CHAPTER 4

INVERSE KINEMATICS OF REDUNDANT ROBOTS IN PRESENCE OF OBSTACLES

4.1 INTRODUCTION

The inverse kinematics problem of a redundant robot has infinite number of solutions since there are infinite number of joint coordinate vectors that correspond to the same end-effector position. A unique solution may be obtained in such cases if a performance criterion, like total joint displacement minimization, is incorporated into the solution scheme. An evolutionary approach based on an elitist real-coded genetic algorithm is used to handle the redundancy resolution. An obstacle-avoidance algorithm is also incorporated into the proposed solution scheme. The obstacle avoidance algorithm is based on the spherization of the robot and the obstacles in the robot workspace. Industrial non-redundant robotic manipulators may be considered redundant for certain tasks wherein their motion could be optimized. The algorithm is tested for the inverse kinematics solution of the Mitsubishi Movemaster RV-M1 robot which is an industrial spatial articulate robotic manipulator. Simulation experiments are carried out to illustrate the efficacy of the proposed approach.

4.2 PROBLEM FORMULATION

For a robot having m joint variables, the position vector \( ^oP_h \) of the robot hand relative to the base coordinate frame can be extracted from the last column of
The matrix \( ^0T_h \) relates the hand coordinate frame of the robot to the base coordinate frame and can be obtained through the use of DH matrices explained in section 3.2.1. For an industrial robot having intersecting last three axes, the number of degrees of freedom can be reduced to \( 'm-3' \) by decoupling the solution of the wrist positioning and the hand orienting joint variables. The position vector of the robot wrist \( ^0P_w \) can be thus be obtained as

\[
\{^0P_w\} = \{^0P_h\} - \{^0R_{m-2}\} \{^m-2P_h\} \tag{4.1}
\]

where \( ^0R_{m-2} \) relates the axes of coordinate frame \( 'm-2' \) to the base coordinate frame and \( \{^0P_h\} \) is a function of the vector of joint variables \( \{q_1, q_2, \ldots, q_{(m-3)}\}^T \). The \( \{^0P_w\} \) vector can alternately be obtained through the last column of the matrix \( ^0T_{m-2} \). In order to achieve a desired position \( \{^0P_{w,des}\} \) of the robot wrist, eqn. (4.1) can be used to form a constraint vector equation of the form

\[
\{^0P_w\} - \{^0P_{w,des}\} = \{0\} \tag{4.2}
\]

For a robotic manipulator working in an obstacle free environment, the performance criterion for redundancy resolution can be taken as the minimization of the total joint displacement

\[
\Delta q = \|\{q\} - \{q_{cur}\}\| \tag{4.3}
\]

where \( \| \| \) denotes the Euclidean distance, \( \{q_{cur}\} = \{q_{1,cur}, q_{2,cur}, \ldots, q_{(m-3),cur}\}^T \) represents the vector of joint variables at the current configuration of the
robotic manipulator and \( \{q\} = \{q_1, q_2, ... q_{m-3}\}^T \) represents the unknown vector of joint variables at the desired goal position of the robotic manipulator.

Further, the limits on the joint variable values can be expressed as

\[
q_k^L \leq q_k \leq q_k^U \quad k = 1, 2, ..., m - 3
\]  

(4.4)

where \( q_k^L \) and \( q_k^U \) represent the lower and upper limits of the joint variables.

The inverse kinematics problem can be stated as the following optimization problem

\[
\begin{align*}
\text{Minimize} \quad & \Delta q \\
\text{subject to} \quad & \{\hat{O}_w^1 - \{\hat{O}_{w,\text{des}}\}\} = 0 \\
& q_k^L \leq q_k \leq q_k^U \quad k = 1, 2, ..., m - 3
\end{align*}
\]

(4.5)

The optimization problem given in eqn. (4.5) is equivalent to the optimization problem given below (Kalra et al., 2003b)

\[
\begin{align*}
\text{Minimize} \quad & w_1 \Delta q + w_2 \left\| \{\hat{O}_w^1 - \{\hat{O}_{w,\text{des}}\}\} \right\| \\
\text{subject to} \quad & q_k^L \leq q_k \leq q_k^U \quad k = 1, 2, ..., m - 3
\end{align*}
\]

(4.6)

The weights \( w_1 \) and \( w_2 \) are dynamically updated as

\[
\begin{align*}
w_2 &= a \exp \left(-\left\| \{\hat{O}_w^1 - \{\hat{O}_{w,\text{des}}\}\}\right\| \right) + b \\
w_1 &= 1 - w_2
\end{align*}
\]

(4.7)

where \( a \) and \( b \) are constants. The current methodology corresponds to a penalty function approach where the ratio \( w_2/w_1 \) represents the penalty parameter. Moreover, the current choice of \( w_2 \) and \( w_1 \) ensures that the search is directed by both the performance criterion (eqn. (4.3)) and the constraint equations (eqn. (4.2)) when the positioning
error is large and by the constraint equations alone towards the end of the search.

To ensure a collision free motion of the robotic manipulator in the presence of obstacles, the optimization problem given in eqn. (4.6) can be modified as

\[ \text{Minimize} \quad F = w_1 \Delta q + w_2 \left\| \{oP_v\} - \{oP_{v,\text{des}}\} \right\| \]

for collision - free configuration

\[ = \text{Large positive value} \]

for configuration involving collision

subject to \( q_k^l \leq q_k \leq q_k^u \quad k = 1, 2, ..., m - 3 \)

The optimization problem stated in eqn. (4.8) can also be re-written as the following maximization optimization problem

\[ \text{Maximize} \quad F' = \frac{1}{1 + w_1 \Delta q + w_2 \left\| \{oP_v\} - \{oP_{v,\text{des}}\} \right\|} \]

for collision - free configuration

\[ = 0 \]

for configuration involving collision

subject to \( q_k^l \leq q_k \leq q_k^u \quad k = 1, 2, ..., m - 3 \)

4.3 COLLISION DETECTION SCHEME

A representation which is well utilized for collision detection is the sphere. Its suitability can be attributed to the trivial and fast distance computation which exists between two spheres, and from the feature that a sphere’s geometry is rotationally invariant in three degrees of freedom.

In this work, the geometry of the manipulator links as well as the geometry of the obstacle is divided into cuboids which are enveloped by spheres. The cuboid shown in
fig. 4.1 can be discretized into elements each of which is circumscribed by a sphere. The spheres are indexed through three indexes \( \lambda_1, \lambda_2, \) and \( \lambda_3 \) in index directions 1, 2 and 3 respectively as \((\lambda_1 \lambda_2 \lambda_3)\). For the cuboid shown in fig. 4.1, \( \lambda_1 = 1, 2, \ldots, 9, 10 \); \( \lambda_2 = 1, 2, 3, 4 \) and \( \lambda_3 = 1, 2, 3, 4 \). Fig. 4.1 shows the cuboid with spheres described through indexes \((4 \ 1 \ 2),(6 \ 3 \ 1)\) and \((2 \ 1 \ 4)\). The radius of the spheres used to envelope the elements can be evaluated using the relation

\[
R = \sqrt{\frac{l_1}{\lambda_{1,\text{max}}}^2 + \frac{l_2}{\lambda_{2,\text{max}}}^2 + \frac{l_3}{\lambda_{3,\text{max}}}^2},
\]

where \( l_1, l_2, \) and \( l_3 \) are the linear dimensions of the cuboid in its local \( x, y, \) and \( z \) coordinate directions respectively and \( \lambda_{1,\text{max}}, \lambda_{2,\text{max}}, \) and \( \lambda_{3,\text{max}} \) represent the maximum value of the indexes \( \lambda_1, \lambda_2, \) and \( \lambda_3 \). The product \( \lambda_{1,\text{max}}\lambda_{2,\text{max}}\lambda_{3,\text{max}} \) gives the number of spheres used to envelope the entire cuboid.

Fig. 4.2 shows the cuboid, shown in fig. 4.1, enveloped by spheres.

The number of spheres enveloping the entire cuboid is determined on the basis of the required modelling tolerance. The modelling tolerance can be defined as the maximum overshoot of the spherical surface beyond the external surface of the surface elements along the lateral dimensions of the link. This can be mathematically evaluated as

\[
\epsilon = \max \left( R - \frac{l_2}{2\lambda_{2,\text{max}}}, R - \frac{l_3}{2\lambda_{3,\text{max}}} \right)
\]

where the dimension \( l_1 \) is assumed to lie along the axial dimension of the link. For a robot link \( k \) divided into...
Fig. 4.1: Cuboid discretized into elements with spheres (4 1 2), (6 3 1) and (2 1 4) circumscribing corresponding elements
Fig. 4.2: Spherization of cuboid shown in fig. 4.1
multiple cuboids, the maximum value of $\varepsilon$ for all cuboids gives the modelling tolerance $\varepsilon_k$ of the link. For the complete robot the modelling tolerance $\chi$ is determined as

$$\chi = \operatorname{Max}(\varepsilon_1, \varepsilon_2, \varepsilon_3, \ldots, \varepsilon_{n_{l-2}}, \varepsilon_{n_{l-1}}, \varepsilon_{n_l})$$  \hspace{1cm} (4.12)

In eqn. (4.12), $n_l$ represents the number of links of the robot which is spherized.

The position vector of the centre of the sphere $(\lambda_1, \lambda_2, \lambda_3)$, used to envelope a cuboid element, is calculated through the following relation

$$\vec{p}_{\lambda_1,\lambda_2,\lambda_3} = \begin{bmatrix} x_1 \ y_1 \ z_1 \end{bmatrix}^T = \begin{bmatrix} x_1 + (\lambda_1 - 1) \left( \frac{x_{1,\max} - x_1}{\lambda_{1,\max} - 1} \right) \\ y_1 + (\lambda_2 - 1) \left( \frac{y_{1,\max} - y_1}{\lambda_{2,\max} - 1} \right) \\ z_1 + (\lambda_3 - 1) \left( \frac{z_{1,\max} - z_1}{\lambda_{3,\max} - 1} \right) \end{bmatrix}$$

$$\lambda_1 = 1, 2, \ldots, \lambda_{1,\max}$$
$$\lambda_2 = 1, 2, \ldots, \lambda_{2,\max}$$
$$\lambda_3 = 1, 2, \ldots, \lambda_{3,\max}$$  \hspace{1cm} (4.13)

In eqn. (4.13), $x_1$, $y_1$ and $z_1$ are the $x$-coordinate, $y$-coordinate and $z$-coordinate of the centre of the sphere. The coordinates are evaluated for all combinations of $\lambda_1$, $\lambda_2$ and $\lambda_3$ to obtain the position vectors of all the spheres which envelope the entire cuboid. These co-ordinates are in the co-ordinate frame of the link of which the cuboid is a part.

The position vector of the centre of the spheres in link $k$, evaluated through eqn. (4.13), is transformed to
obtain the position vector in base coordinate frame, $\vec{p}_{\alpha \beta \gamma}$, as

$$\{\vec{p}_{\alpha \beta \gamma}, 1\}^T = \begin{bmatrix} X_i & Y_i & Z_i & 1 \end{bmatrix}^T$$

$$= \begin{bmatrix} 0_T \end{bmatrix} \{\vec{p}_{\alpha \beta \gamma}, 1\}^T .$$

Fig. 4.3 shows a robot link in the vicinity of an obstacle. The distance between the $i^{th}$ sphere on the robot link (with centre coordinates $(X_i, Y_i, Z_i)$ and radius $R_i$) and the $j^{th}$ sphere on the obstacle (with centre coordinates $(X_j, Y_j, Z_j)$ and radius $R_j$) can be computed as

$$D_{ij} = \left[ (X_i - X_j)^2 + (Y_i - Y_j)^2 + (Z_i - Z_j)^2 \right]^{0.5} .$$

In eqn. (4.15), the coordinates of the centre of the spheres on the robot and the obstacle have to be in the base coordinate frame. Further, the index $i$ would be varied so as to include all spheres that envelope the complete robot. Index $j$ would be varied so as to include all spheres that envelope the obstacle. The collision detection is accomplished through the following:-

1. $D_{ij} > R_i + R_j$ for all combinations of $i,j$ indicates a collision free configuration.
2. $D_{ij} \leq R_i + R_j$ for any combination of $i,j$ indicates a configuration in collision.

### 4.4 DETAILS OF EVOLUTIONARY APPROACH

The evolutionary approach based on a real-coded GA, described in chapter 2, is used as the optimization strategy for the solution of inverse kinematics problem stated in eqn. (4.9). The general basic mechanics of this approach has been described in chapter 2. In this section, the problem-specific parameters of the approach are given.
Spheres enveloping manipulator links

Sphere 'j' (centre $(X_j,Y_j,Z_j)$, radius $R_j$)

Sphere 'i' (centre $(X_i,Y_i,Z_i)$, radius $R_i$)

$D_{ij}$ Spheres enveloping obstacle

Sphere 'j' (centre $(X_j,Y_j,Z_j)$, radius $R_j$)

Fig. 4.3: Collision detection scheme
INDIVIDUAL REPRESENTATION

For a robotic manipulator, the individual in a population would be represented by the real number vector of the joint variables as $\{q_1, q_2, \ldots, q_{(m-1)}\}^T$.

INITIALIZATION

An initial population of robot configurations is generated by random sampling from the variable search space. The goal is to select individuals spread over the entire search space, respecting the limits of joint motion given in eqn. (4.4).

EVALUATION AND SELECTION

The fitness of an individual is a measure of how good a solution it provides to the optimization problem stated in eqn. (4.9). The initialization procedure, along with appropriate recombination and mutation strategies described in sections 2.3.3 and 2.3.4, ensures that all individuals in the population satisfy the limits on joint variable values given in eqn. (4.4). The optimization problem given in eqn. (4.9) thus reduces to an unconstrained optimization problem. The fitness function to be maximized is therefore defined as

$$\text{Fitness} = \frac{1}{1 + w_1 \Delta q + w_2 \left\| \{q_r\} - \{q_{r,\text{des}}\} \right\|}$$

for collision-free configuration \hspace{1cm} (4.16)

$$= 0$$

for configuration involving collision

In the current work, the binary tournament selection operator is used with fitness being evaluated according to its definition given in eqn. (4.16). The detection of a collision for a particular robot configuration is carried out using the methodology described in section 4.3.
RECOMBINATION AND MUTATION

The recombination operator should create children solutions within the range of the joint variables given in eqn. (4.4). The spread factor of the SBX operator, described in section 2.3.3, is therefore evaluated using eqns. (2.5), (2.6) and (2.7).

The mutation operator should also create a solution within the range of the joint variables given in eqn. (4.4). The parameter $\delta$, used in the mutation operator described in section 2.3.4, is therefore evaluated using eqns. (2.10) and (2.11).

ELITISM

In order to preserve and use previously found best solutions in subsequent generations, an elite-preserving operator is often recommended. In an elitist GA, the statistics of the population best solutions cannot degrade with generations. Goldberg (1989) has pointed out that elitism improves the performance of a GA for optimization problems having unimodal surfaces. However, with multimodal functions, it degrades the performance of the GA. This suggests that elitism improves local search at the cost of global perspective. The inverse kinematics problem solution discussed in Chapter 3 involved the evaluation of multiple configurations of non-redundant robotic manipulators through the use of a multimodal objective function. Hence elitism was not introduced there. In this chapter, the redundancy resolution of a redundant robotic manipulator is carried out wherein a single best configuration is desired. Hence an elite preserving operator is used in the solution scheme.
One of the approaches to implement elitism is to create an offspring population by using usual genetic operations. The best N solutions are then chosen from a combined population (of size 2N) of parents and offspring (Deb, 2001). Here N is the population size. This implementation of elitism is utilized in this work.

4.5 SIMULATION EXPERIMENTS

Simulation experiments were performed on the Mitsubishi Movemaster RV-M1 robot shown in fig. 4.4. This figure shows the coordinate frames attached to the robot links. For this robot, the pitch-angle of the hand can be defined as the angle made by the z₅ axis with the X₀Y₀ plane. The DH parameters of the robot are given in Table 4.1 and the limits on the joint variables q₁ through q₅ are:

\[-60° \leq q₁ \leq 240°\]
\[-30° \leq q₂ \leq 100°\]
\[-110° \leq q₃ \leq 0°\]
\[-180° \leq q₄ \leq 0°\]
\[-180° \leq q₅ \leq 180°\]

Using eqn. (3.1), the DH transformation matrices for the different link coordinate frames can be written as

\[
^{0}A₁ = \begin{bmatrix}
C₁ & 0 & S₁ & 0 \\
S₁ & 0 & -C₁ & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\] (4.17)
Fig. 4.4: Mitsubishi Movemaster RV-M1 robot
Table 4.1: DH parameters of Mitsubishi Movemaster RV-M1 robot

<table>
<thead>
<tr>
<th>Link number</th>
<th>$\theta_i$</th>
<th>$d_i$</th>
<th>$a_i$</th>
<th>$a_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$q_1$</td>
<td>0</td>
<td>0</td>
<td>90°</td>
</tr>
<tr>
<td>2</td>
<td>$q_2$</td>
<td>0</td>
<td>250mm</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>$q_3$</td>
<td>0</td>
<td>160mm</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>$q_4$</td>
<td>0</td>
<td>0</td>
<td>-90°</td>
</tr>
<tr>
<td>5</td>
<td>$q_5$</td>
<td>160mm</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Using eqn. (3.2) the homogeneous transformation matrix from the base coordinate frame to the coordinate frame attached to the hand, which is connected to the fifth link, is given by

\[
^{0}{A}_{h} = \begin{bmatrix}
 C_1 & -S_1 & 0 & a_1C_1 \\
 S_1 & C_1 & 0 & a_1S_1 \\
 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 1
\end{bmatrix}
\]

(4.18)

\[
^{1}{A}_{b} = \begin{bmatrix}
 C_2 & -S_2 & 0 & a_2C_2 \\
 S_2 & C_2 & 0 & a_2S_2 \\
 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 1
\end{bmatrix}
\]

(4.19)

\[
^{2}{A}_{b} = \begin{bmatrix}
 C_3 & -S_3 & 0 & a_3C_3 \\
 S_3 & C_3 & 0 & a_3S_3 \\
 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 1
\end{bmatrix}
\]

(4.20)

\[
^{3}{A}_{b} = \begin{bmatrix}
 C_4 & 0 & -S_4 & 0 \\
 S_4 & 0 & C_4 & 0 \\
 0 & -1 & 0 & 0 \\
 0 & 0 & 0 & 1
\end{bmatrix}
\]

(4.21)

The elements of the rotation matrix associated with \( i \) are not shown in eqn. (4.22) and are marked with *. In eqns. (4.17) through (4.22), \( C_i=\cos(q_i) \), \( C_{ij} = \cos(q_i + q_j) \), \( C_{ijk} = \cos(q_i + q_j + q_k) \), \( S_i=\sin(q_i) \), \( S_{ij} = \sin(q_i + q_j) \) and \( S_{ijk} = \sin(q_i + q_j + q_k) \).

The Mitsubishi Movemaster RV-M1 robot may have certain applications in which it can be considered.
redundant. Such applications include situations where the robot hand is required to be positioned at a point with no specific approach vector. In such situations, the first four joints of the robot can be used for positioning the robot hand. Replacing $\{^0P_r\}$ by $\{^0P_h\}$ in eqn. (4.9) and using eqn. (4.22), the inverse kinematics problem for the positioning joint variables is

\[
\begin{align*}
\text{Maximize } & F = \frac{1}{1 + F} \\
\text{for collision-free configuration } & = 0 \\
\text{for configuration involving collision } & = F
\end{align*}
\]

subject to $q_i^l \leq q_k \leq q_i^u$ \hspace{1cm} k = 1, 2, 3, 4

where $F = w_1 \Delta q + w_2 \begin{bmatrix} C_1 (a_1 C_2 + a_3 C_{23} - d_5 S_{234}) \\ S_1 (a_1 C_2 + a_3 C_{23} - d_5 S_{234}) \\ a_2 S_2 + a_3 S_{23} + d_5 C_{234} \end{bmatrix} - \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$

Here X, Y and Z are the coordinates of the desired position of the robot hand, $\Delta q$ is calculated using eqn.(4.3) and $w_1$ and $w_2$ are evaluated using eqn. (4.7). The values of a and b in eqn. (4.7) are taken as 0.5 each.

For the simulation experiments being carried out link one of the robot would not suffer any collision. Hence links two, three, four and five of the robot and the robot hand are spherized. If necessary, link one of the robot can also be spherized. However, the solution scheme remains unchanged. Fig. 4.5 shows the variation of the modelling tolerance for the spherization of the Mitsubishi Movemaster RV-M1 robot with the number of spheres used to envelope the robot links. The modelling tolerance can be fixed on the basis of a sufficient clearance between the robot and the obstacle. Fig. 4.6 (a) shows the robot without spherization whereas fig. 4.6(b) shows the robot
Fig. 4.5: Variation of modelling tolerance with number of spheres used to spherize links 2, 3, 4 and 5 of robot and robot hand
Fig. 4.6: View of Mitsubishi Movemaster robot (a) without spherization (b) showing spherization of links 2, 3, 4, 5 and hand
with links two, three, four and five and the hand enveloped by spheres. The modelling tolerance corresponding to this spherization is 16.23 mm.

### 4.5.1 EXPERIMENT 1 - ROBOT WITH OBSTACLE IN WORKSPACE

Fig. 4.7 shows the robot with an obstacle in its workspace. The desired trajectory of the robot hand, showing the initial point, seven intermediate knot points and the final point, is also shown in the figure. The approach vector of the robot hand is fixed at the initial and final point of the trajectory through a specified pitch angle of 0°. The approach vectors at the intermediate knots are taken to be the same as that at the initial point in case there is no collision. In case there is no collision and the approach vector is specified through the pitch angle, the joint variables are determined through the geometric solution for the inverse kinematics of the Mitsubishi Movemaster RV-M1 robot given in Annexure D. In the event of a collision the joint variables are determined through the problem formulation given in eqn.(4.23). The collision detection scheme given in section 4.3 detects robot obstacle collision at knot points 2 through 6 and a collision-free configuration at knot points 1 and 7. The robot configurations at these knot points, along with the obstacle, can be seen in figs. 4.8(a) through 4.8(g). In these figures the robot along with the obstacle are enveloped by spheres.

The following control parameters were used by the real-coded GA (Deb, 2000) to solve for the positioning joint variables of the robot so as to accomplish obstacle avoidance along with minimum total joint displacement:
Fig. 4.7: Mitsubishi Movemaster RV-M1 robot with obstacle in vicinity of desired trajectory
Fig. 4.8: Spherized robot and obstacle, with robot hand at (a) knot point 1 and (b) knot point 2 of trajectory
Fig. 4.8: Spherized robot and obstacle, with robot hand at (c) knot point 3 and (d) knot point 4 of trajectory
Fig. 4.8: Spherized robot and obstacle, with robot hand at (e) knot point 5 and (f) knot point 6 of trajectory
Fig. 4.8(g): Spherized robot and obstacle, with robot hand at knot point 7 of trajectory
population size=40, crossover probability = 0.9, SBX
distribution index = 5, \( t_{\text{max}} \) (for purpose of calculation of
mutation probability) equal to 800. The convergence
criterion was taken as a positioning error of 0.5mm
subject to a maximum of 100 generations. Table 4.2 gives
the values of the minimum positioning errors and the
minimum total joint displacement obtained using both the
elitist real-coded genetic algorithm and the non-elitist
real-coded genetic algorithm. From this table, it is clear
that elitism improves the performance of the real-coded
genetic algorithm in providing solutions in lesser number
of generations. The non-elitist real-coded genetic
algorithm fails to provide a solution even after 100
generations at knot points 2, 3, 4, 5 and 6. Figs. 4.9,
4.10, 4.11, 4.12 and 4.13 show the variation of average
fitness and best fitness with generations for the solution
at knot points 2, 3, 4, 5 and 6 respectively obtained
using the elitist real-coded genetic algorithm. The
variation of minimum positioning errors with generations,
for the corresponding solutions at knot points 2, 3, 4, 5
and 6, is given in figs. 4.14, 4.15, 4.16, 4.17 and 4.18
respectively. The corresponding figures showing the
variation of minimum total joint displacement with
generations are given in figs. 4.19, 4.20, 4.21, 4.22 and
4.23. Table 4.3 shows the values of the first four joint
variables of the robot at the initial point, intermediate
knot points and the final point of the trajectory.

Figs. 4.24 through 4.28 show the robot and the
obstacle at the intermediate knot points 2 through 6 for
which a collision was detected. These figures show that
the evolutionary approach has been able to achieve
collision avoidance at these knot points.
Table 4.2: Comparison of performance of evolutionary approach (based on real-coded GA) with and without elitism for simulation experiment 1 of Mitsubishi Movemaster RV-M1 robot

<table>
<thead>
<tr>
<th>Knot point</th>
<th>EA without elitism</th>
<th>EA with elitism</th>
<th>Number of generations for convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum positioning error (in mm)</td>
<td>Minimum total joint displacement (in radians)</td>
<td>Whether convergence achieved within 100 generations?</td>
</tr>
<tr>
<td>2</td>
<td>2.24</td>
<td>0.43</td>
<td>No</td>
</tr>
<tr>
<td>3</td>
<td>1.36</td>
<td>2.03</td>
<td>No</td>
</tr>
<tr>
<td>4</td>
<td>1.40</td>
<td>0.35</td>
<td>No</td>
</tr>
<tr>
<td>5</td>
<td>1.64</td>
<td>0.42</td>
<td>No</td>
</tr>
<tr>
<td>6</td>
<td>0.77</td>
<td>0.60</td>
<td>No</td>
</tr>
</tbody>
</table>
Fig. 4.9: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 2 (simulation experiment 1)
Fig. 4.10: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 3 (simulation experiment 1)
Fig. 4.11: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 4 (simulation experiment 1)
Fig. 4.12: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 5 (simulation experiment 1)
Fig. 4.13: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 6 (simulation experiment 1)
Fig. 4.14: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 2 (simulation experiment 1)
Fig. 4.15: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 3 (simulation experiment 1)
Fig. 4.16: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 4 (simulation experiment 1)
Fig. 4.17: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 5 (simulation experiment 1)
Fig. 4.18: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 6 (simulation experiment 1)
Fig. 4.19: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 2 (simulation experiment 1)
Fig. 4.20: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 3 (simulation experiment 1)
Fig. 4.21: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 4 (simulation experiment 1)
Fig. 4.22: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 5 (simulation experiment 1)
Fig. 4.23: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 6 (simulation experiment 1)
Table 4.3: Values of joint variables at different points of trajectory for simulation experiment 1 of Mitsubishi Movemaster RV-M1 robot

<table>
<thead>
<tr>
<th>Point of trajectory</th>
<th>Coordinates of point</th>
<th>q₁ (in rad)</th>
<th>q₂ (in rad)</th>
<th>q₃ (in rad)</th>
<th>q₄ (in rad)</th>
<th>Joint variables obtained using</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>X-coordinate (in mm)</td>
<td>Y-coordinate (in mm)</td>
<td>Z-coordinate (in mm)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Initial</td>
<td>410.00</td>
<td>0.00</td>
<td>-225.00</td>
<td>0</td>
<td>-0.2640</td>
<td>-1.2526</td>
</tr>
<tr>
<td>Knot 1</td>
<td>402.12</td>
<td>79.99</td>
<td>-213.75</td>
<td>0.1963</td>
<td>-0.2171</td>
<td>-1.3170</td>
</tr>
<tr>
<td>Knot 2</td>
<td>378.79</td>
<td>156.90</td>
<td>-202.50</td>
<td>0.3926</td>
<td>0.0452</td>
<td>-1.5527</td>
</tr>
<tr>
<td>Knot 3</td>
<td>340.90</td>
<td>227.78</td>
<td>-191.25</td>
<td>0.5890</td>
<td>0.1170</td>
<td>-1.6037</td>
</tr>
<tr>
<td>Knot 4</td>
<td>289.91</td>
<td>289.91</td>
<td>-180.00</td>
<td>0.7861</td>
<td>0.2723</td>
<td>-1.6014</td>
</tr>
<tr>
<td>Knot 5</td>
<td>227.78</td>
<td>340.90</td>
<td>-168.75</td>
<td>0.9808</td>
<td>0.2197</td>
<td>-1.6687</td>
</tr>
<tr>
<td>Knot 6</td>
<td>156.90</td>
<td>378.79</td>
<td>-157.50</td>
<td>1.1787</td>
<td>0.2175</td>
<td>-1.6987</td>
</tr>
<tr>
<td>Knot 7</td>
<td>79.99</td>
<td>402.12</td>
<td>-146.25</td>
<td>1.3744</td>
<td>0.0550</td>
<td>-1.6235</td>
</tr>
<tr>
<td>Final</td>
<td>0.00</td>
<td>410.00</td>
<td>-135.00</td>
<td>1.5708</td>
<td>0.1001</td>
<td>-1.6631</td>
</tr>
</tbody>
</table>

* - Geometric approach
** - Evolutionary approach
Fig. 4.24: Robot configuration at knot point 2 (simulation experiment 1) obtained using evolutionary approach based on real-coded genetic algorithm.
Fig. 4.25: Robot configuration at knot point 3 (simulation experiment 1) obtained using evolutionary approach based on real-coded genetic algorithm
Fig. 4.26: Robot configuration at knot point 4 (simulation experiment 1) obtained using evolutionary approach based on real-coded genetic algorithm
Fig. 4.27: Robot configuration at knot point 5 (simulation experiment 1) obtained using evolutionary approach based on real-coded genetic algorithm.
Fig. 4.28: Robot configuration at knot point 6 (simulation experiment 1) obtained using evolutionary approach based on real-coded genetic algorithm
4.5.2 EXPERIMENT 2 - ROBOT WORKING THROUGH OPENING IN DOOR

Fig. 4.29 shows the robot working through an opening in a door. In this situation, the door acts as an obstacle in the vicinity of the robot. The desired trajectory of the robot hand, showing the initial point and the final point, is also shown in the figure. The trajectory comprises a straight line segment from (0,320mm,50mm) to (0,400mm,50mm) followed by a straight line segment from (0,400mm,50mm) to (0,400mm,130mm). Eight equi-distant knot points are defined along the first line segment of the trajectory. Point (0,400,50) is also a knot point. Seven equi-distant knot points are defined along the second line segment of the trajectory. The approach vector of the robot hand is fixed at the initial point of the trajectory through a specified pitch angle of 0°. The approach vectors for the intermediate knots and the final point are taken to be the same as that at the initial point in case there is no collision. In case there is no collision and the approach vector is specified through the pitch angle, the joint variables are determined through the geometric solution for the inverse kinematics of the Mitsubishi Movemaster RV-M1 robot given in Annexure D. In the event of a collision the joint variables are determined through the problem formulation given in eqn.(4.23). The collision detection scheme given in section 4.3 detects a collision-free configuration at knot points 1 through 13 and a robot obstacle collision at knot points 14 through 16 and the final point of the trajectory. The robot configurations at the knot points having robot obstacle collision can be seen in figs. 4.30(a) through 4.30(d). In
Fig. 4.29: Mitsubishi Movemaster RV-M1 robot working through opening in door (Robot trajectory is indicated by a solid line)
Fig. 4.30: Robot working at (a) knot point 15 of trajectory through opening in door (b) knot point 14
Fig. 4.30: Robot working through opening in door at (c) knot point 16 (d) final point of trajectory
The following control parameters were used by the real-coded GA (Deb, 2000) to solve for the positioning joint variables of the robot so as to accomplish obstacle avoidance along with minimum total joint displacement: population size = 40, crossover probability = 0.9, SBX distribution index = 5, \( t_{\text{max}} \) (for purpose of calculation of mutation probability) equal to 800. The convergence criterion was taken as a positioning error of 0.5 mm subject to a maximum of 100 generations. Table 4.4 gives the values of the minimum positioning errors and minimum total joint displacement obtained using both the elitist real-coded genetic algorithm and the non-elitist real-coded genetic algorithm. From this table, it is clear that elitism improves the performance of the real-coded genetic algorithm in providing solutions in lesser number of generations. The non-elitist real-coded genetic algorithm fails to provide a solution even after 100 generations. Figs. 4.31, 4.32, 4.33 and 4.34 show the variation of average fitness and best fitness with generations for the solution at knot points 14, 15, 16 and the final point respectively obtained using the elitist real-coded genetic algorithm. The variation of minimum positioning errors with generations, for the corresponding solution at knot points 14, 15, 16 and the final point of the trajectory, is given in figs. 4.35, 4.36, 4.37 and 4.38 respectively. The corresponding figures showing the variation of minimum total joint displacement with generations are given in figs. 4.39, 4.40, 4.41 and 4.42. Table 4.5 shows the values of the first four joint variables of the robot at the initial point, intermediate knot points and the final point of the trajectory.
Table 4.4: Comparison of performance of evolutionary approach (based on real-coded GA) with and without elitism for simulation experiment 2 of Mitsubishi Movemaster RV-M1 robot

<table>
<thead>
<tr>
<th>Knot point number/ final point</th>
<th>EA without elitism</th>
<th>EA with elitism</th>
<th>Number of generations for convergence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum positioning error (in mm)</td>
<td>Minimum total joint displacement (in radians)</td>
<td>Whether convergence achieved within 100 generations?</td>
</tr>
<tr>
<td>14</td>
<td>1.00</td>
<td>0.76</td>
<td>No</td>
</tr>
<tr>
<td>15</td>
<td>2.68</td>
<td>0.37</td>
<td>No</td>
</tr>
<tr>
<td>16</td>
<td>0.90</td>
<td>0.31</td>
<td>No</td>
</tr>
<tr>
<td>Final point</td>
<td>12.93</td>
<td>0.02</td>
<td>No</td>
</tr>
</tbody>
</table>
Fig. 4.31: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 14 (simulation experiment 2)
Fig. 4.32: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 15 (simulation experiment 2)
Fig. 4.33: Variation of best fitness and average fitness with generations during elitist real-coded GA run at knot point 16 (simulation experiment 2)
Fig. 4.34: Variation of best fitness and average fitness with generations during elitist real-coded GA run at final point of trajectory (simulation experiment 2)
Fig. 4.35: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 14 (simulation experiment 2)
Fig. 4.36: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 15 (simulation experiment 2)
Fig. 4.37: Variation of minimum positioning error with generations during elitist real-coded GA run at knot point 16 (simulation experiment 2)
Fig. 4.38: Variation of minimum positioning error with generations during elitist real-coded GA run at final point of trajectory (simulation experiment 2)
Fig. 4.39: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 14 (simulation experiment 2)
Fig. 4.40: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 15 (simulation experiment 2)
Fig. 4.41: Variation of minimum total joint displacement with generations during elitist real-coded GA run at knot point 16 (simulation experiment 2)
Fig. 4.42: Variation of minimum total joint displacement with generations during elitist real-coded GA run at final point of trajectory (simulation experiment 2)
Table 4.5: Values of joint variables at different points of trajectory for simulation experiment 2 of Mitsubishi Movemaster RV-M1 robot

<table>
<thead>
<tr>
<th>Point of trajectory</th>
<th>Coordinates of point</th>
<th>$q_1$ (in rad)</th>
<th>$q_2$ (in rad)</th>
<th>$q_3$ (in rad)</th>
<th>$q_4$ (in rad)</th>
<th>Joint variables obtained using</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial</td>
<td>0 320.00 50</td>
<td>1.5708</td>
<td>0.9862</td>
<td>-2.4189</td>
<td>-0.1381</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 1</td>
<td>0 328.89 50</td>
<td>1.5708</td>
<td>0.9778</td>
<td>-2.3652</td>
<td>-0.1834</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 2</td>
<td>0 337.78 50</td>
<td>1.5708</td>
<td>0.9677</td>
<td>-2.3117</td>
<td>-0.2269</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 3</td>
<td>0 346.67 50</td>
<td>1.5708</td>
<td>0.9562</td>
<td>-2.2581</td>
<td>-0.2689</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 4</td>
<td>0 355.56 50</td>
<td>1.5708</td>
<td>0.9433</td>
<td>-2.2043</td>
<td>-0.3098</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 5</td>
<td>0 364.44 50</td>
<td>1.5708</td>
<td>0.9293</td>
<td>-2.1502</td>
<td>-0.3499</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 6</td>
<td>0 373.33 50</td>
<td>1.5708</td>
<td>0.9142</td>
<td>-2.0957</td>
<td>-0.3894</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 7</td>
<td>0 382.22 50</td>
<td>1.5708</td>
<td>0.8982</td>
<td>-2.0406</td>
<td>-0.4283</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 8</td>
<td>0 391.11 50</td>
<td>1.5708</td>
<td>0.8811</td>
<td>-1.9849</td>
<td>-0.4670</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 9</td>
<td>0 400.00 50</td>
<td>1.5708</td>
<td>0.8632</td>
<td>-1.9284</td>
<td>-0.5056</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 10</td>
<td>0 400.00 60</td>
<td>1.5708</td>
<td>0.8999</td>
<td>-1.9137</td>
<td>-0.5570</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 11</td>
<td>0 400.00 70</td>
<td>1.5708</td>
<td>0.9353</td>
<td>-1.8965</td>
<td>-0.6095</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 12</td>
<td>0 400.00 80</td>
<td>1.5708</td>
<td>0.9691</td>
<td>-1.8768</td>
<td>-0.6631</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 13</td>
<td>0 400.00 90</td>
<td>1.5708</td>
<td>1.0013</td>
<td>-1.8546</td>
<td>-0.7175</td>
<td>GAP</td>
</tr>
<tr>
<td>Knot 14</td>
<td>0 400.00 100</td>
<td>1.5707</td>
<td>0.8899</td>
<td>-1.8798</td>
<td>-0.3317</td>
<td>EA</td>
</tr>
<tr>
<td>Knot 15</td>
<td>0 400.00 110</td>
<td>1.5709</td>
<td>0.8836</td>
<td>-1.8623</td>
<td>-0.2760</td>
<td>EA</td>
</tr>
<tr>
<td>Knot 16</td>
<td>0 400.00 120</td>
<td>1.5704</td>
<td>0.9313</td>
<td>-1.8469</td>
<td>-0.3598</td>
<td>EA</td>
</tr>
<tr>
<td>Final</td>
<td>0 400.00 130</td>
<td>1.5700</td>
<td>0.7758</td>
<td>-1.7612</td>
<td>-0.0002</td>
<td>EA</td>
</tr>
</tbody>
</table>

* - Geometric approach, ** - Evolutionary approach
Figs. 4.43 through 4.46 show the robot at the intermediate knot points 14 through 16 and the final point of the trajectory for which a collision was detected with the door. These figures show that the evolutionary approach has been able to achieve collision avoidance at these points of the trajectory.

4.6 CONCLUSIONS

In this chapter, an evolutionary approach based on a real-coded GA was used to achieve redundancy resolution of a redundant robotic manipulator. The redundancy resolution is based on the minimization of total joint displacement of the robot in the presence of obstacles. Based on the work carried out in this chapter, the following conclusions can be drawn:

1. An evolutionary approach based on an elitist real-coded genetic algorithm is able to solve the inverse kinematics problem of redundant robotic manipulators subject to the restrictions imposed by the joint limits and the positioning errors of the robot hand. Performance criteria like total joint displacement minimization can be included in the approach to achieve a redundancy resolution. The positioning error of the Mitsubishi Movemaster RV-M1 robot hand, obtained using the elitist real-coded genetic algorithm, was within 0.5mm in all the simulation experiments.

2. An obstacle avoidance algorithm based on spherization of the robot and obstacles can be incorporated into the evolutionary approach for the inverse kinematics solution of redundant robots. The modelling tolerance of the spherization can be fixed depending upon the clearance required between the robot and the obstacle. The number of spheres used for the spherization can be
Fig. 4.43: Robot configuration at knot point 14 (simulation experiment 2) obtained using evolutionary approach based on real-coded genetic algorithm
Fig. 4.44: Robot configuration at knot point 15 (simulation experiment 2) obtained using evolutionary approach based on real-coded genetic algorithm
Fig. 4.45: Robot configuration at knot point 16 (simulation experiment 2) obtained using evolutionary approach based on real-coded genetic algorithm
Fig. 4.46: Robot configuration at final point of trajectory (simulation experiment 2) obtained using evolutionary approach based on real-coded genetic algorithm.
decided accordingly. The algorithm is able to provide collision-free configurations of the Mitsubishi Movemaster RV-M1 robot working in the vicinity of obstacles in the simulation experiments.

3. An elitist real-coded genetic algorithm gives a better performance than a non-elitist real-coded genetic algorithm for achieving the redundancy resolution. The former approach provides the solution in fewer number of generations than the latter approach.