Building a Universal Planar Manipulator *

Dan Reznik Emil Moshkovich John Canny

EECS Dept., UC-Berkeley, CA 94720-1776 USA

Abstract

In previous publications we described a novel approach to parallel manipulation in the plane based on a single horizontally-vibrating surface. Our approach is part of an exciting trend in planar manipulation based on non-prehensile methods. Previously, we have described our manipulation algorithm abstractly along with a set of simulation results. Here we describe the challenges involved in building a prototype. Practical solutions which map the algorithm's basic building blocks into the real world are presented and tested. With these building blocks we believe we are significantly closer to a working prototype.

1 Introduction

Recent trends in planar manipulation have been moving towards non-pick-and-place approaches. Novel ideas have sprung up based on non-prehensile manipulation, including part tapping [1], toppling [2] and flow over programmable vector fields [3, 4, 5]. Non-prehensile manipulation has the advantage of often simpler control (sometimes open-loop), and no gripper clutter.

Drawing from ideas of vibration-based part manipulation [6, 7, 8] and minimalist RISC robotics [9], in previous publications we have described an alternative approach to part feeding and parallel manipulation based on a single horizontally-vibrating plate [10, 11, 12]. Our main result has been to show that a single rigid horizontal plate is a universal parallel planar manipulator. One researcher has reported successful experimental results utilizing our method in the simpler case of parts feeding [13]; our own experimental results to date have been only preliminary, due to difficulties in the physical implementation of certain "manipulation primitives".

In this paper we will focus on the implementation issues faced by our project. In Section 2 we concisely review the concepts presented in previous publications. In Section 3 we explore in detail all practical implementation issues and describe how we solved them. In Section 4 we present experimental results testing our new experimental setup, showing significant progress towards a final and working prototype. Conclusions are presented in Section 5. Mathematical derivations and technical details can be found in the Appendices.

* Support provided in part by NSF grant CDA-9726389: "Challenges in CISE: Planning and Control for Massively Parallel Manipulation".

2 Review

2.1 The Manipulation Algorithm

The manipulation problem being addressed is illustrated in Figure 1(a). N parts lie at known positions $P_i,\ i = 1...N$ within a bounded area of the plane. A desired motion $\Delta P_i = (\Delta P_i^x, \Delta P_i^y)^T$ is prescribed for each part, e.g., along a trajectory associated with a high-level task such as part mating, sorting, etc. Define a set $C_j,\ j = ...M,\ M \geq 2N$ of points in the plane.

Our manipulation algorithm can alter parts' positions via a special motion primitive: parts can be "told" to rotate a *constant* distance d about any of the C_j's. This is unlike a rigid rotation for which part's displacements would be proportional to their distance from C. We consider d sufficiently small so the primitive rotation is approximately straight and along the tangent, as shown in Figure 1(b). This primitive causes parts to *flow* along a vector field $\phi_C = (\phi_C^x, \phi_C^y)^T$ defined as:

$$\phi_C = \frac{(P-C)^\perp}{||P-C||}$$

Note that at any point P, ϕ_C is unit and perpendicular to $P-C$. It can be shown that the family of these fields is not closed under addition, i.e., $\{\phi_{C_j}\}, j = 1...M$ will, in general, span an M-dimensional space. Compare this with the linear space of rigid rotations which is closed at dimension 3. [11].

Define a set of small scalars $d_j, j = 1...M$. Define $\Delta P_i'$ as part i's net displacement after it has flowed a distance d_j along ϕ_{C_j}, sequentially, for $j = 1...M$. Since the d_j's are small, ordering effects can be ignored (the Lie-Brackets are negligible [14]), and we can write a linearized expression for $\Delta P_i'$:

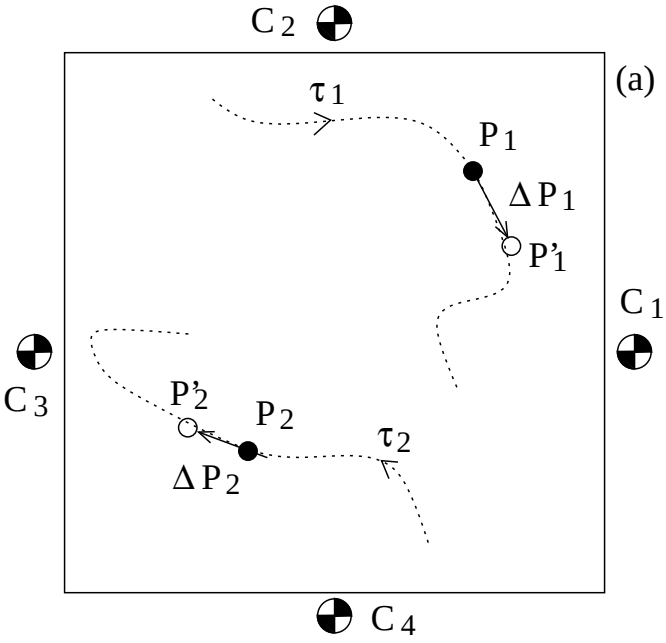
$$\Delta P_i' = \sum_{j=1}^{M} d_j \phi_{ij},\ i = 1...N \qquad (1)$$

where ϕ_{ij} is simply ϕ_{C_j} evaluated at P_i. The above can be expressed succinctly as the following linear system:

$$\Delta P = \Phi . d \qquad (2)$$

With:

$$\Delta P = \begin{bmatrix} \Delta P_i^x \\ - - - - - \\ \Delta P_i^y \end{bmatrix}_{2N \times 1}$$



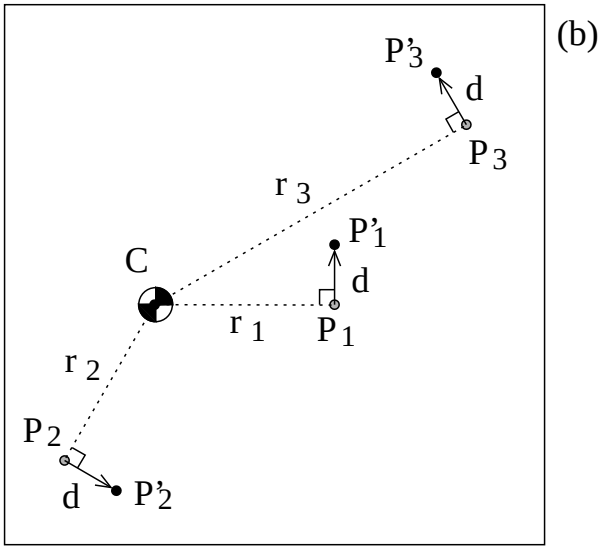


Figure 1: (a) The planar manipulation problem: N parts P_i need to execute a motion ΔP_i, e.g., along trajectories τ_i specified by some high-level task (assembly, sorting, etc.). A set of M points C_j is pre-specified about which the parts can execute a special type of rotation (see below). In the picture, $N = 2$ and $M = 4$. (b) The non-linear rotation primitive used by the manipulation algorithm: all parts P_i flow tangentially with respect to C by a specified d.

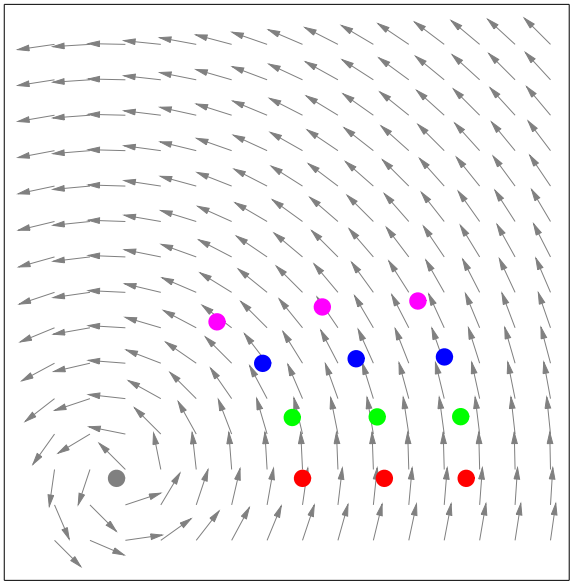


Figure 2: Race Track Experiment: three parts are allowed to "race" simultaneously (i.e., flow) along a non-linear rotation field. Their initial positions are all along a line directly to the right of the center of rotation, which is located at the lower left corner of the field. Four consecutive snapshots of the motion are shown. As expected, the inner parts advance more rapidly than the outer ones.

$$\Phi = \begin{bmatrix} \phi^x_{ij} \\ \text{- - - - -} \\ \phi^y_{ij} \end{bmatrix}_{2N \times M}$$
$$d = [d_j]_{M \times 1}$$

The manipulation algorithm can be summarized as follows: (1) solve Equation 2 for d [i.e., compute $\Phi^{-1}.\Delta P$]; (2) flow parts d_j amount along ϕ_{C_j}, sequentially, for $j = 1...M$. In the end, we expect:

$$\Delta P'_i \cong \Delta P_i, \ i = 1...N$$

Define the approximation error ϵ as:

$$\epsilon = \frac{1}{N} \sum_{i=1}^{N} ||\Delta P'_i - \Delta P_i||$$

Visualization of the concepts discussed in this Section is provided in Figures 2 and 3.

2.2 Time-Asymmetric Motion

Consider a horizontal surface S constrained to move along x. Let the surface's motion be periodic, with velocity profile $\nu_s(t)$, $\nu_s(t) = \nu_s(t+T)$. Consider a part P of mass m lying on S, with velocity ν_p. Assume S's acceleration relative to P is high enough so that (i) the part is always sliding on S and (ii) the part's speed ν_p is constant within one cycle, i.e., frictional forces are negligible compared to inertia. The average Coulomb friction $\bar{f}_{1d}$ applied to the part per cycle is given by:

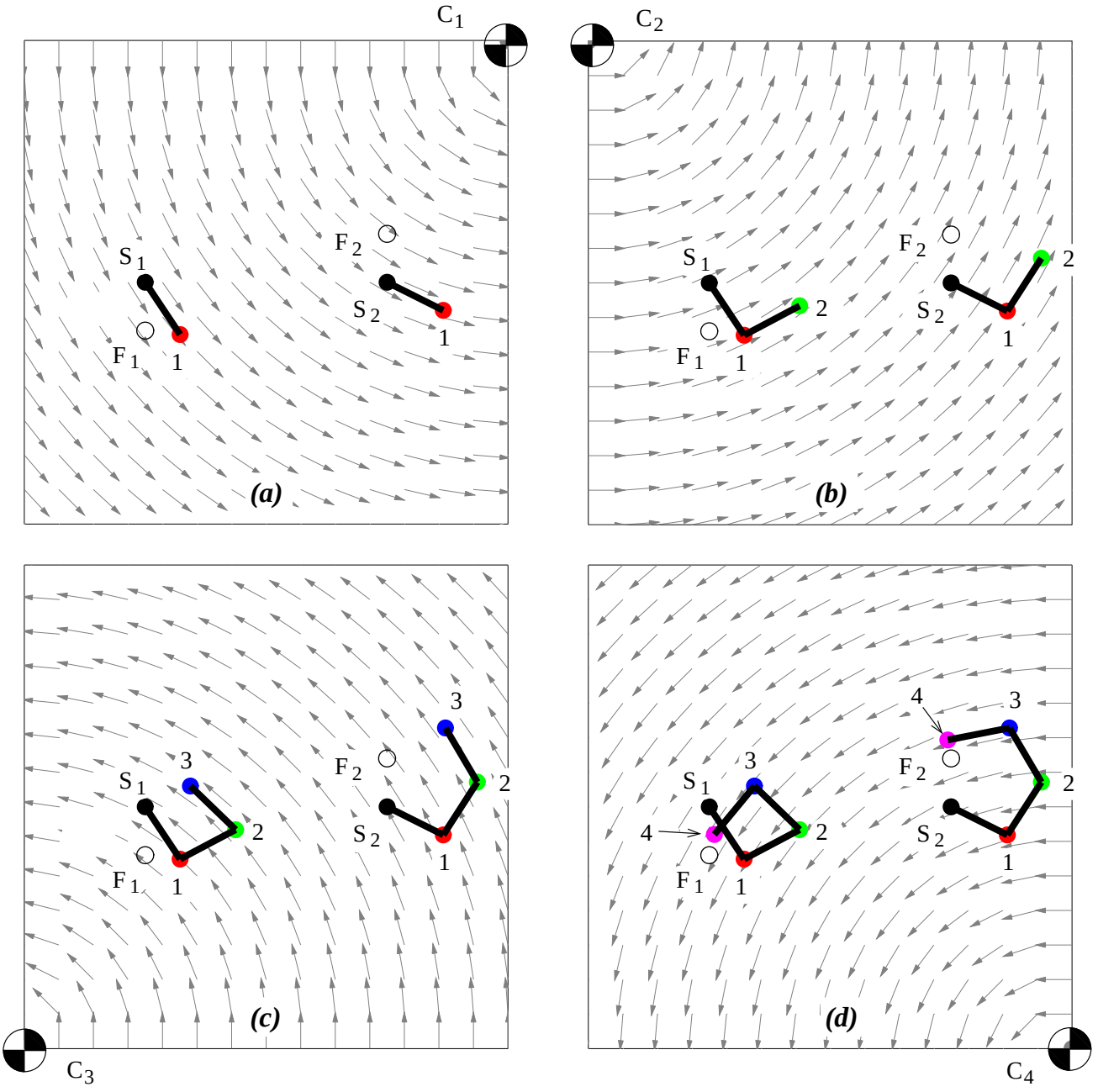


Figure 3: Four snapshots of a 2-part parallel manipulation problem: in (a) two parts are shown lying at starting locations S_1 and S_2; the goal is to move them to final locations F_1 and F_2. Four centers of rotation C_j, $j = 1 \ldots 4$ are specified, each at the corners of a square workspace. The rotations will take place starting with C_1, in counterclockwise order. Snapshots (a) through (d) show the parts' motions incrementally, after each rotation. Intermediate positions are labeled 1 through 4, and connected by a polygonal line. (d) Part's final positions (labeled 4) deviate from the intended destinations F_1 and F_2. This error was made intentionally large by prescribing large desired steps for each part.

$$\bar{f}_{1d} = \frac{\mu mg}{T} \int_0^T sgn[\nu_s(t) - \nu_p]\ dt \tag{3}$$

Define t^+ as the duration of positive $\nu_s(t) - \nu_p$ within one cycle. It can be easily shown [10] that:

$$\bar{f}_{1d} = \mu mg(\frac{2t^+}{T} - 1) \tag{4}$$

While $t^+ > T/2$, $\bar{f}_{1d}$ is positive, and the part will feed. In [10], we considered $\nu_s(t)$ of the form:

$$\nu_s(t) = \cos(wt) - \frac{1}{2}\cos(2wt) \tag{5}$$

With $w = 2\pi/T$. Notice that modulo a phase change, the above is equivalent to functions:

$$\begin{aligned} \pm\sin(wt) &+ \frac{1}{2}\cos(2wt) \\ -\cos(wt) &- \frac{1}{2}\cos(2wt) \end{aligned}$$

This particular function was picked because it contains only two harmonics and delivers a large $\bar{f}_{1d}$ relative to its peak acceleration [10]. In Appendix A we derive that for ν_p small, $\bar{f}_{1d} \cong 0.24\mu mg = \bar{f}_0$.

Consider now a surface S which is constrained to *rotate* about a fixed point C. Let $w_s(t)$ represent the periodic angular velocity of S about C. Let $w_s(t)$ be of the form of Equation 5. Then for a part resting ($||\nu_p|| = 0$) at position P on S, the surface will apply $\bar{f}_0$ average force along $(P - C)^\perp$. Over a time Δt, the part will displace $d \propto \Delta t^2$, regardless of its distance from C (in fact, near C tangential accelerations are too small and the part won't slide). The moral is: vibratory rotation can be used to synthesize the "non-linear rotation primitive" described in Section 2.1.

3 Practical Challenges & Solutions

3.1 Actuation Kinematics

One way to accomplish the oscillatory surface motion prescribed in Section 2.2 is to have the surface's three dof's (x, y, and θ) move in phase with velocities as in Equation 5. Note that the instantaneous velocity of a rigid body in the plane is related to its instantaneous center of rotation by the following map [15]:

$$\begin{bmatrix} c_x \\ c_y \\ w \end{bmatrix} = \begin{bmatrix} -\dot{y}/\dot{\theta} \\ \dot{x}/\dot{\theta} \\ \dot{\theta} \end{bmatrix} \tag{6}$$

The actuation kinematics illustrated in Figure 4 is designed to apply forces along the table's 3 dof's so that C can be easily chosen. As shown, the plate is positioned at the center of a working area. Four linear actuators are used to apply forces to the each of the plate's sides. Shafts connect the table to the motor, allowing the latter to both push and pull on the former. Shafts are stiff along the actuation direction and compliant perpendicularly.

Let X_1, X_2, Y_1, Y_2 denote the force applied to the table the motor positioned to the left, right, bottom, and top of the table, respectively, as shown in Figure 4(a). At the operating frequencies, overall table displacements will be small,

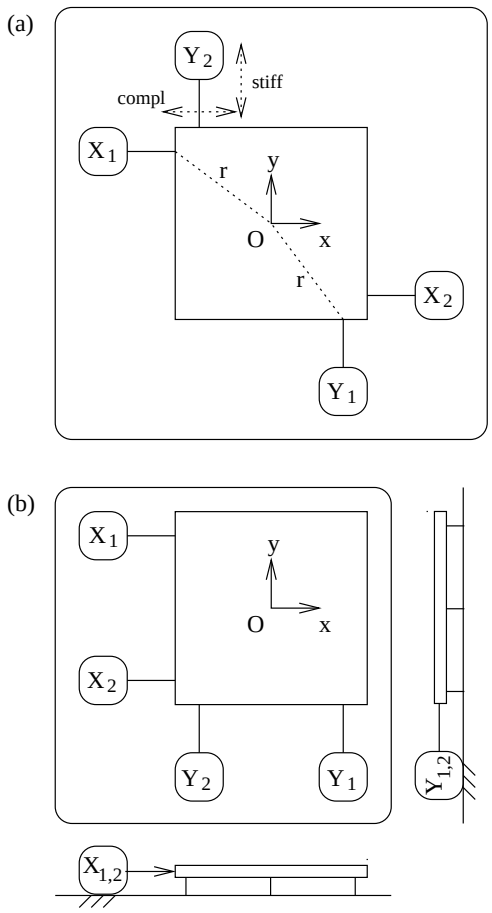


Figure 4: The actuation kinematics: (a) Four linear actuators, labeled $X_i, Y_i, i = 1, 2$, apply force to an individual side of the table through a shaft, attached at a distance r from the center of the table. Shafts are stiff along the driving direction and compliant perpendicularly. In the figure, Y_1's shaft is shown stiff along y and compliant x. (b) A more space-efficient arrangement of motors is shown, along with side views of the table; these show weight-supporting flexible rods under the table (also present in -a-).

so we can decouple cross-talk between dof's. Namely, the table will tend to rotate clockwise if motors at opposite sides push (or pull) *in tandem*, while the table will tend to translate if a given motor pushes while the one on the opposite side pulls (or vice versa). This can be expressed by the following set of equations which relates applied forces to the resultants along the plate's 3 dof's:

$$\begin{aligned} f_x &= X_1 - X_2 \\ f_y &= Y_1 - Y_2 \\ \tau_\theta &= r[(Y_1 + Y_2) - (X_1 + X_2)] \end{aligned} \tag{7}$$

where r denotes the table's center distance to the actuation point on each side. In Figure 4(b), a more space efficient (and kinematically equivalent) arrangement of motors and table is shown, in which the position (and force signs) of X_2 and Y_2 are changed.

We will model the off-axis shaft compliances as linear damped springs. In Appendix D it is shown that if input forces are well above resonance, inertial forces dominate all other ones, i.e., arising from compliance, so that the velocity along each axis becomes the time integral of the resultant forces along that axis. So let each motor apply a force of the type:

$$f(t) = \cos(wt) - \sin(2wt) \tag{8}$$

scaled by chosen constants $X1, X_2, Y_1, Y_2$. Then, because the map in (7) is linear, the force applied to the table along each of its dofs will also be of this form, so that the resulting integrated velocity will be as desired:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}(t) = \begin{bmatrix} \frac{1}{M} f_x \\ \frac{1}{M} f_y \\ \frac{r}{I} \tau_\theta \end{bmatrix} \frac{1}{w}[\sin(wt) + \frac{1}{2}\cos(2wt)] \tag{9}$$

Using Equations 6 and 7, we can choose $X1, X_2, Y_1, Y_2$ to place C at a desired spot and scale the angular velocity about it.

3.2 Signal Generation and COR Visualization

We use *voice coils* [16] for each linear actuator. These devices respond with force along the driving axis proportionally to the current flowing through them. We built a dedicated circuit to generate four independent force waveforms defined in Equation 8; a photo of this device is shown in Figure 6. Two PIC16c73 microcontrollers [17] are used to produce two independent signals each. These are interfaced to a host PC via the parallel port. The PICs firmware allows for the flexible tuning of relative phase, and amplitude between the 1st and 2nd harmonic components in Equation 8. The PC can request that waves be started or stopped with chosen scaling amplitudes X_1, X_2, Y_1, and Y_2. The microcontrollers output the signal in PWM, which are low-pass-filtered, power-amplified, and sent to the motors, as shown in Figure 5.

Instead of calculating C based on a set of known dynamic parameters (input forces, plate mass and geometry, motors' force constants), we take a reverse-engineering approach. We installed ADXL202EB MEMs accelerometers [18] at two opposite corners of the plate (actually glued underneath). Each sensor provides two analog outputs proportional to the acceleration at two perpendicular axes.

In Appendix B we show that by knowing the rigid velocities v_1 and v_2 at two distinct points p_1 and p_2 of a moving plate (e.g., two opposite corners, $p_1 = -p_2$) we can determine the plate's instantaneous center of rotation and angular velocity:

$$|w| = \frac{||v_2 - v_1||}{2||r_1||} \tag{10}$$

$$C = \frac{(v_1 + v_2)^\perp}{2w} \tag{11}$$

There are two problems with the above. (1) the sensors recover acceleration, and not velocity; (2) sensor data is noisy. To address (1) we simply state that under sinusoidal excitation, the rms velocity will be proportional to the rms acceleration, independently for the 1st and 2nd harmonic components. Speaking of RMS, this suggests a solution for (2), i.e., rather than computing C and w based on instantaneous acceleration readings we do that based on average amplitudes over a large number of sampled cycles.

Figure 5 shows the 11-bit A/D converter used to sample the four accelerometer signals simultaneously. This is currently done at a rate of 5 KHz. Samples are passed to the PC via the parallel port in real-time. One such sequence of samples is shown in Figure 7(a). Since the force frequency w is known, the least-squares amplitude and phase of the signal are recovered by dotting the sensor samples with the four orthogonal functions $\cos(wt)$, $\sin(wt)$, $\cos(2wt)$ and $\sin(2wt)$ (essentially a DFT [19]), yielding coefficients c_1, s_1, c_2, s_2, i.e., we fit the following "model" acceleration $a(t)$ to our data:

$$\begin{aligned} a(t) &= c_1\cos(wt) + s_1\sin(wt) \\ &+ c_2\cos(2wt) + s_2\sin(2wt) \end{aligned} \tag{12}$$

which we express succinctly as $a(t) = [c_1, s_1, c_2, s_2]$. A well-registered least-squares fit to the data in Figure 7(a) is shown in Figure 7(b). To visualize the least squares-fit velocity waveform, we simply integrate Equation 12, obtaining an identical waveform expressed as:

$$v(t) = \frac{1}{w}[-s1, c_1, -\frac{s_2}{2}, \frac{c_2}{2}] \tag{13}$$

This is used to generate the velocity waveform shown in Figure 12. Real-time visualization of $v(t)$ allows the user to fine tune relative phase and amplitude parameters between first and second harmonic to compensate for frequency dependent phase and amplitude response (ideally, phase is flat and amplitude roll-off is as $1/w$ when $w >> w_0$, however a bit of pre-compensation is always needed).

The least-squares fit recover unsigned amplitudes to the sinusoidal accelerations along each of the four probed axes. To assign directions to each of these vectors we need to consider the phase relationship between the first and second harmonic of each acceleration signal. For a 1d acceleration profile of the form $\cos(t) + \cos(2t + \phi)$, the average force will be positive iff $\phi \in (0, \pi)$ [10]. In Appendix C, we shown that this corresponds to the following expression in terms of the four free parameters in Equation 12:

$$sgn(\text{force}) = sgn[2s_1c_1c_2 + s_2(s_1^2 - c_1^2)] \tag{14}$$

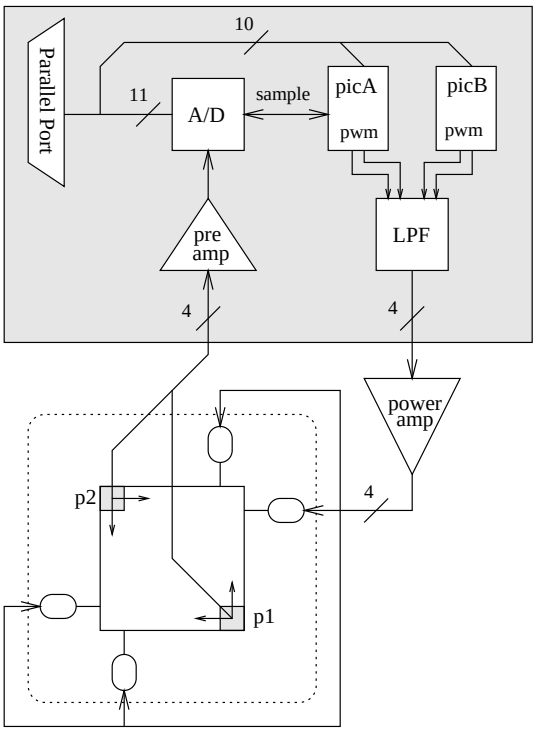


Figure 5: Block diagram of the signal generation and acceleration acquisition hardware: two microcontrollers (picA and picB) generate four independent PWM signals. These are low-pass filtered and power-amplified, and then applied to each motor. Two 2-axis accelerometers are glued under opposite corners of the table. The four acceleration readings are pre-amplified and input to a 4-channel, 11-bit A/D, whose sampling is controlled by one of the PICs. The PC can send commands and/or read samples from the A/D via a parallel port interface.

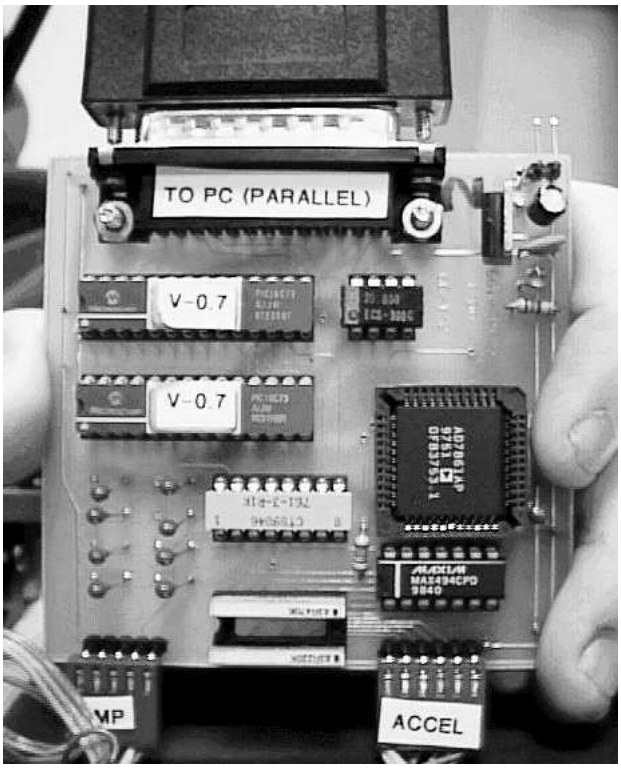


Figure 6: Photograph of the signal-generation and real-time acceleration acquisition hardware.

3.3 Synthesizing Scaled Displacement Fields

Consider a surface rotating periodically about point C with angular velocity $w(t)$ as in Equation 5. Consider N parts P_i lying on S at rest. Define $r_{i,min} = \min_i\{P_i - C\}$. In Appendix A we show that given an r_{min} and other dynamic parameters (such as motor power and coefficient of friction), $w(t)$'s amplitude and frequency can be chosen so that the part's velocity ν_p can be considered constant within one oscillation cycle. In that Appendix it also shown that if the part's velocity is negligible compared to the peak tangential velocity at P_i, call it ν_{max}, the average force $\bar{f}_{1d} = \bar{f}_0$ is constant and independent of ν_p. In particular, as ν_p grows, the force applied to it by the oscillating plate decreases, in linear viscous fashion (see Appendix). At $\nu_p = \frac{2}{3}\nu_{max}$, $\bar{f}_{1d} = 0$.

To simplify control, we make the following key assumptions: (i) A desired displacement field will be generated by a finite-duration pulse. (ii) At the beginning of the pulse all parts will have zero velocity. To ensure this, each pulse will be followed by a sufficiently long rest phase. (iii) By keeping all parts' velocities negligible with respect to the peak of $w(t) \times r_{min}$, all parts will accelerate by the exact same amount, and that amount will be linearly proportional to the pulse's length.

To avoid impulse-response ringing, we will initiate (resp. terminate) the pulse with smooth attack (resp. decay) phases, of identical duration. The pulse's middle part, called its *sustain phase* will be of a much higher duration S. These concepts are illustrated in Figure 9. The final desired displacement d for all parts P_i will be proportional to S^2, i.e.:

$$S \propto \sqrt{d}$$

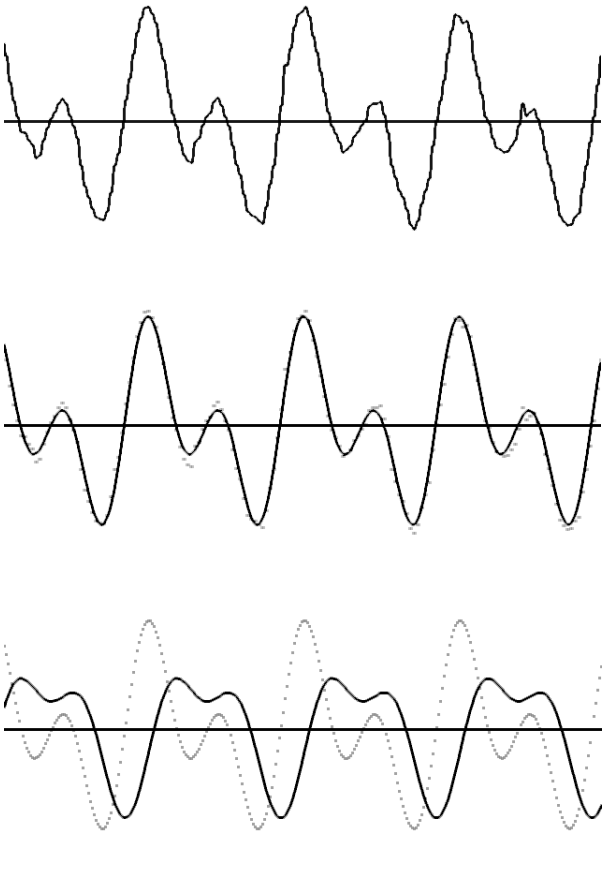


Figure 7: The waveform fitting process: (a) Samples coming from one sensed axis; (b) Least-squares fit (shown solid) and original samples (shown dotted); (c) Least-squares fit (shown dotted) and closed-form integral, i.e., the fitted velocity signal (shown solid).

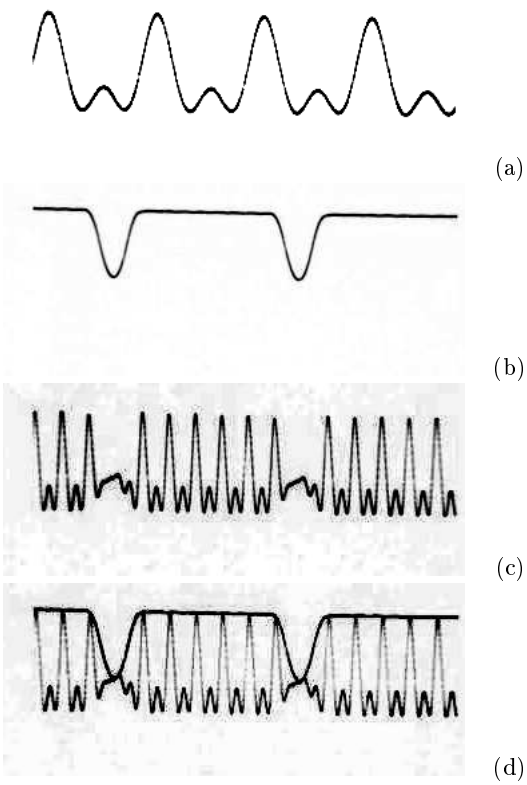


Figure 8: Waveform shaping: (a) four cycles of the original velocity waveform $\sin(t) + \cos(2t)/2$; (b) the attack/sustain/decay envelope; (c) the shaped waveform, i.e., -a- multiplied by -b-; (d) the envelope superimposed on the shaped wave, showing registration. These pictures were taken from an actual oscilloscope (the sweeping rate for -a is four times faster than for the rest).

The signal-generation hardware allows for the easy tuning of attack/decay and sustain durations shaping of the output waveform. Oscilloscope photographs showing actual output is illustrated in Figure 8.

3.4 Tracking Parts

A camera is placed a few feet above the table pointing downward at the latter's center. The ground is black, the table is white and the parts are pennies painted black. An image as seen by the camera is shown in Figure 10(a).

The first step is to determine the table's rotation and translation relative to the image. This is a one-time-only operation, done prior to the task, given that the table itself moves negligibly when it vibrates. We compute the table's edge map using standard procedures [20]. Each edge in the image is then hashed by its distance to the image's center and angle onto a 2d Hough-vote array [20]. Edges making up the table's four sides will cluster at four locations on the Hough-array, as shown in Figures 10(b) and (c). Each of these peaks gives rise to a line. Sorting these lines by angle and then intersecting consecutive line pairs, we obtain the 4 corners of the table and its coordinate frame.

The second step is to locate the coins initial locations.

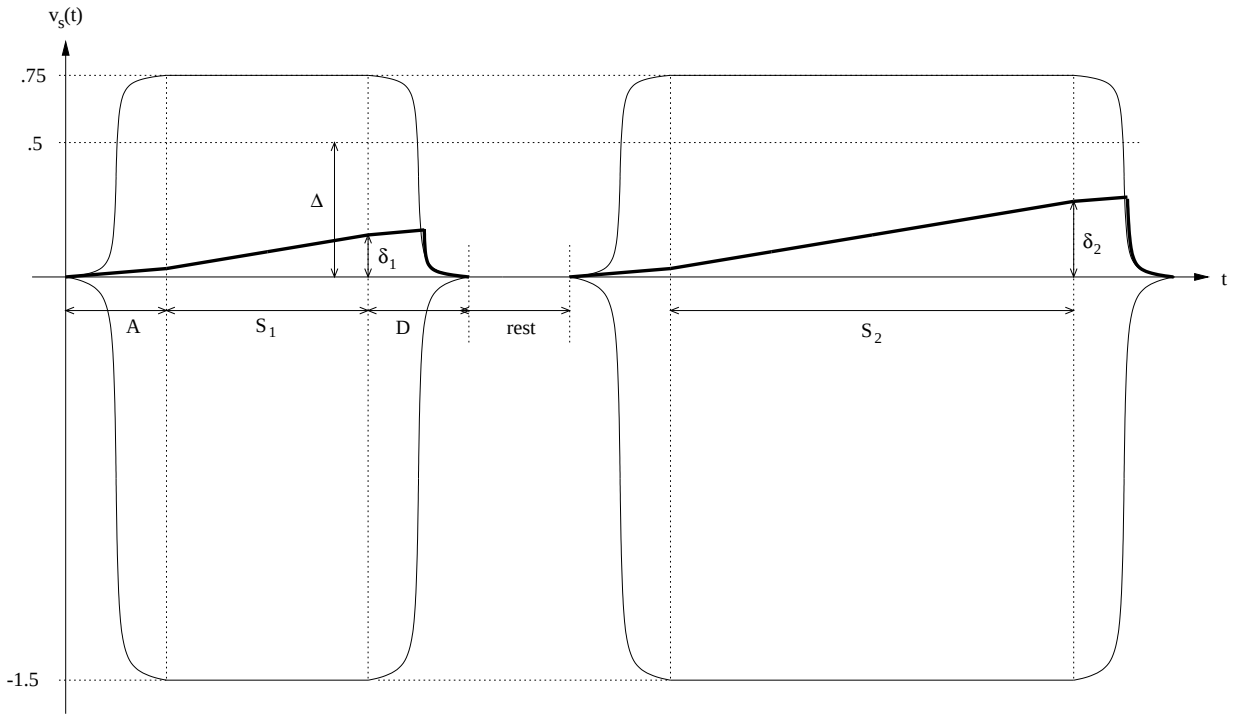


Figure 9: Emulating rotation intensity through pulse duration: two shaped pulses are shown. The pulse is represented by the *outline* of a normalized tangential velocity, covering $[-1.5, 0.75]$ along y (i.e., the range of $\cos(t) - \cos(2t)/2$). Each pulse contains 10s-100s of cycles of the basic driving waveform (not drawn). Each pulse starts (resp. finishes) with a smooth attack (resp. decay) phase lasting A (resp. $D = A$) seconds. The first (resp. second) pulse *sustain* duration is S_1 (resp. S_2). For convenient visualization, $S_2 = 2S_1$. The part's velocity is shown in plotted with a thicker line. Pulses are preceded and followed by a *rest* phase which ensures part velocity is null at the beginning of each pulse. Though not drawn to scale, assume the attack/decay phases are very short compared to the sustain; in this fashion, part speed will increase steadily so that at the end of the pulse, its value (shown as δ_1 and δ_2) is *proportional* to S_1, S_2, i.e., part displacement will be proportional to δ_i^2. To ensure this, i.e., to avoid viscosity effects, the dynamic parameters must be tuned so that δ_i is negligible compared to Δ, the equilibrium velocity.

Having previously determined the table's sides (and their lengths as they appear in the image) we compute a circular kernel (a solid disk) with a pixel-radius proportional to the penny/table-side ratio known a priori. This kernel is convolved with all points in the image interior to the table outline, computed above. The convolved image, shown in Figure 10(d), will contain peaks corresponding to the center of each coin. The detected features (table outline and coin locations) are shown superimposed on the original image in Figure 10(d).

Determine coins' initial locations is done once prior to the task. The actual *tracking* of coins is a much cheaper operation. Once they start moving, once must simply convolve the aforementioned disk-shaped kernel over a 1 or 2 pixel neighborhood of a part's current location; the peak in the convolved neighborhood determines the coin's new position.

3.5 The Control Loop

- Use vision to obtain parts' coordinates P_i
- From task trajectories, specify new motion subgoals ΔP_i
- Given a set of M feasible COR's, solve for rotation scaling $d_j,\ j = 1 \ldots M$.
- Actuate table so it rotates $\sqrt{d_j}$ seconds about C_j (using a shaped pulse), sequentially, for $j = 1 \ldots M$.
- Compare with desired steps (report error), repeat

4 Experiments

4.1 COR Steering and Calibration

Given the actuation kinematics in Figure 4(a), Equations 9 and 6 give rise to the following proportionality laws:

$$\begin{aligned} w &\propto (Y_1 + Y_2) - (X_1 + X_2) \\ c_x &\propto (Y_2 - Y_1)/w \\ c_y &\propto (X_1 - X_2)/w \end{aligned} \tag{15}$$

We used the signal generation hardware to test the table's vibration under six distinct choices for amplitudes $X_i, Y_i, i = 1, 2$, as shown in Table 1. As it is apparent, in all combinations the w control $(Y_1 + Y_2) - (X_1 + X_2)$ is kept constant. By varying the other components, the idea is to "steer" the COR away from its original position in fixed steps along the following axes: $+x$, $+y$, $-x$, $+y$, and $-x$. A program was written which performs real-time acquisition of acceleration data and the simultaneous computation/visualization of the COR's. Figure 11 shows the CORs placement for each of the amplitude combinations sent to the motors; as shown, the COR does get placed at the intended locations. The actual coordinates for C calculated in real-time from the accelerometers' outputs are shown in the last two columns of Table 1.

With this machinery, one can tweak waveform amplitudes input to the four motors until the COR is steered to a convenient location. Repeating this process for enough distinct locations and recording the required amplitudes gives rise to a "COR library" which can then be used by our parallel manipulation algorithm.

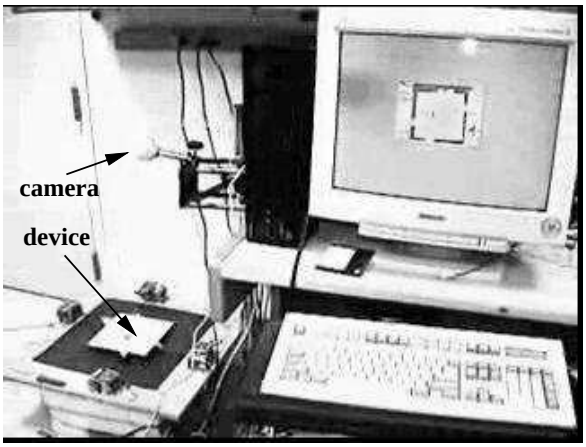


Figure 12: Experimental Setup for the 1-coin experiment: the computer, shaker table, and camera are visible.

4.2 One-Part Trajectory-Following

In order to test the integrity of key parts of the system, namely, the image-processing/part tracking, the interfacing with the signal generation hardware, and the mechanical functionality of our prototype, we designed a simple automated, visually-servoed task involving a single part (a penny painted black). The experimental setup is shown in Figure 12.

(a) The penny is placed at a random location on the table. (b) The image processing system locates it. (c) The penny is brought to the exact center of the table via translations along x and y. (d) The penny will traverse clockwise and indefinitely, the four branches of an imaginary "plus" sign laid over the table. It starts out traversing in the $-x$ direction until it hits the table's edge, at which point it switches directions and returns to the center. After that, the $+y$ branch is explored, and so forth. For this simple task, the system performed robustly and consistently. Eight consecutive snapshots of this experiment, are shown in Figure 13.

5 Conclusion & Future Work

We have described solutions to the challenges facing a practical implementation of our parallel manipulation algorithm. These included the design of a proper actuation kinematics, the ability to flexibly generate signals to the actuators, visualization and calibration of centers-of-rotation, and part localization through image processing. Experimental results involving COR calibration and a simple single-coin manipulation task were described.

We still have not been able to produce a multiple-coin example due to lack of power in the motors currently being used (they are voice coils designed for hard disk drives!). We are currently building a second prototype with a bigger table area and motors which can output two orders of magnitude more force. We expect that this move towards "bigger and better" will allow us to produce an interesting demo involving two or more parts in the very near future.

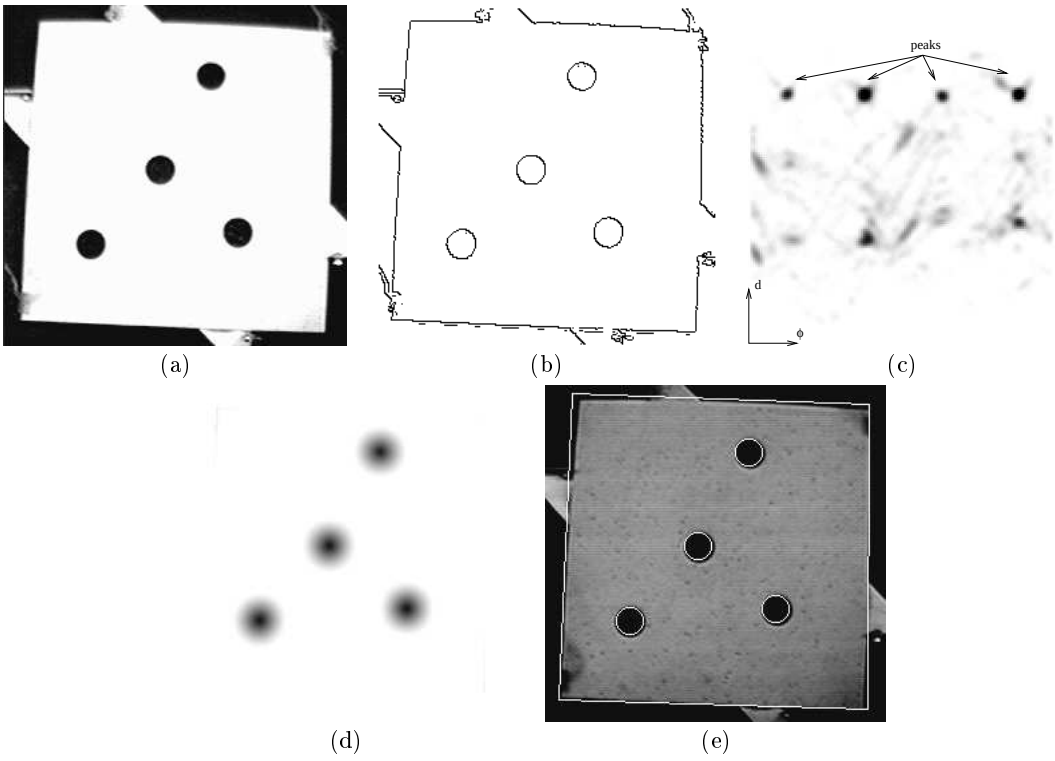


Figure 10: Image processing stages: (a) the original image, as seen from the camera overlooking the table. Four pennies have been placed on the table; (b) after edge detection; (c) Hough-vote space, showing the four peaks corresponding to the table's 4 long sides; (d) image convolved with a penny-sized disk, showing peaks at coins' centers; (e) final result showing detected features (table and coin outlines).

	X_1	X_2	Y_1	Y_2	$Y_1 - Y_2$	$X_1 - X_2$	c_x	c_y	Δc_x	Δc_y
(a)	-128	-128	128	128	0	0	$-.22$	.02		
(b)	-256	0	128	128	-256	0	2.92	.04	3.14	.02
(c)	-256	0	0	256	-256	256	2.72	2.52	$-.20$	2.48
(d)	-128	-128	0	256	0	256	$-.18$	2.66	-2.90	.14
(e)	-128	-128	-128	384	0	512	$-.14$	5.34	.02	2.68
(f)	0	-256	-128	384	256	512	-3.25	5.54	-3.11	.2

Table 1: Motor amplitudes $X_i, Y_i, i = 1, 2$ as they were passed to the hardware waveform generator. Real-time accelerometer output was used to compute the coordinates c_x and c_y of the associated center of rotation, displayed in inches with respect to the table's center (the table is an 8"x8" square). Notice that the last two CORs lie outside the table's surface. The $\Delta c_{x,y}$ show the COR displacement with respect to its location given the controls in the preceding row. As seen, the device is fairly "balanced" on both axis, responding linearly to changes in the control as predicted by Equation 15

.

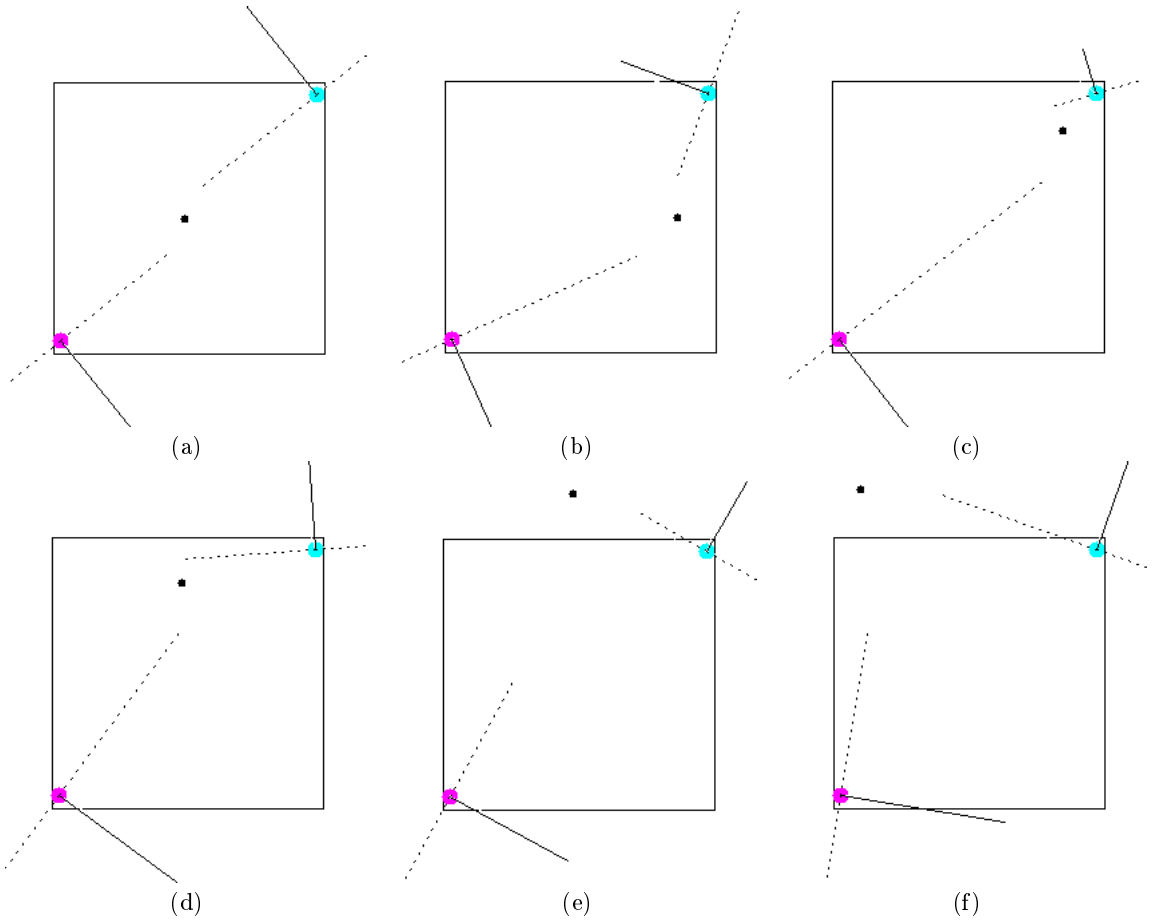


Figure 11: Steering the COR with the 6 amplitude combinations shown in Table 1. The table is drawn as an outline; the accelerometers are drawn centered at their actual locations near the lower-left and upper-right corners of the table. The actual magnitude of acceleration measured by each two-axis accelerometer is shown along with the perpendicular (the COR is supposed to fall at the intersection of these). The actual computed COR is shown as a black dot. In (a) through (d), the COR lies inside the table's surface; in (e) and (f) it falls outside.

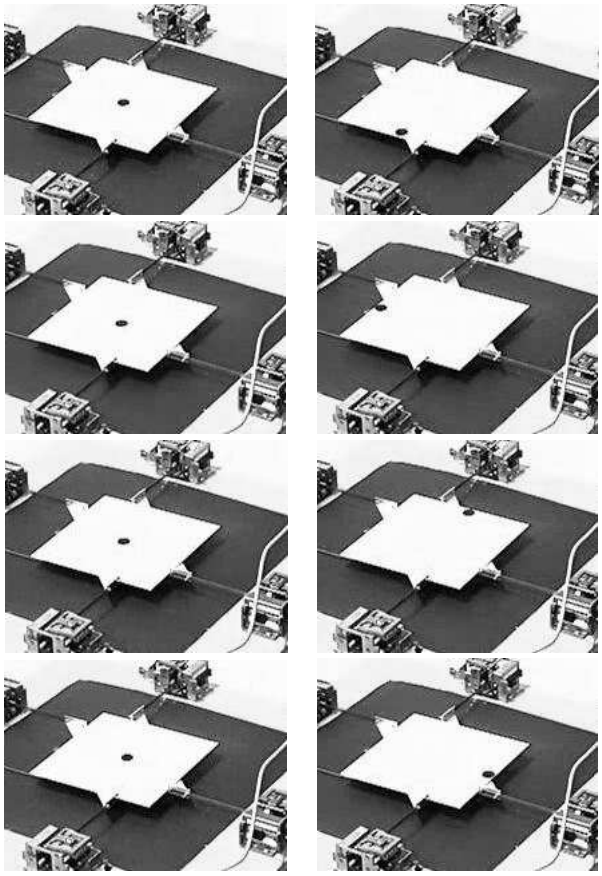


Figure 13: Eight consecutive snapshots (to be read left-to-right, top-to-bottom) of a simple visually-servoed trajectory-following task involving a single part (black penny). The plate is vibrated along x and y to steer the coin along the branches of an imaginary "plus" sign centered on the board. It does so in clockwise order, starting with the $-y$ branch. For each branch, the coin advances from the table's center to its edge at which point visual-servoing commands the motors to reverse feeding direction.

A 1d Feeding Analysis

Consider a horizontal surface S constrained to move along x. The surface vibrates periodically with velocity profile $\nu_s(t)$:

$$\nu_s(t) = A[\cos(wt) - \frac{1}{2}\cos(2wt)] \tag{16}$$

where A, and w are gain, and frequency parameters, respectively.

Consider a part P lying on the surface at some known position and at known current speed ν_p. Let $T = 2\pi/w$, and μ, g represent the part-surface coefficient of dynamic friction and the acceleration of gravity, respectively. Choose A, w high enough, so (i) part and surface's relative accelerations are high enough to ensure permanent sliding-mode (no stiction), and (ii) the part's change in speed per cycle, bounded by $\mu g T$ is negligible compared to A, i.e., $2\pi\mu g/(wA) \ll 1$. The average Coulomb friction $\bar{f}_{1d}$ is then given by Equation 3. Define t^+ as the duration of positive $\nu_s(t) - \nu_p$ within one cycle. When $\nu_p \in [0, A/2]$, the graph of $\nu_s(t) - v_p$ will have two symmetric roots r and $-r$. Solving $\nu_s(r) - v_p = 0$ yields:

$$r = \cos^{-1}\left(\frac{1 - \sqrt{3 - 4\nu_p/A}}{2}\right) \tag{17}$$

Since the roots are symmetric, $t^+ = 2r$. From Equation 4 derive:

$$\bar{f}_{1d} = \mu mg\left(\frac{2}{\pi}r - 1\right) \tag{18}$$

At $\nu_p = 0$, $\bar{f}_{1d} \cong 0.24\mu mg = \bar{f}_0$. With $\nu_p = A/2$, the total applied force is zero, and the part is at the *equilibrium velocity* ν_{eq}. As apparent from the graph of $\bar{f}_{1d}$ vs. ν_p (Figure 14), this function can be closely approximated by a straight line connecting its two extreme values, i.e.:

$$\bar{f}_{1d} \cong \left(\frac{1}{2} - \frac{\nu_p}{A}\right) 2\bar{f}_0 \tag{19}$$

The approximation in Equation 19 suggests that the part perceives the oscillating plate as a viscous fluid flowing forward at speed $A/2$ (the equilibrium velocity), i.e., with $\dot{\nu}_p(t) = \bar{f}_{1d}/m$, and $\nu_p(0) = 0$:

$$\nu_p(t) = k_1(1 - e^{-k_2 t}) \tag{20}$$

Using conditions $\nu_p(\infty) = A/2$ and $\dot{\nu}_p(0) = \bar{f}_0/m$, derive:

$$\begin{aligned} k_1 &= A/2 \\ k_2 &= \frac{2\bar{f}_0}{mA} \cong .48\frac{\mu g}{A} \end{aligned} \tag{21}$$

If $t \ll 1/k_2$, Equation 20 linearizes to:

$$\nu_p(t) \cong \frac{Ak_2}{2}t = \frac{\bar{f}_0}{m}t = .24\mu g t$$

Let f and a_{max} denote the surface's vibration frequency (in Hz) and peak acceleration a_{max} (in g's). Simple integration of sinusoidal acceleration into velocity yields $A = a_{max}/2\pi f$, and from Equation 21 obtain:

$$\frac{1}{k_2} \cong \frac{a_{max}}{\mu\pi f}$$

Example: at typical $\mu = 0.3$, $a_{max} = 2$ g and $f = 60$ Hz, $1/k_2 = 35$ ms.

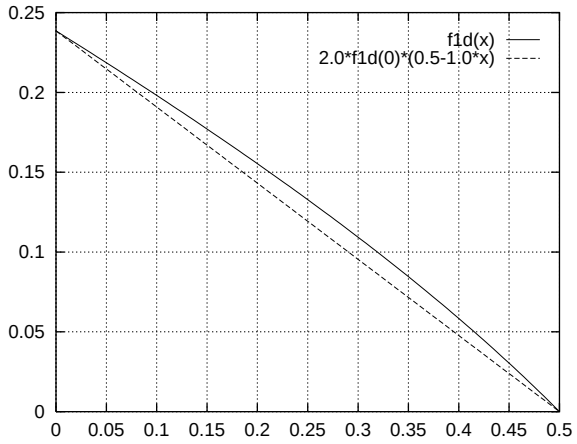


Figure 14: Solid: the average frictional force $\bar{f}_{1d}$ applied to the part per cycle of $\cos(wt) - \cos(2wt)/2$, assuming $\mu mg = 1$, plotted against $\nu_p \in [0, 1/2]$, the part's velocity. Dashed: linear approximation to the same function, showing a nearly viscous effect (see text).

B COR Calculation

Assume the table is a rigid square with center O. Assume the instantaneous velocities v_1 and v_2 at points r_1 and r_2 are known. These quantities are illustrated in Figure 15. The goal is to compute the table's instantaneous center of rotation c and the associated instantaneous angular velocity w measured about c. We can write:

$$v_1 = w(r_1 - c)^\perp \quad (22)$$
$$v_2 = w(r_2 - c)^\perp \quad (23)$$

Taking the difference (22)-(23) eliminates c, i.e.:

$$v_2 - v1 = w(r_2 - r_1)^\perp = -2wr_1 \quad (24)$$

Which implies:

$$|w| = \frac{||v_2 - v_1||}{2||r_1||} \quad (25)$$
$$sgn(w) = sgn[(v_2 - v_1) \times r_1]$$

Taking the sum (22)+(23) yields:

$$v_1 + v_2 = w(r_1 + r_2)^\perp - 2wc^\perp$$

Since $r_1 + r2$ vanishes in the above, we proceed with:

$$v_1 + v_2 = -2wc^\perp$$
$$(v_1 + v_2)^\perp = 2wc$$
$$c = \frac{(v_1 + v_2)^\perp}{2w} \quad (26)$$

With w computed as in (25). Equations 25 and 26 are then the final results. An alternative method to compute c is to find the intersection of infinite lines L_1, L_2 passing thru r_1, r_2, which are perpendicular to v_1, v_2, respectively (see Figure 15). This method is inconvenient since the intersection is ill-defined with nearly parallel v_1 and v_2.

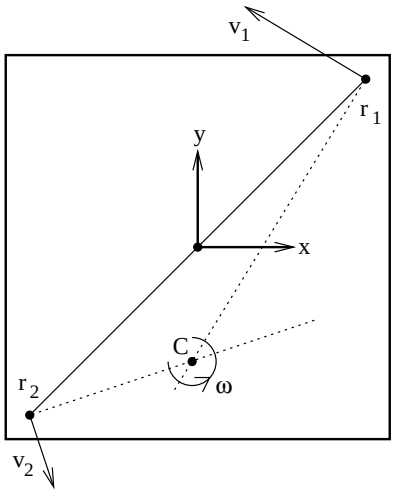


Figure 15: The shaker table is shown with two accelerometers placed at r_1 and r_2, with $r_2 = -r_1$. The instantaneous velocities at these points are v_1 and v_2, respectively. Lines L_1 and L_2 pass through r_1 and r_2, and are perpendicular to v_1 and v_2, respectively. The instantaneous center of rotation C and angular velocity w are also shown. Notice that C will lie at $L_1 \cap L_2$.

C Positive Force Test

Assume plate's acceleration relative to part is of the form:

$$a_p(t) = \cos(t) + 2b\cos(2t + \phi) \quad (27)$$

In [10] we show that under the above plate motion, the part's equilibrium velocity is:

$$\nu_{eq} = b\sin(\phi), \ |b| < 1/2 \quad (28)$$

Though a closed-form expression was not derived for the average force applied to the part per cycle (assuming zero part velocity) in terms of b and ϕ, Equation 28 implies that the *sign* of the average force is given by $sgn[b\sin(\phi)]$.

An alternative representation for Equation 27 is:

$$a_p(t) = c_1\cos(t) + s_1\sin(t) + c_2\cos(2t) + s_2\sin(2t) \quad (29)$$
$$= m_1\cos(t - \alpha_1) + m_2\cos(2t - \alpha_2) \quad (30)$$
$$(m_i, \alpha_i) = (\sqrt{c_i^2 + s_i^2}, \ \tan^{-1}\frac{s_i}{c_i}), \ i = 1, 2$$

Let $t' = t - \alpha_1$, then Equation 30 can be rewritten as:

$$a_p(t) = m_1[\cos(t') + \frac{m_2}{m_1}\cos(2t' + 2\alpha_1 - \alpha_2)] \quad (31)$$

Modulo the m_1 scaling factor, Equation 31 is in the form of Equation 27, with $\phi = 2\alpha_1 - \alpha_2$ and $b = \frac{m_2}{2m_1} > 0$. So the force will be positive when $\sin(2\alpha_1 - \alpha_2) > 0$, i.e.:

$$2\alpha_1 - \alpha_2 \in (0, \pi) \tag{32}$$

Define complex numbers $z_i = c_i + js_i, i = 1, 2$. Then $2\alpha_1$ and α_2 are the angles under $z_1^2 = c_1^2 - s_1^2 + 2jc_1s_1$ and z_2, respectively. So Equation 32 is equivalent to stating $z_1^2 \times z_2 > 0$, or equivalently:

$$2s_1c_1c_2 + s_2(s_1^2 - c_1^2) > 0$$

D Table Dynamics

Let each actuation degree-of-freedom of the table be, under small displacements, modeled as a forced mass-damped-spring system, obeying the following linear differential equation [15]:

$$f_{ext} = m\ddot{q} + b\dot{q} + kq \tag{33}$$

Where "q" represents the axis under question (x, y, or θ), and m, b, k are the mass (moment of inertia for θ), damping, and coefficient of friction along the appropriate axis. The resonance frequency w_0 of such a system is given by [15]:

$$w_0 \cong \sqrt{\frac{k}{m} - \left(\frac{b}{2m}\right)^2} \tag{34}$$

In general, each axis will have distinct w_0's, i.e., $w_{0,x} \neq w_{0,y} \neq w_{0,\theta}$, so define:

$$w_{0,max} = \max\{w_{0,x}; w_{0,y}; w_{0,\theta}\}$$

For $f_{ext} = \sin(wt)$, the solution to Equation 33 is $q = \alpha(w)\sin(wt + \phi)$ [15], which implies w and $-w^2$ coefficients in the $\dot{q}$ and $\ddot{q}$ derivatives, respectively. So sufficiently above the system's highest resonance, i.e., $w >> w_{0,max}$, the inertial term in Equation 33 dominates over the damped-spring forces, and $f_{ext} \cong m\ddot{q}$, i.e., the velocity along that d.o.f. is proportional to the integral of the external force:

$$w \gg w_{0,max} \quad \Rightarrow \quad \dot{q}(t) = \frac{1}{m}\int f_{ext}(t)\ dt$$

References

[1] W. Huang and M. Mason. Mechanics, planning, and control for tapping. In P. Agarwal, L. Kavraki, and M. Mason, editors, *3rd Workshop on Algorithmic Foundations of Robotics: Robotics: the algorithmic perspective*. A. K. Peters, Natick, MA, 1998.

[2] K. Lynch. Toppling manipulation. In *IEEE International Conference on Robotics and Automation*, Detroit, MI, May 1999.

[3] J. Luntz, W. Messner, and H. Choset. Velocity field design for the modular distributed manipulator system (mdms). In P. Agarwal, L. Kavraki, and M. Mason, editors, *3rd Workshop on Algorithmic Foundations of Robotics: Robotics: the algorithmic perspective*. A. K. Peters, Natick, MA, 1998.

[4] L. Kavraki. Part orientation with programmable vector fields: Two stable equilibria for most parts. In *IEEE International Conference on Robotics and Automation*, Albuquerque, NM, April 1997.

[5] K. Böhringer et al. Parallel microassembly using electrostatic force fields. In *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[6] G. Boothroyd. *Assembly automation and product design*. Marcel Dekker, Inc., New York, NY, 1991.

[7] N. Tran, K. Chan, and V. Hayward. Object behavior using a vibrating plate testbed for part presentation research. EE304-494 project writeup, McGill University, December 1995.

[8] K. Böhringer, V. Bhatt, and K. Goldberg. Sensorless manipulation using transverse vibrations of a plate. In *IEEE International Conference on Robotics and Automation*, Nagoya, Japan, May 1995.

[9] J. Canny and K. Goldberg. RISC for industrial robots: Recent results and open problems. In *IEEE International Conference on Robotics and Automation*, San Diego, CA, May 1994.

[10] D. Reznik and J. Canny. The coulomb pump: A novel parts feeding method using a horizontally-vibrating surface. In *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[11] D. Reznik and J. Canny. A flat rigid plate is a universal planar manipulator. In *IEEE International Conference on Robotics and Automation*, Leuven, Belgium, May 1998.

[12] D. Reznik and J. Canny. Universal part manipulation in the plane with a single horizontally-vibrating plate. In P. Agarwal, L. Kavraki, and M. Mason, editors, *3rd Workshop on Algorithmic Foundations of Robotics: Robotics: the algorithmic perspective*. A. K. Peters, Natick, MA, 1998.

[13] A. Quaid. A miniature movile parts feeder: operatin principles and simulation results. In *IEEE International Conference on Robotics and Automation*, Detroit, MI, May 1999.

[14] R. Murray, Z. Li, and S. Sastry. *A mathematical introduction to robotic manipulation*. CRC Press, Boca Raton, FL, 1994.

[15] J. Craig. *Introduction to robotics, mechanics and control*. Addison-Wesley, Reading, MA, 2nd edition, 1989.

[16] BEI Kimco Magnetic Systems. Voice coil actuators, an applications guide, 1999. Information booklet.

[17] Microchip Technology. Pic16/17 microcontroller data book, 1996. Catalog.

[18] Analog Devices. Winter 1999 short form designer's guide, 1999. Catalog.

[19] S. Haykin. *An introduction to analog and digital communications*. John Wiley & Sons, Inc., New York, NY, 1989.

[20] S. Russell and P. Norvig. *Artificial Intelligence, a Modern Approach*. Prentice Hall, New Jersey, NJ, 1995.