2013 IEEE International Conference on Robotics and Automation (ICRA)
Karlsruhe, Germany, May 6-10, 2013

Development and Control of a Three DOF Spherical Induction Motor

Masaaki Kumagai and Ralph L. Hollis

***Abstract*— This paper reports, to our knowledge, the first spherical induction motor (SIM) operating with closed loop control. The motor can produce up to 4 Nm of torque along arbitrary axes with continuous speeds up to 300 rpm. The motor's rotor is a two-layer copper-over-iron spherical shell. The stator has four independent inductors that generate thrust forces on the rotor surface. The motor is also equipped with four optical mouse sensors that measure surface velocity to estimate the rotor's angular velocity, which is used for vector control of the inductors and control of angular velocity and orientation. Design considerations including torque distribution for the inductors, angular velocity sensing, angular velocity control, and orientation control are presented. Experimental results show accurate tracking of velocity and orientation commands.**

I. INTRODUCTION

Many multi-DOF spherical actuators have been proposed using electromagnetic forces or ultrasonic vibrations. However, the application of those actuators have been limited due to restricted range of motion and/or output power. Notably, there have been no practical spherical actuators with enough power, *i.e.,* speed and torque, to serve as a prime mover for mobile robots.

Both of the authors have previously developed mobile robots whose single wheel is a sphere, *ballbot* [1], [2] and *BallIP* [3], [4]. Each of these robots has a mechanism to drive their ball wheels, *e.g.,* inverse mouse ball drive or special omniwheels, which made the robots mechanically complex and caused problems due to friction. One of us, Hollis, has had the plan to use spherical motor wheels for ballbots. A direct drive spherical motor has only one moving part reducing an omnidirectional mobile robot to a body and a ball. What could be simpler?

This plan notwithstanding, spherical motors are not off-the-shelf items. As the single wheel of a balancing robot, the motor would need to have enough torque for balancing and enough speed for mobility, with response times short enough to permit good feedback control.

Spherical ultrasonic motors have short response times and moderate torque, whereas the speed is rather slow. A spherical stepper motor [5] can perform absolute rotation control although it requires complex spatial control of magnetic fields using many electromagnets, and lacks a practical level of power and smooth torque output. Several different spherical induction motors (SIMs) [6], [7] were proposed and prototyped decades ago, but their rotors only operated over limited angular ranges and they also lacked closed loop control. Certainly, no one had previously imagined such motors could be used as ball wheels for balancing mobile robots. The authors believe SIMs are promising for these robots because of their simple rotor construction, and mechanical simplicity. Successful operation of such SIMs, however, will depend heavily on modern drive and sensing electronics as well as the ability to perform fast realtime computation.

Masaaki Kumagai is with the Faculty of Engineering, Tohoku Gakuin University, Tagajo 985-8537 Japan. Ralph L. Hollis is with The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA.
kumagai@tjcc.tohoku-gakuin.ac.jp,
rhollis@cs.cmu.edu

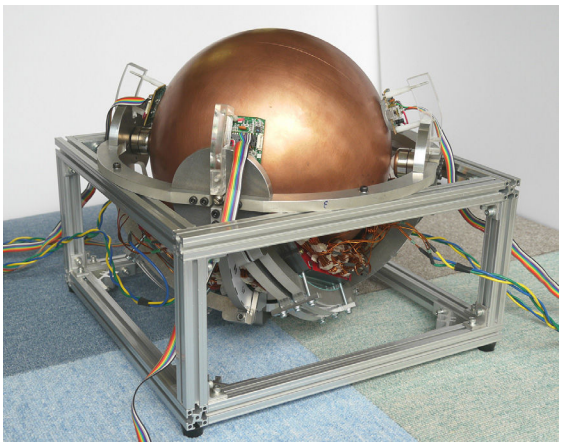


Fig. 1. Spherical induction motor.

To realize the SIM described in this paper, the authors had previously investigated vector control methods for linear induction motors (LIMs) [8], developed a spherical motion sensing method using optical mouse sensors [9], and developed a closed-loop planar induction motor as a flat-type 3-DOF induction motor (a SIM of infinite radius) [10]. Here, we report the developed SIM based on these previous efforts. The prototype implementation and control methods of the SIM are presented first, followed by experimental results, and conclusions.

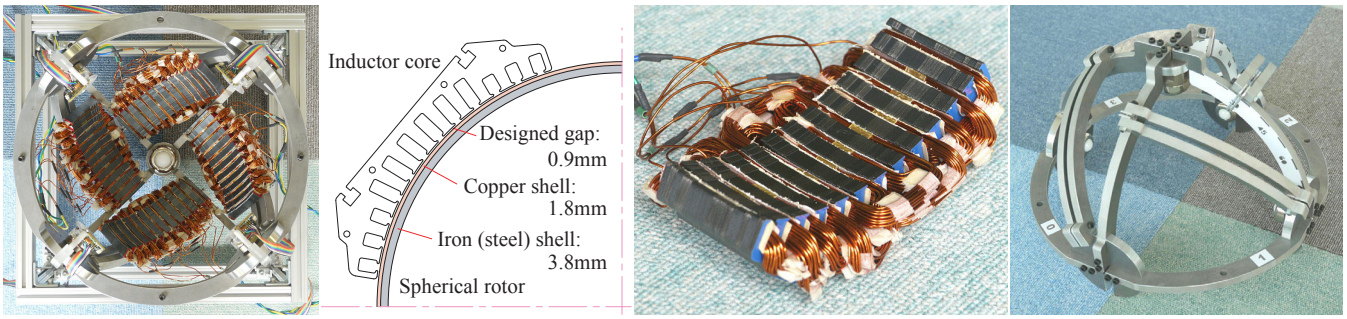
II. IMPLEMENTATION OF THE MOTOR

A. Overview of the developed motor

Our developed closed-loop SIM is shown in Fig. 1. The motor, as a 3-DOF actuator, is capable of 300 rpm rotation in arbitrary axis with 4 Nm torque on the spherical rotor of 246 mm diameter. There are four curved inductors in close proximity to the rotor that are each operated by a vector control method.

This SIM is not just an actuator. It has four optical mouse sensors that measure the local surface velocities of the rotor from which its angular velocity can be estimated. This enables, in turn, precise control of angular velocity and orientation (rotational angle). For example, the measured

978-1-4673-5642-8/13/$31.00 ©2013 IEEE



(a) stator (b) rotor (c) inductor (d) frame

Fig. 2. The stator of the SIM, cross-section of the rotor, constitutive inductor and the frame to mount inductors.

response times to a 180°/s step velocity command and a 22.5° step rotation command were approximately 0.1 s and 0.2 s, respectively.

B. Hardware of the system

The SIM consists of the spherical rotor and the stator shown in Fig. 2(a), *i.e.* four inductors fixed on the supporting frame, and four optical mice sensors. The rotor is constrained by ball bearing transfer devices.

The rotor is a two-layered spherical shell whose inner layer is made of iron and outer layer is of copper as in Fig. 2(b). The inner iron layer forms closed magnetic circuits with the inductor cores. It is also a structural part of the rotor needed to resist strong attractive forces from the inductors in addition to any external load. Eddy currents are induced in the outer conductive copper layer by the traveling magnetic field, which generates the thrust traction force (*i.e.* torque on the rotor) by interaction with the impressed magnetic fields.

The rotor was made by Kitajima Shibori Seisakusho Co., Ltd. First, hemispheres of the iron and copper, two for each, were produced by spinning. Then, iron hemispheres were welded together to form the inner shell. The copper hemispheres were attached to the iron shell using adhesives. The thickness of the iron shell is around 3.8 mm, and the copper shell is 1.8±0.1 mm (the thickness of the copper is important for uniform characteristics of the SIM). The outer diameter of the rotor is 246.2 mm and it weighs 8.2 kg, with moment of inertia estimated to be 0.080 kgm^2.

The inductor shown in Fig. 2(c) has almost the same dimension as we previously used for the PIM [10], except for being curved in two dimensions. The inductor consists of coils and the lamination core that has 12 slots of 12.5 mm pitch with 9 coils of 25 turns each. The width of each inductor is approximately 50 mm with 92 0.53 mm thick non-oriented JIS 50A470 magnetic steel sheets. To fit the surface of the spherical rotor, the teeth shape facing the rotor form a part of a sphere, by combining five types of sheets having different radii. The resulting magnetic gap is approximately 1 mm.

The inductors were fixed on the structural frame (Fig. 2(d)), made of 10 mm thick A7075 aluminum alloy. The frame assembly has an orthogonal circular frame part and four inductor fixture frames. Each fixture frame can slide over the orthogonal frame to change torque-related mechanical parameters described later and to fit inductors into the available space.

Each inductor is driven by a vector controller we had previously developed. The parameters needed for vector control were calibrated manually through experiments, to control the thrust force generated by its associated inductor, which is commanded by a main control computer.

The SIM also has four mouse sensors (Avago ADNS6010, obtained from mice) that measure the surface velocity of the rotor. Each sensor can measure speeds up to 1 m/s on the surface, which is about 3π rad/s, or 1.5 rev/s. The SIM can rotate faster, but closed loop control of angular velocity and position is currently limited to this ceiling.

III. CONTROL OF SPHERICAL INDUCTION MOTOR

The system block diagram of the SIM is shown in Fig. 3, which mainly consists of the SIM actuation part and the rotor velocity sensing part. The torque of the SIM is controlled through the vector controller of each inductor. Vector control requires feed forward of the rotor velocity component along the inductor's force-generating direction [8], measured by the optical mouse sensors tracking scratches and imperfections on the rotor's outer layer and the angular velocity estimation process. The rotor's angular velocity and orientation are controlled with feedback. Resulting values can be passed to higher level application controls.

We first describe how a commanded arbitrary rotor torque vector results in the distribution of force output commands to the four inductors. Then, the angular velocity estimation process is briefly described with a method that is an improvement over our previous method [9]. Finally, we describe how the rotor's angular velocity and orientation is controlled.

A. Control of output torque

The output torque of the SIM is generated by the combination of thrust forces generated between each inductor and the spherical rotor. As shown in Fig. 2(a), four inductors are used in our implementation. Let the position vector of the inductor i ($i = 1 \cdots n$) on the surface be $\boldsymbol{p}_i$ and the unit vector denoting its thrust generating tangential direction be $\boldsymbol{s}_i$ as in Fig. 4. Note that we assume that the output thrust force

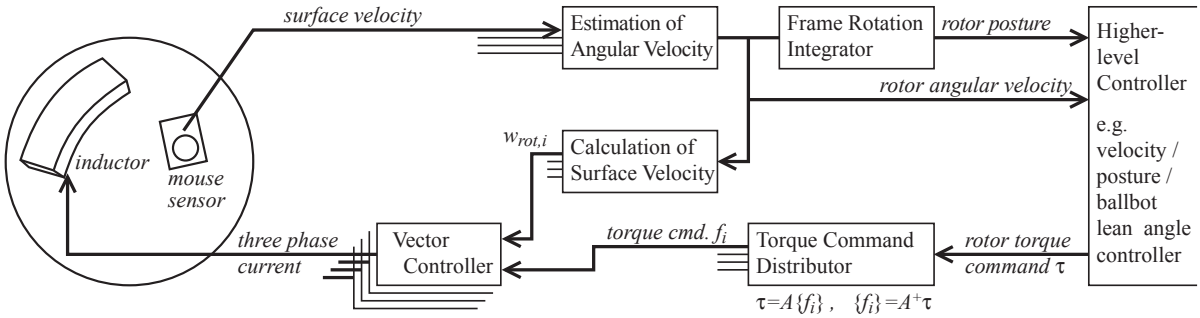


Fig. 3. Overall control block diagram of the SIM.

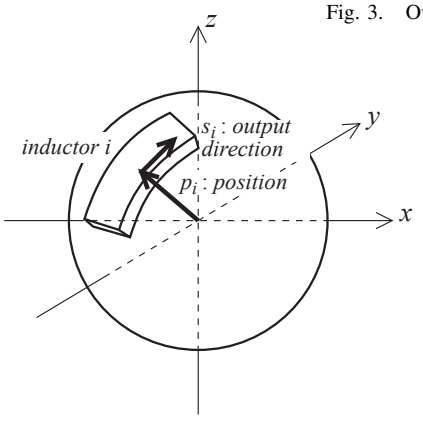


Fig. 4. Coordinate and parameter definition for torque generation at one inductor.

is generated at the point $\boldsymbol{p}_i$ whereas in actuality the force is distributed in a complex way over the inductor surface facing the rotor. (This approximation does not seem to cause a significant difference between theory and experiment.)

The output torque of inductor i is formulated by the outer product:

$$\boldsymbol{\tau}_i = \boldsymbol{p}_i \times (f_i \boldsymbol{s}_i) = (\boldsymbol{p}_i \times \boldsymbol{s}_i) f_i, \tag{1}$$

where f_i is the force output command of inductor i.

The total output of the SIM becomes

$$\boldsymbol{\tau} = \sum_{i=1}^{n} \boldsymbol{\tau}_i = \sum_{i=1}^{n} (\boldsymbol{p}_i \times \boldsymbol{s}_i) f_i. \tag{2}$$

The goal of the torque control is to determine f_i for each inductor i to generate an arbitrary torque $\boldsymbol{\tau}$.

If we let the outer product $\boldsymbol{p}_i \times \boldsymbol{s}_i$ be $\boldsymbol{t}_i$, the equation can be re-written in the matrix form:

$$\begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix} = \begin{pmatrix} t_{1,x} & \cdots & t_{n,x} \\ t_{1,y} & \cdots & t_{n,y} \\ t_{1,z} & \cdots & t_{n,z} \end{pmatrix} \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} = \boldsymbol{A} \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix}, \tag{3}$$

where $\boldsymbol{A}$ is the $3 \times n$ actuation matrix depending on the geometrical arrangement of the inductors.

If there are only three inductors, whose $\boldsymbol{t}$s are linearly independent, this equation can be solved to decide f_i:

$$\begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} = \boldsymbol{A}^{-1} \begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix}. \tag{4}$$

In the case of more than three inductors, the output f_i can be decided in several different ways. One straightforward idea is to use a pseudo inverse matrix of the A.

$$\begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} = \boldsymbol{A}^{+} \begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix}, \quad \boldsymbol{A}^{+} = \boldsymbol{A}^T (\boldsymbol{A}\boldsymbol{A}^T)^{-1}. \tag{5}$$

We used this equation in our implementation.

An alternative idea is to select fewer than n inductors to be activated when lower torque is needed, and to use the above equations for selected inductors to reduce total power consumption and fluctuation in output. Activated inductors consume current, incurring i^2R losses, to maintain magnetization which can be saved if they are turned off. Note that this scheme would require longer times to activate the inductor, so frequent switching may not be practical.

B. Inductor arrangement

As described above, the inductors of the SIM can be arranged arbitrarily on condition that $\boldsymbol{A}$ in (3) has an inverse or pseudo inverse, *i.e.,* the rank of $\boldsymbol{A}$ is three. There are two policies of arrangement. One is to arrange inductors orthogonally along $x-y, y-z, z-x$ planes, *i.e.* on the equator and meridians. This is the arrangement proposed in most of the previous work, in which output commands are basically independent to each other; there are specific inductors for rotation around the x, y or z axis. Another arrangement is to make each inductor have non-axis-specific driving ability, wherein commanded torque about specific axes are achieved by a combination of several inductors.

We chose the latter arrangement for our prototype SIM as in Fig. 2(a), referred to as a "skewed" arrangement, for several reasons. Our primary application is a ballbot that requires rotation in x and y axes for balancing and traveling about and an additional z, or yaw axis, rotation for pivoting to a specified azimuth. In that case, dedicated inductors z on equators require space, weight, power, and control systems even if they are seldom used. Second, the skewed arrangement permits the space occupied by the inductors to be smaller, or the surface area of the inductors to be wider for greater traction force. Finally, the arrangement appears to be novel and not used before which is better to show the effectiveness of our torque control method.

For the arrangement of the inductors shown in Fig. 2(a), there are three angles associated with each inductor as shown in Fig. 5. The angle ϕ is the angular position around the z

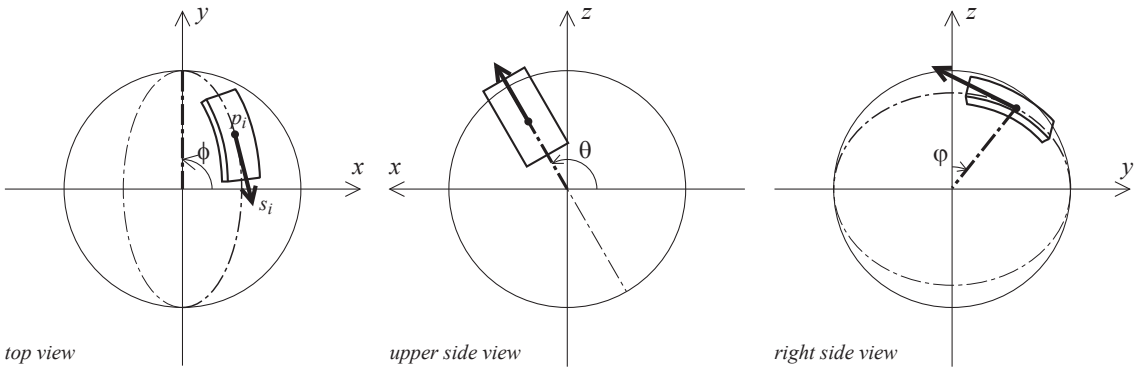


Fig. 5. Three angular parameters for positioning each inductor.

axis; 0°, 90°, 180°, 270° respectively for the four inductors. The angle θ is the angular tilt (the skew) from a meridian line. In our implementation, it was 30°. Note that it is zero in the orthogonal arrangement policy. The angle ψ determines the angular position of the inductor along the great circle. Because the thrust force at an inductor is generated along this great circle, ψ has no affect on torque generation, but is determined by mechanical design. (For example, in our application it is desirable to leave a large portion of the rotor free of inductors and sensors which favors concentrating the inductors near the north pole.)

The parameters $\boldsymbol{p}$ and $\boldsymbol{s}$ in (1) are derived as follows:

$$\begin{pmatrix} p_x \\ p_y \\ p_z \end{pmatrix} = \begin{pmatrix} \cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi \\ \sin\phi\sin\psi - \cos\phi\sin\theta\cos\psi \\ \cos\theta\cos\psi \end{pmatrix},$$
$$\begin{pmatrix} s_x \\ s_y \\ s_z \end{pmatrix} = \begin{pmatrix} \cos\phi\cos\psi - \sin\psi\sin\theta\sin\psi \\ \sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\cos\theta\sin\psi \end{pmatrix}. \quad (6)$$

Note that these are $(0,0,1)^T$ and $(1,0,0)^T$ after consecutive rotation ψ around y, θ around x, and ϕ around z. We used four inductors which have the same θ and ψ, and have 90° rotational symmetry as mentioned above. Using (1) through (5), the force output distribution can be obtained numerically:

$$\begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{pmatrix} = \begin{pmatrix} 0.000 & 0.577 & 0.500 \\ -0.577 & 0.000 & 0.500 \\ 0.000 & -0.577 & 0.500 \\ 0.577 & 0.000 & 0.500 \end{pmatrix} \begin{pmatrix} \tau_x \\ \tau_y \\ \tau_z \end{pmatrix}. \quad (7)$$

The equation means, *e.g.,* if $\tau_x = 2\,\mathrm{Nm}$ is commanded, inductors 2 and 4 should output 1.15 Nm, and all the inductors will output 1 Nm if $\tau_z = 2\,\mathrm{Nm}$. The angle θ is a contradictory parameter for torque generation. The SIM can output in x and y axes more effectively if θ is smaller while it is weak in z. If θ is larger it is better for torque around z. In our case, because we are focused on using the SIM for mobile robots, we chose $\theta = 30°$ which is the smallest value for a practical mechanical frame design.

C. Angular velocity measurement

We previously reported an optical angular motion sensing scheme for a sphere [9]. A typical optical mouse sensor has two velocity measurement axes. For each axis, let the sensor position be $\boldsymbol{q}_i$ with the unit vector along the sensing direction given by $\boldsymbol{u}_i$ $(i = 1 \ldots m)$ as in Fig. 6. When the rotor rotates at the angular velocity $\boldsymbol{\omega}$, each sensor measures its local surface velocity:

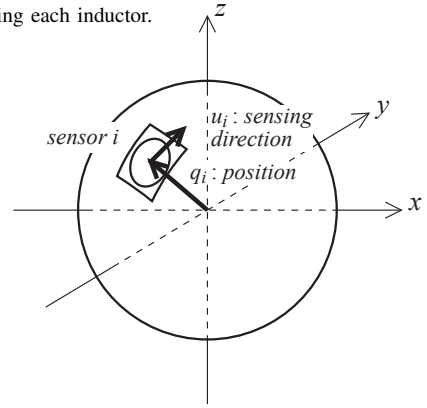


Fig. 6. Coordinate and parameter definition for angular velocity sensing.

$$v_{si} = \boldsymbol{u}_i \cdot (\boldsymbol{\omega} \times \boldsymbol{q}_i) = \boldsymbol{\omega} \cdot (\boldsymbol{q}_i \times \boldsymbol{u}_i). \quad (8)$$

To calculate the rotor angular velocity from the sensor readings, only 3 (i, j, k) axes are needed out of all m measuring axes.

$$\begin{pmatrix} v_{si} \\ v_{sj} \\ v_{sk} \end{pmatrix} = \begin{pmatrix} \boldsymbol{q}_i \times \boldsymbol{u}_i \\ \boldsymbol{q}_j \times \boldsymbol{u}_j \\ \boldsymbol{q}_k \times \boldsymbol{u}_k \end{pmatrix} \boldsymbol{\omega} = \boldsymbol{S}\boldsymbol{\omega}, \quad (9)$$

$$\boldsymbol{\omega} = \boldsymbol{S}^{-1} \begin{pmatrix} v_{si} \\ v_{sj} \\ v_{sk} \end{pmatrix}, \quad (10)$$

where $\boldsymbol{S}$ is a 3×3 matrix defined by the geometrical arrangement of sensors. In our previous method [9], we chose several meaningful triplets to calculate $\boldsymbol{\omega}$, and derived the estimation by using the weighted mean of those $\boldsymbol{\omega}$.

Although this method provided good angular measurements, we introduce here another method to improve the result in the case of noisy, saturated conditions. Equation (10) is extended by using a pseudo inverse matrix:

$$\begin{pmatrix} v_{s1} \\ \vdots \\ v_{sm} \end{pmatrix} = \begin{pmatrix} \boldsymbol{q}_1 \times \boldsymbol{u}_1 \\ \vdots \\ \boldsymbol{q}_m \times \boldsymbol{u}_m \end{pmatrix} \boldsymbol{\omega} = \boldsymbol{S}\boldsymbol{\omega}, \quad (11)$$

$$\boldsymbol{\omega} = \boldsymbol{S}^{+} \begin{pmatrix} v_{s1} \\ \vdots \\ v_{sk} \end{pmatrix}, \quad \boldsymbol{S}^{+} = (\boldsymbol{S}^T \boldsymbol{S})^{-1} \boldsymbol{A}^T. \tag{12}$$

Next, a masking variable d_i(=0,1) is introduced to (11):

$$\begin{pmatrix} d_1 v_{s1} \\ \vdots \\ d_m v_{sm} \end{pmatrix} = \begin{pmatrix} d_1(\boldsymbol{q}_1 \times \boldsymbol{u}_1) \\ \vdots \\ d_m(\boldsymbol{q}_m \times \boldsymbol{u}_m) \end{pmatrix} \boldsymbol{\omega} = \boldsymbol{S}\boldsymbol{\omega}. \tag{13}$$

If $d_i = 0$, the row i becomes $0 = 0$, which is equivalent to being removed from the equation. Before calculating the angular velocity using the pseudo inverse, if the sensor value v_{si} is considered to be reliable, set $d_i = 1$, otherwise, $d_i = 0$. Because the angular velocity of the rotor will not change much during the (10 millisecond) control period, v_{si} can be estimated from the previously obtained angular velocity, which is used to judge whether the current reading is within a reasonable range or should be ignored by appropriately setting the value of d_i.

The estimated angular velocity thus determined is used as the feed forward term in the vector control of the inductors,

$$\omega_{rot,i} = \boldsymbol{s}_i \cdot (\boldsymbol{\omega} \times \boldsymbol{p}_i). \tag{14}$$

The angular velocity is also used for higher-level control such as velocity and orientation control of the rotor.

D. Control of angular velocity

The angular velocity of the rotor can be controlled by, *e.g.,* proportional integral derivitive (PID) feedback. The torque command is calculated for each axis, for example in the x axis:

$$e_{\omega,x} = \omega_{cmd,x} - \omega_{act,x}\ , \quad \tau_x = K_{VP} e_{\omega,x} + K_{VI} \int e_{\omega,x} dt + K_{VD} \frac{d}{dt} e_{\omega,x}, \tag{15}$$

where ω_{cmd} and ω_{act} are the commanded and measured angular velocities around the x axis, and K_{VP}, K_{VI}, K_{VD} are the PID gains. PI control was used for our experiments.

E. Control of rotor orientation

Because there is no definite rotation axis in the SIM whereas a traditional 3-DOF joint has three rotationally driven axes, the Euler angles or the roll-pitch-yaw notations, which does not have uniformity, are not best way to express the rotation of the rotor. Therefore, we used a rotation matrix directly to command orientation of the rotor. (Note that the above three-angles-notation can be interpreted as a matrix, and the proposed method is applicable.)

Let the rotation matrix of the current orientation of the rotor be $\boldsymbol{R}_a$ and the commanded reference be $\boldsymbol{R}_c$. The idea is to force the three axes of the rotor to align with the reference axes. First, differences between the rotor axes and the reference axes are calculated:

$$(\boldsymbol{r}_{ax}, \boldsymbol{r}_{ay}, \boldsymbol{r}_{az}) = \boldsymbol{R}_a, \tag{16}$$

$$(\boldsymbol{r}_{cx}, \boldsymbol{r}_{cy}, \boldsymbol{r}_{cz}) = \boldsymbol{R}_c, \tag{17}$$

$$\theta_{px} = \cos^{-1} \frac{\boldsymbol{r}_{ax} \cdot \boldsymbol{r}_{cx}}{|\boldsymbol{r}_{ax}||\boldsymbol{r}_{cx}|} = \cos^{-1}(\boldsymbol{r}_{ax} \cdot \boldsymbol{r}_{cx})\ , \tag{18}$$

$$\boldsymbol{e}_{px} = \boldsymbol{r}_{cx} \times \boldsymbol{r}_{ax}, \tag{19}$$

where θ_{px} is the angle between $\boldsymbol{r}_{ax}$, the unit vector of axis x in $\boldsymbol{R}_a$, and $\boldsymbol{r}_{cx}$, and $\boldsymbol{e}_{px}$ indicates the rotational axis needed to move $\boldsymbol{r}_{ax}$ into alignment of $\boldsymbol{r}_{cx}$ using the nature of the outer product.

The angular velocity required to direct the x axes into alignment is given by:

$$u_{px} = K_{PP}\theta_{px} + K_{PI} \int \theta_{px} dt + K_{PD} \frac{d}{dt} \theta_{px}, \quad \boldsymbol{\omega}_{cmd} = u_{px} \frac{\boldsymbol{e}_{px}}{|\boldsymbol{e}_{px}|}, \tag{20}$$

where K_{PP}, K_{PI}, K_{PD} are the PID gains for feedback, u_{px} is the magnitude of required feedback, which is proportional to the difference between $\boldsymbol{r}_{ax}$ and $\boldsymbol{r}_{cx}$. The total angular velocity command will then be given by:

$$\boldsymbol{\omega}_{cmd} = u_{px} \frac{\boldsymbol{e}_{px}}{|\boldsymbol{e}_{px}|} + u_{py} \frac{\boldsymbol{e}_{py}}{|\boldsymbol{e}_{py}|} + u_{pz} \frac{\boldsymbol{e}_{pz}}{|\boldsymbol{e}_{pz}|}. \tag{21}$$

The angular velocity of the rotor is then controlled using the previously discussed velocity feedback control to rapidly align the rotor axes with the commanded axes.

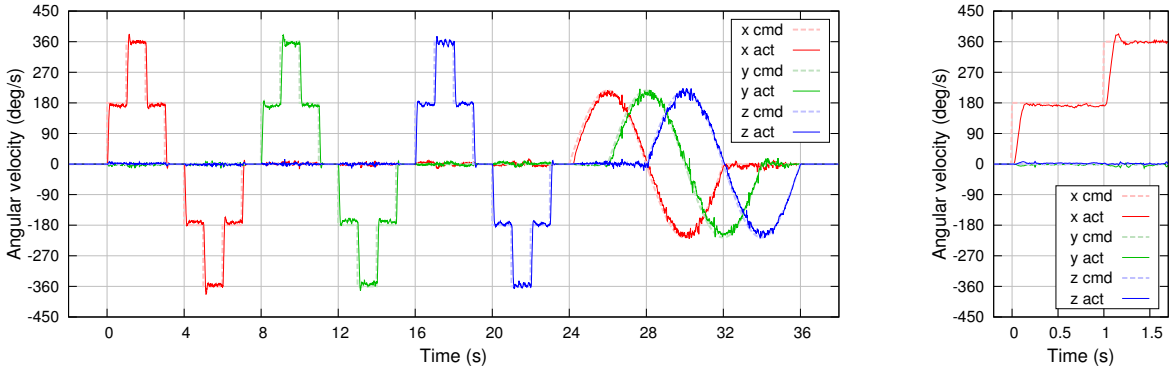
Note that in case of $\sin\theta_{px(yz)} = 0$, *i.e.,* $\theta_{px(yz)} = 0$ or π, this method does not work. In the first case there is nothing to do because the axes of the rotor already coincide with those of the reference. The latter is a singular dead point to be treated by some exceptional action which will hopefully rarely occur in practical situations. Though the proposed method lacks stability analysis, it worked experimentally.

IV. Experimental results

We carried out several experiments to measure the performance the SIM using our control methods, which are included in an accompanying video. The vector control parameters, the conservative PID gains for feedback were chosen empirically.

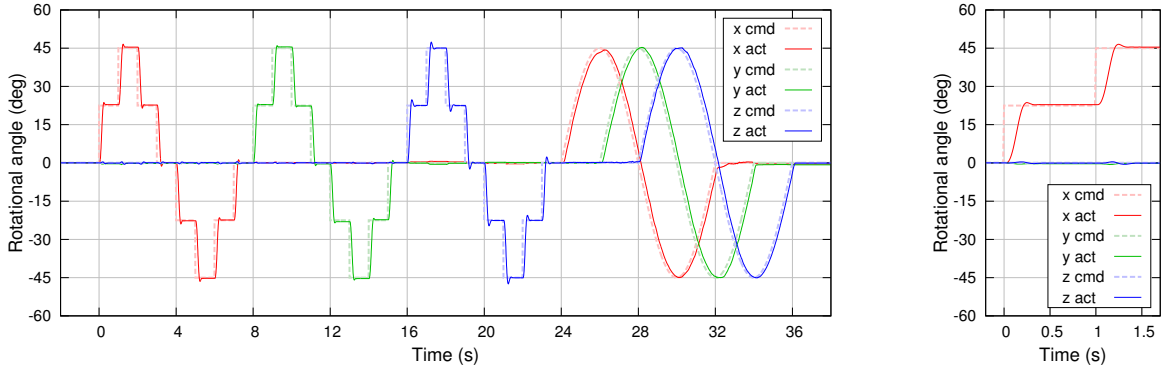
First, the output torque was measured. The thrust force on the surface of the sphere was measured using a force gauge in the orientation controlled, "locked rotor" condition and found to be up to approximately 40 N. A stick was attached to apply the force as in a scene "orientation keeping" in the video. With the radius of the sphere, it was equivalent to torque in the x and y axes, which are theoretically smaller than in the z axis, was approximately 4.5 Nm.

Second, the response of the SIM in angular velocity control and angular position control was examined, results of which are shown in Fig. 7 and Fig. 8, respectively. Both figures show good tracking control during 36 s of operation in three axes, with magnified views of the transient response shown in the beginning of each run. The reference commands include steps in three independent axes (x, y, and z) and sinusoidal waveforms combining three axes. Note that the amplitude of the sinusoidal velocity was limited due to the sensor limitation previously mentioned.



(a) Responses in whole period (b) Transient

Fig. 7. Responses of angular velocity control. "cmd" and "act" are for commanded reference and measured value.



(a) Responses in whole period (b) Transient

Fig. 8. Responses of angular position control. "cmd" and "act" are for commanded reference and measured value.

For velocity control, the step response time was about 0.1 s with a peak angular acceleration of 50 rad/s^2, and tracking results were good except for small amount of noisy vibration, which seemed to be caused by sensing fluctuation and torque ripple of each inductor. In position control, the step response time was about 0.2 s and smooth tracking results with small overshoots were observed. The motor consumed 200 W (50V, 4A) in steady, lower output state, and up to 1 kW in above maximum torque condition. It is rather high, meaning low efficiency, which should be improved in future works.

V. Conclusions

We have presented control methods for a spherical induction motor including torque distribution, sensing, angular velocity control, and orientation control. The torque control method allows inductors to be arranged freely around the rotor to optimize designs for particular applications. A similar statement obtains for the arrangement of sensors. A new prototype implementation was also developed, with experimental results showing the practical achievement of up to 4 Nm torque and 300 rpm rotation speed. Closed loop control of angular velocity and orientation was achieved with good response times. We believe the closed-loop SIM presented here can be a potential new prime mover for mobile robots.

VI. Acknowledgements

A part of this work was performed in the Microdynamic Systems Laboratory, The Robotics Institute, Carnegie Mellon University, as a part of the dynamically stable mobile robots project. This work was supported in part by NSF grant ECCS 1102147 and by KAKENHI(23760234) in Japan.

References

[1] T.B.Lauwers, G.A.Kantor, R.L.Hollis, "A Dynamically Stable Single-Wheeled Mobile Robot with Inverse Mouse-Ball Drive," *Proc. ICRA 2006*, pp 2884–2889, 2006

[2] U.Nagarajan, B.Kim, R.Hollis, "Planning in high-dimensional shape space for a single-wheeled balancing mobile robot with arms," *Proc. ICRA 2012*, pp 130–135, 2012

[3] M.Kumagai, T.Ochiai, "Development of a robot balanced on a ball – Application of passive motion to transport –," *Proc. ICRA 2009*, pp 4106–4111, 2009

[4] M.Kumagai, T.Ochiai, "Development of a Robot Balanced on a Ball – First Report, Implementation of the Robot and Basic Control –," *Journal of Robotics and Mechatronics*, vol.22 no.3, pp 348–355, 2010

[5] K.-M. Lee and C.-K. Kwan, "Design concept development of a spherical stepper motor for robotic applications," IEEE Trans. on Robotics and Automation, vol. 7, pp. 175-181, 1991.

[6] F. C. Williams, E. R. Laithwaite, and J. F. Eastham, "Development and design of spherical induction motors," Proc. IEE, vol. 47, pp. 471–484, 1959

[7] B. Dehez, G. Galary, D. Grenier, and B. Raucent, "Development of a spherical induction motor with two degrees of freedom," IEEE Trans. on Magnetics, vol. 42, no. 8, pp. 2077–2089, 2006

[8] M. Kumagai: "Development of a Linear Induction Motor and a Vector Control Driver", *SICE Tohoku chapter workshop material*, pp. 262–9 (in Japanese language), 2010

[9] M. Kumagai, R.L. Hollis, "Development of a three-dimensional ball rotation sensing system using optical mouse sensors", *ICRA 2011*, pp. 5038–5043, 2011

[10] M. Kumagai, R.L. Hollis, "Development and Control of a Three DOF Planar Induction Motor", *ICRA 2012*, pp. 3757–3762, 2012