CHAPTER 3

KINESTHETIC PERCEPTION OF CLOSENESS TO OBJECTS AND INNOVATIVE DYNAMICS SENSITIVE PERCEPT
The model mediated approach referred in chapter 2, employ the remote environment model to locally generate feedback effects. We intend to use a fluidics based virtual environment model as mediator model and so attempt to develop a suitable model in following sections..

3.1 Haptics Oriented Model Design

Our goal is to form perception of an object’s presence by rendering some form of force in kinaesthetic domain to act as percept. Though notion of nearness has been used in several potential based approaches [21, 22] the objective of these approaches are to form a method of control of robot that takes it away from contact. The stress in these approaches is on forming some control law to avoid contact by treating them as obstacles [61]. We intend to develop a method for dissuading the operator to move toward object.

3.1.1 Fluidics Inspired Model

Consider fluid motion in a closed piston-cylinder assembly (figure 3.1). The cylinder has a side branch tube connected at tail of cylinder through which the fluid can flow back to the head end of the cylinder. The side assembly consists of one outlet let at.

![Figure 3.1: The single bypass cylinder behaviour.](image)
head end and a single inlet at tail end. A piston pushes the fluid from head end which flows back into the cylinder through the side loop To simplify the model we made some assumptions such as no friction between cylinder wall and fluid. Further, we consider no turbulence conditions and the flow to be in laminar regime. Volume flow rate through the cylinder [62] is

$$Q = \frac{(\nu \pi D^2)}{4}$$  \hspace{1cm} (3.1)

Where \(\nu\) = piston velocity, \(D\) = cylinder diameter

Flow resistance of pipe:

$$R = \frac{(128 \mu_f l)}{\pi d^4}$$  \hspace{1cm} (3.2)

Where \(\mu_f\) = fluid viscosity, \(d\) = pipe diameter, \(l\) = pipe length

Force on piston:

$$F = \Delta P \cdot A = Q' \cdot R$$  \hspace{1cm} (3.3)

Where \(Q'\) = volume flow rate through pipe, \(\Delta P\) = pressure difference between two sides of the piston

$$F = \nu \cdot \left(\frac{D}{d}\right)^2 \cdot \left(\frac{128 \mu_f l}{\pi d^4}\right) \cdot \left(\frac{\nu \pi D^2}{4}\right)$$  \hspace{1cm} (3.4)

$$F = \nu \cdot C \cdot \rho$$  \hspace{1cm} (3.5)

Where constant \(C = \left(\frac{D}{d}\right)^2\) and \(\rho = \frac{(128 \mu_f l)}{\pi d^4} \cdot \left(\frac{\nu \pi D^2}{4}\right)\)

\(C\) and \(\rho\) are functions of cylinder dimensions and fluid viscosity \(\mu_f\) which are constant for a given set up, leaving the force \(F\) as a function of velocity. The
opposing force $F$ on cylinder varies linearly with the velocity of the piston (figure 3.1). An imaginary object with such assembly placed normal to its surface and tail end in contact, opposes motion towards it (figure 3.2). The width $W$ represents velocity of piston i.e. velocity of approach towards the body and $F$ represents the opposition force.

![Figure 3.2: Opposing force to an approaching robot](image)

Figures 3.2a, 3.2b and 3.2c use width ‘$W$’ of the approaching vector as depiction of required opposition force ‘$F$’ presented to an approaching robot with increasing velocity of approach.

### 3.1.1a Properties of the Model

The model generates more opposition to movement at higher velocity. The model dissuades the piston if approaching at high speed but conversely allows it to reach the body at very low speed. This ensures reaching without high impact to the object. Its function is same along full length ‘$L$’. A model that can simultaneously make the opposing force ‘$F$’ dependant on nearness to tail end can also serve the additional purpose of developing a kinesthetic feel of nearness to end for the pusher.
3.1.1b Creating Position Sensitivity in Model

Consider a cylinder with modified return path as depicted in figure 3.3. Major part of the return is of large diameter. Here \( n \) segments of small diameter ‘\( d \)’ and length ‘\( l \)’ are connected in parallel. Thus \( n \) paths with flow resistance \( \rho_1 \) to \( \rho_n \) act as parallel.

Figure 3.3: Force behaviour in Multiple Pass Cylinder – ‘MPC’
return paths. Remaining part of loopback path has very large cross section compared to these and hence has ‘negligible resistance’ to flow. For a constant velocity ‘V’ of piston, ‘F’ is now dependent on the piston position in cylinder as available return paths change along the traverse. In case of travel from start to position 1, all n parallel paths are available (refer figure 3.3a and 3.3b).

\[ F = v \times (\frac{D}{d})^2 \times \left( \frac{1}{\frac{1}{\rho_1} + \frac{1}{\rho_2} + \frac{1}{\rho_3} + \cdots + \frac{1}{\rho_n}} \right) \]  \hfill (3.7)

In present example n is 5. As the piston moves inward in cylinder, return paths keep falling out of loop. At point 4 (figure 3.3c) resistance to flow is

\[ \gamma_4 = \frac{1}{\frac{1}{\rho_1} + \frac{1}{\rho_2} + \frac{1}{\rho_3} + \frac{1}{\rho_4}} \times C \]  \hfill (3.8)

At point 3 (figure 3.3(d)) the effective R increases to

\[ \gamma_3 = \frac{1}{\frac{1}{\rho_1} + \frac{1}{\rho_2}} \times C \]  \hfill (3.9)

the general relation is

\[ \gamma_n = \frac{1}{\frac{1}{\rho_1} + \frac{1}{\rho_2} + \frac{1}{\rho_3} + \cdots + \frac{1}{\rho_n}} \times C \]  \hfill (3.10)

figure 3.3e shows the complete behaviour of F for a constant velocity v of the piston. An approaching piston faces opposition force F that varies with distance remaining to tail end and its velocity v (figure 3.4). Since C is constant we have a general equation

\[ F = v \times \gamma_n \]  \hfill (3.11)
Chapter 3:
Kinesthetic Perception of Closeness to Objects & Innovative Dynamics Sensitive Percept

3.1.2 Model Function

The model enhances properties of single pass loop back cylinder by adding the capability to produce object vicinity based increase of $F$. A push action on piston at constant speed causes stepped increase at each bypass crossing. The effect is more enhanced for piston moving at higher velocity (figure 3.5). It retains the property of

Figure 3.4: The multi-pass cylinder responds to speed as well as distance from object

Figure 3.5: a; Conceptual piston-cylinder assembly attachment to an object’s outer surface for causing reaction to approaching robot arm, b; $F$ developed by multi-pass cylinder and its dependence on piston velocity
velocity dependence of earlier version. The position dependant change in F can be used to generate perceivable increase of F with nearness to a body part while approaching it and the property can be used to ensure reaching body without high impact. We call this model Multiple Pass Cylinder or ‘MPC’

3.1.3 MPC Model Evaluation

The model can be imagined as an arrangement placed with the closed end of cylinder touching the specimen part and the axis of cylinder aligned with normal to the surface (figure 3.5a). A point robot, ‘R’, approaching the specimen has to push the piston to reach the body surface. Such an arrangement achieves velocity based opposition to ‘R’ move in neighbourhood of body. It hinders move along ‘II’ in figure 3.5b, which has high velocity, but permits approach at a very low speed, reducing risk of damage by its impact as evident for move along path ‘I’. F, computed by (3.7) is treated as Response Parameter ‘RP’ for developing perceivable effect to operator. The stepped variations in \( \gamma_n \) is preferred (3.10) by some users as the resulting sharp change in F at positions 1,2,3 etc, in figure 3.5, cause easily perceivable feel of F. A smoothened F too can be mathematically formulated by using interpolations to intermediate positions.

3.1.3a Force Rendering from the MPC Model

For evaluation of the model in the proposed form, figure 3.5, of use in telecontrol application, output of MPC emulation process is used with an experimental force transducer. The MPC function is formed in two phases. In pre-run phase, a desired model is formed by choosing \( n, d \) and \( l \). Resulting impedances \( \gamma_n \) are held as an array (as in figure 3.6), facing a planar part in workspace. In this arrangement, \( \gamma_1 \) is
the impedance closest to part surface and is assigned from the last segment of the cylinder and $y_5$ is assigned 5th from first segment (figure 3.6) and so on.

In run time the ‘$\mathcal{R}$’ position is computed from the kinematics model using the present joint status of slave arm. This ‘$\mathcal{R}$’ position is binned as per the ‘distance to object’, in workspace giving applicable ‘$L_n$’, and corresponding $y_n$ value is identified and retrieved from array. The probe velocity ‘$v$’ is computed from discrete time stamped positions at time intervals or from velocity sensors on slave robot arm whose tool tip represents ‘$\mathcal{R}$’ (figure 3.6). Applicable Response Parameter ‘$R_P$’ is computed by (3.7). $R_P$ is converted in physical form by using servo motor operating in torque control mode for creating force-feel on Active Joystick (AJS) as opposition. Figure 3.7 outlines method for response parameter generation.

Figure 3.7: Experimental response parameter estimation for MPC model
In the evaluation experiments, the feedback is a force-feel created by an active joystick AJS, designed and developed for the experiments. AJS has 2 degrees of motion and renders force by using servomotors working in torque mode and coupled directly to the handle joint. The AJS is well balanced to appear passive without excitation of torque transducers. It has low inertia. (figure 3.8)

Figure 3.8( a and b): The balanced active joystick ‘AJS’ developed for validation of MPC model.

The torque demand signal vs. torque output of the motors is linear with programmable gain selection. It is dependent on the sensor output dynamic range. AJS has been formed using 100 W motor on \( \theta \) shown as axis 1 and 400 W motor on \( \phi \) axis shown as axis 2. The \( \theta \) axis has 0.32 N-m torque capabilities up to 50 rps with torque constant 0.36N-m/A. and \( \phi \) axis has torque 1.27 N-m up to 50 rps with torque constant 0.49N-m/A.

Figure 3.9(a and b): Rotation and balancing arrangement for the two axes.
3.1.3b Response Parameter Estimation and Rendering

The applicable $\gamma$ for the robot position is computed from the joint sensor values and robot kinematics model. The method followed is outlined in figure 3.7. For RP conversion to ‘force-on hand’, the servo motor and controller combination is configured in ‘torque-mode’ and operating sensitivity is designed as illustrated in figure 3.10 and figure 3.11. Note that the transducer formed by the motor and controller combination has very good linearity.

![Figure 3.10](image1.png)

**Figure 3.10:** a; $\theta$ axis command Voltage-Torque Characteristics, b; $\theta$ motor characteristics (torque control mode)

![Figure 3.11](image2.png)

**Figure 3.11:** a; $\phi$ axis command Voltage-Torque Characteristics, b; $\phi$ motor characteristics (torque control mode)
3.1.4 Experimental Verification of Model Performance

Experiments were carried out for different design of the MPC model as described below.

3.1.4a Experimental Setup

A 5 DOF robot’s (figure 3.12) only translation stages x and y were used keeping z constant in experiments for $R_P$ generation. In the experiment, time stamping was used for unified referencing. The legend AJC represents joy-stick controller.

Figure 3.12 a; 5 DOF Cartesian robot experimental set-up, b; Signal flow in a single axis test setup. AJC- joystick controller.

In the experiments, slave motor (in 5DOF robot) worked on velocity control mode. Velocity control signal to the controller was formed by 14 bit accuracy D/A output and updated at 100 Hz. Shaft encoder tracking was achieved by the slave joint controller and time stamped position was acquired. For high speed approaches, the object surface was used as data coded limit and real object was removed for avoiding inadvertent damage during experiments.
3.1.5 MPC Model Variety

The MPC model can be differently designed by changing physical parameters of the bypasses. The interval between the bypasses plays an important role in forming different models of MPC. This is explored below.

3.1.5a Regular Size ‘$g$’ Based MPC Model

The probe robot arm was moved along a fixed ‘Y’ at different constant velocities which were sensed by the robot joint sensors. The $RP$ values generated by the model are shown in figure 3.13. $RP$, developed by system for same conditions but constant accelerations and constant deceleration are shown in figure 3.13b and figure 3.13c.

![Graphs showing response parameter generated by the model under controlled speed operation](image)

3.1.5b Effect of Variation of gap $g$ between Bypasses in MPC Model

The parameter ‘$g$’ i.e. gap (figure 3.3a) is mapped as $k$ (integer) numbers of voxel of length $V_x$. Therefore $k$ is scale factor between real workspace and model. For $k = 2$, fluid resistance $\gamma$ forms an array that appears as

$$[\gamma_1, \gamma_1, \gamma_2, \gamma_2, \gamma_3, \gamma_3, \gamma_4, \gamma_4, \ldots, \gamma_{n-1}, \gamma_{n-1}, \gamma_n, \gamma_n]$$
Figure 3.14: a; Coding for narrow ‘\(G\)’, b; for unequal gap ‘\(G\)’

profile is depicted by different colour code in figure 3.14a and is repeated for a set of

\(Y\) values for facilitating single axis motion along \(X\) axis. If value of gap \(G\) is reduced

then \(k\) reduces. Lowest value can be \(k = 1\). The coding is shown in figure 3.14a.

**3.1.5c Performance of the Transducer**

The servo controller of AJS motors provides instantaneous torque proportional

signals at monitoring output.

\[
\tau = \lambda \ast RP
\]  

(3.12)

Where, \(\lambda\) is the torque sensitivity to control signal for AJS

For a nominal holding position at 75mm from the AJS shaft, the force appears as in

figure 3.15 for ‘\(R\)’ motion executed by controller at zero acceleration, constant

acceleration and constant deceleration.

Figure 3.15: Transducer performance (\(\theta\) axis) for constant \(G\) and fixed, accelerating

and decelerating motions. In experiments, \(x = 10cm\). (Plots are for fraction of \(x\))
3.1.6 MPC Evaluation for Narrow Gaps and Varying Gaps

The rendered effect from unequal gap design (figure 3.14) is depicted in figure 3.16b. The advantage of narrow gap is evident in closer vicinity where a responsive operator reduces speed causing ‘R’ slowing down but bypasses being nearer still senses slowdown owing to frequent change of γ rather than change of ν which may not be appreciable over short travel. An interesting way of enhancing closeness perception is to have gap 𝐺 of different sizes. At periphery of vicinity 𝐺 can be

![Graphs](image)

Figure 3.16: Effect of changing 𝐺 on force feedback, a; regular gap with  𝑘 = 2, b; regular gap with  𝑘 = 1, c; initially  𝑘 = 2 and later  𝑘 = 1. (In experiments,  𝑥 = 10cm. Plots are for fraction of  𝑥)
longer and close to object it can be reduced. For such case the \( \gamma_n \) coding for the same part is as shown in figure 3.14b. The resulting \( RP \) is shown in figure 3.16c. As \( \mathcal{R} \) approaches closer to the object the transducer produces more frequent \( RP \) change giving feel of increasing proximity by increasing frequency of step changes on AJS.

### 3.1.7 Observations from MPC Evaluation.

The AJS produces stepped force opposing operator hand which increases in magnitude with advance of \( \mathcal{R} \) towards object. For higher velocity of approach, at same relative distances from object surface it produces higher force magnitude step. On frequency scale the faster move generates increasing frequency of steps. A design with lower \( G \) near cylinder end offers relatively faster step occurrence at low velocity also and enhances perception of approach to object more effectively. The \( RP \) in experiments produced perceivable opposition force to operator’s hand placed at nominal distance of 75 mm from rotation axes in manner expected using \( M_1 \) on 100% dynamic torque range. For constant speed move, generated by computer controlled servo, appreciable stepped changes in force were produced for low speeds when high torque gains were set. But for speeds above 0.3 m/sec. force was high enough for reasonably strong push back to operator. Response parameter developed here works well (Table 3.1). The variety of the response makes its rendering very effective for conveying the proximity based object presence to AJS operator. Individual’s sensitivity being different, range tuning is desirable and selection of \( RP \) to force ratio’ by adjusting torque gain of \( M_1 \) or \( M_2 \) as per their sensitivity. Experimenters perceived the approach phenomenon in spite of not being allowed to see the robot and workspace.
Chapter 3:
Kinesthetic Perception of Closeness to Objects & Innovative Dynamics Sensitive Percept

3.1.8 Observations on Model Function

The function of ‘virtual transduction’ is to create a feedback parameter for use at master end based on telerobot’s proximity to parts in workspace and its own dynamics as depicted in figure 3.5. While the basic model is virtual, the rendered effect achieved by the transduction has to be real. The function of MPC model is to create a kinaesthetic feedback parameter for use at master end based on telerobot’s proximity to workspace parts and its own dynamics. The method must mitigate abrupt feedback to the operator on surface contact, it must offer programmability of the modelled parameter in an unambiguous manner, opposition must appear in close vicinity of specimen surface and must depend on closeness to body. The functional

<table>
<thead>
<tr>
<th>Motor</th>
<th>Mode</th>
<th>Control signal (volts)</th>
<th>Torque Usage</th>
<th>Experimental observation on force rendering</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1</td>
<td>Vicinity at normal speed</td>
<td>0 – 3.6 (+/-)</td>
<td>Normal demand</td>
<td>Effective force feedback dependant on vicinity. AJS operation is low inertia, quick response. Transducer action very good. The perception developed is well suited for delicate manipulation.</td>
</tr>
<tr>
<td></td>
<td>High Velocity</td>
<td>3.6 - 8 (+/-)</td>
<td>Overdrive 250%</td>
<td></td>
</tr>
<tr>
<td>M2</td>
<td>Vicinity at normal speed</td>
<td>0 – 3.8 (+/-)</td>
<td>Normal demand</td>
<td>Effective force feedback dependant on vicinity with wider force range. Transducer action very good. AJS feels heavier on hand and can resist operator motion apart from only forming perception</td>
</tr>
<tr>
<td></td>
<td>High Velocity</td>
<td>3.8 - 7 (+/-)</td>
<td>Overdrive 200%</td>
<td></td>
</tr>
</tbody>
</table>

Table 3.1: Performance of MPC driven percept rendering using AJS
requirements dictate that it should be suitable to real time working and must leave the slave side control unaltered.

The simple model offers following features

i. It is well suited for transduction and rendering haptic effect on a force feedback joystick.

ii. The model has sensitivity to position that is of discrete nature.

iii. A constant speed move generates distinct force steps.

iv. The force steps, nonlinearly rise in magnitude when closing in to object and are more distinct near object.

v. The step weights are amplified by speed of piston motion.

vi. The step interval on time axis is constant for a fixed speed but reduces with higher speed. In effect the step frequency increases with speed of piston motion.

vii. The model offers flexibility of sizing the gap ‘G’, allowing a frequency based change of step rate. A frequency modulation steps wherein a piston sees more frequent force steps closer to object when moving at a fixed speed.

3.2 Viability of MPC Model Application

In figure 3.17 an articulated arm is treated as example. The joint configuration is illustrated by stick model. It has $\phi$ rotation around vertical rotation axis and $\theta_1, \theta_2, \theta_3$ rotations around tilt axes that cause rotation of links. The length of the vectors formed between the points on previous state of arm and corresponding present instance of the respective points are the move vectors $P_1P_1', P_2P_2', ... P_uP_u'$. 
These approximate motion directions of the points on body when extrapolated, intersect the body at $P_n^\prime$.

Figure 3.17: (a, b) Move vectors on various points on an articulated arm; (c, d) The MPC model on a linear approach to the object ‘O’

If $P_n^\prime P_n^\prime$ length is equal to piston stroke length along the cylinder i.e. $L$, the model action needs to be emulated. Binning of $P_n^\prime P_n^\prime$ is done between $\lambda_6$ to $\lambda_1$ and applicable $\gamma_n$ is found from the model design based on $d, D, l, G$ and $\mu_f$.

The velocity is computed by:

$$V = \frac{|P_n^\prime P_n^\prime|}{\Delta T}$$ (3.13)

Where $\Delta T$ is encoder sampling interval. The $F$ is computed by (3.7)

Therefore, the model application is primarily viable, but in the case below, $P_n P_n^\prime$ does not hit object ‘O’ but grazes past it and in such case also the vicinity based drag must act on the robot. Therefore there is a need to probe the condition by using transverse vectors ‘$TD$’ normal to $P_n^\prime P_n^\prime$” and the distance $TD$ as shown in the diagram determines applicable $\lambda_n$ by binning $TD$ instead of $P_n^\prime P_n^\prime$. The procedure has to be followed in all direction as shown in figure 3.18d.
3.2.1 Probe Penetration Based Approach

Alternatively objects modelled in any manner referred earlier can be intersected by a probe vector and applicable $\gamma$ can be estimated. For planar objects and other objects approximated by planar surfaces at the point of penetration, the applicable MPC model can be estimated by projecting a vector in direction of its move with length representing the next move distance. But large number of surfaces has to be tested.

3.2.2 Search Scope Control

The concepts of section 2.8.2 can be used in vector - object intersection for limiting the inclusion of object or surface entities within search process [63, 64]. The method can be seen as ‘Search Scope’ control. It can be done by use of ‘Smallest Encapsulating Sphere Object’ ‘SESO’ with radius ‘$r$’ figure 3.19. It achieves reduction by excluding objects (and their component surfaces) that are farther than ‘$r$’ from current location $P_n$ through simple test of comparing object centre distance from $P_n$. 

![Figure 3.18: Cases needing different treatment](image)
3.2.3 Probing ‘CAD’ Generated Objects

Several tests have been chained to reduce computation cost of testing vector penetration in a part. These are following. ‘Search scope’ control extension to Stereo Lithographic surfaces that are planar triangular surfaces called ‘STL’ by using ‘Smallest Encapsulating Sphere for Surface’ patch ‘SESS’. The working is similar to SESO but it leads to data enlargement problem as each surface triangle must have an associated CG and radius stored. Data structure also needs redefinition of object CAD files for this data inclusion or attachments of these data as linked data. SESS for large surface is big. In a typical situation (figure 3.19d) ‘R’ approaching towards the object sees two SESS formed by surfaces 2,3 and by 7,8. But SESS 2,3 being on the other side, will not be intersected in reality whereas surfaces 7,8 will be. The basis of this test is that since the $P_n'$ is on robot body, it is naturally outside any object and so any surface with exposed side’s normal, not opposite to $PP_n'$ cannot be reached without crossing the object body and so is not reachable. We refer to this test as Surface Direction Test ‘SDT’. Surface direction is coded as order of the three nodes in STL triangle data-set in the triangle list. In this work surface direction is used to eliminate surface 2-3 from SESS at position ‘k’ as it is not facing move.
vector $\mathbf{P}'_n \mathbf{P}_n$ (figure 3.17). The test is named ‘surface direction test’ SDT. Algorithm is similar to detecting occluded surface in graphics rendering [43]. It uses angle between the vector $\mathbf{P}'_n \mathbf{P}_n$ and STL surface normal ‘$n$’ to test whether the exposed side is towards $P$ or not.

### 3.2.4 Intersection Determination and Computation Reduction

A ray (infinite length) in probe direction ‘$\overrightarrow{pq}$’ is tested for intersection with STL’s plane ‘$S$’, formed by nodes $S1, S2$ and $S3$. Except for the case when ray is parallel to this plane an intersection occurs (figure 3.20a) somewhere in space at $p''$.

For a penetration to be ‘true’ two conditions are necessary. i) $p''$ should be within the STL triangle $S1S2S3$ and ii) the distance of this intersection from $p$ i.e $\|pp''\|$ should be $< L_6$, for the MPC to touch or penetrate the object to which the STL belongs. $L_6$ is the probe length that is equal to longest MPC piston stroke. Only intersection determination with plane ‘$S$’ is called intersection determination procedure ‘IDP’ in further discussions.

**Range Test** : The second test listed as (ii) above is ‘Range test’ (RT) and is done first to reduce computation in case a failure was to happen.

![Figure 3.20: Optimization by test order selection – minimizing late fails. The graded bar is aligned along move vector $pq$ which intersects plane of STC on $p''$](image)

J. K. Mukherjee, HBNI, Mumbai

Ph.D. Thesis
### Table 3.2: Computations in vector- STL object interaction

<table>
<thead>
<tr>
<th>Sr no.</th>
<th>Procedure</th>
<th>Operation Performed</th>
<th>Floating Point Operations Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Within SESO “SESO”</td>
<td>a) Non-Optimized (3 S, 3 M, 2 A, 1 SR, 1 C)</td>
<td>(12~35) * $N_{max}$ $N_{max}$ = total objects in Workspace. “SESO (Load)”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Optimized (3 S, 3 M, 2 A, 1 C)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Within SESS “SESS”</td>
<td>a) Non-Optimized (3 S, 3 M, 2 A, 1 SR, 1 C)</td>
<td>(12~35) * $\sum_{i=1}^{N_{q}} N_{max}(i) N_{max}$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Optimized (3 S, 3 M, 2 A, 1 C)</td>
<td>(i) is number of triangles in an $i^{th}$ object within the object CG radius. “SESS (Load)”</td>
</tr>
<tr>
<td>3</td>
<td>Surfaces Facing Observer “SDT”</td>
<td>a) Non-Optimized (3 S, 3 M, 2 A, 2 SR, 1 D, 1 T, 1 C)</td>
<td>(12~82)* $N_{tq}$ where $N_{tq}$ is number of triangles within the surface CG radius “SDT (Load)”</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Optimized (3 S, 3 M, 2 A, 1 C)</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>“IDP”</td>
<td>(12 M, 9 A, 1 D, 1 C)</td>
<td>(57)*$N_{ft}$ where $N_{ft}$ is number of triangles facing vector. “IDP (Load)”</td>
</tr>
<tr>
<td>5</td>
<td>Range Test “RT”</td>
<td>4M 5S 1 C</td>
<td>(14)* $N_{ray}$ where $N_{ray}$ is number of plane intersect within radial distance from observer “RT (Load)”</td>
</tr>
<tr>
<td>6</td>
<td>Vector Intersect Test “VIT”</td>
<td>(24M, 18S, 13A, 1 L, 2C)</td>
<td>(82)* $N_{rayq}$ where $N_{rayq}$ is number of triangles having intersection with the observer probe length. “VIT (Load)”</td>
</tr>
</tbody>
</table>

**Notes:** This table assumes there are ‘$N_{max}$’ numbers of objects in the workspace. Here, IDP=Intersection Determination Procedure, S=Subtraction (1 FPC), A=Addition (1 FPC), E=Exponent, SR=Square Root (23 FPC), M=Multiplication (2 FPC), T=Trigonometric (1 FPC, considering lookup tables), C=Comparator (1 FPC), D=Division (=23 FPC), L =Logical Operation (1 FPC), CG=Centre of Gravity, ‘$N_{max}$’ = Total numbers of objects in the workspace.
Vector Intersection Test: Vector intersection test ‘VIT’ finds whether the intersection point $q''$ is within triangle STL on surface ‘$S$’ (figure 3.20c). This computation is costly (case 6, Table 3.2). A failure may happen even after this test if the virtual cylinder falls short of touching ‘$S$’. Therefore a wiser approach is to change order of testing and first test $\delta < L_6$ condition (figure 3.20b) and then test ‘inside $S$’ condition. It is termed vector intersect test here. It can result in a minimum cost equal to that of ray test and maximum cost equal to that of vector test.

In summary the determination of probe contact with surface comprises IDP+VIT+RT. Ordering VIT and RT is important. Ordering IDP followed by RT and then VIT is beneficial. Process IDP+RT is called vector intersection procedure (VIP). These methods can be seen as filters (figure 3.21) that terminate the test loop early by predicting non-intersection condition and save time. The time ‘$t_s$’ represents time within which sensing should be complete and procedure should start for the next position $P_{n+1}$. Time $t_R$ is available to react in true vicinity. For reducing computations, reusable data like ‘bounding box extents, centre and radius of bounding spheres etc. are computed once at start up and associated with part models as additional persistent data.

Figure 3.21: Filters for computation reduction in vector object intersection
3.2.5 Tests for Assessment of Vector Object Intersection Optimization:

A software test environment has been built to apply the filtering methods on line and access the parameters indicating computations passing through the filter processes at various locations of a point robot on selected locus. Objects (CAD in STL forms) and their pose is user definable and locus too can be chosen as 3D vector. The setup operates through interactive console.

3.2.5 a Test set-up:

In the test set up a complex section cylinder comprising 221 STL triangles is used to form a work space scenario by placing two instances of the same object (figure 3.22). The object instance B is rotated 90° around Z (vertical axis) from pose of object A. The point robot ‘R’ is moved from a to h. The point robot has a sense vector starting from the point and emanates in direction of move and has a length ‘L6’. We refer this as ‘p’. As explained in earlier section the vicinity computing process avoids finding intersection of p with the objects A and B directly and applies filtering chain (figure 3.21) as it would otherwise require test of intersection with 442 STL surfaces. First it checks the objects using ‘SESO’ test by test of presence of ‘R’ sufficiently near spheres embodying A and B. For this { r0 + L6 > δ} test is used. Where ‘r0’ is radius of sphere enclosing the object, and ‘δ’ is computed Euclidian distance from container sphere centre to the position of ‘R’.
3.2.5 b Experiments:

The setup executes probe interaction at each location of ‘\( \mathcal{R} \)’ by subjecting the procedure to the filter (figure 3.21) and the tests described above. It graphically shows the various parameters namely SESO, SESS, SDT, IDP (Intersection Determination Procedure), RT query, VIT query and total computation. The display cluster is aligned to show all these parameters as bar-graph on sequence of operation.

3.2.5 c: Computation Loads in Scope Control Approach:

Consider two objects A and B in workspace as shown in figure 3.22. Consider an object following a path from point ‘a’ to ‘h’.

The number of computation for profile path ‘a’-‘h’ along with their load is tabulated in Table 3.3 and 3.4 for un-optimized and optimized cases.

The optimized version in table 3.4 achieve significant reduction in computation for achieving same results as done by the processes in table 3.3. Maximum gains result in case ‘e’ for the test case where the test location is near multiple objects.

The major advantage comes from following -

1. In SESO and SESS computation optimization is achieved by comparing Euclidean distance square rather than computing distance which involves square root (23 FPC) as extra load.

2. In SDT computation optimization is done by calculating dot product between normal to \( S \) and vector \( PnPn \)” which is an intermediate step of calculating the angle between vectors and comparing the result rather than computation of exact angle. Only those faces which have \( \text{dotproduct} \leq 0 \) are considered. The actual angle magnitude calculation that costs (3M, 2A, 1SR) is avoided.
## Table 3.3: Types of tests and their loads at different position of ‘R’ with un-optimized methods

<table>
<thead>
<tr>
<th>Position</th>
<th>Cost by SESO</th>
<th>Cost by SESS</th>
<th>Cost by SDT</th>
<th>Cost by IDP</th>
<th>Cost by RT</th>
<th>Cost by VIT</th>
<th>Σ Load</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Cost</td>
<td>Units (35U)</td>
<td>Units (12U)</td>
<td>Units (12U)</td>
<td>Units (57U)</td>
<td>Units (14U)</td>
<td>Units (82U)</td>
</tr>
<tr>
<td>SESO</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SESS</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SDT</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IDP</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RT</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VIT</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*a Column header indicates type of tests performed and their respective loads*

Note: SESO = Smallest Encapsulating Sphere for Object, SESS = Smallest Encapsulating Sphere for Surface, SDT = Surface Direction Test, IDP = Intersection Determination Procedure, RT = Range Test, VIT = Vector Intersection Test. U = Unit load
Table 3.4: Types of tests and their loads at different position of ‘R’ with optimized methods

<table>
<thead>
<tr>
<th>Position</th>
<th>Cost by SESO</th>
<th>Cost by SESS</th>
<th>Cost by SDT</th>
<th>Cost by IDP</th>
<th>Cost by RT</th>
<th>Cost by VIT</th>
<th>Σ Load</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SESO Query</td>
<td>Units (12U)</td>
<td>SESS Query</td>
<td>Units (12U)</td>
<td>SDT Query</td>
<td>Units (12U)</td>
<td>IDP Query</td>
</tr>
<tr>
<td>a</td>
<td>2</td>
<td>24</td>
<td>220</td>
<td>2640</td>
<td>16</td>
<td>192</td>
<td>8</td>
</tr>
<tr>
<td>b</td>
<td>2</td>
<td>24</td>
<td>220</td>
<td>2640</td>
<td>14</td>
<td>168</td>
<td>4</td>
</tr>
<tr>
<td>c</td>
<td>2</td>
<td>24</td>
<td>220</td>
<td>2640</td>
<td>8</td>
<td>96</td>
<td>0</td>
</tr>
<tr>
<td>d</td>
<td>2</td>
<td>24</td>
<td>220</td>
<td>2640</td>
<td>2</td>
<td>24</td>
<td>0</td>
</tr>
<tr>
<td>e</td>
<td>2</td>
<td>24</td>
<td>440</td>
<td>5280</td>
<td>17</td>
<td>204</td>
<td>6</td>
</tr>
<tr>
<td>f</td>
<td>2</td>
<td>24</td>
<td>220</td>
<td>2640</td>
<td>11</td>
<td>132</td>
<td>11</td>
</tr>
<tr>
<td>g</td>
<td>2</td>
<td>24</td>
<td>220</td>
<td>2640</td>
<td>17</td>
<td>204</td>
<td>16</td>
</tr>
</tbody>
</table>

* Column header indicates type of tests performed and their respective loads

Note:  
SESO = Smallest Encapsulating Sphere for Object,  
SESS = Smallest Encapsulating Sphere for Surface,  
SDT = Surface Direction Test,  
IDP = Intersection Determination Procedure,  
RT = Range Test,  
VIT = Vector Intersection Test.  
U = Unit load
Figure 3.22: Experimental results showing computation loads for vector intersection with STL CAD modelled objects in workspace.
Notable details are as following:

i. First row in the figure 3.22 shows the total no of objects in the workspace undergoing SESO test i.e. to find out the number of bounding entities which are in range of point ‘p’.

ii. The SESO test gives the number of objects which will subsequently undergo SESS test. The SESS test will take all the surfaces of the objects in SESO’s within range into account and finds out which STL surface’s are within point ‘p’ range. The number of triangles undergoing SESS test is shown in second row.

iii. After SESS test only few triangles remain which need to undergo SDT (Surface Direction Test) and only those surfaces facing the vector are considered for further computation. Other faces are discarded. The third row shows the number of triangles undergoing SDT.

iv. The fourth row shows the number of intersection point calculated on the surface planes using IDP (Intersection Determination Procedure).

v. The fifth row shows the surfaces undergoing Range Test (RT) where the surfaces which are not in the ‘$L_6$’ range of the point ‘p’ are discarded.

vi. The surfaces which pass the Range Test (RT) are then checked for Vector Intersection Test (VIT) where the intersection point’s inclusion within STL triangle is tested. Based on VIT test the hit is qualified. The surfaces undergoing VIT test are shown in sixth row.

vii. The Last row shows the sum of all the tests performed.
3.2.6 Results and Discussions

The results of above experiments are analysed to form an insight on their usefulness towards MPC model application. A detailed display in figure 3.23 uses computation quantum (along axis Z) for each filter procedure along filter axis (X) at each test point appearing along location axis (Y). ‘Filter’ axis corresponds to rows of the figure 3.22. Other axis corresponds to positions a to h and magnitude shows quantum of computation for different types of tests at the location of ‘.getResource’ while moving on test path.

3.2.6a Salient Observations

As we see the profile the following are noted:

1. For profile ‘a’-‘d’ the second row shows no change as we can see in the figure 3.22 that up to position ‘d’ the point ‘p’ is in the vicinity of only one object i.e. ‘A’.

2. In position ‘e’, there is a sudden jump in the number of candidate STLs for SESS test as we can see now the point ‘p’ is in the bounding sphere of two objects A and B. Hence the total computations rise.

3. The SDT profile in third row abruptly comes down to zero after point ‘d’ because the surfaces which were initially facing the observer ray have gone past the point ‘p’. So the total computations also reduce.

4. Sixth row, which shows the VIT computation, indicate high load as VIT is computation expensive. Out of all the points from ‘a’ to ‘h’ VIT was invoked only at ‘a’ and ‘h’. Note that otherwise all STL triangles in all the objects of workspace would have required VIT test or at least IDP and RT.
Chapter 3: Kinesthetic Perception of Closeness to Objects & Innovative Dynamics Sensitive Percept

Figure 3.23: Plots of computation minimization filter in tests. Legend gives test type plotted along ‘x’ axis as ‘filter’. Test position sequence is shown along ‘y’ corresponds to the test locations a to h.

The test position of ‘R’ is along Y axis of the plot. At each location ‘a’ to ‘h’, the query for vector \( P_nP_n' \) intersect with workspace object proceeds along X axis with dotted and solid pairs of same color show the number of tests and their computation load respectively. The gaps on axis ‘x’ is used only for showing sliced view.

The number of tests performed at location (‘a’ to ‘h’) reduces along X axis. Note that at ‘a’ and at ‘h’ (near grid = 0 and grid = 13) a few STL triangles eventually reach STL hit stage. On other points, STLS are filtered out. Note that ‘IDP’ is not a test but a mandatory ray-plane intersects computation that has to be done once a STL triangle qualifies SESS and SDT. Therefore it is important to have homogenous sided STLS.
3.2.6 b On Performance of Search Scope Control approaches

SEBO, SESO tests are equal to number of objects but cost of test is small. But objects with large number of STL surface segments such as those with NURBS surfaces bring in large SESS candidates needing elimination by SDT or RIT (Ray Intersect Test). Both are computationally costly. The basic criteria is that free space inside SESO or SEBO should be minimised (figure 3.19b, 3.19c), mutual overlap should be minimised.

Note that the depicted phenomena is for single vector ‘𝑝’, but a large number of vectors emanating from ‘ﺮ’ at entire front hemisphere has to be considered and generally are more than 32 in number. The computations are as many times of that plotted here. Real time performance failure happens as computation load grows to extremely high figures near object when time overshoot can result in hit.

![Figure 3.24: STL model of complex cylinder with same accuracy, a; has heterogeneous side triangles, b; better STLs than (a), c; homogeneous STL in large numbers.](image)

The effectiveness of SESS lies in homogeneity of STL sides’ size and their magnitude. In example figure 3.24 model error is same as the top surface polygon is converted to same polygon in all the cases and the object being cylinder, does not have any effect of STL triangle heights along cylinder axis. The (b) version needs
double the numbers of triangles and corresponding data size. The representation in figure 3.24c has higher number of STLs than first. The SESS are smaller and so fewer of them qualify for further tests when tested at locations of ‘p’ farther than the small radius of their SESS. They qualify for the costlier intersection test much later during point robot approach towards the wall. Therefore reducing total numbers of STL surfaces does not definitely yield overall advantage as number of STL test increase while intersection test reduce.

As a consequence, for case figure 3.24c, 8 times the STLs shall be subjected to SESS test as compared to that for case ‘a’, but only a fraction will be subjected to ray test. However a good fraction of the STL qualifying in case ‘a’ will be eliminated by ‘SDT’. On the other hand those STL qualifying from case ‘c’ may not be filtered much unless the object is very thin and SESS of opposite side also qualify.

### 3.3 Observations on Model Implementation

The optimisations described in the earlier section need pre-processing of STL model to attach data for SESS and reconstruct the data in different form. Further the effectiveness is not guaranteed to yield real time performance owing to search dependent approach which are inherently non deterministic. The filters do not help when the probe is in concave area near object and large STL’s qualify for further test. The main function of filters is to free up a processor from one vector computation to take up another in multiple vector tests running on parallel threads on a computer.
Closure:

The MPC model conforms to haptic design criteria well, by using a physical model whose haptics model is easy to understand and adequate information is available. The model succeeds in rendering the proximity and the robot-dynamics effects as kinaesthetically perceivable effect on operator hand. The model is apparently very compact in terms of computation load but its attempted implementation routes are not viable owing to uncertainty in computation load. Various techniques from the domain of obstacle avoidance are promising. The task ahead is to identify suitable modelling technique that can circumvent the presently encountered problem. Of particular interest is to evolve techniques that reduce search or eliminate search. Chapter 4 is devoted on overcoming this limitation.