2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Generalized Virtual Fixtures for Shared-Control Grasping in Brain-Machine Interfaces

Samuel T. Clanton, *Student Member, IEEE*, Robert G. Rasmussen,
Zohny Zohny, and Meel Velliste, *Member, IEEE*

I. Introduction

In brain-machine interface (BMI) prosthetic systems, recordings of brain activity are used to control external devices such as computers or robots. BMI systems that have shown the highest fidelity of control use neural signals recorded directly from microelectrodes in the brain to control upper-limb prostheses. These have progressed from allowing control of 2 and 3 dimensional movement of a cursor on a computer screen [1], [2] to control of robot arms in first four [3], [4] and more recently seven degrees-of-freedom (DoF) (Fig. 1) [5], [6]. These types of systems require methods to train users to control large numbers of DoF simultaneously.

Shared control is a concept originating in telerobotic and robotic surgery applications in which operator control of a robot is modified to keep robot movements within safe boundaries or enhance task performance [7], [8]. A kind of shared control system called the “Virtual Fixture” (VF) was introduced by Rosenburg [9] to describe the concept of embedded guidance or constraint within human-machine interactive systems. In systems that use virtual fixturing, a user controls the general movement of an effector robot but the virtual fixturing algorithm modulates that movement to enhance task performance. These algorithms guide the robot endpoint towards or away from specified areas of a workspace [10]–[12].

Shared control has been applied to task training in general for human-operated robots, where user-controlled robot movements are corrected towards goal trajectories as humans learn to control devices independently [13], [14]. The concept of shared control in brain-computer interfaces has been employed for manipulator stabilization [15], for cooperative task completion using reinforcement learning [16], for actuating coordinated robotic hand postures [17], and for assisting in control of an EEG-guided wheelchair [18]. In previous BMI work in our laboratory, the related concept of “deviation gain” was applied to 4-DoF control [3].

In this paper we present a new method for shared-control guidance. This method of “Positive-Span” Virtual Fixturing extends the concept of Virtual Fixtures to guide both translational and rotational DoF of a brain-controlled robot hand toward whole sets of robot poses that would allow an object to be grasped. This system was used to successfully train monkeys to operate the 7-DoF BMI [5], leading directly to the simplified system of “ortho-impedance” used to guide human subject BMI control in a similar experiment [6].

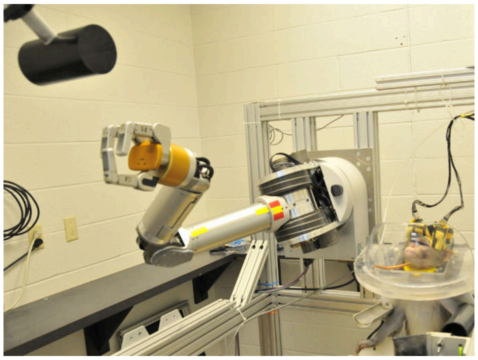

Fig. 1: 7-DoF brain-machine interface experiment [5].

The specific advantages of the Positive-Span Virtual Fixturing method include

1) Extension of Virtual Fixtures to coordinated motion in high-dimensional control spaces, e.g. to include control of both manipulator translation and rotation.
2) Motion towards or along irregular fixture shapes or clouds of points that can be used directly in the positive-span VF control law, without decomposition of constraint surfaces to lines or planes.

The underlying motivation for the creation of the Positive-Span Virtual Fixturing method comes from the desire to move towards integration of BMI- or otherwise user-controlled robotic systems with automated grasp or movement planning systems. These hybrid systems could be used for training or to enhance overall system performance in high-dimensional control applications. Grasp planning systems commonly produce an n-dimensional point set representing individual poses within the manifold of poses that would effectively grasp an object. The Positive-Span Virtual Fixturing system is compatible with direct constraint of robot movement towards these n-D point sets or manifolds.

II. Positive-Span Virtual Fixture Control Law

In a well-known formulation for Virtual Fixturing [10], a control law is described that constrains the admittance

S.T. Clanton is with the Rehabilitation Institute of Chicago, Chicago, Illinois. M. Velliste is with the Department of Neurobiology and Systems Neuroscience Institute, University of Pittsburgh, Pittsburgh, Pennsylvania. R.G. Rasmussen is with the Department of Bioengineering, University of Pittsburgh. Z. Zohny is with the Washington University Department of Neurosurgery, St. Louis, Missouri. This work was supported by NIH grants NS050256, NS070311, F30N060530, and DARPA grant W911NF-06-1-053.

978-1-4673-6357-0/13/$31.00 ©2013 IEEE

of a human-controlled system in directions defined instantaneously by a line or plane. The control law was applied as a "virtual contact" that governs an anisotropic admittance for a system that transduces human input force ($\boldsymbol{f}$) to robot output velocity ($\boldsymbol{v}$). The virtual fixture equation

$$\boldsymbol{v} = c(\boldsymbol{f_\delta} + c_\tau \boldsymbol{f_\tau}) \tag{1}$$

decomposes input into its projection to a line or plane representing the "preferred directions" of an input fixture ($\boldsymbol{f_\delta} = \boldsymbol{D_\delta f}$), and its residual $\boldsymbol{f_\tau} = \boldsymbol{f} - \boldsymbol{f_\delta}$, where $\boldsymbol{D_\delta}$ is the projection operator over the input fixture set $\boldsymbol{\delta}$, s.t. $\boldsymbol{D_\delta} = \boldsymbol{\delta}(\boldsymbol{\delta}^T\boldsymbol{\delta})^{-1}\boldsymbol{\delta}^T$. In this formulation, $\boldsymbol{\delta}$ contains a $3 \times m, 0 < m < 3$ time-varying matrix with column rank corresponding to a 1-D or 2-D orthonormal basis for a line or plane fixture. Output projections of input vectors to the line or plane defined by $\boldsymbol{\delta}$ were governed by the admittance c, while orthogonal movements were additionally attenuated by c_τ.

We propose a related control law to generalize fixtures over complex shapes in n-dimensional control space. Consider the corresponding control law (c omitted)

$$\boldsymbol{\nu} = \boldsymbol{\phi_\Delta} + c_\tau \boldsymbol{\phi_\tau} \tag{2}$$

which is identical to the control law above except that the projection of the $n \times 1$ input $\boldsymbol{\phi}$ is over the "positive span" of the fixture vector set $\boldsymbol{\delta}^*$. The positive span as defined by Mason [21] is $pos(\{\boldsymbol{z}_i\}) = \{\sum k_i \boldsymbol{z}_i | k_i \geq 0\}$ for a set of vectors $\{\boldsymbol{z}_i\}$, in this case the subspace of the input reachable by the basis vectors in $\boldsymbol{\delta}^*$ applied with positive multiplicative coefficients only. $\boldsymbol{\delta}^*$ is an $n \times m$ matrix of any number of constraints m in any number of dimensions n. The remaining control equations are

$$\boldsymbol{\phi_\Delta} = D_{\boldsymbol{\Delta}}\boldsymbol{\phi} \quad \boldsymbol{\phi}_\tau = \boldsymbol{\phi} - \boldsymbol{\phi_\Delta} \tag{3}$$

$$D_{\boldsymbol{\Delta}} = \boldsymbol{\Delta}(\boldsymbol{\Delta}^T\boldsymbol{\Delta})^+\boldsymbol{\Delta}^T \tag{4}$$

$$\boldsymbol{\Delta} = pos\_vectors(\boldsymbol{\delta}^*, \boldsymbol{\phi}) \tag{5}$$

where $\boldsymbol{\Delta}$ represents a selection of vectors from $\boldsymbol{\delta}^*$ that positively span the input command vector $\boldsymbol{\phi}$. $D_{\boldsymbol{\Delta}}$ is a projection operator onto these positively spanning vectors $\boldsymbol{\Delta}$. No rank restrictions are imposed on $\boldsymbol{\delta}^*$ and thus $\boldsymbol{\Delta}$. This new control law has the effect of preserving the portion of input directed toward the polytope described by $\boldsymbol{\delta}^*$ while attenuating its orthogonal component. Instead of constraining movement toward a 1-d or 2-d subspace, movement toward irregular fixture geometry in 3 dimensions or more can now be embedded within the VF control law.

As this algorithm relies on the projection of the input to some set of basis vectors $\boldsymbol{\Delta}$, the basic stability arguments presented in [10] apply to the fixturing system here, while noting that in high-dimensional control spaces input in one domain or type of movement (e.g. translational velocity) can result in projection to output in another domain (e.g. rotational velocity). If not desired, individual columns can take values only over dimensions in which coordinated motion is desired (for example see Section III-A).

A. Positive Span Vector Determination

The modified VF control law does not specify a particular projection of the input to the positive span of the $\boldsymbol{\delta}^*$ matrix. One formulation for a closest-point projection of $\boldsymbol{\phi}$ to the fixture vectors is to the surface of the n-dimensional convex cone bounded by the columns of $\boldsymbol{\delta}^*$. This is a quadratic programming problem to maximize the magnitude of the projection of $\boldsymbol{\phi}$ to the $\boldsymbol{\delta}^*$ vectors in

$$\begin{array}{ll} \text{minimize } \boldsymbol{\lambda} & \|\boldsymbol{\lambda\delta}^* - \boldsymbol{\phi}\|^2 \\ \text{subject to} & \boldsymbol{\lambda} > 0 \end{array}$$

where $\boldsymbol{\lambda}$ is a diagonal matrix of coefficients corresponding to each column vector in $\boldsymbol{\delta}^*$. When $\boldsymbol{\phi}$ is outside of the convex cone formed by the columns of $\boldsymbol{\delta}^*$ this can be performed by taking the convex hull of the origin and extended points on the lines indicated by $\boldsymbol{\delta}$, then using a closest-point algorithm to n-D convex hulls such as GJK [22]. Because of the poor performance of convex hull algorithms beyond the third dimension, a different projection to the positive span was used here in which a "greedy" set of fixture vectors for inclusion in $\boldsymbol{\Delta}$ were selected in turn to each maximally project onto the input. The algorithm to determine these vectors is similar to Gram-Schmidt orthonormalization except that each basis vector is chosen for its individual maximal projection on $\boldsymbol{\phi}$. The first two basis vectors are chosen as

$$\boldsymbol{\Delta}_1 = \text{colmax}_{\boldsymbol{\delta}^*}(\boldsymbol{\phi}^T\ \boldsymbol{\delta}^*)$$

$$\hat{\boldsymbol{\Delta}}_i = \frac{\boldsymbol{\Delta}_i}{\|\boldsymbol{\Delta}_i\|} \tag{6}$$

$$\boldsymbol{\phi}' = \boldsymbol{\phi} - (\boldsymbol{\phi} \cdot \hat{\boldsymbol{\Delta}}_1)\hat{\boldsymbol{\Delta}}_1 \tag{7}$$

$$\boldsymbol{\delta}^{*\prime} = \boldsymbol{\delta}^* - (\boldsymbol{\delta}^* \cdot \hat{\boldsymbol{\Delta}}_1)\hat{\boldsymbol{\Delta}}_1 \tag{8}$$

$$\boldsymbol{\Delta}_2 = \text{colmax}_{\boldsymbol{\delta}^{*\prime}}(\boldsymbol{\phi}'^{\,T}\ \boldsymbol{\delta}^*)$$

$$\boldsymbol{\phi}'' = \boldsymbol{\phi}' - (\boldsymbol{\phi}' \cdot \hat{\boldsymbol{\Delta}}_2)\hat{\boldsymbol{\Delta}}_2 \tag{9}$$

$$\boldsymbol{\delta}^{*\prime\prime} = \boldsymbol{\delta}^{*'} - (\boldsymbol{\delta}^{*'} \cdot \hat{\boldsymbol{\Delta}}_2)\hat{\boldsymbol{\Delta}}_2 \tag{10}$$

$$\tag{11}$$

where $\text{colmax}_{\boldsymbol{\delta}}(y)$ returns the column of matrix $\boldsymbol{\delta}$ on which y projects maximally. Numerical subscripts refer to matrix columns. The algorithm terminates when the maximum projection of $\boldsymbol{\phi}'^{\cdots\prime\prime}$ on any column in $\boldsymbol{\delta}^{*\prime\cdots\prime\prime} \leq 0$.

Simple illustrative examples of selecting fixture vectors $\boldsymbol{\delta}^*$ include movement toward a region of arbitrary shape and movement along a path. As an example of movement toward a shape, Fig. 2 (left) shows some translational $\boldsymbol{\delta}$ vectors directed at the vertices of a cube-shaped spatial target. For rotations, angular velocity vectors in $\boldsymbol{\delta}$ can be generated as infinitesimal rotations from the current manipulator orientation leading towards target orientations. For any type of control space, if the robot is located on the interior of a target region, it will move freely if fixture vertices are included in $\boldsymbol{\delta}$ to span control dimensions, as shown in Fig. 2 (right). The system will behave similarly for movement towards or inside more complex shapes or point clouds requiring more $\boldsymbol{\delta}$ vectors to define (see examples in Results).

To place a fixture that constrains a robotic manipulator toward and along a path, column vectors in $\boldsymbol{\delta}$ can be set to point towards the nearest point on the path and points along it in desired directions of translation. The tightness with which the path is followed is dictated by the spread of the basis vectors pointing along the path (Fig. 3). Similar reasoning can be followed to place angular velocity constraint vectors toward more closely or widely spread rotations on a path of desired orientations. Combinations of region pursuit and path following can be performed by choosing $\boldsymbol{\delta}^*$ vectors pointing to selected local regions of target geometry.

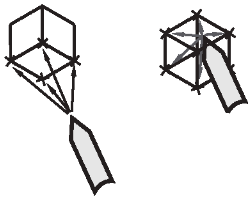

Fig. 2: Illustration of a 4-column $\boldsymbol{\delta}^*$ matrix directing motion towards a cube (vectors on interior of convex cone not shown).

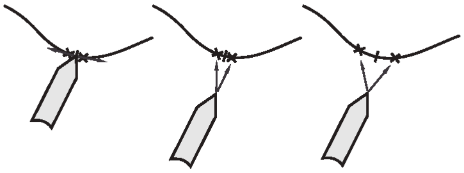

Fig. 3: (Left) Tight bidirectional path-following with vectors pointed at closely-spaced points on the path, (Center) vectors through the same points on the path guide the robot when the manipulator leaves the path, and (Right) more forgiving path following with more widely-spaced $\boldsymbol{\delta}^*$ vectors.

III. Results

A. On-Line Virtual Fixturing Experiment

An experiment was devised in which the performance of the modified VF system was tested for online robot control assistance in a grasping task. The shared control and robotic systems used in this experiment were identical to that in the BMI experiments reported in [5], but the control input was provided by human subjects using hand-held controllers rather than a brain-machine interface. The purpose of this experiment was to prototype the fixturing system to be used for BMI training and verify that the modified VF system was able to adjustably aid subjects in real time as designed. Experimental trials were approved by the Institutional Review Boards at the University of Pittsburgh and Carnegie Mellon University.

In this experiment, a Barrett WAM robotic arm (Barrett Technologies, Cambridge, Mass., USA) was mounted across from a DENSO (DENSO Robotics, Long Beach, CA, USA) industrial 6-DoF arm (Fig. 4) with a cylindrical grasp target attached to it. Subjects stood in the room with the robots and controlled the WAM using a handheld game controller (Sony Sixaxis Controller, Sony Corporation, NY, NY). Four degrees of freedom were input through the movement of two thumb-controlled joysticks, and an additional 2 input DoF were provided by two sets of pressure-sensitive trigger buttons. The control space of the robot was the set of 3-D linear and 3-D angular velocities centered at the palm of a Barrett Hand (Barrett Technologies) mounted at the end of the WAM, with no control over the fingers in this experiment. Each input DoF was mapped to a movement DoF that spanned either the 3 linear or 3 rotational degrees of freedom at the palm, so that for instance movement of one of the sticks upward would move the robot toward a corner of the workspace, movement of the other stick left would move it towards an orthogonal corner, and pushing one of the buttons would rotate the robot endpoint in the direction of the "corner" of angular velocities with a simultaneous yawing, pitching, and rolling movement.

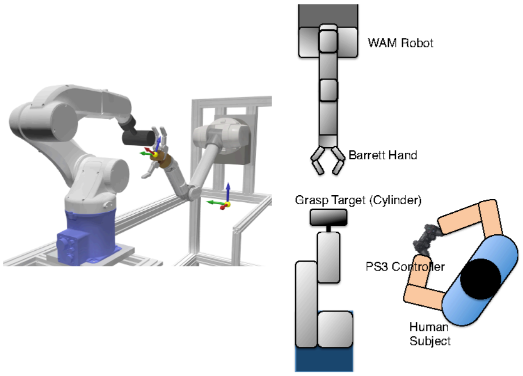

Fig. 4: (Left) A view of the robotic setup in the human subject task, (Right) A schematic of the setup in the room including the human subject.

At the beginning of each experimental trial, one of 30 combinations of linear and rotational targets was selected and the robots moved to corresponding starting positions. A target region was defined as the WAM robot reaching within 12 cm and 40 total degrees of orientation from being directly in front of the target cylinder[*].

As each trial began, subject movement commands input from the controller were sampled at 30 ms intervals, filtered by the Positive-Span Fixturing algorithm, and passed to the robot. Commanded WAM velocities output by the positive span VF system were achieved by the WAM using PID control implemented in custom low-level control software. Each trial would register as a failure after 5-7 seconds[†] unless the WAM endpoint entered the target region during this time. At each 30 ms interval, vectors in the $\boldsymbol{\delta}^*$ matrix were generated pointing from the current robot pose to the linear and rotational target boundary points generated at the beginning of the trial. Linear and rotational examples of this are shown in Fig. 5. Here, each column embedded either

[*] Target sizes were chosen such that the rotational and translational difficulty were subjectively similar to the experimenters. Rotational targets were generated as random quaternions of 40 degrees of total rotation from the target. [†] corresponding to primate experiment scheme to minimize "lazy" delayed control strategies.

linear or rotational constraints but not both, i.e. $\boldsymbol{\delta}^* = [\boldsymbol{\delta}_t^* \ \boldsymbol{\delta}_r^*]$, where columns in $\boldsymbol{\delta}_t^* = \begin{bmatrix} x & y & z & 0 & 0 & 0 \end{bmatrix}^T$ and columns in $\boldsymbol{\delta}_r^* = \begin{bmatrix} 0 & 0 & 0 & r_x & r_y & r_z \end{bmatrix}^T$ such that subjects could move in ranges of linear and rotational DoF without forcing the manipulator toward individual 6-D grasps (see Section III-C where coordinated 6-D poses were used).

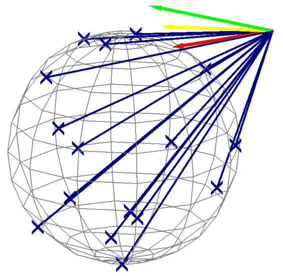

Fig. 5: Translation target set with $\boldsymbol{\delta}^*$ vectors. Vectors show positive span VF output from single input for $c_\tau = \{1.0, 0.5, 0\}$ in {green, yellow, red}.

Nine subjects who had not previously controlled the robot each performed 80 experimental trials with the Positive Span Virtual Fixturing-enabled robot control system. c_τ values for all 6 dimensions were initially set to 0.2 such that 80% of control error was attenuated. During the first 60 trials, $\boldsymbol{c_\tau}$ values were increased at a constant rate per trial until the robot was fully controlled by each subject starting on the 61st trial. An example of how the Positive Span Virtual Fixturing method could affect trajectories over a whole trial when applied with different levels of c_τ is shown in Fig. 7.

Subject performance was measured using the average deviation from direct movement from the starting position to the target. Rotational deviation from the closest path to target was calculated using the Slerp algorithm [23]. The relationship between error attenuation coefficients c_τ and control error are indicated in Fig. 8 and 9. Subjectively, users felt that smaller amounts of shared-control assistance ($c_\tau = 0.8$) helped to keep the system under control while learning the task, especially for rotational DoF. Higher amounts of shared control ($c_\tau < \ 0.5$) were felt by subjects to be too restrictive and not allow subjects to explore the control space, even if success rates were higher. These results influenced the policy regarding adjustment of shared control parameters during the monkey BMI experiments (see below).

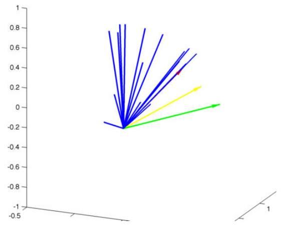

Fig. 6: Rotational constraint. The target region is difficult to represent, but angular velocity $\boldsymbol{\delta}^*$ vectors derived from this target space are shown (blue). Positive span VF output from single input are shown for $c_\tau = \{1.0, 0.5, 0\}$ in {green, yellow, red}.

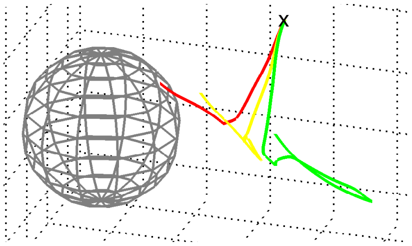

Fig. 7: Translational trajectory for $c_\tau = \{1, 0.5, 0\}$ in {green, yellow, red} given identical whole-trial input recorded from the experiment.

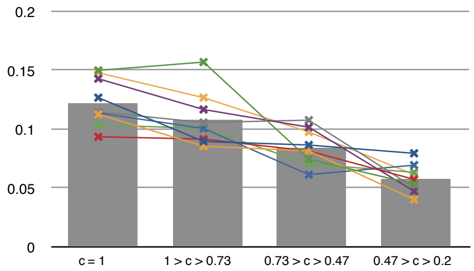

Fig. 8: Error attenuation level c_τ vs. mean trial translational error. Points (x) show performance averaged over trials for each subject at different assistance levels, and average over subjects as bars. Improvements in individual performance vs. no assistance are significant at $p < 0.05$ for 0/10 subjects with low assistance, 5/10 subjects with medium assistance, and 10/10 with high assistance.

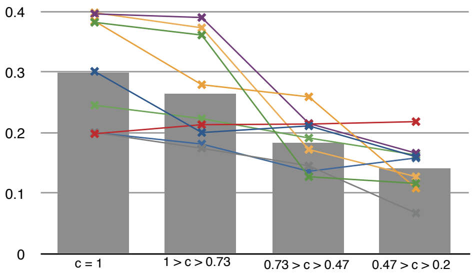

Fig. 9: Error attenuation level c_τ vs rotational error. Differences in performance from no assistance are significant at $p < 0.05$ for 1/10 subjects with low assistance, 4/10 with medium assistance, and 7/10 with high assistance.

B. *On-Line Brain-Machine Interface Experimentation*

In the 7-DoF monkey BMI experiment, the modified VF system used in the online human control experiment was applied to monkeys learning to control the device using a BMI. Instead of control commands originating with a hand-held controller, the activity of neurons was directly recorded and output as 6-DoF motion commands that were fed directly into the modified VF system during training. Due to the difficulty of training monkeys to perform oriented grasping tasks, small $\boldsymbol{c_\tau}$ values were used during initial control trials in order to allow the monkeys to associate trial completion with a juice reward delivered at the end of each successful trial. As the monkeys began to understand the task goal, the shared control values were adjusted so that success rates were kept high enough ($>\sim 40\%$) for the monkeys to maintain attention to the task. Because of the subjective results from the human experiment, $\boldsymbol{c_{\mathcal{T}}}$ was raised to around 0.8 as soon as possible. $\boldsymbol{c_{\mathcal{T}}}$ then remained at $0.8 - 0.95$ until monkeys mastered the task. These brain-machine interface results are beyond the scope of this paper and reported in [5]. A simplified version of this system was later used successfully in a human BMI experiment [6].

C. *Offline Grasp Pose Manifold Simulation*

A simulation was created to show how this type of fixturing can be used for more complex grasping tasks. The GraspIt! grasp planning software system [19], [20] was used to generate a point set of 6-D grasp translation and orientation poses that would allow a virtual Barrett Hand (Barrett Technologies, Cambridge, MA) to be in a form-closure grasp upon flexing the fingers (see [24]). 10000 random 6-DoF simulated user commands were generated in the interval $u_i = -0.5 \ldots 0.5$. To focus the fixturing method to grasp pose orientations on the proximal side of the target object, a heuristic method was used to select the $d/3$ most proximal target points as the fixturing points, where d is the distance from robot to the closest target point (in cm). The set of 10000 random commands was applied to the robot starting with the pose shown in Fig. 10 using vectors from the current robot position to the heuristically chosen points as VF fixtures for at each time step. Resulting trajectories at different levels of $c_{\mathcal{T}}$ are shown in Fig. 13. The highest $c_{\mathcal{T}}$ at which the hand converged on a target was 0.5 (results shown in Table I). Result poses from the same control inputs at different $c_{\mathcal{T}}$ levels are shown in Fig. 13.

$c_{\mathcal{T}}$	0.0	0.1	0.2	0.3	0.4	0.5
Steps	3712	4210	4792	5849	6788	8451

TABLE I: Steps to convergence for different levels of $c_{\mathcal{T}}$, higher levels did not converge in 10000 steps.

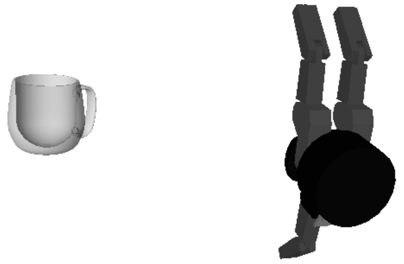

Fig. 10: Initial pose in mug grasping simulation with Barrett Hand model.

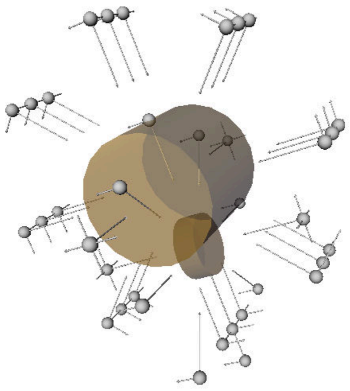

Fig. 11: Example initial poses generated by grasp planner. Spheres are at the center of the palm, lines indicate the direction normal to the palm and the direction of the thumb.

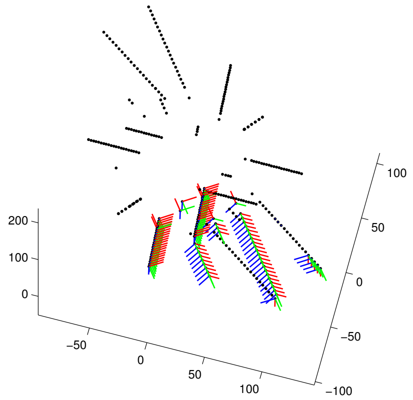

Fig. 12: Form-closure grasps exported from the grasp planner into the Positive-Span Virtual Fixturing system. Orientations chosen by the heuristic mechanism at the first simulation time step are shown.

IV. Conclusions and Future Work

In this work we present a new control law that generalizes the concept of virtual fixtures to high dimensional control tasks and to irregular fixture shapes. We show results from using this novel VF system in an online control task for simple grasping, then an offline simulation that illustrates its use in a more complex task. The application of our VF system has thus far been in 6 and 7-DoF tasks to be controlled using a brain-machine interface. For control of robots that could perform fine dextrous grasping and manipulation through BMI or any control interface, the VF method described here prototypes the integration of virtual fixturing with real-time grasp planning that includes coordinated finger motions [20]. Its generalization of VF over large numbers of DoF indicates its potential for application to guidance in advanced robotic surgical and telepresence applications [25]. We intend this algorithm to provide a basic framework for extension of shared control to new task domains.

Acknowledgments

This work was completed under the advisorship of Andrew Schwartz in his laboratory at the University of Pittsburgh. We would like to thank Yin Zhang for his assistance in data collection.

References

[1] M. D. Serruya, N. G. Hatsopoulos, L. Paninski, M. R. Fellows, and J. P. Donoghue, "Instant neural control of a movement signal." *Nature*, vol. 416, no. 6877, pp. 141–142, 2002.

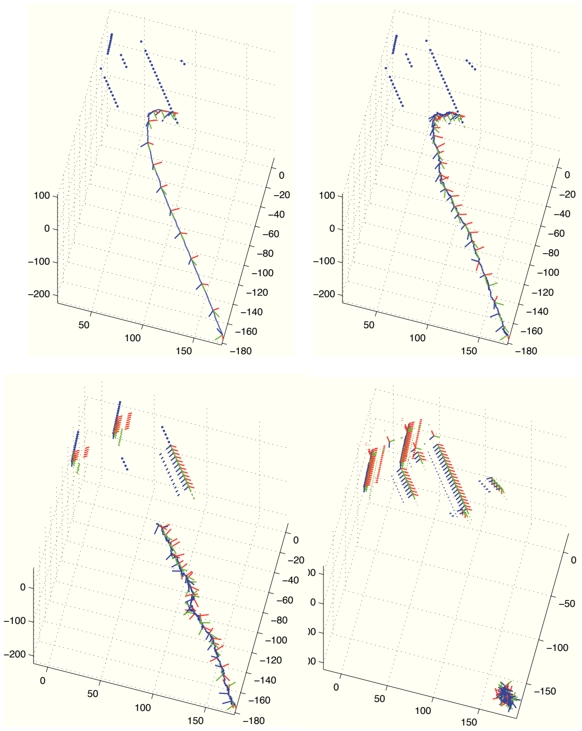

Fig. 13: Trajectories of simulated robot for identical command set with c_τ set to $0.0, 0.5, 0.7, 1.0$ from top left to bottom right. Target orientations are shown for heuristically chosen targets during the last time step.

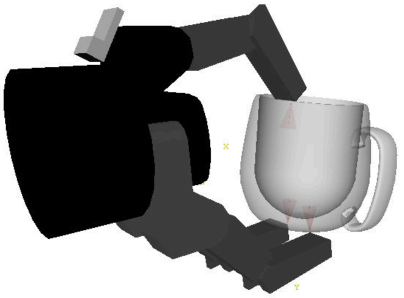

Fig. 14: Final grasp output from the planner with $c_\tau = 0$.

[2] D. M. Taylor, S. I. H. Tillery, and A. B. Schwartz, "Direct cortical control of 3D neuroprosthetic devices." *Science*, vol. 296, no. 5574, pp. 1829–1832, 2002.

[3] M. Velliste, S. Perel, M. C. Spalding, A. S. Whitford, and A. B. Schwartz, "Cortical control of a prosthetic arm for self-feeding," *Nature*, vol. 453, no. 7198, pp. 1098–1101, 2008.

[4] L. R. Hochberg, D. Bacher, B. Jarosiewicz, N. Y. Masse, J. D. Simeral, J. Vogel, S. Haddadin, J. Liu, S. S. Cash, P. van der Smagt, and J. P. Donoghue, "Reach and grasp by people with tetraplegia using a neurally controlled robotic arm." *Nature*, vol. 485, no. 7398, pp. 372–375, 2012.

[5] S. T. Clanton, "Brain-Computer Interface Control of an Anthropomorphic Robotic Arm," Ph.D. dissertation, Citeseer, 2011.

[6] J. L. Collinger, B. Wodlinger, J. E. Downey, W. Wang, E. C. Tyler-Kabara, D. J. Weber, A. J. McMorland, M. Velliste, M. L. Boninger, and A. B. Schwartz, "High-performance neuroprosthetic control by an individual with tetraplegia," *The Lancet*, vol. 381, no. 9866, pp. 557–564, Dec. 2012.

[7] O. Khatib, "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots," *The international journal of robotics research*, vol. 5, no. 1, pp. 90–98, 1986.

[8] R. Kumar, G. Hager, A. Barnes, P. Jensen, and R. Taylor, "An Augmentation System for Fine Manipulation BT - Lecture Notes in Computer Science," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2000*, S. Delp, A. DiGoia, and B. Jaramaz, Eds. Berlin, Heidelberg: Springer Berlin / Heidelberg SN -, 2000, vol. 1935, no. Chapter 99, pp. CH404–CH404.

[9] L. B. Rosenberg, "Proceedings of IEEE Virtual Reality Annual International Symposium," in *IEEE Virtual Reality Annual International Symposium*. IEEE, 1993, pp. 76–82.

[10] A. Bettini, P. Marayong, S. Lang, A. M. Okamura, and G. D. Hager, "Vision-assisted control for manipulation using virtual fixtures," *Robotics, IEEE Transactions on*, vol. 20, no. 6, pp. 953–966, 2004.

[11] D. Kragic, P. Marayong, M. Li, A. M. Okamura, and G. D. Hager, "Human-Machine Collaborative Systems for Microsurgical Applications," *The international journal of robotics research*, vol. 24, no. 9, pp. 731–741, 2005.

[12] J. Abbott, P. Marayong, and A. Okamura, "Haptic Virtual Fixtures for Robot-Assisted Manipulation BT - Springer Tracts in Advanced Robotics," in *Robotics Research*, S. Thrun, R. Brooks, and H. Durrant-Whyte, Eds. Berlin, Heidelberg: Springer Berlin / Heidelberg SN -, 2007, vol. 28, no. Chapter 5, pp. 49–64.

[13] Y. Li, J. Huegel, V. Patoglu, and M. O'Malley, "Progressive shared control for training in virtual environments," *EuroHaptics conference, 2009 and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems. World Haptics 2009. Third Joint*, pp. 332–337, 2009.

[14] M. O'Malley, "Progressive haptic and visual guidance for training in a virtual dynamic task," *Haptics Symposium*, pp. 343–350, 2010.

[15] H. K. Kim, J. Biggs, W. Schloerb, M. Carmena, M. A. Lebedev, M. A. L. Nicolelis, and M. A. Srinivasan, "Continuous shared control for stabilizing reaching and grasping with brain-machine interfaces," *Biomedical Engineering, IEEE Transactions on*, vol. 53, no. 6, pp. 1164–1173, 2006.

[16] J. C. Sanchez, B. Mahmoudi, J. DiGiovanna, and J. C. Principe, "Exploiting co-adaptation for the design of symbiotic neuroprosthetic assistants," *Neural Networks*, vol. 22, no. 3, pp. 305–315, 2009.

[17] C. Cipriani, F. Zaccone, S. Micera, and M. C. Carrozza, "On the Shared Control of an EMG-Controlled Prosthetic Hand: Analysis of User-Prosthesis Interaction," *Robotics, IEEE Transactions on*, vol. 24, no. 1, pp. 170–184, 2008.

[18] X. Perrin, R. Chavarriaga, F. Colas, R. Siegwart, and J. d. R. Millán, "Brain-coupled interaction for semi-autonomous navigation of an assistive robot," *Robotics And Autonomous Systems*, vol. 58, no. 12, pp. 1246–1255, 2010.

[19] A. Miller and P. Allen, "GraspIt!" *IEEE Robotics & Automation Magazine*, vol. 11, no. 4, pp. 110–122, Dec. 2004.

[20] M. T. Ciocarlie, S. T. Clanton, M. C. Spalding, and P. K. Allen, "Biomimetic grasp planning for cortical control of a robotic hand," in *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on Intelligent Systems and Robotics*. IEEE, 2008, pp. 2271–2276.

[21] M. T. Mason, *Mechanics of Robotic Manipulation*. MIT Press, 2001.

[22] E. Gilbert, D. Johnson, and S. Keerthi, "A fast procedure for computing the distance between complex objects in three-dimensional space," *IEEE Journal on Robotics and Automation*, vol. 4, no. 2, pp. 193–203, Apr. 1988.

[23] K. Shoemake, "Animating rotation with quaternion curves," in *SIGGRAPH '85*. New York, NY, USA: ACM, 1985, pp. 245–254.

[24] A. Miller, S. Knoop, H. Christensen, and P. Allen, "Automatic grasp planning using shape primitives," *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, pp. 1824–1829, 2003.

[25] F. Ryden and H. J. Chizeck, "Forbidden-region virtual fixtures from streaming point clouds: Remotely touching and protecting a beating heart," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, Oct. 2012, pp. 3308–3313.