or a score to the new speaker setting (step **906).** The cost function may assign a penalty or a score for differences between calculated and targeted sound characteristics, such as sound pressure, in sound spots, wherein the calculated sound characteristics are calculated on the basis of the new speaker setting. Similarly, a penalty or score may be assigned to differences between calculated speaker locations, which deviate from the user specified and common-sense constraints, such as speaker locations outside the speaker envelopes. The assigned penalties and scores may be added to a total cost value, which is then used to evaluate a new speaker set up with respect to an existing speaker-set up in the population. Better scoring solutions (i.e. low penalties, high score) are replaced with earlier solutions in the population. This optimization process may be repeated until a predetermined condition is met (e.g. max number of iterations or optimized solutions match the desired 3D sound distribution within a certain accuracy.

[0057] Then, one or more optimized speaker set- ups may be selected (step **908**) . These one or more selected optimized speaker set- ups include filter coefficients, which may be sent to the (real- life) electronically configurable speakers (step **910**) . The filter coefficients may be used to configure a digital audio filter in each of the speakers such that the speakers produce a desired sound pressure distribution. Filter coefficients may be determined on the basis of known filter algorithms. For example, in one embodiment, a filter coefficients may be generated on the basis of a digital FIR filter algorithm as described in the article by Karam et. al., "Design of optimal digital FIR filters with arbitrary magnitude and phase responses", ISCAS 1996. This particular algorithm may allow design of optimal Chebyshev digital FIR filters with arbitrary magnitude and phase specifications and is thus particular suited for use with the acoustical optimization method as described with reference to **Fig. 9.**

[0058] The computer-implemented acoustic simulation method will be discussed hereunder in more detail on the basis of parts of pseudo-codes of the evolutionary algorithm. The algorithm may start with the initialization of some important simulation parameters including, the frequency bands in the sound spectrum for which the sound pressure distributions are determined, the maximum number of speakers used during the simulation and the positions and dimensions of the speaker envelopes.

[0059] In some embodiments, a directionality function may be associated with one or more of the speakers. A directionality function may be associated with a particular speaker design. The directionality function defines the speaker directivity, i.e. a speaker characteristic determining in which directions sound pressure waves of a particular frequency are transmitted.

[0060] The cost function in the algorithm revolves around a so-called Transfer Function. This is a mathematical function, which takes the speaker positions, their associated filter settings (coefficients) and a position of interest in the 3D venue space as input and returns a sound pressure integrated over the sound spectrum. This calculated value is then compared with the desired sound pressure (integrated over the sound spectrum). The directionality function of a speaker may be part of this Transfer Function, as is any diffusion. Basically the Transfer Function describes anything between the electrical current into the speaker and the pressure wave at the spatial point of interest. This includes behavior of a speaker, the speaker cabinet, speaker directionality, the medium through which sound pressure waves propagate (e.g. air, which can be modeled as being linear or non-linear diffusive medium), reflections and/or interferences.

```
// ------------------------------------------
// >> INIT 1 - VENUE SETUP
// ------------------------------------------
// define central frequencies of interest

frequencies = new Array(30, 50, 80); // Hz
hardware = 8 delays //in ms, phase delay

// define speakers 1-n with position(x, y, z) and set-
tings (frequency, phase delay, amplitude)

Speakers = new Array(new Array());

Speakers[0] = new Array(0,0,0);
Speakers[1] = new Array(0,0,0);

for (i=0 ; i<Speakers.length ; i++) {
    Speakers[i][4] = new Array();
    for (j=0 ; j<Frequencies.length ; j++) {
        Speakers[i][4].push(0); // amplitude
        Speakers[i][4].push(0); // phase delay
    }
}

// define directionality function

var directionality_function = 1
```

[0061] Further, during the execution of the 3D venue editor, speaker envelopes may be defined in the 3D venue space. Defining speaker envelopes using the GUI of the 3D venue editor is described in detail with reference to **Fig. 7** and **8.** When 3D sound pressure distributions are calculated for different speaker locations, solutions associated with speakers positioned in the speaker envelopes are strongly preferred over those solutions where speakers are outside the envelopes. A speaker envelope may comprise one speaker or, alternatively, it may comprise several speakers. To that end, speaker dimensions may be defined so that an initial distribution of the speakers over the envelopes may be calculated.

```
// define speaker dimensions (w, h, d)
 Speaker_dimensions = (1, 1, 1) ;
//define envelopes 1- n with bounds (x1, x2, y1, y2, z1, z2)
//and volume, to be used as a scaling factor for drawing
//random speaker positions in the case of multiple envelopes
 Envelopes = new Array(new Array() ) ;
// i.e.
 Envelopes[0]- new Array(0, 20, 0, 20, 0, 10) ;
```

[0062] Predetermined desired sound pressures may be set by defining predetermined volumes in the venue space and assigning predetermined one or more sound pressure values to these volumes and/or planes. As already described with reference to **Fig. 5** and **6,** so- called sound spots may be defined in the 3D venue space, which are associated with a particular sound pressure, e.g. in one embodiment sound pressures targets may be selected from a range between 0 and approx. 110 dB. In an embodiment, a sound spot may be used to define a "silent region" i.e. a region in which the sound pressure stays below a predetermined maximum value.

```
// define sound spots 1- n with position(x, y, z) radius and
sound targets 1- n (frequency, amplitude) )
var radius, targets;
Soundspots = new Array(new Array() ) ;
// i.e.
Soundspots[0]- new Array(5, 5, 5, 1, new Array() ) ;
// targets: silence at all frequencies
for (i=0 ; i<Soundspots.length ; i++) {
 for (j=0 ; j<Frequencies.length ; j++) {
 Soundspots[i] [4] [Frequencies[j] ] = 0;
  }
 }
```

[0063] Similarly, a listening plane, i.e. a sound spot with an audible sound pressure amplitude at listening height, may be defined in the venue space. A desired sound pressure may be assigned to this plane. During the simulation, the algorithm will seek for 3D sound pressure distributions for which the calculated sound pressure at the locations of sound spots and/or listening planes closely match the desired sound targets.

[0064] Thereafter, each sound spot may be divided in a grid of cubes each having a predetermined position. For each speaker configuration, the algorithm will calculate sound pressures for each cube and compare the calculated maximum and minimum sound pressure values with the sound pressure values associated with the sound spot. The grid size may determine the accuracy of the algorithm. The smaller the grid size, the more accurate a solution may be determined. Typically, the grid size will be a compromise between desired accuracy and computational costs.

```
// ----------------------------------------
// >>  INIT 2 - SOUND SPOTS
// ----------------------------------------
```

```
        gridsize = 1;

// define individual grid cubes as a list of positions (x,y,z)

        gridcubes = new Array(new Array());

// loop over the sound spots

        for (i=0 ; i<soundspots.length ; i++) {
            radius = soundspots[i][2];
            for (a=0 ; a<radius ; a+=gridsize) {
                for (b=0 ; b<radius ; b+=gridsize) {
                    for (c=0 ; c<radius ; c+=gridsize) {
                        gridcubes.push(new Array(a,b,c, new Ar-
ray()));

// loop over de targets of this sound spot and
// copy the targets to the grid cube

                        for (target in Soundspots[i][4]) {

                            gridcubes[gridcubes.length-
1][3][target] = Soundspots[i][4][target];
                        }

// if so desired, store an 'importance' factor

                        gridcubes[gridcubes.length-1].push(1);
                    }
                }
            }
        }
```

[0065] An importance factor may be assigned to a sound spot or a listening plane. The importance factor may be used by the simulation algorithm in order to make a selection between two or more "conflicting" solutions, e.g. first speaker settings which produce a sound pressure distribution closely matching the conditions of a first sound spot and second speaker settings which procedure a sound pressure distribution closely matching a second sound spot. When building a 3D venue, the 3D venue editor may allow a user to assign an importance factor to one or more sound spots so that the sound spots are prioritized allowing the algorithm to select a particular solution which closely matches the conditions of a sound spot with a high importance factor.

[0066] After initialization of the venue, an evolutionary algorithm may be initialized. In one embodiment, a differential evolution algorithm may be used in order to calculate solutions, i.e. one or more speaker set-ups, each being associated with a 3D sound pressure distribution, which closely match a desired sound pressure distribution as defined by the sound spots and/or listening planes, and to determine the speaker locations and audio filters associated with these solutions.

[0067] For a predetermined number of frequency bands within the acoustic spectrum, approx. 20 Hz - 20 kHz, the algorithm will seek for sound pressure distributions within the virtual 3D venue space, which match the conditions as set by the user. The venue may be modeled by elements defining a bounded 3D space, e.g. floor(s), walls, ceiling(s) etc., and objects within that space, e.g. a stage or tribunes, wherein acoustic properties such as absorption and/or reflection coefficients may be assigned to these elements and objects.

[0068] Each sound pressure distribution represents a solution of a number of interfering sound pressure waves generated by a speaker set-up in the venue space. Here, a speaker set-up is associated with predetermined a number of speakers, each being associated with a speaker position and a (digital) audio filter. A audio filter may include phase delay and/or amplitude information.

[0069] Sound pressure waves are generated by the speakers, travelling through the medium, typically air, of the 3D venue space and reflected and/or diffracted via the walls, floors, ceiling and/or objects in the virtual venue space wherein reflection and diffraction is determined by the acoustic properties of the walls, floor, ceilings and/or objects and the medium. Through constructive and destructive interference of travelling and reflecting pressure waves a complex 3D sound pressure distribution in the virtual 3D venue space will be formed. In certain areas constructive interference will result in high sound pressure areas and in other areas destructive interference will result in low sound pressure areas. The amount, size and locations of these high and low sound pressure areas in the venue space depend on the particular speaker set-up.

[0070] Finding solutions, i.e. virtual speaker set-ups associated with desired 3D sound pressure distributions, is a very complex problem. The evolutionary algorithm will look for solutions by evaluating the evolution of a population of virtual speakers set-ups on the basis of a cost function. In particular, the evolutionary algorithm uses this cost function to optimize the problem of finding specific solutions, i.e. speaker set-ups, which are associated with 3D sound pressure distributions which match the conditions set for a predetermined virtual 3D venue as close as possible. The algorithm may execute the evolution of a population of candidate solutions, i.e. candidate speaker set-ups, by generating predetermined changes (mutations) in certain speaker set-ups and creating new set-ups (mutants) on the basis of existing speaker set-ups and the mutated speaker set-up according to a predetermined mutation/recombination scheme. A score or fitness may be associated with the existing and new speaker set-up and those having the best score or fitness are selected and kept. The evolutionary process of mutations, recombination and selection is iteratively repeated until the solutions are evolved within a predetermined number of recombinations (iterations) into solutions, which closely match desired conditions in terms of sound spots and/or listening planes.

[0071] During the initialization phase of the algorithm, a first initial random population comprising a number of NP solutions, i.e. speaker set- ups, may be created wherein the parameter NP determines the size of the population. The parameters defining a particular speaker set- up, i.e. the locations of the speakers and the phase delay information and the amplitude information per frequency band for each speaker may be represented in the form of a vector m. In general, m may have n parameters ($m_1, ..., m_n$) wherein n represents the dimensionality of the acoustic problem to be optimized. This way, vector m^1 defines the speaker set- up i (i=1, ..., NP) in a population P comprising NP different speaker set- ups.

[0072] The initial speaker locations may be selected on the basis of the speaker envelopes and the (maximum allowable) speakers in a speaker envelope such that the evolutionary process starts with a population of speaker set-ups having speaker locations somewhere within the envelopes. Further, during the initialization phase, the maximum number of recombination cycles gen_max may be determined.

```
// -------------------------------------------
// >> INIT 3 - DIFFERENTIAL EVOLUTION
// -------------------------------------------

// maximum nr of generations

        gen_max = 1000;

// set population size
// this can be viewed as the number of virtual speaker setups

        NP = 100;

// define x which will hold the population of speaker setups

        x = new Array(NP);

// set dimensionality (x, y, z + frequencies * ( phase, ampli-
tude ))

        D = 3 + Frequencies.length * 2;

// randomly generate first population and store linearized in x

    // calculate the nr of speakers per envelope

        Speakers_per_Envelope = new Array();
        var positioned_Speakers = 0;
        for (i=0 ; i<Envelopes.length ; i++) {
            Speakers_per_Envelope[i] = Math.round(NP * (Enve-
lopes[6] / Envelopes_TV) );
            positioned_Speakers += Speakers_per_Envelope[i];

        }

    // compensate for rounding errors
```

```
        Speakers_per_Envelope[Speakers_per_Envelope.length] += NP
    - positioned_Speakers;

    // draw speaker positions from the assigned envelope space

        for (i=0 ; i<Speakers_per_Envelope.length ; i++) {
            for (j=0 ; j<Speakers_per_Envelope[i] ; j++) {
                x = Math.Min(Envelopes[i][1], Envelopes[i][0]) +

                    Math.Rand() * (Math.Abs(Envelopes[i][1] -
    Envelopes[i][0]) - Speaker_dimensions[0]);

                y = Math.Min(Envelopes[i][3], Envelopes[i][2]) +

                    Math.Rand() * (Math.Abs(Envelopes[i][3] -
    Envelopes[i][2]) - Speaker_dimensions[1]);

                z = Math.Min(Envelopes[i][5], Envelopes[i][4]) +

                    Math.Rand() * (Math.Abs(Envelopes[i][5] -
    Envelopes[i][4]) - Speaker_dimensions[2]);

                x[i].push(x,y,z);

                for (k=0 ; k<(Frequencies.length * 2) ; k++) {
                    x[i].push(0,0); // phase delay, amplitude
                }
            }
        }
```

[0073] In one embodiment, the predetermined changes (mutations) in speaker set-ups and the creation of new speaker set-ups on the basis of existing speaker set-ups and the mutated speaker set-ups (mutants) is implemented using a differential evolutionary (DE) algorithm. The DE algorithm uses mutations, which are generated on the basis of differences between randomly selected solutions in the population.

[0074] In particular, the DE algorithm may randomly select three vectors m^a, m^b and m^c from the speaker settings population P, select an index $R \in [1,...,n]$ wherein n is the dimensionality of the problem and define a donor vector $d^i = m^a + F^* (m^b + m^c)$, wherein F is a weighting factor.

[0075] A new target vector $p^i = (p_1, ..., p_n)$, a recombination, may be determined by determining for each k=1, ..., n: select a uniformly distributed number $r_k \in (0, 1)$; if $r_k < CR$ or $k=R$ then $p^i_k = m^a_k + F^* (m^b_k + m^c_k)$ else $p^i_k = m^i_k$; if $F_{cast} (p^i) < F_{cast} (m^i)$ then m^i is replaced by p^i in the population P. This process is repeated for each vector in the population until one or more vectors in the population generate a sound pressure distribution which matches a desired sound pressure distribution (in terms of sound spots and/or listening planes) within a certain acceptance range or until a maximum number of recombinations (iterations) is executed.

[0076] Here, F_{cost} represents a cost function, which is configured to evaluate a vector to what extend the sound pressure distribution matches or differs from the desired sound pressure distribution as defined by the sound spots and/or listening planes and to what extend the speaker locations associated with a vector are within the desired speaker envelopes.

[0077] The cost function is configured to calculate sound pressure distributions, in particular sound pressure in the sound spots and/or listening planes on the basis of a vector in the population. When calculating the sound pressures, the function takes into account as much factors as possible that significantly influence the sound pressure, including the location of the speakers, the direction in which the speakers transmit the sound pressure waves into the venue space and/or the sound pressure waves, which are reflected and/or diffracted by objects in said venue space and/or boundaries of said venue space. Thereafter, the cost function assigns a cost score to the vector.

[0078] The cost function is further configured to compare calculated sound pressures and desired sound pressure targets in the sound spot and to assign penalties or a score for differences and matches respectively between the calculated and desired (targeted) sound pressures in a sound spot and for maxima in the frequency domain. Further, it may assign an additional penalty for speaker locations, which do not conform to the constraints as defined by the user, e.g. speaker positions which lie (partially) outside the speaker envelopes. In an embodiment, a score may also be assigned to the number of speakers used and/or for the filter length in order to minimize both and push the optimization towards more economically attractive solutions for the end-user.

```
// define a trial vector
        m = new Array(D);

// define a weighing factor
        F = 1;

// set crossover true / false
        crossover = true;
        CR = 0.3; // crossover probability

// set current generation

        gen = 0;

// -------------------------------------------
// ------------- optimization algorithm  ----
// -------------------------------------------

// perform differential evolution
// for max_gen generations

        while (gen < max_gen) {
            gen++;

// loop over the population

            for (i = 0; i < NP; i++) {

// draw three different vectors from the population

                a = b = c = gen;
                while (a == gen) { a = Math.floor( Math.random()
* NP ); }
                while (b == gen || b == a) { b = Math.floor( Ma-
th.random() * NP ); }
                while (c == gen || c == a || c == b) { c = Ma-
th.floor( Math.random() * NP ); }

// generate a trial vector based upon them
// including crossing over if so required
```

```
                    for (j=0 ; j<D ; j++) {
                        if (crossover && Math.random() > CR) {
                            m[j] = x[i][j];
                        } else {
                            m[j] = x[a][j] + F * (x[b][j] -
x[c][j]);
                        }
                    }

// compare the trial vector to the original
// and replace it if so required

                    if (cost(m) [trail vector] < cost(x[i])) { //
would be faster to store calculated costs...

                        for (j=0 ; j<D ; j++) {
                            x[i][j] = m[j];
                        }
                    } else {
                        //
                    }
                }
            }

// ------------------------------------------
// -------------- cost function  ----------
// ------------------------------------------

            function cost(vector) {

                var cost_score = 0;
                var ci, cj;
                var distance;
                var soundpressure;
                var angular_momentum, speaker_phase_delay, spea-
ker_amplitude, frequency;
                var upper, lower;

// loop over the gridcubes and
// calculate the sound pressure in each of them

                for (ci=0 ; ci<gridcubes.length ; ci++) {
                    soundpressure = 0;

// loop over the vector (= speaker setup)

                    for (cj=0 ; cj<vector.length ; cj+=D) {
                        speaker_x = vector[cj];
                        speaker_y = vector[cj+1];
                        speaker_z = vector[cj+2];
                        distance = Math.Pow((gridcube[i][0]-
speaker_x)*(gridcube[i][0]-speaker_x) + (gridcube[i][1]-
```

```
speaker_x)*(gridcube[i][1]-speaker_x) + (gridcube[i][2]-
speaker_x)*(gridcube[i][2]-speaker_z),0.5);

// loop over the 'settings' per speaker

                    for (ck=0 ; ck<Frequencies.length; ck++) {

                        frequency = Frequencies[ck];
                        speaker_phase_delay = vector[cj+2*ck];
                        speaker_amplitude = vector[cj+2*ck+1];

                        angular_momentum = 2 * Math.PI * fre-
quency;

                        upper = Mathf.Cos(speaker_phase_delay -
angular_momentum * distance_to_speaker );

                        lower = Mathf.Sin(speaker_phase_delay -
angular_momentum * distance_to_speaker );

                        soundpressure +=
eval(directionality_function) * speaker_amplitude / dis-
tance_to_speaker * ( upper / lower );

                    }
                }

                soundpressure = Math.Pow( Ma-
th.Abs(soundpressure), 2);

// if we only want silent spots...

            cost_score += soundpressure;

            }
            return cost_score;
        }
```

[0079] Once an optimized population of speaker set-ups is generated, a speaker set-up may be evaluated graphically as shown in **Fig. 10** showing a simple virtual space similar to the one depicted in **Fig. 6** comprising a simple stage and two speaker modules located next to the stage. The program may illustrate the sound pressure distribution **1002** at certain listening planes for a predetermined frequency **1004** (in this case 63 Hz). A user may browse through the audio frequencies using a selection tool, e.g. a mouse or touch screen button, which may be manipulated by sliding the area in which the frequency is displayed up or down, so that sound pressure distributions associated with a certain audio frequency band may be graphically displayed.

[0080] Further, the program may allow to graphically visualizing at least part of an optimized speaker setting. For example, in **Fig. 11** the amplitude settings for a configured virtual speaker are shown wherein e.g. in a certain frequency range the amplitude of the signal is amplified (amplitude coefficients **1002)** and in other ranges the amplitudes may be attenuated (amplitude coefficients **1004) .** A user may manipulate the amplitude settings by frequency rules **1108- 1112** and amplitude rules **1106.**

[0081] One embodiment of the disclosure may be implemented as a program product for use with a computer system. The program(s) of the program product define functions of the embodiments (including the methods described herein) and can be contained on a variety of computer-readable storage media. The computer-readable storage media can be a non-transitory storage medium. Illustrative computer-readable storage media include, but are not limited to: (i) non-writable storage media (e.g., read-only memory devices within a computer such as CD-ROM disks readable by a CD-ROM drive, ROM chips or any type of solid-state nonvolatile semiconductor memory) on which information is permanently stored; and (ii) writable storage media (e.g., floppy disks within a diskette drive or hard-disk drive or any type of solid-state random-access semiconductor memory, flash memory) on which alterable information is stored.

[0082] It is to be understood that any feature described in relation to any one embodiment may be used alone, or in combination with other features described, and may also be used in combination with one or more features of any other of the embodiments, or any combination of any other of the embodiments. Furthermore, equivalents and modifications not described above may also be employed without departing from the scope of the invention, which is defined in the accompanying claims.

Claims

1. Computer-implemented acoustic simulation method for generating one or more speaker set-ups for electronically configuring speakers in an acoustical system such that, when configured, said speakers produce a desired sound pressure distribution in an acoustical space of a venue:

providing an at least a partially bounded virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue;
determining one or more sound spots by selection of one or more volumes at one or more sound spot locations in said virtual space and assigning said one or more sound pressure values to said one or more volumes;
determining at least part of said speaker set-ups settings by executing an evolutionary algorithm, preferably a differential evolutionary algorithm, said algorithm comprising:

generating a population of speaker set-ups, preferably a speaker set-up including at least one of: speaker location information, speaker phase delay information and/or speaker amplitude information; and,
optimizing said population of speaker set-ups on the basis of said one or more sound pressure values associated with said one or more sound spots.

2. Method according to claim 1 wherein said algorithm further comprises:

creating a further speaker set-up on the basis one or more speaker set-ups in said population; and,
optimizing said population of speaker settings by replacing a speaker set-up in said population with said further speaker set-up on the basis of a fitness function configured for comparing sound pressure values associated with said one or more sound spots with sound pressure values calculated on the basis of said further speaker set-up.

3. Method according to claims 1 and 2 wherein said algorithm further comprising:

determining one or more speaker envelopes by selecting one or more volumes in said displayed virtual space, a speaker envelope defining a volume for locating one or more configurable speakers; and,
generating said population of speaker settings on the basis of said one or more speaker envelopes.

4. Method according to claims 3 wherein defining said one or more speaker envelopes further includes:

selecting a predetermined number of speakers;
associating speaker locations to said predetermined number of speakers on the basis of said speaker envelopes.

5. Method according to claims 3 or 4 wherein optimizing said population of speaker set-ups further comprises:

checking whether the speaker locations associated with said further speaker setting are at least partly located within said one or more speaker envelopes.

6. Method according to any of claims 1-5 wherein determining one or more sound spots includes selecting said one or more volumes in said virtual space using a graphical user interface or selecting one or more sound spots from a database comprising a list of stored sound spots.

7. Method according to any of claims 3-5 wherein determining one or more speaker envelopes includes selecting said one or more volumes in said virtual space using a graphical user interface or selecting one or more speaker envelopes from a database comprising a list of stored sound spots.

8. Method according to claims 6 or 7, wherein, in case of using a graphical user interface, selecting said one or more sounds spots or one or more speaker envelopes further including:

rendering of a graphical selector, preferably a 3D wire-frame, in said displayed virtual space;
selecting a sound spot or speaker envelope location by locating said graphical selector in said virtual space;
selecting a sound spot or a speaker envelope volume by graphically modifying the 3D shape of said selector.

9. Method according to any of claims 1-8 further comprising:

selecting at least one speaker set-up from said optimized population of speaker set-up
sending said at least one speaker set-up, preferably said speaker comprising filter coefficients, to at least part of said electronically configurable speakers.

10. An acoustic simulation system for configuring one or more electronically configurable speakers in an acoustical system such that said configurable speakers produce a desired sound pressure distribution in an acoustical space of a venue, said acoustic simulation system being configured for:

providing an at least partially bounded virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue;
determining one or more sound spots by selecting one or more volumes at one or more sound spot locations in said virtual space and assigning said one or more sound pressure values to said one or more volumes;
determining at least part of said speaker set-ups settings by executing an evolutionary algorithm, preferably a differential evolutionary algorithm, said algorithm comprising:

generating a population of speaker set-ups, preferably said speaker set-ups including at least one of: speaker location information, speaker phase delay information and/or speaker amplitude information; and,
optimizing said population of speaker set-ups on the basis of said one or more sound pressure values associated with said one or more sound spots.

11. A configurable acoustical source comprising:

a digital signal processor configured for receiving at least one optimized speaker set-up from an acoustical simulation system according to claim 10, said digital signal process further being configured to configure at least part of a digital audio filter on the basis of filter coefficients in said speaker set-up; or,
a digital signal processor configured for executing a computer-implemented acoustic simulation method according to claims 1-9.

12. A graphical user interface for use in an acoustical simulation system according to claim 10, said graphical user interface being configured for graphically defining boundary conditions for said evolutionary algorithm, wherein said graphical user interface is configured for:

displaying a partially bounded 3D virtual space on a graphical display, wherein at least part of said virtual space substantially matches the acoustic space of said venue;
rendering of a 3D shaped graphical selector, preferably a 3D wire-frame, in said displayed virtual space, said 3D shaped graphical selector being associated with location coordinates in said virtual space and wherein said 3D shaped graphical selector comprising one or more shape manipulation markers wherein a user may interact with said 3D shaped graphical selector such that:

a sound spot location is selected by locating said graphical selector at predetermined location coordinates in said virtual space;
a sound spot volume is selected by graphically modifying the 3D shape of said selector so that it defines volume of a predetermined size in said virtual space.

13. A computer program product, implemented on computer-readable non-transitory storage medium, the computer program product configured for, when run on a computer, executing the method steps according to any of claims 1-9.

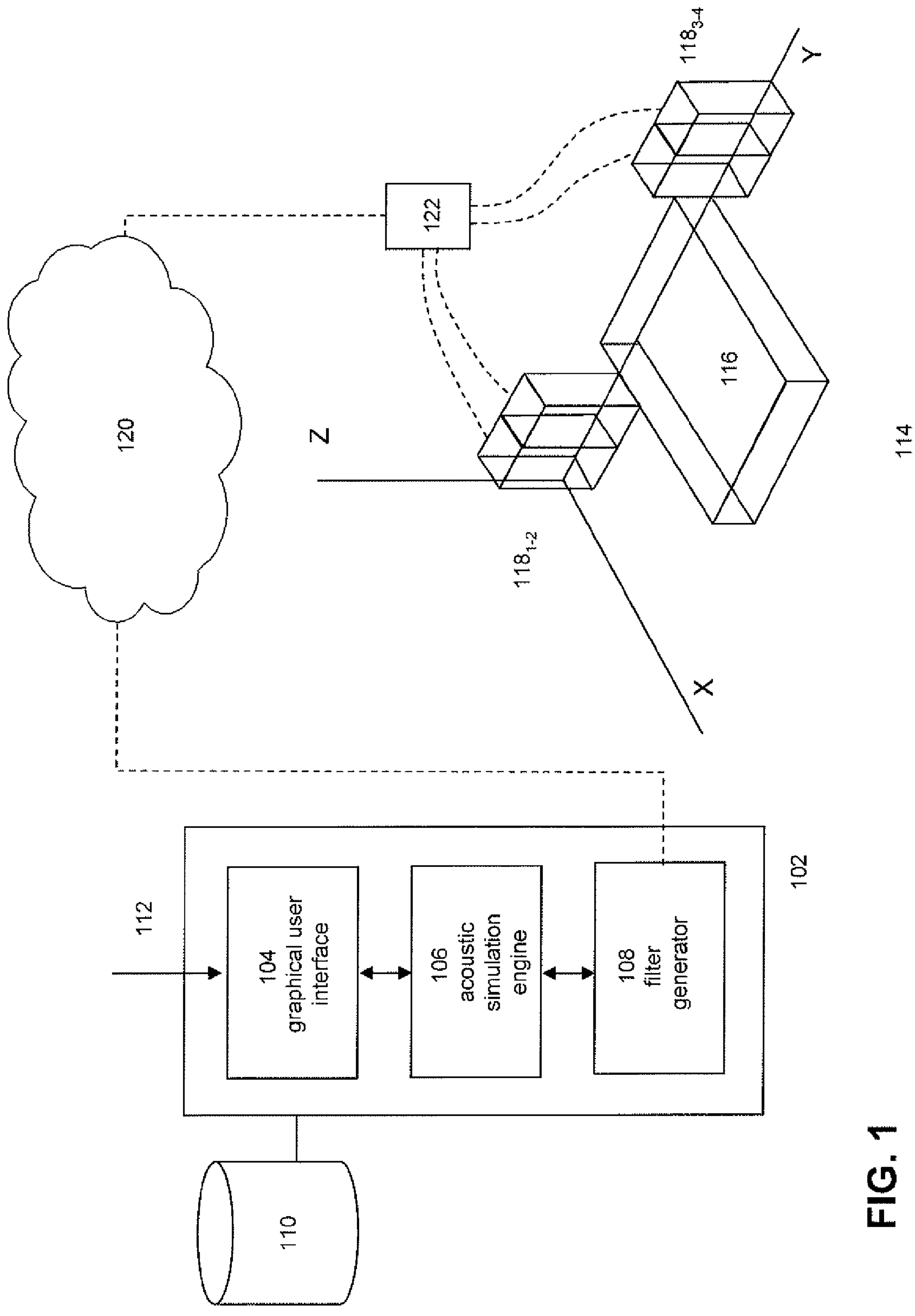
120
122
118_{3-4}
Y
Z
116
118_{1-2}
X
114
112
104
graphical user
interface
106
acoustic
simulation
engine
108
filter
generator
102
110

FIG. 1

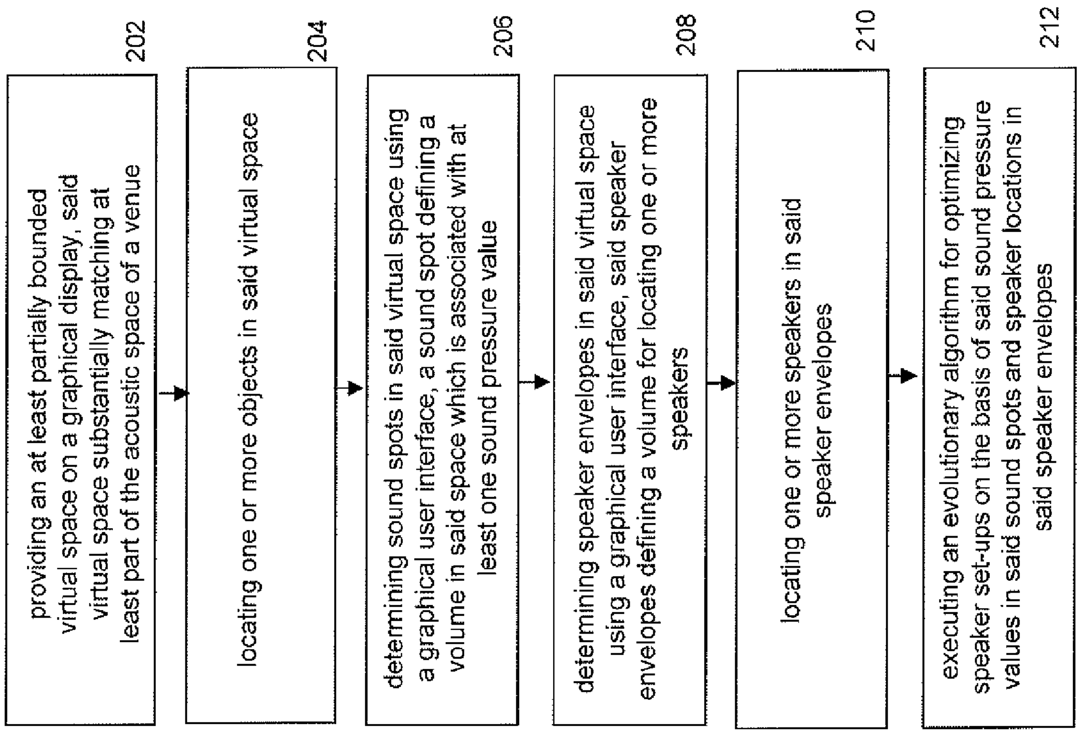
providing an at least partially bounded virtual space on a graphical display, said virtual space substantially matching at least part of the acoustic space of a venue
202
locating one or more objects in said virtual space
204
determining sound spots in said virtual space using a graphical user interface, a sound spot defining a volume in said space which is associated with at least one sound pressure value
206
determining speaker envelopes in said virtual space using a graphical user interface, said speaker envelopes defining a volume for locating one or more speakers
208
locating one or more speakers in said speaker envelopes
210
executing an evolutionary algorithm for optimizing speaker set-ups on the basis of said sound pressure values in said sound spots and speaker locations in said speaker envelopes
212

FIG. 2

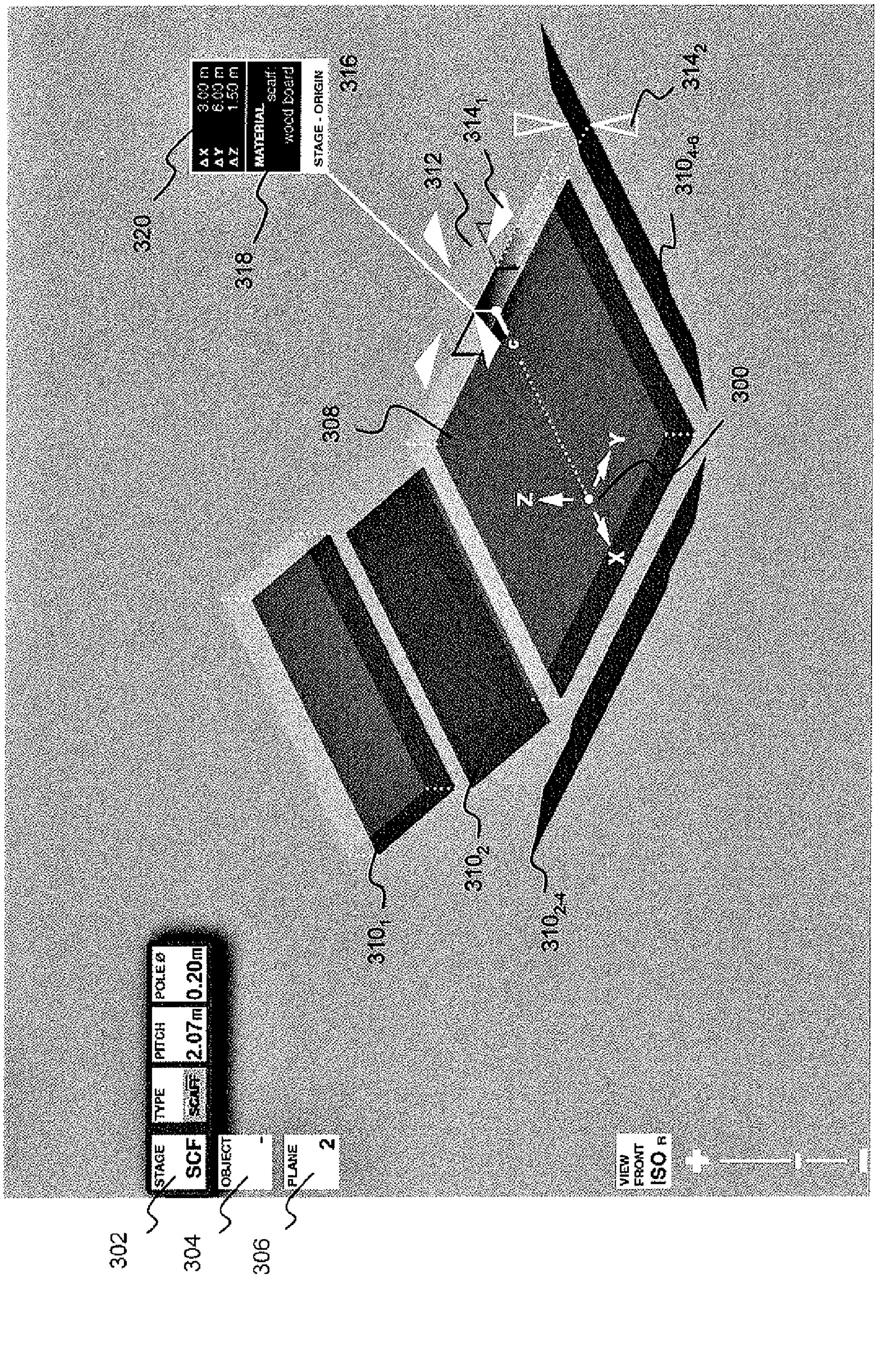
302
304
306
STAGE
SCF
TYPE
SCAFF
PITCH
2.07m
POLE Ø
0.20m
OBJECT
-
PLANE
2
VIEW
FRONT
ISO R
310_1
310_2
310_{2-4}
308
300
312
314_1
314_2
310_{4-6}
320
318
316
ΔX 3.00 m
ΔY 6.00 m
ΔZ 1.50 m
MATERIAL scaff. wood board
STAGE - ORIGIN
X
Y
Z

FIG. 3

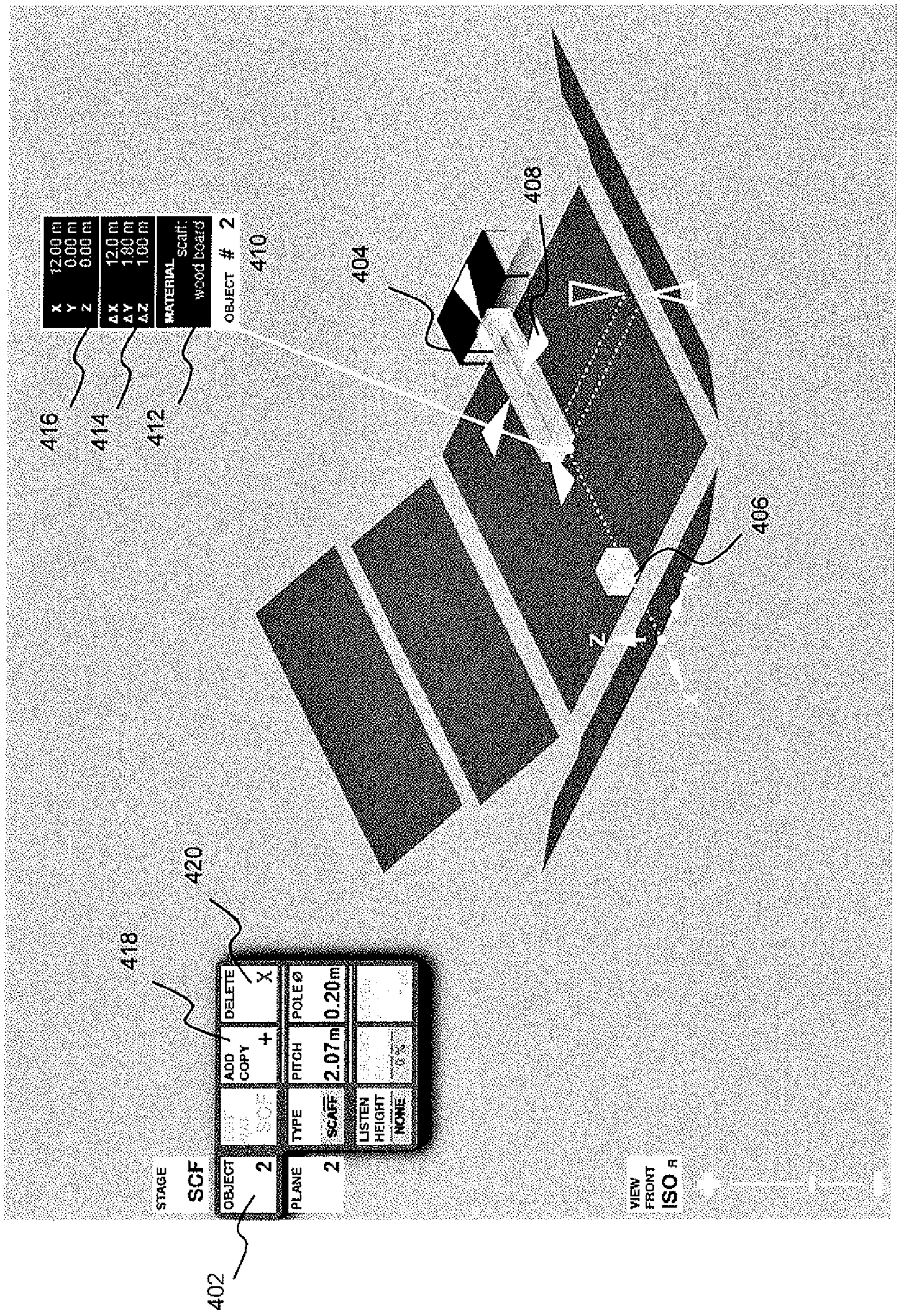
STAGE
SCF
OBJECT 2
PLANE 2
TYPE
SCAFF
PITCH
2.07m
POLE Ø
0.20m
ADD COPY +
DELETE X
LISTEN HEIGHT
NONE
VIEW
FRONT
ISO R
X 12.00 m
Y 0.00 m
Z 0.00 m
ΔX 12.0 m
ΔY 1.80 m
ΔZ 1.00 m
MATERIAL scaff: wood board
OBJECT # 2
402
404
406
408
410
412
414
416
418
420

FIG. 4

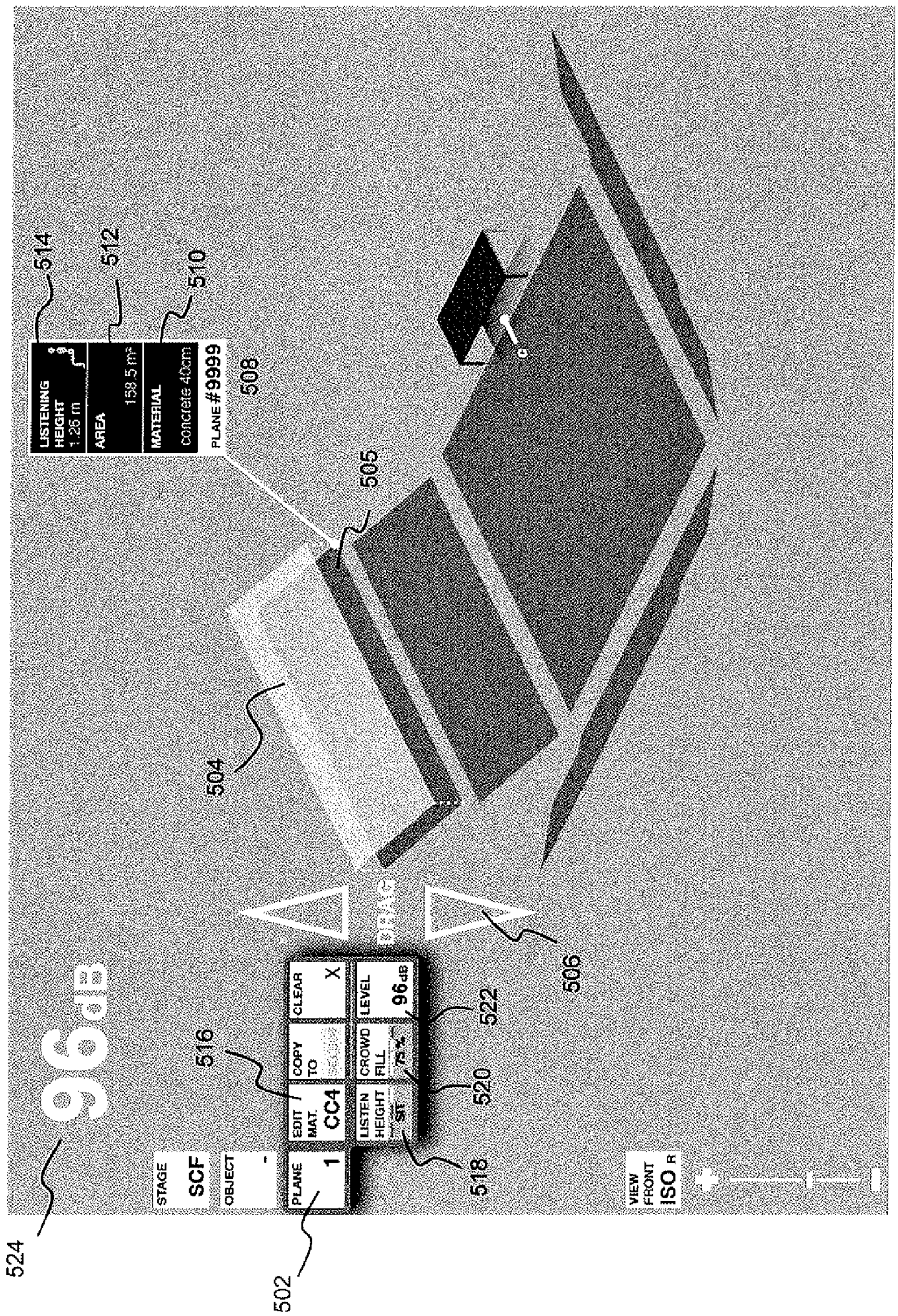
96 dB
STAGE
SCF
OBJECT
PLANE
1
EDIT MAT.
CC4
COPY TO
CLEAR
X
LISTEN HEIGHT
CROWD FILL
75%
LEVEL
96 dB
DRAG
VIEW FRONT
ISO R
LISTENING HEIGHT
1.26 m
AREA
158.5 m²
MATERIAL
concrete 40cm
PLANE #9999
502
504
505
506
508
510
512
514
516
518
520
522
524

FIG. 5

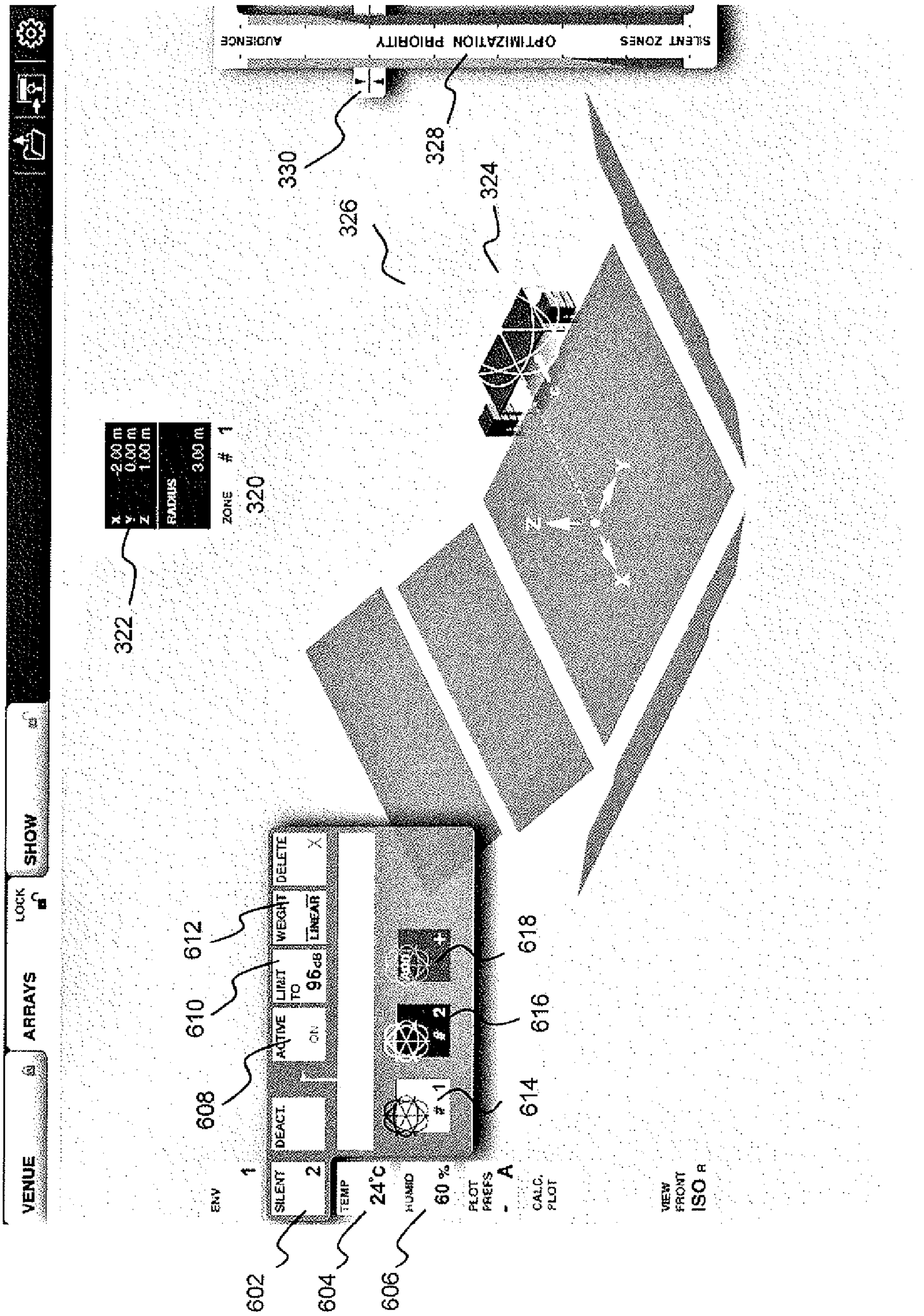
VENUE
ARRAYS
LOCK
SHOW
AUDIENCE
OPTIMIZATION PRIORITY
SILENT ZONES
330
328
326
324
322
X -2.00 m
Y 0.00 m
Z 1.00 m
RADIUS 3.00 m
ZONE # 1
320
ENV 1
SILENT 2
DEACT.
ACTIVE ON
LIMIT TO 96dB
WEIGHT LINEAR
DELETE
TEMP 24°C
HUMID 60 %
PLOT PREFS A
CALC. PLOT
VIEW FRONT ISO R
1
2
602
604
606
608
610
612
614
616
618

FIG. 6

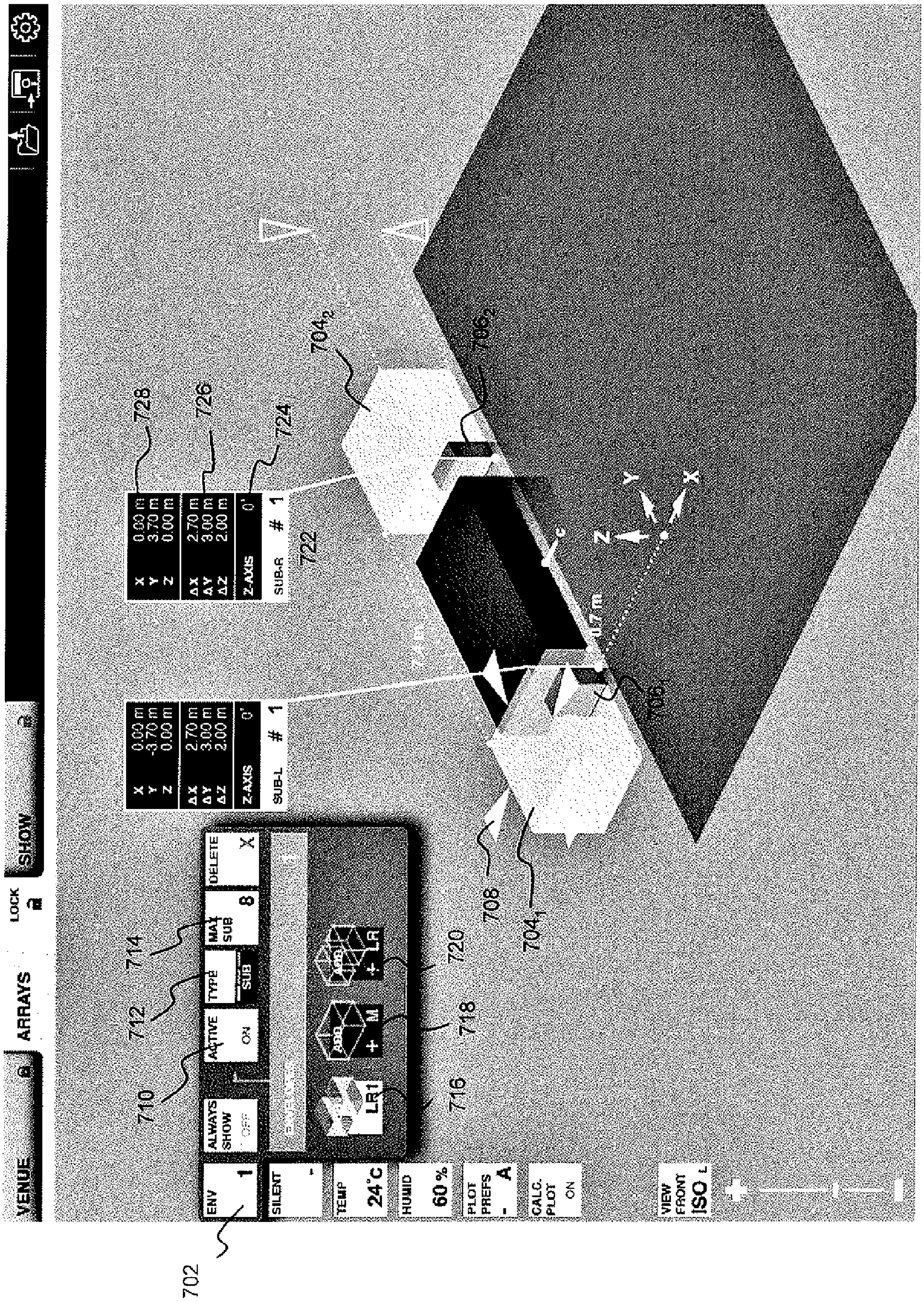
VENUE
ARRAYS
LOCK
SHOW
ENV
1
SILENT
TEMP
24°c
HUMID
60 %
PLOT PREFS
A
CALC. PLOT
ON
VIEW FRONT
ISO
ALWAYS SHOW
OFF
ACTIVE
ON
TYPE
SUB
MAX SUB
8
DELETE
X
LR1
M
LR
SUB-L # 1
SUB-R # 1
Z-AXIS 0°
X
Y
Z
702
704₁
704₂
706₂
708
710
712
714
716
718
720
722
724
726
728

FIG. 7

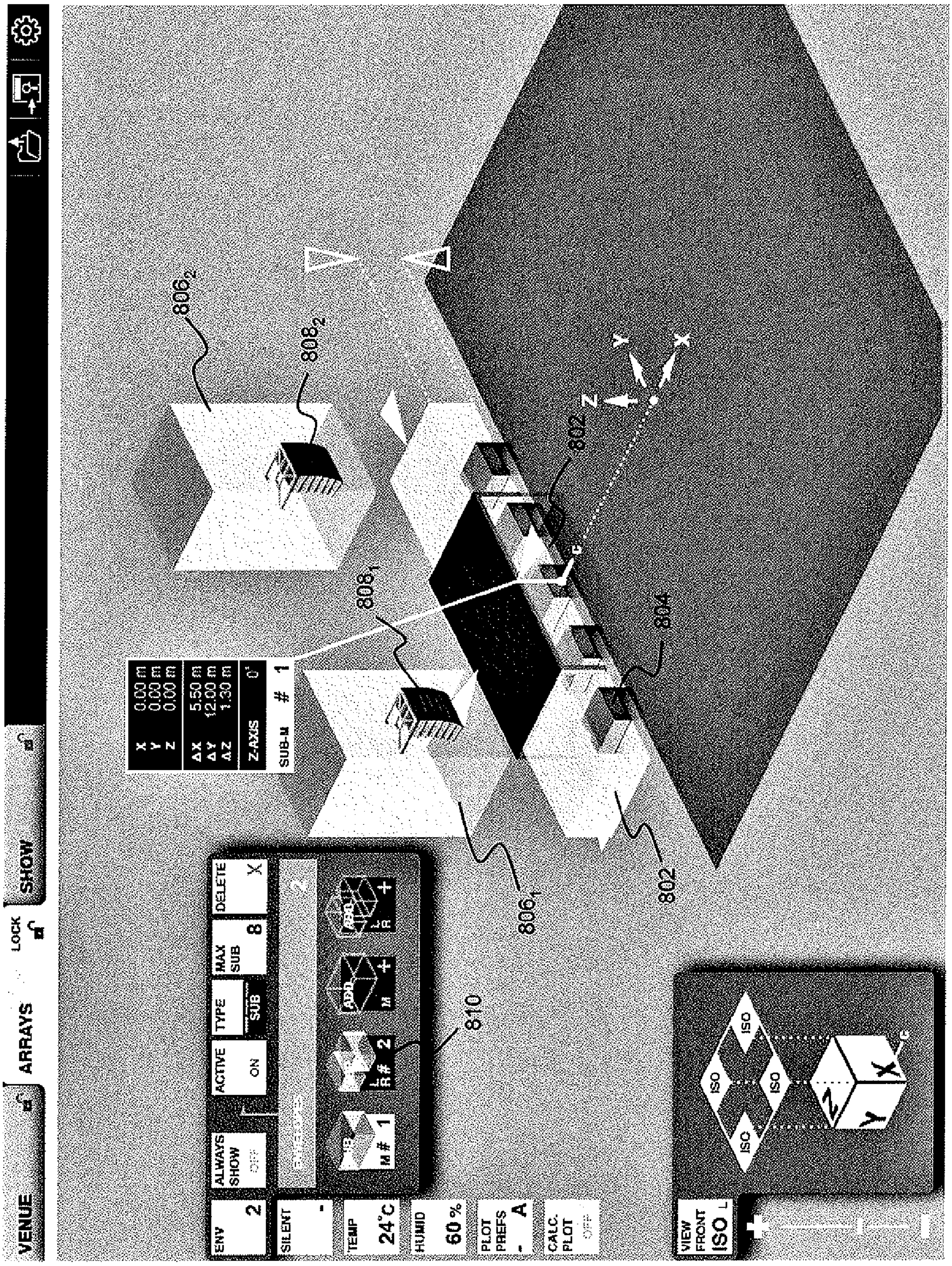
VENUE
ARRAYS
LOCK
SHOW
ENV
2
SILENT
TEMP
24°C
HUMID
60 %
PLOT
PREFS
A
CALC.
PLOT
OFF
ALWAYS
SHOW
OFF
ACTIVE
ON
TYPE
SUB
MAX
SUB
8
DELETE
X
M# 1
LR# 2
ADD
810
X 0.00 m
Y 0.00 m
Z 0.00 m
ΔX 5.50 m
ΔY 12.00 m
ΔZ 1.30 m
Z-AXIS 0°
SUB-M # 1
806₁
806₂
808₁
808₂
802
804
X
Y
Z
ISO
VIEW
FRONT
ISO

FIG. 8

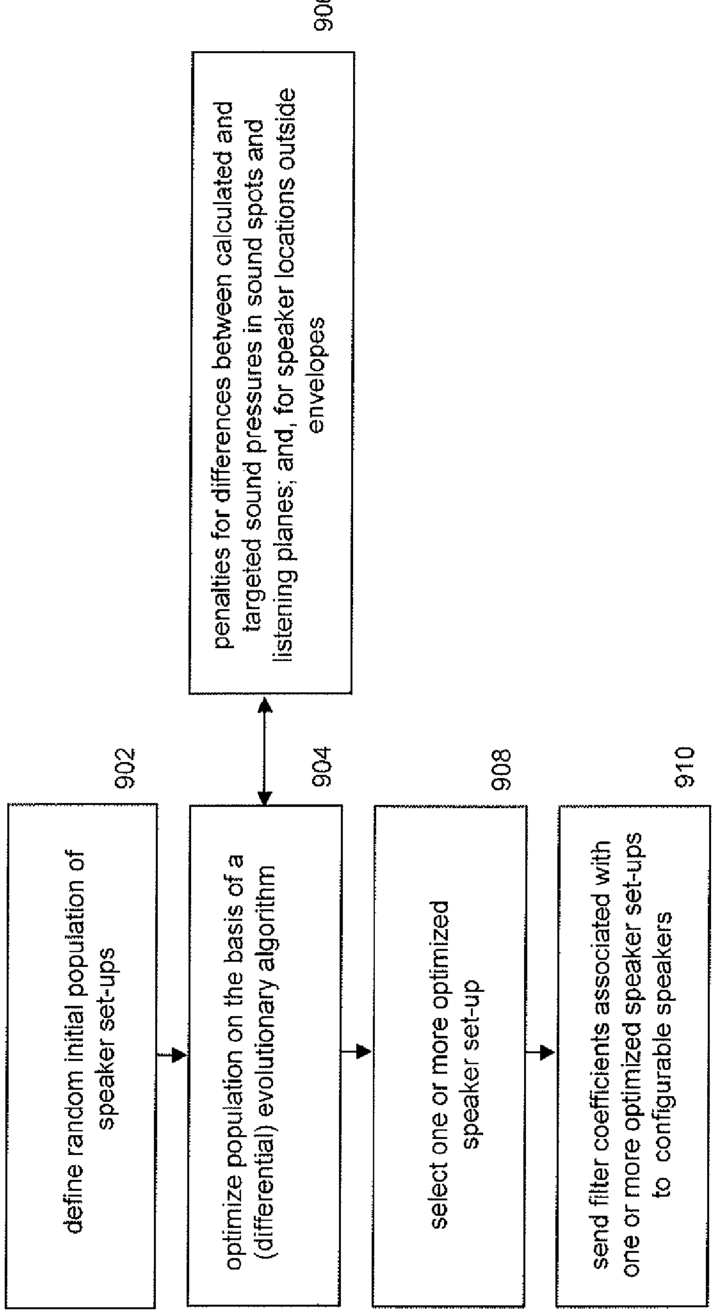
define random initial population of speaker set-ups
902
optimize population on the basis of a (differential) evolutionary algorithm
904
penalties for differences between calculated and targeted sound pressures in sound spots and listening planes; and, for speaker locations outside envelopes
906
select one or more optimized speaker set-up
908
send filter coefficients associated with one or more optimized speaker set-ups to configurable speakers
910

FIG. 9

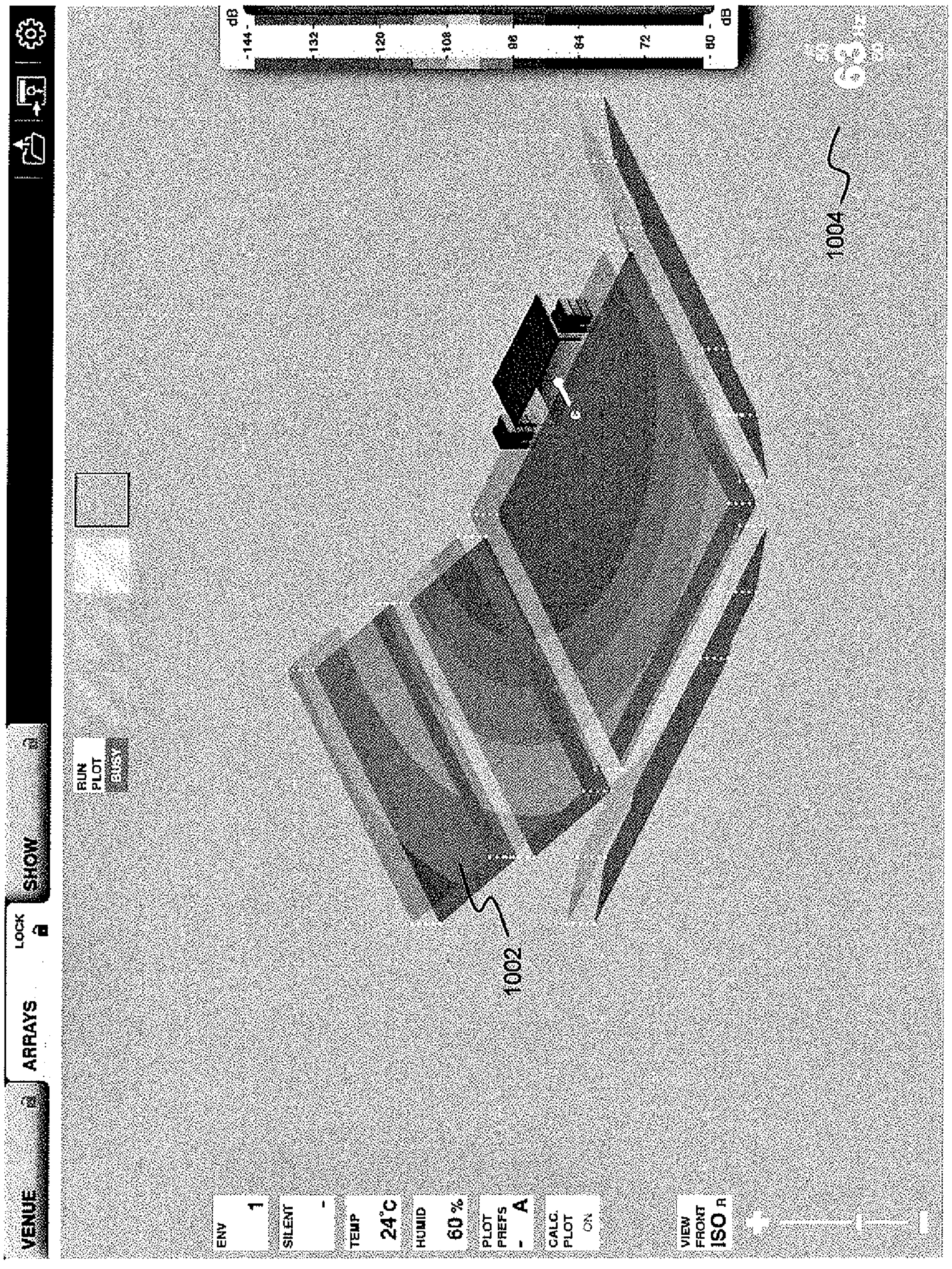
VENUE
ARRAYS
LOCK
SHOW
RUN
PLOT
BUSY
ENV
1
SILENT
-
TEMP
24°C
HUMID
60 %
PLOT PREFS
- A
CALC. PLOT
ON
VIEW
FRONT
ISO
dB
144
132
120
108
96
84
72
60
dB
1002
1004

FIG. 10

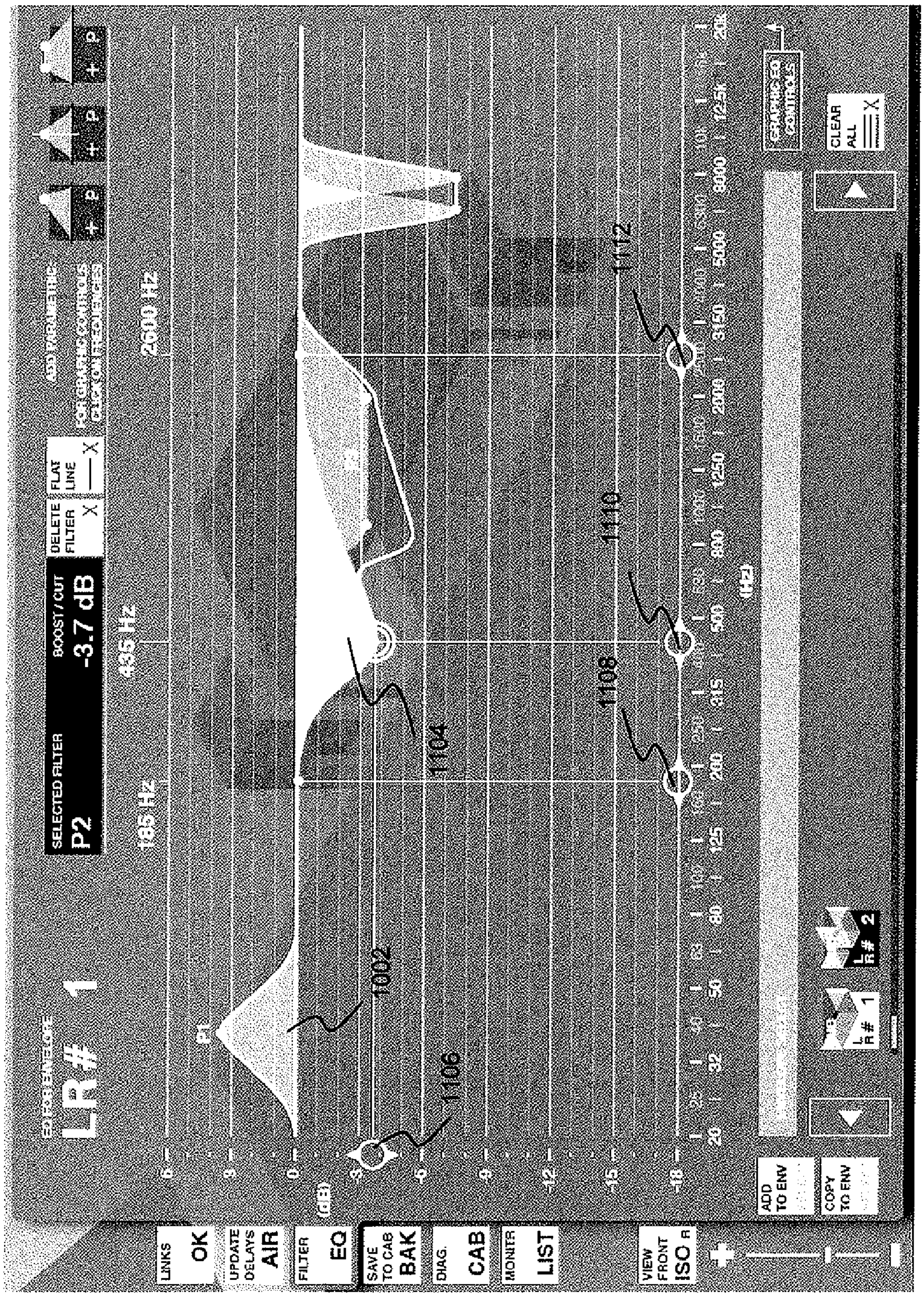
LR# 1
SELECTED FILTER
P2
BOOST / CUT
-3.7 dB
DELETE FILTER
FLAT LINE
185 Hz
435 Hz
2600 Hz
P1
1002
1104
1106
1108
1110
1112
(dB)
(Hz)
LINKS OK
UPDATE DELAYS AIR
FILTER EQ
SAVE TO CAB BAK
DIAG. CAB
MONITR LIST
VIEW FRONT ISO
ADD TO ENV
COPY TO ENV
CLEAR ALL
GRAPHIC EQ CONTROLS

FIG. 11

Europäisches Patentamt
European Patent Office
Office européen des brevets

EUROPEAN SEARCH REPORT

Application Number
EP 12 16 3054

DOCUMENTS CONSIDERED TO BE RELEVANT

Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X	US 2010/305725 A1 (BRANNMARK LARS-JOHAN [SE] ET AL BRAENNMARK LARS-JOHAN [SE] ET AL) 2 December 2010 (2010-12-02) * the whole document *	1-13	INV. H04R27/00
A	EP 1 205 863 A1 (HONDA R & D EUROP DEUTSCHLAND [DE]) 15 May 2002 (2002-05-15) * page 3, paragraph 6 - page 6, paragraph 33 *	1-13	
A	US 2003/007648 A1 (CURRELL CHRISTOPHER [US]) 9 January 2003 (2003-01-09) * the whole document *	1-13	
A	US 2005/220309 A1 (HIRATA MIKIKO [JP] ET AL) 6 October 2005 (2005-10-06) * page 3, paragraph 61 - page 10, paragraph 145 *	1-13	

TECHNICAL FIELDS SEARCHED (IPC)

H04R
H04S

The present search report has been drawn up for all claims

Place of search	Date of completion of the search	Examiner
Munich	12 July 2012	Coda, Ruggero

2

CATEGORY OF CITED DOCUMENTS

X : particularly relevant if taken alone
Y : particularly relevant if combined with another document of the same category
A : technological background
O : non-written disclosure
P : intermediate document

T : theory or principle underlying the invention
E : earlier patent document, but published on, or after the filing date
D : document cited in the application
L : document cited for other reasons
& : member of the same patent family, corresponding document

EPO FORM 1503 03.82 (P04C01)

ANNEX TO THE EUROPEAN SEARCH REPORT ON EUROPEAN PATENT APPLICATION NO. EP 12 16 3054

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

12-07-2012

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2010305725 A1	02-12-2010	NONE	
EP 1205863 A1	15-05-2002	EP 1205863 A1 JP 2002203228 A US 2002099929 A1	15-05-2002 19-07-2002 25-07-2002
US 2003007648 A1	09-01-2003	NONE	
US 2005220309 A1	06-10-2005	JP 2005286903 A US 2005220309 A1	13-10-2005 06-10-2005

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 20090144036 A **[0003]**
- US 7206415 B **[0003]**

Non-patent literature cited in the description

- **KARAM.** Design of optimal digital FIR filters with arbitrary magnitude and phase responses. *ISCAS,* 1996 **[0057]**