Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Robot Phonotaxis with Dynamic Sound-source Localization

Sean B. Andersson, Amir A. Handzel, Vinay Shah, and P.S. Krishnaprasad
Electrical and Computer Engineering and Institute for Systems Research
University of Maryland, College Park, MD 20742
sanderss@deas.harvard.edu {handzel,krishna}@isr.umd.edu vinayshah01@yahoo.com

Abstract—**We address two key goals pertaining to autonomous mobile robots: one, to develop fast accurate sensory capabilities — at present, the localization of sound sources — and second, the integration of such sensory modules with other robot functions, especially its motor control and navigation. A primary motivation for this work was to devise effective means to guide robotic navigation in environments with acoustic sources. We recently designed and built a biomimetic sound-source localization apparatus. In contrast to the popular use of time-of-arrival differences in free field microphone arrays, our system is based on the principles observed in nature, where directional acoustic sensing evolved to rely on diffraction about the head with only two ears. In this paper we present an integrated robot phonotaxis system which utilizes the robot's movement to resolve front-back localization ambiguity. Our system achieves high angular localization acuity (± 2°) and it was successfully tested in localizing a single broadband source and moving towards it within a cluttered laboratory environment.**

I. Introduction

The survival of animals depends on their effectiveness in collecting sufficient and timely information about their ever-changing environment and on their ability to act upon sensory information by moving towards food and away from danger. Mobile autonomous robots, especially those intended for operation in hazardous or hostile environments, necessitate the design of analogous capabilities. Acquisition of optical information by cameras has become standard and a corresponding body of theories and practical knowledge has been developed. Other modalities – sound, ultrasound, chemical sensing ("olfaction") – are receiving increasing attention, either as independent sensory means, or to close gaps and augment optical sensing. As in the animal world, one wants to utilize the sensory information to guide the robot towards points of interest, such as sources of sound — *phonotaxis*, or along gradients of concentration of chemical compounds, known as *chemotaxis* [7], [13].

To date, a few phonotactic robots have been constructed based on various paradigms: a robot equipped with a semi-circular sonar array was used to explore the issues of target representation and analogies to human cognition [1]; another project aimed to emulate neural circuitry underlying insect phonotaxis and chemotaxis [14]; closer to applications, the Army Research Laboratory has developed robot platforms which function well in outdoor environments [16]. Most sound localization systems, whether stationary [4], [6], [15] or mounted on mobile robots [16], comprise free field microphone arrays. Localization is achieved by measuring differences in time of arrival between microphone pairs.

In nature, directional acoustic sensing evolved to rely on diffraction about the head with only two sensors — the ears. The impinging sound waves are modified by the head in a frequency and direction dependent way, and additional complex filtering is performed by the external ears (*pinnae*). The cochlea decomposes the sound pressure signal into frequency bands. The brain then uses interaural differences in phase (IPD) and intensity level (ILD) in the various frequency bands to infer the location of a source [2], [10]. Inspired by human sound localization, we recently conceived the design of artificial systems according to similar principles [8]. We then built and successfully tested such a stationary apparatus made of an artificial head with microphones placed antipodally on its surface [9]. We developed a suitable algorithm to extract the directional acoustic information, based on the fact that the sound pressure at each microphone can be computed analytically by modelling the head as a sphere [3]. Due to the spherical symmetry of the device, stationary localization in azimuth is possible only up to a front-back ambiguity. In [8] we proposed generating dynamic localization cues through rotation of the apparatus. Here we report on an implementation of this idea by using the robot's own movement, obviating the need for active device rotation.

A key motivation for our work is to devise effective means to guide robotic navigation in environments with acoustic sources, namely phonotaxis. Central in our approach to the integration of sensory modalities with other robot functions is a modular software architecture structure, the *Modular Engine*, coupled with a movement control framework called *Motion Description Language – extended* (MDLe). A motion description language provides a means for abstracting from the low-level details of a control system. The motion programs written in such a language combine feedback control laws and logic into strings that have meaning nearly independently of the underlying system. See [5], [12] for relevant background. The modular engine, described in [11], provides a software architecture to integrate the control of a physical system through MDLe with the system hardware and with additional sensors.

The paper is organized as follows. In Section II we prescribe the computations needed to extract the direction of a source

0-7803-8232-3/04/$17.00 ©2004 IEEE

from two consecutive measurements of pairs of sound signals. A brief description of MDLe is given in Section III while the modular engine is outlined in Section IV. The physical apparatus and the robot are described in Section V and the integration of the components into a phonotactic robot is presented in Section VI. Results are presented in Section VII followed by a brief conclusion.

II. Localization algorithm

In this work we restrict ourselves to localizing a source in *azimuth only*. The measured sound pressure at each microphone is a complex response to the excitation by a source:

$$p = Ae^{i\alpha - i\omega t} \tag{1}$$

where ω is the angular frequency, t is time and α is the part of the phase containing spatial information. With pressure measured at the right (R) and left (L) microphones, we define the Interaural Level Difference and Interaural Phase Difference:

$$\mathrm{ILD} = \log A_L - \log A_R \qquad \mathrm{IPD} = \alpha_L - \alpha_R, \tag{2}$$

which are both smooth functions of frequency. We consider the ILD-IPD plane as a basic feature space in which localization is performed. Every source direction and emission frequency induces an "active" point in the ILD-IPD plane. Since ILD and IPD depend smoothly on frequency, a broadband sound source generates a whole curve $\sigma(\cdot)$ in this plane which is its specific *signature* depending on the source location [8].

The picture can be summarized as follows: a source at position $\mathbf{r}_0$ in the plane emits sound which is mapped through the scattering process, S, to a pair of sound pressure measurements, i.e. a pair of smooth complex functions of some frequency interval Ω. Extracting the binaural, i.e. relative, phase and intensity, reduces them to a pair of Real functions:

$$\begin{tikzcd} \mathbb{R}^2 \arrow[r, "S"] & C_{\mathbb{C}}(\Omega) \times C_{\mathbb{C}}(\Omega) \arrow[r, "I"] & C_{\mathbb{R}}(\Omega) \times C_{\mathbb{R}}(\Omega) \\ \mathbf{r}_0 \arrow[r, mapsto, "S"] & (p_L, p_R) \arrow[r, mapsto, "I"] & (\mathrm{ILD}, \mathrm{IPD}) \end{tikzcd} \tag{3}$$

The task is to prescribe a localization operator that would, in effect, invert the above to recover the source direction. We do so by defining the squared L^2 norm distance between the measured interaural functions ($\mathrm{ILD}(\omega)$,$\mathrm{IPD}(\omega)$) and the theoretical functions ($\mathrm{IPD}(\theta,\omega), \mathrm{ILD}(\theta,\omega)$). The theoretical functions are calculated using the analytical solution of the full 3D acoustic scattering problem [8] and stored in a table. To perform static localization, we pick the angle whose interaural functions are closest to those measured in the following sense. The metric for IPD is:

$$\begin{aligned} \mathbf{D}_2^{\mathrm{IPD}}(\theta) &\equiv \| \mathrm{IPD}(\theta,\omega) - \mathrm{IPD}(\omega) \|_2^2 \\ &= \textstyle\sum_\omega (\mathrm{IPD}(\theta,\omega) - \mathrm{IPD}(\omega))^2 \end{aligned} \tag{4}$$

and similarly for ILD. We normalize each of the two metrics with respect to its maximal value (over θ):

$$\mathbf{D} \longrightarrow \frac{1}{M}\mathbf{D} \qquad M = \max_{(\theta)} \mathbf{D}(\theta), \tag{5}$$

and combine the normalized metrics for ILD and IPD to produce a combined distance function:

$$\mathbf{D}_2^{\mathrm{Comb}} = \mathbf{D}_2^{\mathrm{IPD}} + \mathbf{D}_2^{\mathrm{ILD}}. \tag{6}$$

The static source direction is the one for which the combined metric is at minimum. An implementation of the above has been recently reported [9].

As explained in [8], the symmetry of the apparatus and the interaural functions allows plane localization up to an ambiguity between the front and back relative to the inter-microphone axis. In order to overcome this limitation, we suggested gleaning additional spatial information by rotating the apparatus relative to the source, similarly to human sound localization performance [8]. Such rotation induces a flow of the signature curve in the ILD-IPD plane; (see plot in [8]). Infinitesimally, a gradient vector of the signature is generated at each active point in the feature plane. Its components are the derivatives of the interaural functions w.r.t. the direction angle θ (denoted by primes):

$$\mathrm{ILD}' = (\log A_L)' - (\log A_R)' \qquad \mathrm{IPD}' = \alpha_L' - \alpha_R'. \tag{7}$$

In practice, we approximate the gradient by the difference between two consecutive measurements over a short time interval. There is no need, however, to construct a dedicated rotation mechanism for the localizing head: as the robot moves about, its movement can be exploited for dynamic localization. Since the derivatives are approximated by finite differences, we need to divide the measured signals by the appropriate finite rotation angle of the robot, e.g. $\Delta\alpha/\Delta\theta$. The angle $\Delta\theta$ is obtained from the robot odometry. In order to use the dynamic cues, we then define additional metrics:

$$\begin{aligned} \mathbf{D}_2^{\mathrm{IPD}'}(\theta) &\equiv \| \mathrm{IPD}'(\theta,\omega) - \mathrm{IPD}'(\omega) \|_2^2 = \\ &\sum_\omega (\mathrm{IPD}'(\theta,\omega) - \mathrm{IPD}'(\omega))^2 \end{aligned} \tag{8}$$

and similarly for ILD'. Finally, the total metric comprises four terms:

$$\mathbf{D}_2^{\mathrm{Tot}} = \mathbf{D}_2^{\mathrm{IPD}} + \mathbf{D}_2^{\mathrm{ILD}} + \mathbf{D}_2^{\mathrm{IPD}'} + \mathbf{D}_2^{\mathrm{ILD}'}. \tag{9}$$

The angle which obtains the minimum of the metric is assigned to be the source direction.

III. Motion description language

We present a brief synopsis of the MDLe language specification; details can be found in [11], [12]. The underlying physical system (in this case a mobile robot) is equipped with a set of sensors and actuators and is assumed to be governed by a differential equation of the form

$$\dot{x} = f(x) + G(x)u; \quad y = h(x) \in \mathbb{R}^p \tag{10}$$

where $x(\cdot) : \mathbb{R}^+ \rightarrow \mathbb{R}^n$ is the state of the system, $u(\cdot) : \mathbb{R}^p \times \mathbb{R}^+ \rightarrow \mathbb{R}^m$ is a control law of the type $u = u(h(x), t)$, and G is a matrix whose columns are vector fields in $\mathbb{R}^n$. The simplest element of MDLe is the *atom*, defined to be a triplet of the form $\sigma = (u, \xi, T)$, where u is as defined earlier,

$\xi : \mathbb{R}^p \rightarrow \{0,1\}$ is a boolean *interrupt* function defined on the space of outputs from p sensors, and $T \in \mathbb{R}^+$ denotes the value of time (measured from the time the atom is initiated) at which the atom will expire. To *evaluate* the atom is to apply the control law u until the interrupt ξ returns zero or until T units of time have elapsed. Atoms can be composed into a string, called a *behavior*, that carries its own interrupt function and timer. Behaviors can in turn be composed to form higher-level strings (called *partial plans*) and so on. We will use the term *plan* to refer to a generic MDLe string independent of the number of nested levels it contains.

IV. Modular Engine

In order to execute a motion control program on a physical system, software is needed to read data from the sensors and to translate the MDLe plan into actuator commands. As different systems have different actuators and sensors, one desires an architecture which is as independent of the underlying hardware as possible. This is achieved through the Modular Engine (ME). Under this architecture, each system component is handled by its own device driver, known as a **module**. These modules are compiled separately into run-time libraries and are loaded as needed by the ME at execution time.

The ME uses a system timer to enforce a pseudo-periodic control cycle comprising two parts: the **turn** segment and the **turn break** segment. Each module provides two special methods to the ME, namely a turn method and a turn break method. At the beginning of each control cycle, the ME initiates a turn and spawns a thread for each module's turn method. In this segment all I/O, data processing, and computation take place. When each module has completed its tasks, the turn ends. The ME then calls each turn break method, allowing the modules to share their data. For example, a sensor module would access the physical sensor to sample the data and perform any desired signal processing during the turn. The results are then made available to other modules during the turn break.

The ME provides one special module, the MDLe module, which translates MDLe plans first down to the current atom and then down to the executable code that implements the atom. As the MDLe module runs, it parses the current plan and locates the first atom. Pointers to the control and interrupt functions are retrieved and placed in the module's turn method. During each turn, the ME calls the MDLe turn method which executes the feedback loop and evaluates the interrupt for the current atom. In the next turn break, the module checks the value of the last interrupt evaluation. If any level of interrupt has been triggered, then the module advances down the plan (skipping to the next atom, behavior, etc. based on the level of interrupt) to find the next atom to be executed. During this same turn break the control values from the previous turn are passed to the appropriate hardware module.

The user must supply at least one module, usually termed the Robot module, which encapsulates all the device drivers necessary to interface with the robot's actuators and built-in sensors. During each turn this module translates the control values determined by the MDLe module in the previous turn into a machine specific command and then writes this command to the actuators. It also reads the data from all the active on-board sensors. During each turn break this data is made available to the other modules.

Additional modules may be written to incorporate custom sensors, actuators, or software functionality.

V. Hardware

A biomimetic "head" for sound-source localization was created by mounting a pair of Knowles FG-3329 microphones antipodally on a hard, spherical shell of radius 10 cm, shown in Figure 1. This device improves upon an earlier model, described in [9], by utilizing a nearly spherical head. The head was filled with styrofoam to dampen internal resonances. Output signals from the microphones are fed to a small circuit board which contains an anti-aliasing filter and an output amplifier. Since the algorithm relies on interaural phase and amplitude differences, it is vital that the filters and amplifiers have as similar a response across the two channels as possible. For this reason a flat phase-response filter was chosen (a sixth-order low-pass Bessel filter with a cutoff frequency of approximately 11 kHz) and the gain of the amplifier on each channel was selected so as to equalize the average background signal level on each channel. The head and circuit board were then mounted on a mobile robot equipped with a sound card. The amplified microphone signals were connected to the line-in input of the sound card.

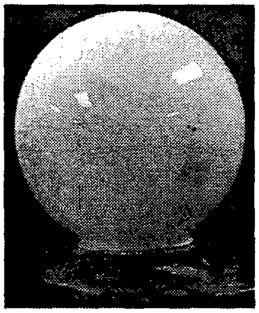


Fig. 1. Sound-localizing head

The robot used is a direct-drive wheeled robot obeying the following nonholonomic kinematics.

$$\begin{aligned} \dot{x} &= u_f \cos(\theta), \\ \dot{y} &= u_f \sin(\theta), \\ \dot{\theta} &= u_\theta \end{aligned} \tag{11}$$

where (x, y, θ) describe the position and orientation of the robot with respect to a fixed lab frame and u_f and u_θ are the forward and turning velocities of the robot. The output is given by the state of the robot itself as determined by the on-board odometry. This odometry drifts from the true position over time due to errors such as wheel slippage. However, it is reasonably accurate over a few turns and in the controls for phonotaxis (see Section VI) only the difference in the odometry between successive turns is used. The robot is equipped with a sonar array to detect obstacles in the environment and a bump sensor to detect collisions with objects. The on-board sound card is initialized to sample two channels of data at its maximum rate of $f_s = 22$ kHz. Physically, the A/D converter samples the two channels sequentially at a rate of 44 kHz and thus introduces a frequency-dependent phase shift in the signals given by

$$\Delta\phi_{sampling}(f) = \frac{2\pi f}{f_s}. \qquad (12)$$

VI. A PHONOTACTIC ROBOT

The ME was used to integrate the head and the robot with the sound-localization algorithm and an MDLe plan combining motion towards the sound-source and obstacle avoidance.

A. Modules

Two new modules for the ME were designed and implemented.

Heading module: The heading module separates the task of determining the location of a target, in this case a sound-source, from the task of moving towards or tracking a target by providing a place for target-finding algorithms to write an estimated direction (in radians) and range (in meters) to the target and a place for control algorithms to read the data from.

Sound-localization module: The sound-localization module implements the algorithm described in Section II. During each turn segment, a 46 ms sample of sound is captured from each microphone. These data are passed through a fast Fourier transform and the IPD and ILD are calculated from (2). The frequency-dependent shift in the IPD due to the alternating sampling of each channel is corrected and the IPD and ILD signals are smoothed using a nine-point moving average filter. The robot was found to produce a low frequency signal and, therefore, the IPD and ILD data are high-pass filtered at 200 Hz. The internal odometry of the robot is read from the Robot module and compared against the odometry data stored during the previous turn to determine the relative rotation of the robot between measurements of the sound-source. This relative rotation, together with the processed IPD and ILD information from the current turn and the previous turn, is then used in the front-back symmetry breaking algorithm to determine the direction to the sound source. When calculating the distance function in (9), only frequencies with amplitudes beyond an experimentally determined threshold level are used. The estimated direction is then written to the heading module.

B. The `Taxis` Plan

As described in Section V, the robot was equipped with a set of sonar sensors and a bump sensor. Using these built-in devices, the following interrupt functions were designed.

- `(bumper)` : returns 0 if the bump sensor is contacted and 1 otherwise.
- `(obstacleFront` d`)` : returns 0 if an obstacle is detected within d meters of the robot and 1 otherwise.
- `(noObstacleFront` d`)` : returns 0 if no obstacle is detected within d meters of the robot and 1 otherwise.

To create a robot capable of moving towards the target specified in the heading module while avoiding intervening obstacles, the following two control functions were defined.

- `(follow` k_f k_θ`)` : Sets $u_f = k_f(\frac{\pi}{2} - \text{abs}(\hat{\theta}))$ and $u_\theta = -k_\theta \hat{\theta}$ where $\hat{\theta}$ is the heading to the target read from the heading module.
- `(avoid` d`)` : Steers the robot around an obstacle within d meters of the front of the robot. In the absence of an obstacle, sets $u_f = u_\theta = 0$.

Under the `follow` function, the robot steers towards the target at a rate dependent on the angular distance of the target away from the straight forward direction. In particular, if the target is behind the robot the control will drive the system backwards.

From these control and interrupt functions two atoms were defined and from them the following behavior was composed.

- `TaxisBehavior = {`
  ((follow (k_f k_θ)), (obstacleFront 1))
  ((avoid d), (noObstacleFront 1)) }

where the notation for each atom is ((control), (interrupt)).

The MDLe plan `Taxis` was defined by setting `TaxisBehavior` to loop *ad infinitum* and associating the `bumper` function as the plan level interrupt. Under this plan, the robot will move in the direction indicated in the heading module so long as there is no nearby obstacle detected. In the presence of an obstacle, the robot will instead attempt to move around it until its path is clear again, at which time it will continue to move towards the target.

VII. EXPERIMENTAL RESULTS

We present the results of a set of experiments to determine the accuracy of the algorithm and apparatus described in Sections II and V and then discuss the behavior of the robot under the `Taxis` plan.

A. Localization accuracy

In these experiments, the head was removed from the robot base, mounted on a tripod, and placed in a small room whose walls were covered in foam. While the room was not an anechoic chamber, it did insulate the apparatus from external sources of noise.

A broadband signal was generated in Matlab and presented over a speaker placed in the room at a fixed radial distance from the center of the head. The heading angle between the speaker and the head was adjusted by rotating the head using the tripod. The true angle to the source was measured with a protractor fixed to the bottom of the head. Ten 43 ms data samples were collected at every 2° from −90° (laterally at the right) to 90° (laterally at the left); 0° marks the front ahead direction. Due to the symmetry of the apparatus, sound signals for source directions behind the head were the same as for the mirrored positions in the front and were thus not sampled.

To simulate motion of the robot, data from two headings separated by $\Delta\theta = 6°$ were used. These data were processed as described in Section VI and then the algorithm described in Section II was applied to produce the heading estimate.

In Figure 2 we show two representative distance curves (using the total metric in (9)) for two nearby sources, one at −14° and one at −16°. The figure clearly shows a sharp global minimum in both curves. A closer view of the vicinity of the global minima is shown in Figure 3 and it is clear that the two signals are easily differentiated. One may expect that as the direction to the sound-source moves farther away from the centerline, the resolution of the algorithm may decrease due to the "bright spot," a pronounced constructive diffraction in the front scatter direction which substantially reduces the ILD at ±90° relative to the other source locations. In Figure 4 we show a close view of the global minima for sources at −76° and −78°. This figure reveals that the distance curves remain sharp even as the direction to the sound-source approaches the lateral direction.

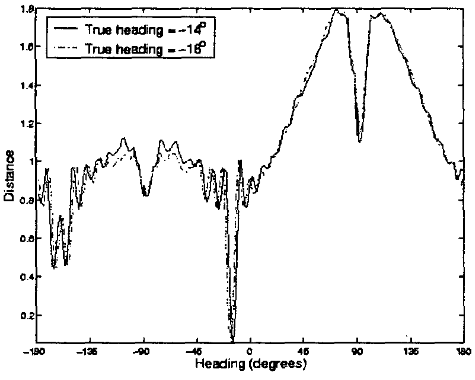


Fig. 2. Distance functions for sources at −14° and −16°

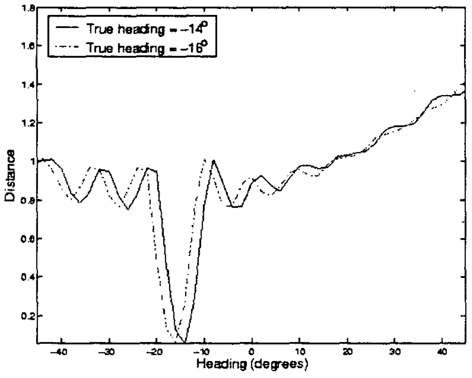


Fig. 3. Zoomed-in distance functions for sources at −14° and −16°

Ten heading estimates were produced from the ten data sets for all source directions. Figure 5 shows a scatter plot of these estimates as a function of the true heading to the source. There is both very little error and very little variation in the estimated direction to the source and the performance is strikingly good over the full range.

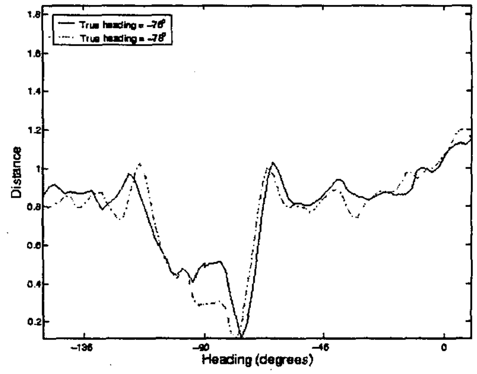


Fig. 4. Zoomed-in distance functions for sources at −76° and −78°

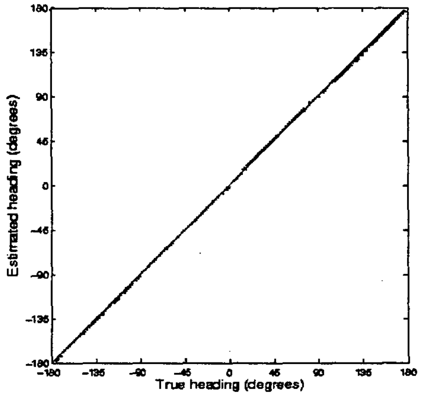


Fig. 5. Localization performance

A scatter plot of the error is shown in Figure 6. Almost all the readings are within ±2°. Interestingly, there is greater error in the right half plane than in the left half plane. Over the range $[0°, 90°]$ the estimate appears to be shifted by 2° while over the range $[90°, 180°]$ the estimate appears to be shifted by $-2°$. As this error is reflected about 90° it is likely to be due to a systematic error in the data collection or the experimental setup. Indeed, upon closer inspection we found that the mounting fixture attaching the head to the tripod was not perfectly rigid and some reorientation of the head occurred when it was rotated to the right.

B. Phonotaxis experiments

For this set of experiments, the head was once again mounted on the robot. A speaker was placed in a fixed location in the laboratory and the robot placed in the room at various initial positions and orientations. A broadband signal was presented through the speaker. Using the ME, the sound localization and heading modules were started, the `Taxis` plan was loaded into the MDLe module, and the MDLe module was started. The laboratory contained tables, desks, bookshelves and chairs. In every trial the robot successfully

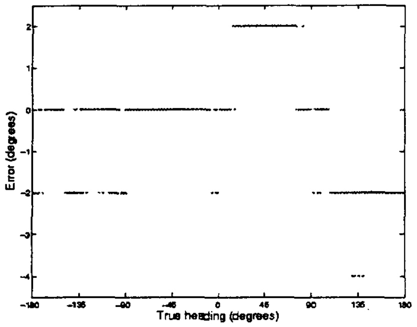
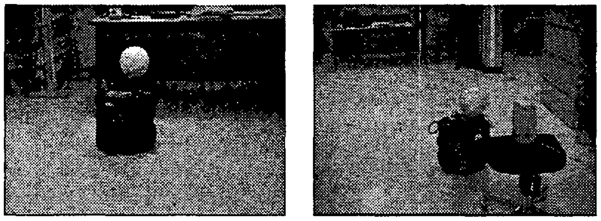


Fig. 6. Scatter plot of error

navigated to the sound-source while avoiding these obstacles.

In Figure 7.a the initial position of the robot from one trial run is shown. The sound-source was located off the lower-right corner of the image and the robot was initially facing away from the target. Upon starting, the heading module initializes the direction to the target as 0°. Consequently, the robot initially moved forward while the sound localization module first gathered data and then determined the direction to the sound-source and wrote this direction to the heading module. On the third turn this data became available to the control function. The sound-source was actually behind the robot and thus the system was driven in reverse and rotated until the sound-source was in front. At this point the robot moved forward and approached the speaker. Figure 7.b shows the robot once the speaker has been reached. Since no range information to the target is determined by the localization algorithm, the robot cannot know when the target is reached. However, the obstacle avoidance part of the `Taxis` plan prevents the robot from colliding with the chair upon which the speaker sits. The full movie of this experiment can be found at `www.deas.harvard.edu/~sanderss/movies.html`.



a. Beginning of run

b. End of run

Fig. 7. Trial run images

VIII. Concluding Remarks

In this paper we have presented a robot phonotaxis system which uses the ME to smoothly integrate a biomimetic sound-source localization apparatus with a simple taxis behavior coded in the motion description language MDLe. This system fits well with the concept of sensory-motor integration: sensory information guides robot movement and navigation, which in turn enhances the sensory capability by utilizing the motion of the robot to resolve the front-back localization ambiguity. The experiments presented here show that the apparatus and algorithm are accurate to within ±2° over the entire 360° sensing range. When the apparatus was coupled to the mobile platform, the robot was able to successfully navigate in a cluttered environment and move towards the sound-source.

IX. Acknowledgements

The authors gratefully acknowledge Fumin Zhang for the use of his obstacle avoidance routine. This research was supported in part by NSF Learning and Intelligent Systems Initiative Grant CMS9720334, by the ARO ODDR&E MURI01 Program Grant No. DAAD19-01-1-0465 to the Center for Communicating Networked Control Systems (through Boston University) and by ONR ODDR&E MURI97 Program Grant No. N000149710501EE to the Center for Auditory and Acoustics Research. S. Andersson was also supported by a fellowship from the ARCS Foundation and V. Shah was supported by the REU program at the University of Maryland through NSF.

References

[1] Estela Bicho, Pierre Mallet, and Gregor Schöner. Target representation on an autonomous vehicle with low-level sensors. *International Journal of Robotics Research*, 19(5):424–447, May 2000.

[2] Jens Blauert. *Spatial Hearing*. MIT Press, revised edition, 1997.

[3] J.J. Bowman, T.B.A. Senior, and P.L.E. Uslenghi. *Electromagnetic and Acoustic Scattering by Simple Shapes*. North-Holland, 1969.

[4] Michael Brandstein and Darren Ward, editors. *Microphone Arrays*. Digital Signal Processing. Springer-Verlag, 2002.

[5] R. W. Brockett. Hybrid models for motion control systems. In H. Trentelman and J. Willems, editors, *Perspectives in Control*, pages 29–54. Birkhäuser, Boston, 1993.

[6] Huang Jie *et al.* A model-based sound localization system and its application to robot navigation. *Robotics and Autonomous Systems*, 27(4):199–209, 1999.

[7] F.W. Grasso, T.R. Consi, D.C. Mountain, and J. Atema. Biomimetic robot lobster performs chemo-orientation in turbulence using a pair of spatially separated sensors: progress and challenges. *Robotics and Autonomous Systems*, 30(1-2):115–131, 2000.

[8] Amir A. Handzel and P.S. Krishnaprasad. Biomimetic sound-source localization. *IEEE Sensors Journal*, 2(6):607–616, 2002.

[9] Amir Aharon Handzel, Sean Bertil Andersson, Martha Gebremichael, and P.S. Krishnaprasad. Biomimetic apparatus for sound-source localization. In *Proc. IEEE Conf. on Decision and Control, to appear*. IEEE, 2003. submitted.

[10] William M. Hartmann. How we localize sound. *Physics Today*, pages 24–29, November 1999.

[11] D. Hristu-Varsakelis, P.S. Krishnaprasad, S. Andersson, F. Zhang, L. D'Anna, and P. Sodre. The MDLe engine: A software tool for hybrid motion control. Technical Report TR2000-54, The Institute for Systems Research, 2000.

[12] V. Manikonda, P. S. Krishnaprasad, and J. Hendler. Languages, behaviors, hybrid architectures and motion control. In J. Baillieul and J.C. Willems, editors, *Mathematical Control Theory*, pages 199–226. Springer, 1998.

[13] R.A. Russell. Survey of robotic applications for odor-sensing technology. *International Journal of Robotics Research*, 20(2):144–162, February 2001.

[14] Barbara Webb. Robots, crickets and ants: models of neural control of chemotaxis and phonotaxis. *Neural Networks*, 11:1479–1496, 1998.

[15] Juyang Weng and Kamen Y. Guentchev. Three-dimensional sound localization from a compact non-coplanar array of microphones using tree-based learning. *J. Acoustical Society America*, 110(1):310–323, July 2001.

[16] Stuart H. Young. Detection and localization with an acoustic array on a small robotic platform in urban environments. Technical Report ARL-TR-2575, Army Research Laboratory, Adelphi, MD, January 2003.