Sensor-Based Planning for Grasping and Manipulation with Multifingered Robot Hands[1]

Roderic A. Grupen and Richard S. Weiss

Laboratory for Perceptual Robotics

COINS Technical Report 90-58
July 13, 1990

Abstract

This report presents an architecture for reasoning about a general class of interactions between a robot and its environment. We propose distributed control agents, or *behaviors* which express some of the force domain objectives in grasping and manipulation. The behaviors are implemented as feedforward control laws which employ offline models of the manipulator and multiple resolution models of the object in wrench space. This system architecture permits concurrent sensing and control in an incremental, closed-loop planner. Such a planning mechanism suggests that the system may be capable of *learning* object models by active sensory exploration whose goal it is to accurately model relevant force domain features of the object.

[1]Work reported in this document was supported in part by NSF/CER DCR8500332 and NSF/CII CDA-8922572.

Contents

1 Introduction — Control for Multifingered Robot Hands

Considerable progress in grasp planning has been published in the 1980's. Salisbury developed the Grip Transform to express the transformation from a set of contact forces or velocities to a net object frame force or velocity[31, 37]. These results were extended by formulating the grasp force solution as a mathematical programming problem employing frictional constraints[24, 19]. Li and Sastry extended the computed torque control technique to include object dynamics and null space grasp forces. The result is a means of regulating the position and orientation of an object within the workspace of the hand[27]. Nguyen develops the concept of a force closure grasp[33, 34]. Nguyen also analyzes the effective stiffness impressed on an object by the grasp and suggests a means of designing compliance spaces [35]. However, all of this work requires a predefined grasp configuration consisting of contact locations on the surface of the object and corresponding contact types.

The problem of synthesizing the grasp configuration has likewise received a good deal of attention, but has produced far fewer definitive results. Several researchers have attempted to formulate a grasping paradigm motivated by human grasp taxonomies[12, 22, 39], but the results are constrained by the discrete selection of "canonical" grasps. Nguyen and Stephanou extended these results to permit linear combinations of the canonical grasp types [32] — but these approaches are limited to manipulators which approximate the kinematic qualities of the human hand and to prehensile tasks. Moreover, these techniques represent modes of coordinated hand control and discard some of the flexibility of individual finger control schemes. Manipulation tasks which require re-grasp solutions while maintaining object stability will undoubtedly find such flexibility necessary. Other researchers have attempted to formulate objective functions which express the goals of grasping tasks. Jameson and Leifer describe a grasp goal function representing the *distance* of the grasp state from instability[23]. Several researchers have investigated the use of contact geometry, friction and compliance to generate object reorientations within the hand [14, 30]. These results demonstrated that it is possible to take advantage of the properties of the contact to reorient an object by rolling or sliding the contact location over the surface of the object. However, the solution in 3D awaits future developments.

There is no currently available planning methodology which exploits the full complexity of multifingered robot hands to perform grasping tasks. Moreover, current technology does not promise to scale up to the manipulation task involving regrasping strategies — incremental modifications to existing grasps subject to stability constraints. Current approaches depend on simple manipulators, simplified modes of control for complex manipulators, or employ inflexible objective functions. It is valuable to discover more general properties of

the interaction between a robot and its environment. Tasks involving arm/torso *grasps*, co-operating kinematically independent robots, or legged locomotion are strikingly similar to that of grasping. Our approach is potentially useful in the control of general mechanisms which interact with their environment.

Most work on path planning for complex manipulators does not take into account the real-time constraints or consider the efficacy of programming methodologies in this domain. Canny has derived an algorithm for finding a path with algebraic constraints that is exponential in the number of degrees of freedom [9]. The dimensionality of the state space for grasp planning is $(R^l \times R^{2m} \times R^n)$, where $l = 3$ represents the geometry of the object, $m = 6$ is the dimensionality of the screw spaces in which simultaneous wrench and twist tasks are expressed, and $n = 32$ (for the Utah/MIT hand) represents the independent actuator degrees of freedom in the manipulator. It is clear that traditional combinatoric search techniques which consider the entire state space will be computationally cumbersome. Moreover, the quality of grasp solutions is extremely sensitive to uncertainty. Combinatoric techniques will, in general, be required to discard partial solutions and re-plan. From the systems viewpoint, real-time grasping applications must be scheduled on both robotic and computational hardware. Scheduling is most effective for processes which have low mean computational time, with low variance and which are relatively data independent[38].

Our approach is based on *incremental planning techniques.* Several researchers have noted the utility of incremental planning in such domains[5, 21, 40]. Traditional feedback control methods make the relationship between sensing and action explicit: servo systems continually suppress errors between the state of the system with the reference input. Closed loop planning systems can be constructed by combining incremental planning techniques with multiple resolution models of the state space. Such approaches may provide a general methodology for implementing "anytime" algorithms[2]. Such characteristics permit real-time operating systems to make feasible schedules for resource allocation in a time constrained system.

2 Incremental Control Architecture for Manipulation Systems

The subsumption architecture was developed to construct systems which require composite behaviors[7, 8]. Raibert also adopted this *control factorization* methodology. The state space of complex systems is partitioned into disjoint regions, each with an associated control law [36]. This system was represented by a finite state automaton where state transitions are triggered by sensory events.

[2]algorithms which guarantee progress proportional to CPU time allocated

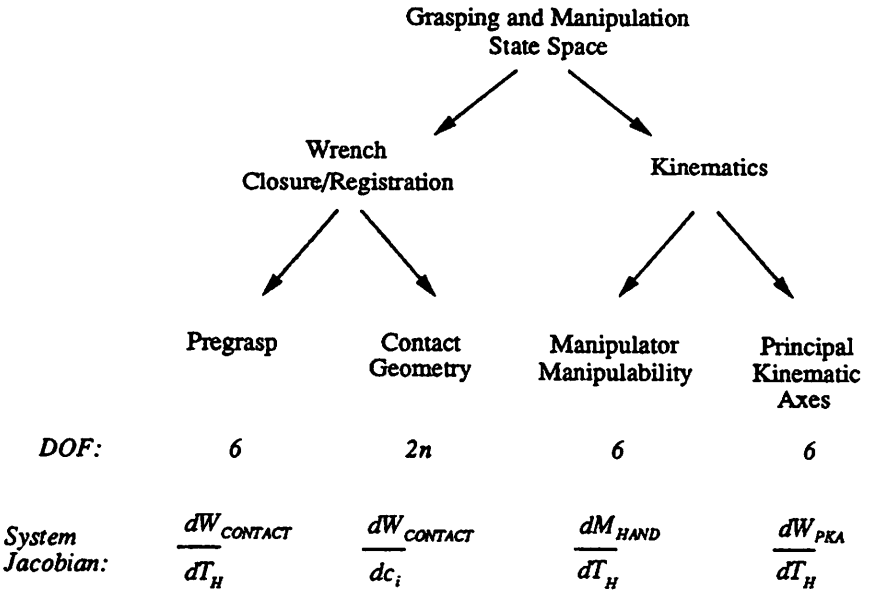

Figure 1: The Definition of System Jacobians for Grasp Synthesis

We have adopted a similar strategy. Figure 1 illustrates the control decomposition for the synthesis of hand/object configurations moderated by force-domain grasp requirements. In this domain, the state space is characterized by:

Object wrench spaces which can be generated by the manipulator, W_{PKA}, where PKA stands for the *Principal Kinematic Axes*[43] describing (among other things) the mapping from manipulator joint torques to Cartesian forces.

Object wrench spaces which can be transmitted by the surface of the object, $W_{CONTACT}$. The contact is modeled as one of the *canonical* contact types which is aligned with local surface coordinates and which employ prespecified coefficients of static friction.

A Manipulability Index for the hand. We have, in fact, employed a scalar field defined as the product of the manipulator manipulability and a scalar weighting coefficient reflecting joint range limits.

Figure 1 describes four behaviors in the system. Each of these system Jacobians describes how a particular grasp descriptor changes with respect to variations in a subset of the system degrees of freedom.

Pregrasp Hand Jacobian - $\frac{dW_{CONTACT}}{dT_H}$ — The sensitivity of the net wrench space, $W_{CONTACT}$, spanned by the set of contacts (with associated contact models) with respect to the

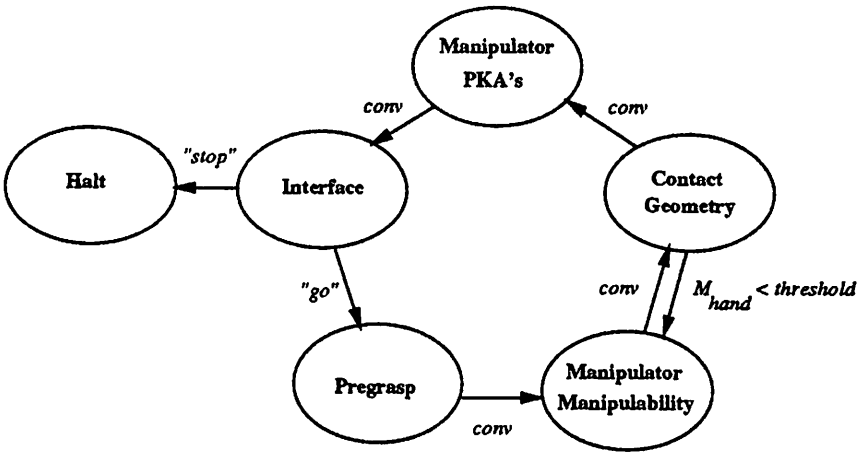

Figure 2: The Finite State Machine for Integrating Grasp Behaviors

position and orientation of the hand, T_H, is described while holding the configuration of the hand fixed.

Contact Geometry Jacobian - $\frac{dW_{CONTACT}}{dc_i}$ — The sensitivity of the net wrench space, $W_{CONTACT}$, spanned by the set of contacts (with associated contact models) with respect to contact positions, c_i, is described while holding the position and orientation of the hand frame fixed.

Manipulator Manipulability Jacobian - $\frac{dM_{HAND}}{dTH}$ — When contact positions vary while the hand's position and orientation is held constant, the kinematic isotropism of the hand degrades. The sensitivity of the Manipulability index of the hand, M_{HAND}, with respect to the position and orientation of the hand, T_H, is described while holding fingertip positions on the surface of the object fixed.

Manipulator PKA Jacobian - $\frac{dW_{PKA}}{dT_H}$ — The sensitivity of the net wrench space, W_{PKA}, spanned by the set of contacts (with associated relative, kinematically dependent force limits) with respect to the position and orientation of the hand, T_H, is described while holding fingertip positions fixed.

The result is a grasp strategy which *staircases* through the state space toward a solution. Grasp behaviors are selected to suppress state errors relative to a goal specification, or to depart from nearby grasp quality constraints in the state space.

Figure 2 illustrates a finite state machine in which each state has an associated control law. Each system Jacobian is the basis for a grasp behavior which controls a grasp quality metric by employing orthogonal subsets of the system DOFs.

We have shown that this form of control architecture consisting of: control states, abstract state transition events, and associated pre- and post-conditions for behavior application can be applied to a general class of grasping tasks *and* to manipulation tasks involving regrasp with stability constraints[18, 20]. Video taped sequences of regrasp manipulation strategies which consider simultaneous stability constraints have also been produced. The balance of this text will discuss the details of the grasping behaviors employed to date. After a brief discussion of kinematic behaviors for grasping, we will focus on the role of the object geometry in grasp synthesis.

2.1 Manipulator Kinematics

Several metrics have been suggested to describe the conditioning of a redundant manipulator[25, 28, 29, 37]. The *manipulability ellipsoid* is used in this work to characterize a manipulator's ability to generate forces and velocities[43]. The Principal Kinematic Axes (PKAs) of each finger may be identified by examining the transformation from joint space to Cartesian space expressed by the Jacobian. The singular value decomposition is employed to represent the character of the transformation. For a complete and detailed description of the singular value decomposition see Golub *et al.* [17].

The manipulator Jacobian is a linear transform describing Cartesian fingertip velocities resulting from joint velocities. The Jacobian representing a four DOF robot finger appears in Equation 1.

$$J = \begin{bmatrix} \frac{\partial x}{\partial \theta_1} & \frac{\partial x}{\partial \theta_2} & \frac{\partial x}{\partial \theta_3} & \frac{\partial x}{\partial \theta_4} \\ \frac{\partial y}{\partial \theta_1} & \frac{\partial y}{\partial \theta_2} & \frac{\partial y}{\partial \theta_3} & \frac{\partial y}{\partial \theta_4} \\ \frac{\partial z}{\partial \theta_1} & \frac{\partial z}{\partial \theta_2} & \frac{\partial z}{\partial \theta_3} & \frac{\partial z}{\partial \theta_4} \end{bmatrix} \tag{1}$$

The Jacobian is a linearized transform from joint space (R^4 is this case) to Cartesian space (R^3 in this example). If we were to map the unit sphere in R^4

$$\|\dot{\theta}\| = \dot{\theta}_1^2 + \dot{\theta}_2^2 + \dot{\theta}_3^2 + \dot{\theta}_4^2 \tag{2}$$

into Cartesian R^3 space through the Jacobian, the result is the manipulability ellipsoid proposed by Yoshikawa[42]. The ellipsoid describes the Cartesian character of the *amplification* in the Jacobian transform from joint velocities to Cartesian velocities. If the singular value decomposition of the Jacobian is computed, the manipulability ellipsoid is described by the Principal Kinematic Axes (PKAs):

$$\{\sigma_1 \vec{u_1}, \sigma_2 \vec{u_2}, \sigma_3 \vec{u_3}\} \tag{3}$$

where the σ_i are the singular values and the $\vec{u_i}$ are the corresponding singular vectors.

Chiu[10] points out that the singular vectors for the force relationship $\vec{\tau} = J^T \vec{f}$ are the same as those describing the velocity domain, but that the singular values are reciprocal. This

suggests that favorable velocity amplification is orthogonal to favorable force amplification. Moreover, since accuracy is inversely proportional to amplification, a reciprocal relationship also exists between accuracy and amplification in both the velocity and force domains as well.

The manipulability index of the finger is a scalar metric describing the conditioning of the kinematic transformation [43]. The index is proportional to the ellipsoidal volume spanned by the PKAs. This volume is approximated in practice by the product of the singular values. Since the volume of an ellipse increases as the ellipse becomes more spherical, there is a correlation between high manipulability indices and isotropic conditioning.

Joint space limitations can be considered by computing the product of the scalar manipulability index with a weighting coefficient which penalizes extreme joint positions. One such weighting coefficient is presented in Equation 4.

$$W(\theta_i) = \cos\left\{2\pi \left(\frac{\theta_i - \theta_{i,nom}}{\theta_{i,max} - \theta_{i,min}}\right)^2\right\} \tag{4}$$

This coefficient penalizes extreme joint configurations for robots which have ranges of motion from θ_{min} to θ_{max} and which have nominal joint positions midway through this range. The product of the weighting coefficient over all joints in a finger results in a scalar field defined over the workspace of the manipulator. The product of this scalar field with the manipulability index results in the weighted manipulability isograms presented in Figure 3. In practice, this field can be computed offline to quickly quantify the isotropic quality of a

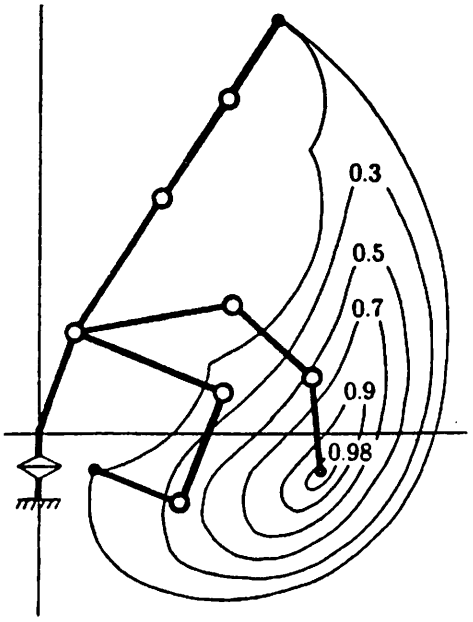

Figure 3: A Scalar Field Combining Manipulability and Joint Limits

particular hand configuration. This is important when the hand is required to comply to

changing contact geometries. The finger's workspace is discretized into $0.25in$ bins. The joint configuration which maximizes this conditioning metric, and the maximal value of the metric are recorded in the model. To evaluate the hand configuration an index for the hand can be computed using the parallel resistance analogy expressed by Equation 5.

$$M_{hand} = \frac{1}{\sum_i \frac{1}{M_i}} \tag{5}$$

This expression tends to associate the index for the hand with the smallest finger index. Behaviors derived from the hand index have been developed which enumerate candidate hand frame movements and compare the resulting hand indices. The candidate grasp which improves the the hand index most significantly is chosen as the successor state. Such hand configurations are capable of complying to object geometry and task specifications.

2.2 Object Representation, Planning and Control in the Force Domain

When the goal of a planning system to to reason about how to interact physically with the environment, it seems clear to first define what nature of interaction do we desire. We will describe a task specification which expresses a general class of force domain goals. Such goals include: pushing, stable grasping, manipulation, and legged mobility. We will not consider the ability of the robot to generate forces on the environment, that is the topic of Section 2.1. In this section, we will consider how a robot might reason about which features of the environmental geometry can be used to advantage in an interaction task.

2.2.1 The Task, the Grasp State, and the State Error

We express the task relative to an object coordinate frame as a set of wrenches (forces and torques).

$$\mathcal{T} = \{(\alpha_i^+, \alpha_i^-) \cdot \hat{t}_i\} \tag{6}$$

Where $\hat{t}_i \in R^m$ (m is the dimensionality of the wrench space) describes a basis for the task, and the α_i are the associated magnitudes in both positive and negative directions along $\hat{t}_i$.

Given a contact geometry with associated contact models we enumerate both positive and negative sense forces that may be transmitted by the surface at the contact locations. The set of these forces on the surface,

$$\mathbf{F} = \{\vec{f} | \vec{f} \in R^n\}, \tag{7}$$

maps to a set of object frame wrenches,

$$\mathbf{W} = \{\vec{w} | \vec{w} \in R^m\}, \tag{8}$$

$$\mathbf{W} \subset R^m, \tag{9}$$

using a linear mapping $\mathcal{G}$. If we restrict ourselves to the case where no contact torques are applied to the object's surface, then the mapping is defined by Equation 10.

$$\begin{aligned} &\mathcal{G} : R^n \mapsto R^m \textit{ such that} \\ &\mathcal{G}(\vec{f}_i) = (\vec{f}_i, (\vec{r}_i \times \vec{f}_i)) \\ &\mathcal{G}(\mathbf{F}) = \mathbf{W} \end{aligned} \tag{10}$$

where $\vec{r}$ is the position vector of the contact location relative to the task frame. The computation of an error with respect to the task specification, $\mathcal{T}$, requires that $\mathbf{W}$ be projected along the task basis, $\hat{t}$. In doing so we define the *utility* of the contact system for this task:

$$\mathcal{W} = \{(\beta_i^+, \beta_i^-) \cdot \hat{t}_i\}. \tag{11}$$

The error of the contact system with respect to the task is then the difference of magnitude limits along each of the task basis vectors,

$$\mathcal{E} = \{(e_i^+, e_i^-) \cdot \hat{t}_i\}. \tag{12}$$

The solution consists of simultaneously reducing the components of $\mathcal{E}$ by modifying the set of contact sites.

2.2.2 Grasp Synthesis

Given the error specification expressed by Equation 12, can one develop grasp strategies which eliminate these error components? We have cast this problem as an A* search in the state space described above. The cost of a solution is expressed as the sum total of contact displacements over the surface of the object from an initial contact configuration. The distance remaining to the goal is estimated by the sum of the squared error components expressed in Equation 12. The search space is the space of all possible contact discretized configurations. The root node of the search tree is the initial contact configuration and successors are defined to be the set of configurations obtained by moving exactly one contact to an adjacent bin. Thus, the branching factor is $B \times N$ where B is the number of bins adjacent to any contact and N is the number of contacts in the grasp.

An example of the A* behavior is presented in Figure 4. The grasp positions are designed to apply a net moment in both directions about the z axis. The three left most figures are solutions to the $+M_z$ task, the right most three are for the $-M_z$ task. This combinatoric grasp solution produces useful solutions for a wide variety of grasping tasks, however it fails to satisfy many of the specifications outlined in Section 1:

1. It requires complete geometrical descriptions of the object.

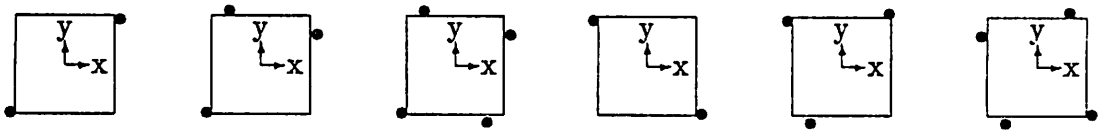

Figure 4: A* grasp solutions for a square peg with net moment tasks.

2. It produces strategies which are not robust with respect to uncertainties in position or geometry. Deviations from the plan require that large parts of the plan be discarded and re-planned.

3. A* (and all combinatoric techniques, in general) often produces long, unpredictable latencies before partial solution strategies emerge.

Grasp Stability: Building Null Spaces in the Grasp A necessary (but not sufficient) condition for grasp stability requires that the union of the wrench subspaces generated by the contact system and external forces (environmental) must be six dimensional. If this condition is met, infinitesimal perturbations of the object can be resisted by forces generated in the manipulator and/or environment. The robustness of this stability is related to the magnitude of perturbations that the contact system can resist. The Contact Geometry behavior considers the placement of contacts on the object's surface to make stability robust. The null space in the grasp is that space in which grasp forces can increase arbitrarily without producing net forces on the object. We may, therefore, scale the wrench subspace associated with frictional forces in the grasp by *squeezing* the object within the null spaces comprised of normal forces.

Principle 2.1 (Null Space Sufficiency) *IF a contact geometry is constructed such that there is a null space in the Grip Transform which includes wrenches derived from normal forces at each contact, AND IF the contact wrench system has rank* 6, *THEN arbitrary forces can be applied to the object via the contact system, AND the required contact forces can satisfy the friction cone at each contact by "squeezing" the object within the null space.*

Therefore, assuming soft finger contacts, infinitely strong hands and objects, we could theoretically grasp any object provided a one dimensional null space which includes normal forces can be constructed. This observation establishes a formal basis for an equivalence class of grasp geometries which can meet task (force domain) specifications. However, large internal grasp forces may exceed actuator capabilities and can contribute to instabilities in

the presense of position errors or force perturbations. To discriminate between *sufficient* grasps, we introduce the concept of grasp *efficiency.*

Principle 2.2 (Null Space Efficiency) *Grasp forces required to meet the task objectives can be minimized by aligning the contact wrench space generated by normal contact forces with the task wrench specifications.*

Unlike other approaches to grasp synthesis, we have adopted a strategy of computing incremental state space trajectories which construct null spaces in the contact system. The Contact Geometry behavior constructs sufficient grasps and the Pregrasp behavior focuses on grasp efficiency.

Contact Geometry Behavior Suppose that we are presented with an object and an existing set of contact positions. We wish to define a trajectory for one of these contacts to construct a null space in the grasp. From a given configuration of the contact system, the total squared error as a function of the surface trajectory of the selected contact, Δu, can be estimated using the following first order approximation.

$$\begin{aligned} \| \vec{E_T}^{\,t+1} \|^2 = \| \sum_{i=1}^{2N} \vec{e_i}^{\,t+1} \|^2 &\leq \sum_{i=1}^{2N} \| \vec{e_i}^{\,t} - \frac{d\vec{w}}{du}\Delta u \|^2 \\ &\leq \sum_i \langle \frac{d\vec{w}}{du}, \frac{d\vec{w}}{du} \rangle \Delta u^2 - 2 \sum_i \langle \frac{d\vec{w}}{du}, \vec{e}^{\,t} \rangle \Delta u + \sum_i \langle \vec{e_i}^{\,t}, \vec{e_i}^{\,t} \rangle, \end{aligned}$$

where $\frac{d\vec{w}}{du}$ expresses how the contact wrench changes in the neighborhood of the current contact position as a function of surface coordinate, u. This expression may be written as:

$$\| \vec{E_T}^{\,t+1} \|^2 \leq A(u)\Delta u^2 + B(u)\Delta u + C(u). \tag{13}$$

Since this estimate is quadratic in Δu we may solve for the surface trajectory which minimizes the estimated error.

$$\Delta u_{opt} = -\frac{B(u)}{2A(u)} = \frac{\sum_i \langle \frac{d\vec{w}}{du}, \vec{e_i} \rangle}{\sum_i \langle \frac{d\vec{w}}{du}, \frac{d\vec{w}}{du} \rangle} \tag{14}$$

Equation 14 defines a locally optimal surface trajectory for the selected contact. The contact migration behavior is repeated until the sum of projections $\langle \vec{e_i},\ d\vec{w}/du \rangle$ over all elements of $\mathcal{E}$ is zero. Since our error specification includes simultaneous positive and negative sense components along each task basis, this control policy results in contact geometries which produce null space in the contact wrench vector space. If we assume frictionless point contacts, then the solution attempts to build *sufficient* null space grasps (note that sufficiency also specifies full rank in the contact system, this property must be checked in general

to determine whether a sufficient null space has been constructed). Equation 14 is easily extended to 3D objects, where there are two linearly independent surface coordinates.

To illustrate this procedure we present the solution for a planar grasp on a circular object which has only 2 translational degrees of freedom in Figure 5. For this two dimensional

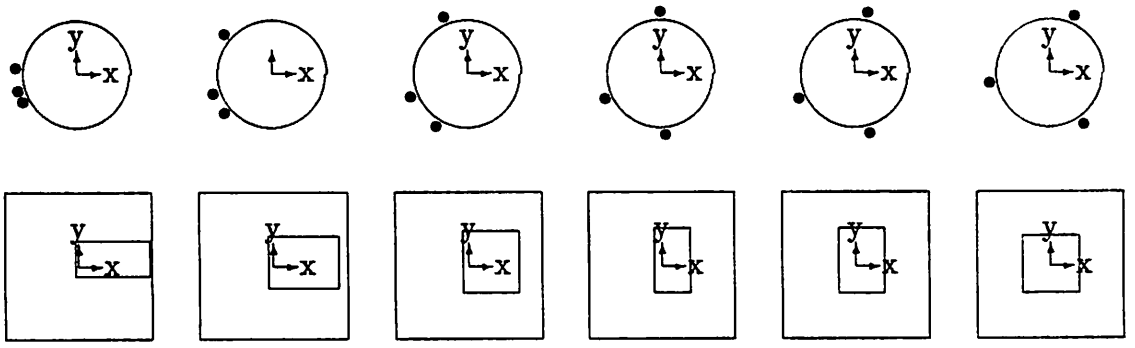

Figure 5: A three fingered grasp on a circular object using the local null space gradient.

demonstration we illustrate the task and contact basis (*volumes* in wrench space) as the grasp evolves. The larger of the two rectangles in the bottom half of Figure 5 represents the task volume and the smaller rectangle depicts the contact system volume which changes over successive frames. From left to right the contacts migrate from initial to final positions which construct null space in the contact system.

2.2.3 Multiple Resolution Object Models in the Force Domain

In practice, gradients based on local derivatives will vanish in contact positions which are not null spaces in the grasp. Local derivatives in the wrench surface do not encode global information, which can lead to myopic solutions where valuable features in the force space are not considered or to conflicts between the multiple components of the error specification. Figure 6 illustrates the $du(u)$ control policy expressed in Equation 14 for the 2D square shape. The task for this example is:

$$\begin{aligned} \mathcal{T} = \{ \quad & ((1,-1)(1,\vec{0},0)) \\ & ((1,-1)(0,\vec{1},0)) \\ & ((1,-1)(0,\vec{0},1)) \quad \} \end{aligned}$$

and the solution considers the placement of a second contact on the surface of the object given a fixed contact at $u = 0$ given a (hyper)cubic wrench task. Both contacts are modeled

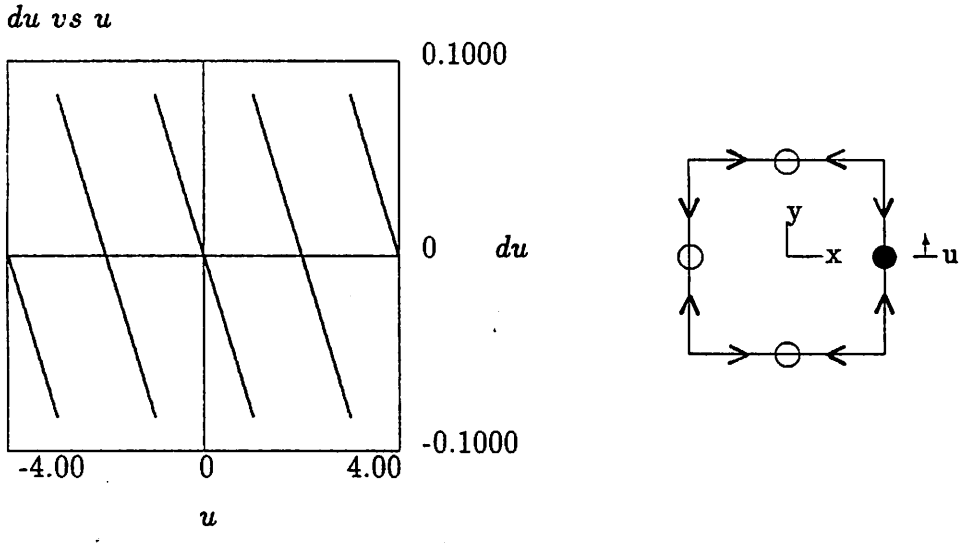

Figure 6: The Contact Migration Policy for the square shape.

as frictionless point contacts. The solid circle represents the fixed contact location, the open circles represent attractors in the migration policy for the second contact. Depending on where the second contact is placed initially, it will migrate toward one of four separate negative sloping zeros in the $du(u)$ policy. Positive sloping zeros in the control policy are unstable — causing the contact position to depart.

This problem is addressed by employing multiple resolution force domain knowledge in the surface migration strategy. A geometric model of the object is mapped into wrench space as a function of surface coordinate and contact type. These force diagrams are approximated as a finite series solution employing the fourier basis functions.

$$g(u) = \frac{A(0)}{2.0} + \sum_{n=1}^{m}[A(n) * cos(nu) + B(n) * sin(nu)] \quad (15)$$

The wrench subspace spanned by a contact and its derivative may now be expressed as a fourier series. We have shown that for 2D, regular polygons with the coordinate frame located at the centroid, the fundamental frequency is 1 cycle per circumference in the force diagrams and is n cycles per circumference in the moment diagram. The proof can be formulated in terms of the winding number of the force and moment functions. Intuitively, the force vector transmitted by a surface element for such objects accumulates rotations at the vertices of the polygons of $\Delta\theta = \frac{2\pi}{n}$, where n is the number of edges in the polygon. In order to generate a full cycle in the force component of the wrench, therefore, the contact location must traverse n edges of the closed curve. Equivalently, the fundamental frequency of the force diagram is 1 cycle per circumference about any section through an object. The moment diagram,

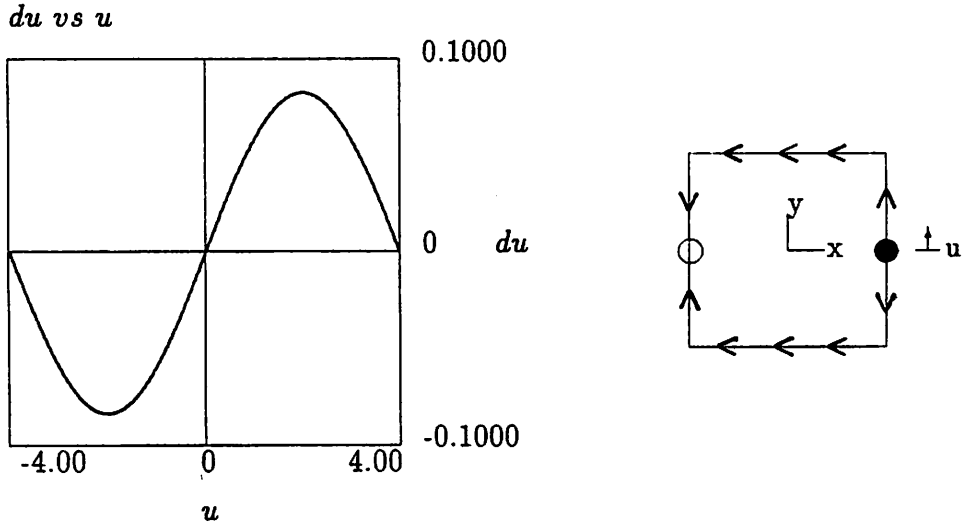

Figure 7: The Contact Migration Policy using 1 harmonic force models.

however, goes through an entire cycle along each edge of a polygon resulting in a fundamental frequency in the moment diagram of n cycles per circumference. Since the objects considered here are regular (symmetric) polygons, there is no energy in frequencies of less than n cycles per circumference. This observation extends to more general objects which exhibit n-fold cyclic symmetries. An example of this force domain signature for regular polygons can be found in Section A.1.

Figure 7 illustrates the control policy resulting from the 1^{st} harmonic model of the wrench space for the square. Consider, the placement of a second contact on the surface of the object given a fixed contact at $u = 0$. Figure 7 illustrates that the $du(u)$ policy defined by Equation 14 exhibits a single stable zero. It is clear that regardless of where the second contact is placed initially, it will tend to migrate toward $u = 4$ of the object's surface. We can show that the control policy expressed by Equation 14 exhibits at most two zeros when only 1 harmonic is considered. These locations represent singular points in a surface coordinate gradient field for the contact considered. One of these singular points will be unstable (positive sloping) and the other is stable (negative sloping $du(u)$ function). The trajectory resulting from this low resolution force model is a function solely of global attributes of the object. Successive harmonics create additional zeros in the control policy and displace existing zeros. As more detail is added to the force surface (more harmonics in the fourier series) existing stable zeros are bracketed by at most one stable and one unstable zero, causing the gradient field to deform.

Figures 8 and 9 describe what happens as the force domain object model is estimated

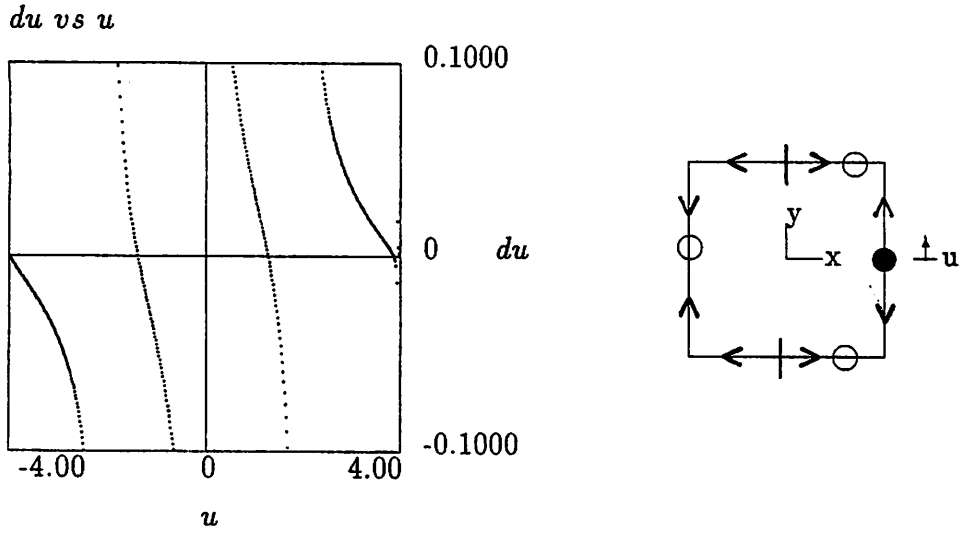

Figure 8: The Contact Migration Policy using 3 harmonic force models.

using 3 and 4 harmonic Fourier series approximations. This control policy seeks surface coordinates where the net projection of the error set onto local wrench derivatives vanishes. Since the error set includes both positive and negative sense objectives along each task basis vector (see Equation 12), this control policy can be viewed as constructing *null spaces* in the grasp— it suppresses errors in the contact wrench such that a migrating contact complements the balance of the set of contacts.

Pregrasp Behavior From a given configuration of the contact system, the total squared error as a function of the trajectory of the hand relative to the object, ΔT_H, can be estimated using the following first order approximation.

$$\begin{aligned} \| \vec{E_T}^{\,t+1} \|^2 &\leq \sum_{i=1}^{2N} \| \vec{e_i}^{\,t} - \frac{d\vec{w}}{dT_H}\Delta T_H \|^2 \\ &\leq \sum_i \langle \frac{d\vec{w}}{dT_H}, \frac{d\vec{w}}{dT_H} \rangle \Delta T_H^2 - 2\sum_i \langle \frac{d\vec{w}}{dT_H}, \vec{e}^{\,t} \rangle \Delta T_H + \sum_i \langle \vec{e_i}^{\,t}, \vec{e_i}^{\,t} \rangle \end{aligned}$$

or,

$$\| \vec{E_T}^{\,t+1} \|^2 \leq A(T_H)\Delta T_H^2 + B(T_H)\Delta T_H + C(T_H) \tag{16}$$

Here, T_H represents the transform that describes the position and orientation of the hand relative to the object. As such, there are then six candidate transforms we may wish to consider (just as there were n contacts we may wish to consider in the Contact Geometry behavior). Just as in the Contact Geometry behavior, we will consider only one such DOF at a time. We will discuss how to select that DOF such that the wrench registration problem is

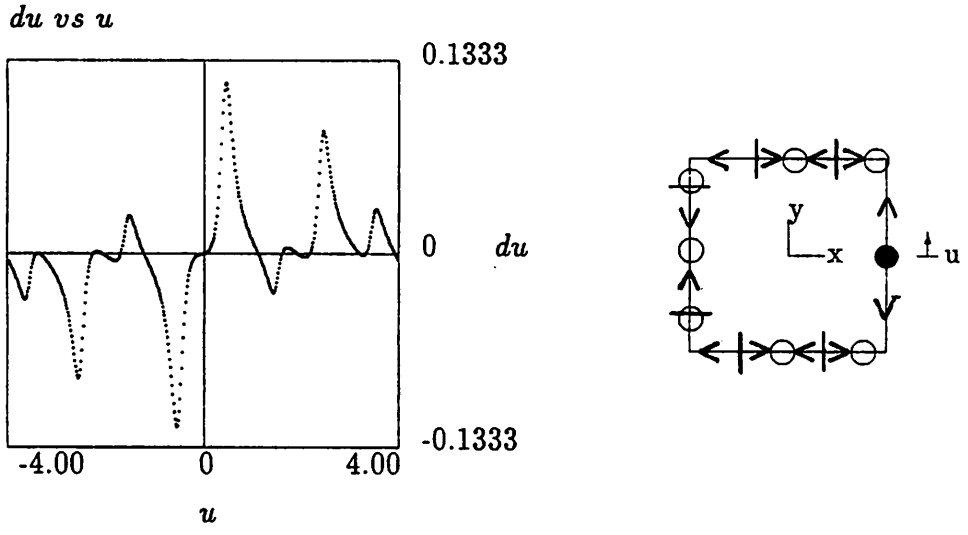

Figure 9: The Contact Migration Policy using 4 harmonic force models.

addressed in a latter section. Give this simplification, we may then solve for the ΔT_H which reduces the expression for the squared total error to a minimum.

$$\Delta T_{MSE} = -\frac{B(T_H)}{2A(T_H)} = \frac{\sum_i \langle \frac{d\vec{w}}{dT_H}, \vec{e}_i \rangle}{\sum_i \langle \frac{d\vec{w}}{dT_H}, \frac{d\vec{w}}{dT_H} \rangle} \tag{17}$$

The hand frame migration procedure continues until the sum of all projections of $\vec{e}_i$ onto $d\vec{w}/dT_H$ over all elements of $\mathcal{E}$ is zero. This approximation defines a locally convex policy for selecting a ΔT_H as a function of the position and orientation of the hand,T_H.

2.2.4 An Example: Regrasping the Square

Figure 10 shows a sequence of responses of this system to input wrench space tasks. A square object is illustrated in each frame along with a bar diagram depicting the wrench space volume of the task (empty rectangles) and the current contact wrench volume (cross hatched areas). The current wrench space is computed using truncated fourier series representations of the force diagrams. As successive harmonics are added, local features in the force space are incorporated into the solutions. This example also illustrates the Pregrasp behavior mentioned in Section 2.2.3. This behavior uses the same control policy expressed by Equation 14 but employs a different disjoint set of system degrees of freedom — namely, the 3 DOF defining the position and orientation this planar hand relative to the object. All other system DOF are held constant. In accordance with the finite state machine behavior composition, the task is always addressed by first executing the Pregrasp behavior followed

by the Contact Geometry behavior. A sequence of (re)grasp tasks is submitted to the system: the first is simply to construct a null space within a cubic wrench volume centered on the origin in force space.

The second task de-emphasizes the component of the force task in the x direction which results in the Pregrasp solution illustrated. This solution is refined by employing the Contact Geometry behavior with successively more harmonics. The third task submitted to the system represents a grasp which imparts a net positive moment to the object. Note that during the Contact Geometry behavior, the moment solution is not addressed until the wrench models contain at least the first 4 harmonics.

3 Discussion: Closed Loop Planning for Complex Robots

When considering the value of complex robot hands, it is worthwhile to examine the utility (direct and indirect) of the ancestor of these mechanisms — the parallel jaw gripper. This device may only be applied to certain kinds of grasping tasks, cannot address manipulation, cannot implement compliant motion control directly, and constrains enormously the geometry of objects with which the robot may interact. The value of the parallel jaw gripper is the extent to which it constrains the state space facing planners.

We have presented elements of a control architecture under development for multifingered robot hands. The promise of such devices is the general class of tasks and objects to which they may be applied. The challenge is the complexity of the mechanism and the expanded dimensionality of the state space afforded to them. In this section, we will discuss the features of the architecture under development which may help to achieve this goal and extrapolate to the form of a system which meets these objectives.

3.1 Graduated Optimality

It is valuable, in the context of our introduction, to build a controller which produces one of an equivalence class of solutions to the specified task. The multiple resolution force models we suggest have several positive attributes in this regard. Frequency encoding distinguishes between higher frequency components (local features) of the force model and low frequency (global) features. In our distributed control architecture, it is important for control behaviors which employ these models to be resilient to external constraints, i.e., manipulator kinematic constraints and constraints expressing what portions of the surface may be used when selecting a grasp (pick and place constraints). As successive harmonics are added to the models the control policy optimizes locally. The control behaviors which employ such multiple resolution models will be able to *do the best they can* when faced with arbitrary, externally defined constraints. Consider the following two examples:

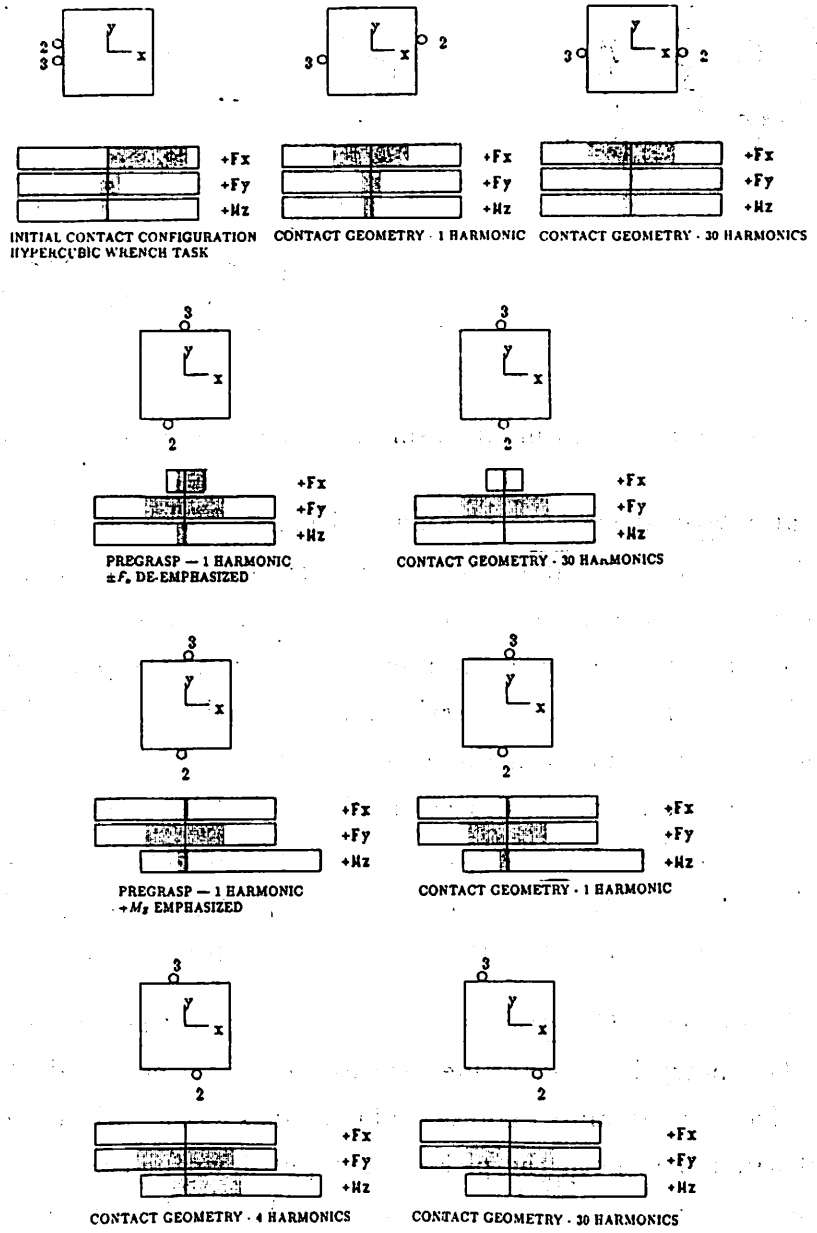

Figure 10: A trace of the (re)grasp solution over a sequence of force space tasks.

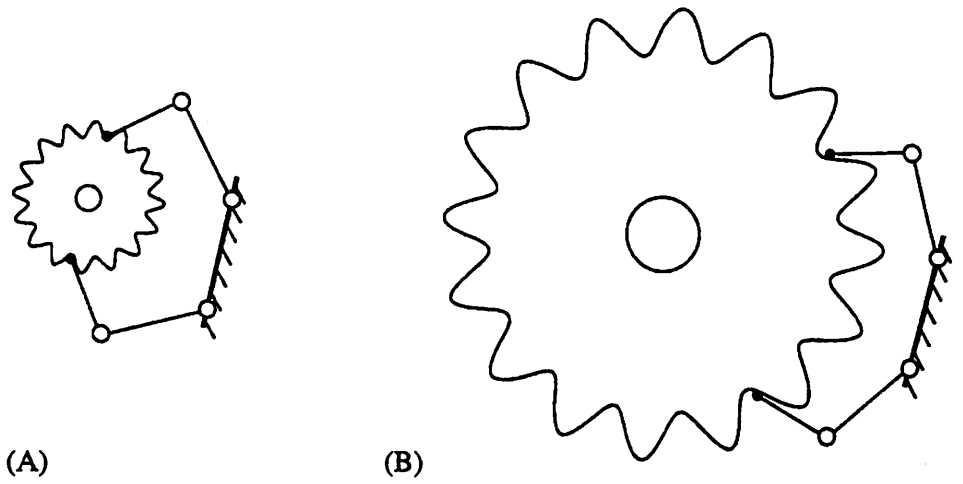

Figure 11: Graduated optimality using multiple resolution force domain models

1. Manipulator Workspace Limits - Figure 11 illustrates an object with k-fold cyclic symmetry. Such objects exhibit fundamental frequencies of 1 *cycle/circumference* and *k cycles/circumference* for the force and moment diagrams, respectively. The contact migration policy for the second contact is identical to the policy depicted by Figure 7. In the absence of external constraints, the second contact will move to oppose the fixed contact as illustrated in Figure 11(A). If, however, control policy is inhibited by kinematic constraints as illustrated in Figure 11(B), the subsequent consideration of higher frequency features of the force surface permit the contact geometry to optimize locally. If the result produces a sufficient null space (by Principle 2.1), then the control policy has succeeded in generating a grasp in the presence of external constraints.

2. Surface Accessibility Constraints - The task may be the source of constraints to the Contact Geometry behavior. One such example are the so-called *pick and place* constraints. These constraints express configurations of the object and manipulator which are unobstructed in both the pick-up and put-down phases of the task. Another instance of this form of constraint is the assembly of the grasped object to other sub-assemblies. The manipulator must not be permitted to uses surface elements which contribute to *fit* relationships with the sub-assembly. In these instances, the surfaces are simply marked inaccessible and the controller is forced to optimize by either incorporating more local features of the force surface or by selecting different contacts for migrations.

3.2 Nested Optimality

In practice, the Contact Geometry behavior (as presented) requires a means of selecting which contact should undergo a surface trajectory. We have recognized the utility of using secondary criteria to select which contact will be selected to undergo a surface trajectory. Selection criteria used to date include:

1. least registered contact

$$max E^2 = \sum_{i=1}^{2N} \langle \vec{e}_i, \vec{e}_i \rangle$$

The contact which individually produces the largest squared error is selected to migrate. The resulting null space is a more *efficient* solution.

2. steepest descent

$$max \sum_{i=1}^{N} \langle (e^+ + e^-)\hat{t}_i, \frac{d\vec{W}}{du} \rangle$$

The contact which most effectively addresses the instantaneous error is selected to move. The result is the *closest* null space.

3.3 Incremental Planning

Figure 12 is a schematic of the proposed approach which serves to highlight many of the scientific issues we address in this research program. There are two form of input to this system: a task specification and sensor data from multiple sources. Sensor systems extract information from the workspace environment in order to interpret the geometry of the environment to predefined levels of certainty. During this exploration, two models of the environment are constructed. The first is a local geometric model. This model is maintained to facilitate the consistent interpretation of sensor data. The second model is a global force domain model of the objects discovered in the environment. This model supports reasoning about general classes of potential contact interactions with the object. The force model represents a hypothesis about how general contact types map to object wrenches through the surface geometry. When a manipulation task is submitted to the system (as in Section 2.2.2), a planning agent interprets the current state of the workcell with respect to the task and employs a sequence of incremental, feedforward control laws.

3.4 Control-Based Active Sensing

While the control agents are active, the sensor systems are focused on particular objects in the workspace and on particular features of the objects which are critical to the developing interaction strategy. The proposed system constitutes an adaptive controller in which object

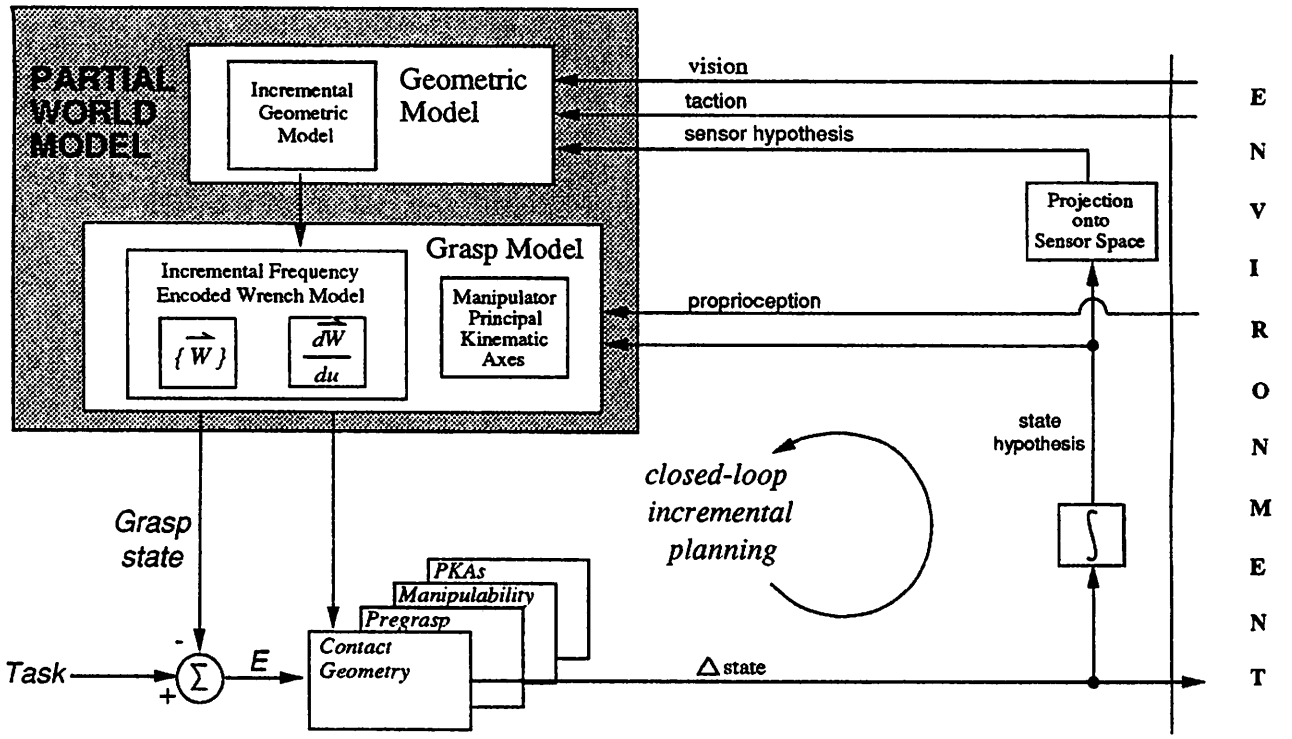

Figure 12: Closed loop planning using incremental sensing and control

model parameters are modified to improve the correlation between the predicted state and the measured state. We observe that low frequency information is extracted relatively easily from sparse visual information. This observation was used by Allen [2, 3] where occluding contours were used to generate initial object surface estimates. Subsequently, Allen demonstrated that surface and volumetric descriptions could be derived from tactile exploratory strategies[4]. The control processes described earlier begin to reason about the object using only these coarse force domain models. Concurrent sensing strategies extract additional geometric information and revise the existing geometric and force domain models. Moreover, actions derived from partial/uncertain models provide a mechanism for *opportunistic* tactile sensing — the fortuitous detection of object geometry during grasp formation. This is a potentially powerful technique for optimizing the use of computational resources since model sufficiency is determined by meeting the grasp objectives and not by achieving uniform certainty. Moreover, features of the object which are particularly relevant to manipulation goals should draw the attention of the grasp control agents and should also, therefore, contribute more detailed opportunistic sensor data to the evolving models. Since objects are often engineered for specific force domain tasks, one can imagine that the geometrical density of sensed data may well be related to the function of the object rather than the global certainty

of a surface parameterization.

3.4.1 Incremental Geometric Modeling

The primary requirements of the geometric model are: that it support the construction of a force model, that it be incremental, and that new information effects only local surface elements. Planar patches can be mapped easily into wrench space. In addition, they are a local representation that can be modified incrementally with new sensor data which is obtained either from visual or tactile sensors. A model which segments data into local, triangular surface patches meets these specifications. Extremal boundaries (occluding contours) provide estimates on the maximal extent of a surface. An extremal boundary defines a *cone* from the camera focal point, and the surface must lie within this cone [11]. At any time, the set of cones from multiple views can be used to detect inconsistency of the model with the visual data. Similarly, a tactile sense datum defines a normal to the surface, which constrains the extent of the surface. This is particularly useful for concave surface elements where visual data may not be available.

Visual data is obtained from a sequence of images taken from known camera motion and sequence produces 3D polygonal curves which are linked into patches of triangles by the following algorithm:

- Extract straight lines from each image.
- Match lines from successive images, using an epipolar constraint derived from the camera motion. The basis for this step is the algorithm developed by Williams and Hanson [41].
- Compute 3D lines and surface curvatures using the Giblin-Weiss algorithm [16]. This approach can also be used to distinguish extremal boundaries from creases and surface markings.
- Construct planar triangles using line endpoints.

There are two ways in which this tactile data could be integrated. In the case where a triangular planar patch has already been produced from the visual data, the uncertainty in the position of that triangle can be represented by a prism whose cross-section is the triangle and whose height is the uncertainty in the position of the triangle. A tactile measurement made within such a prism, can be used to subdivide the triangle or to reposition the triangle in the prism. If the measurement is made outside all such prisms then a new triangle is created. A truncated tetrahedron around that triangle representing its uncertainty in position and orientation is constructed. The evolving geometric data model is used to update the force model.

3.4.2 Incremental Force Domain Modeling

Section 2.2.2 demonstrated the process which maps geometry into a wrench space model. For the 2D square, each edge is mapped into wrench space using a contact force model. The composite function formed by summing the force image over each edge is then approximated using the Fourier series. This approach generalizes to surfaces which are only partially modeled. By modeling the surface as collection of planar patches, each patch can be mapped independently into wrench space.

The Fourier coefficients can be computed incrementally by considering the definition of the discrete Fourier transform:

$$F_{mn} = \sum_u \sum_v f(u,v) e^{-i(\frac{mu}{2\pi} + \frac{nv}{2\pi})}, \tag{18}$$

where (u,v) designates a object surface coordinate and (m,n) represents a discrete combination of spatial frequencies. A set of existing set of Fourier coefficients, F_{mn}, $(m = 0, M \ and \ n = 0, N)$, represents a hypothesis for an element of the force diagram. Sense data, $s(u,v)$, which are inconsistent with this hypothesis require each Fourier coefficient to be updated:

$$\Delta F_{mn} = [f_s(u,v) - f(u,v)] e^{-i(\frac{mu}{2\pi} + \frac{nv}{2\pi})}, \tag{19}$$

where, $f_s(u,v)$, represents the sensed datum, $s(u,v)$, mapped into the force domain as described in Section 2.2.2. ΔF_{mn}'s must be summed over the extent of the newly instantiated surface patch. Since there are $M \times N \times 6$ coefficients ($M \times N$ for each of the six wrench components) the complexity of the model update is polynomial in the number of harmonics maintained in the model. Moreover, the number of harmonics required is limited by the characteristic dimension of the contact. Thus, the force model update is effectively constant time.

Another approach for computing the Fourier coefficients is to use an optimal estimator based on the difference between the image in wrench space and the truncated Fourier series:

$$\Sigma w_j [\mathcal{G}(\vec{n}_j) - \Sigma (A_m \sin m\theta_j + B_m \cos m\theta_j)]^2 \tag{20}$$

where w_j is a weight based on the area of triangle j and n_j is the normal for that triangle, and $\mathcal{G}$ is the mapping from surface forces to object wrenches described in Section 2.2.1. We plan to investigate both methods for updating the force domain object model.

4 Acknowledgements

The authors with to gratefully acknowledge the contribution of Dave Oskard and Ted Schnackertz whose comments have improved the presentation.

References

[1] P.K. Allen. Surface descriptions from vision and touch. In *Proceedings of the 1984 Conference on Robotics*, pages 394–397, Atlanta, GA, March 1984. IEEE.

[2] P.K. Allen. Sensing and describing 3-d structure. In *Proceedings of the 1986 Conference on Robotics and Automation*, volume 1, pages 126–131, San Francisco, CA, April 1986. IEEE.

[3] P.K. Allen. Mapping haptic exploratory procedures to multiple shape representations. In *Proceedings of the 1990 Conference on Robotics and Automation*, volume 3, pages 1679–1684, Cincinnati, OH, May 1990. IEEE.

[4] R.J. Anderson. Dynamic sensing in a ping-pong playing robot. *IEEE Transactions on Robotics and Automation*, 5(6):728–739, 1989.

[5] R.A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2(1):14–23, March 1986.

[6] R.A. Brooks. Intelligence without representation. In *Proceedings of the Workshop on the Foundations of Artificial Intelligence*. MIT Press, 1987.

[7] J.F. Canny. *The Complexity of Robot Motion Planning*. MIT Press, Cambridge, MA, 1988.

[8] S.L. Chiu. Kinematic characterization of manipulators: An approach to defining optimality. In *Proceedings of the 1988 Conference on Robotics and Automation*, volume 2, pages 828–833, Philadelphia, PA, April 1988. IEEE.

[9] C.I. Connolly and J.R. Stenstrom. Construction of polyhedral models from multiple range views. In *Proceedings of the 8th International Conference on Pattern Recognition*, pages 85–87, October 1986.

[10] M.R. Cutkosky. On grasp choice, models, and the design of hands for manufacturing tasks. *IEEE Transactions on Robotics and Automation*, 5(3), June 1989.

[11] R.S. Fearing. Implementing a force strategy for object re-orientation. In *Proceedings of the 1986 Conference on Robotics and Automation*, volume 1, pages 96–102, San Francisco, CA, April 1986. IEEE.

[12] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. In *Proceedings of the First International Conference on Computer Vision*, pages 136–144, London, ENGLAND, December 1987.

[13] G.H. Golub and C.F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore, MD, 1983.

[14] R.A. Grupen. *General Purpose Grasping and Manipulation with Multifingered Robot Hands.* PhD thesis, University of Utah, Merrill Engineering Building, Salt Lake City, UT 84112, August 1988.

[15] R.A. Grupen, K. Biggers, T.C. Henderson, and S. Meek. Task defined internal grasp wrenches. Technical Report UUCS-88-001, Department of Computer Science, University of Utah, 1988.

[16] T. Henderson and R. Grupen. Logical behaviors. *Journal of Robotics Systems*, 7(3):309–336, 1990.

[17] S. Hutchinson and A. Kak. Planning sensing strategies in a robot work cell with multi-sensor capabilities. *IEEE Transactions on Robotics and Automation*, 5(6):7765–783, 1989.

[18] T. Iberall. Grasp planning for human prehension. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1153–1156, 1987.

[19] J.W. Jameson and L.J. Leifer. Automatic grasping: An optimization approach. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-17(5):806–814, September 1987.

[20] J. Kerr and B. Roth. Analysis of multifingered hands. *Journal of Robotics Research*, 4(4):3–17, Winter 1986.

[21] C. Klein and B. Blaho. Dexterity measures for the design and control of kinematically redundant manipulators. *Journal of Robotics Research*, 6(2):72–83, Summer 1987.

[22] Z. Li, P. Hsu, and S. Sastry. Grasping and coordinated manipulation by a multifingered robot hand. *International Journal of Robotics Research*, 8(4):33–50, August 1989.

[23] Z. Li and S. Sastry. Task-oriented optimal grasping by multifingered robot hands. *IEEE Journal of Robotics and Automation*, 4(1):32–44, February 1988.

[24] Z. Li and S. Sastry. Task-oriented optimal grasping by multifingered robot hands. *IEEE Transactions Systems, Man, and Cybernetics*, 11(10):681–689, 1988.

[25] M.T. Mason. *Manipulator Grasping and Pushing Operations.* PhD thesis, Massachusetts Institute of Technology, Cambridge, Mass, June 1982.

[26] M.T. Mason and J.K. Salisbury. *Robot Hands and the Mechanics of Manipulation.* The MIT Press, Cambridge, MA, 1985.

[27] T.N. Nguyen and H.E. Stephanou. A topological algorithm for continuous grasp planning. In *Proceeding of the 1990 IEEE Conference on Robotics and Automation*, pages 670–675, Cincinatti, OH, 1990. IEEE.

[28] V.D. Nguyen. The synthesis of stable grasps in the plane. In *Proceedings of the 1986 Conference on Robotics and Automation*, volume 2, pages 884–889, San Francisco, CA, April 1986. IEEE.

[29] V.D. Nguyen. Constructing force closure grasps. *International Journal of Robotics Research*, 7(3):3–16, 1988.

[30] V.D. Nguyen. Constructing stable grasps. *International Journal of Robotics Research*, 8(1):26–37, 1989.

[31] M.H. Raibert. *Legged Robots that Balance.* MIT Press, Cambridge, MA, 1986.

[32] J.K. Salisbury. *Kinematic and Force Analysis of Articulated Hands.* PhD thesis, Stanford University, May 1982.

[33] J.A. Stankovic. Misconceptions about real-time computing. *COMPUTER Magazine*, October 1988.

[34] R. Tomovic, G. Bekey, and W. Karplus. A strategy for grasp synthesis with multifingered robot hands. In *Proceedings of the 1987 Conference on Robotics and Automation*, pages 83–89, Raleigh, NC, April 1987. IEEE.

[35] J.P. Trevelyan. Sensing and control for sheep-shearing robots. *IEEE Transactions on Robotics and Automation*, 5(6):716–727, 1989.

[36] L. Williams and A. Hanson. Translating optical flow into token matches and the recovery of depth from looming. In *Proceedings of the International Conference on Computer Vision*, Tarpon Springs, FL, December 1988.

[37] T. Yoshikawa. Analysis and control of robot manipulators with redundancy. In *Robotics Research: The First International Symposium*, pages 735–747, 1984.

[38] T. Yoshikawa. Manipulability of robotic mechanisms. *Journal of Robotics Research*, 4(2):3–9, Summer 1985.

A Frequency Encoded Wrench Models for 2D Object Geometries

A geometric model of the object is mapped into wrench space as a function of surface coordinate and the contact type. These force diagrams are approximated as a finite series solution employing the fourier basis functions.

$$g(u) = \frac{A(0)}{2.0} + \sum_{n=1}^{m}[A(n) * cos(nu) + B(n) * sin(nu)] \quad (21)$$

A.1 Square

The fourier coefficients for the F_x, F_y, *and* M_z components of the wrench as a function of surface coordinate are presented in Table 1. The wrench subspace spanned by a contact and

Table 1: The first ten series coefficients approximating wrenches on the SQUARE

	F_x :		F_y :		M_z :	
I	A(I)	B(I)	A(I)	B(I)	A(I)	B(I)
0	0.000000	0.000000	0.000000	0.000000	0.125000	0.000000
1	-0.899593	0.044194	-0.044194	-0.899593	0.000000	0.000000
2	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
3	-0.297933	0.044194	0.044194	0.297933	0.000000	0.000000
4	0.000000	0.000000	0.000000	0.000000	0.125000	0.628417
5	0.176433	-0.044194	0.044194	0.176433	0.000000	0.000000
6	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
7	0.123514	-0.044194	-0.044194	-0.123514	0.000000	0.000000
8	0.000000	0.000000	0.000000	0.000000	-0.125000	-0.301777
9	-0.093441	0.044194	-0.044194	-0.093441	0.000000	0.000000
10	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

its derivative may now be expressed as a fourier series. Using only the first harmonic, we note that the force diagrams are approximated by a sinusoid with a frequency of one cycle per object circumference. Considering only one harmonic for this object, the moment diagram is featureless (with only a spurious DC component). When this lowest resolution wrench model is employed, grasp behaviors focus on forces and ignore the moment components of a task. When four terms in the series are considered, a non-zero amplitude appears in the moment function, and it thus effects the control policy. Figure 13 depicts wrenches transmitted to the object by a frictionless point contact as a function of the surface coordinate, u, on a 2D square object. The permitted contact wrenches are shown to be discontinuous at the corners

of the object and exhibit a higher characteristic frequency in the moment diagram than is present in the force diagrams.

A.2 L-shape

Figure 14 illustrates the result of frequency encoding the force space of a 2D L-shape. In this example, the moment diagram has been normalized — it is clear that the actual magnitudes of the moment about the z axis ranges between +/- 3 for unit normal contact forces. Once again we note the preference at the lower harmonics for the relatively large gradients in the F_x and F_y domains.

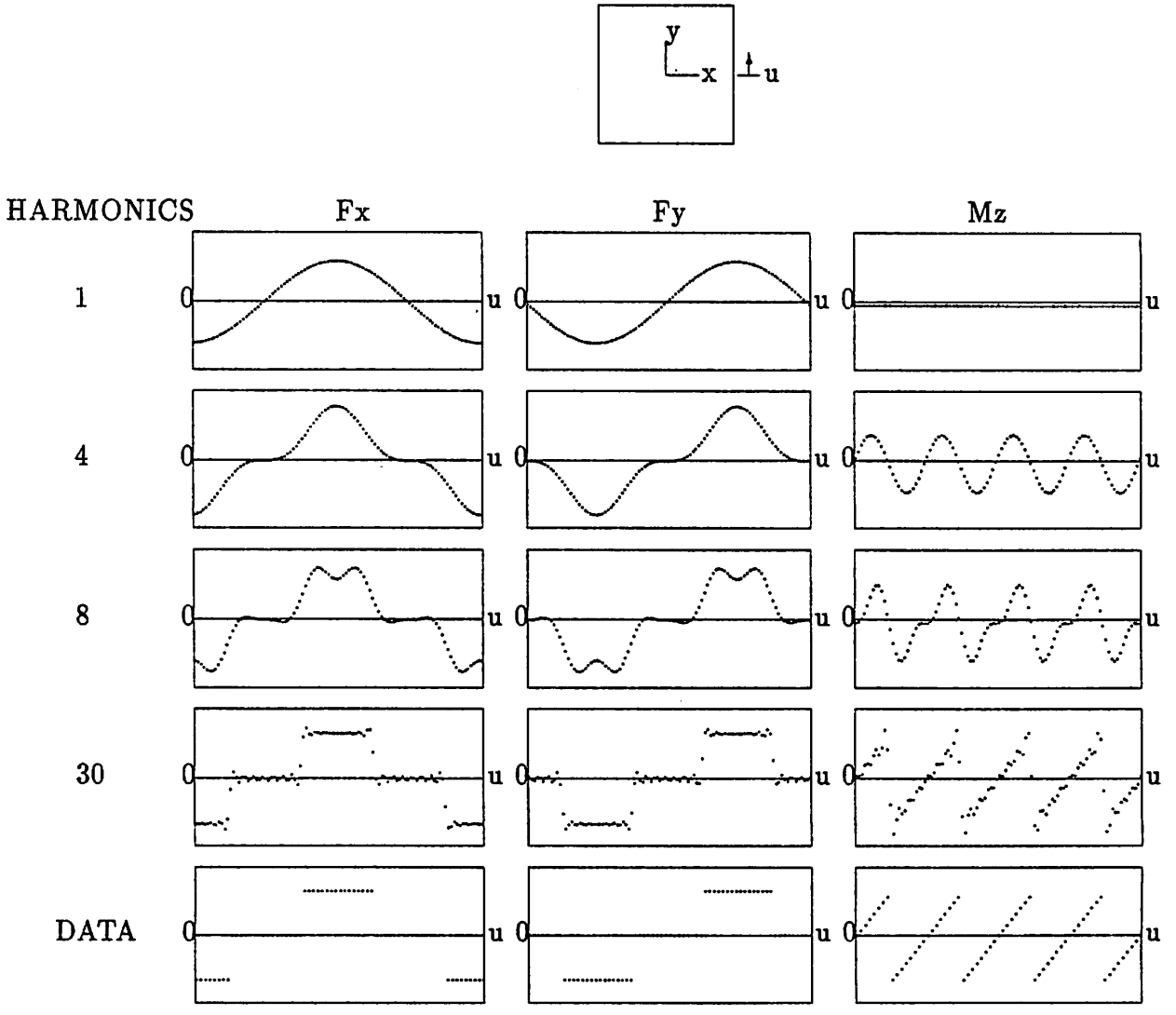

Figure 13: The 2D square geometry and the frequency encoded force and moment diagrams for a frictionless point contact as a function of surface coordinate.

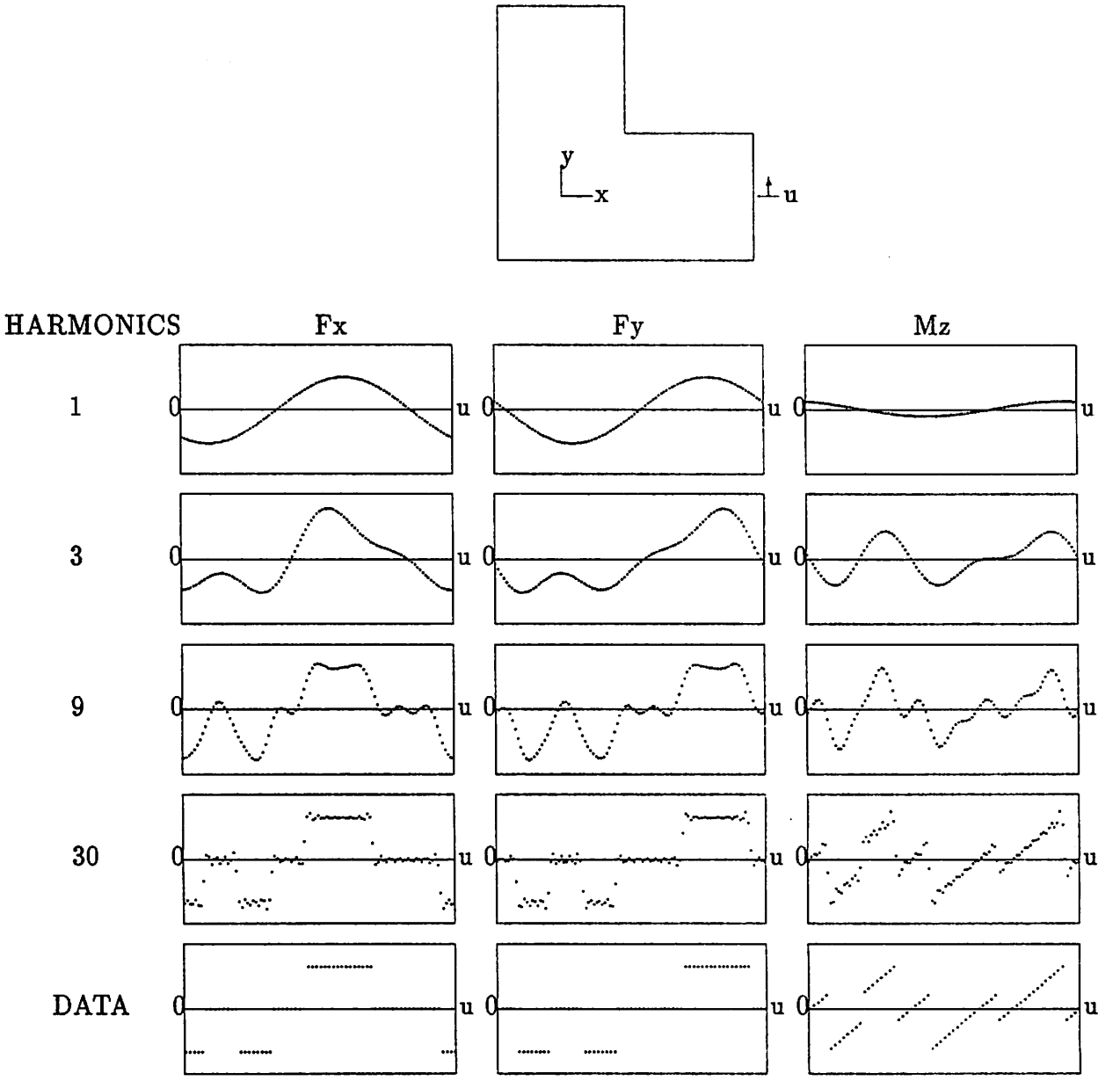

Figure 14: The 2D L-shape geometry and the frequency encoded force and moment diagrams for a frictionless point contact as a function of surface coordinate.