Redundancy utilization for obstacle avoidance of planar robot manipulators

U Sezgin, **L D Seneviratne** and **S W E Earles**
Department of Mechanical Engineering, King's College London

Abstract: Two obstacle avoidance criteria are developed, utilizing the kinematic redundancy of serial redundant manipulators having revolute joints and tracking pre-determined end effector paths. The first criterion is based on the instantaneous distances between certain selected points along the manipulator, called configuration control points (CCP), and the vertices of the obstacles. The optimized joint configurations are obtained by maximizing these distances. Thus, the links of the manipulator are configured away from the obstacles. The second criterion uses a different approach, and is based on Voronoi boundaries representing the equidistant paths between two obstacles. The optimized joint configurations are obtained by minimizing the distances between the CCP and control points selected on the Voronoi boundaries. The validities of the criteria are demonstrated through computer simulations.

Keywords: obstacle avoidance, redundant manipulators, redundancy resolution, configuration optimization

NOTATION

CCP	configuration control points
DI	maximization objective function
DM	minimization objective function
$\mathbf{I}$	identity matrix
$\mathbf{J}$	manipulator Jacobian
$\mathbf{J}^+$	pseudo-inverse of the Jacobian
$\mathbf{J}^{-1}$	inverse of the Jacobian
l_i	ith link length
m	dimension of the workspace
n	number of DOF
VB	Voronoi boundary
$\boldsymbol{X}$	end effector position vector
$\dot{\boldsymbol{X}}$	end effector velocity vector
$\dot{\boldsymbol{\theta}}$	joint velocity vector
$\|\cdot\|$	Euclidian norm

1 INTRODUCTION

One of the major advantages of the human arm is its superior manoeuvring capability in complex workspaces cluttered with obstacles. One key to having this advantage is its kinematic redundant nature; i.e. it has more degrees of freedom (DOF) than required to perform an operation. This observation has motivated researchers to investigate manipulators with kinematic redundancy as industry demands more versatile and dexterous manipulators for advanced robotics applications. The additional degrees of freedom of redundant manipulators can be utilized to avoid obstacles and improve performance.

Task descriptions for redundant manipulators usually specify the priority of the goals (**1**, **2**). For instance, the manipulation of an object in a cluttered environment may have a primary goal, to manipulate the object to the desired location, and a secondary goal, to avoid obstacles (**3**). Alternatively, the secondary goal may involve achieving low-energy motion (**4**) or maintaining maximum manipulatory ability (**5**).

The important analytical problem for a redundant manipulator is its inverse kinematics, i.e. the problem of finding the joint coordinate vectors given the end effector position and orientation (POSE). For a redundant manipulator there are an infinite number of joint coordinate vectors that correspond to the same end effector POSE. This is due to the manipulator DOF being greater than the dimensions of the workspace. A unique solution may be found if some performance criterion is imposed on the manipulator kinematics, known as a redundancy resolution. In the literature two main redundancy resolution methods have emerged: the velocity-based resolved motion method (**6**) and the position-based Lagrangian multiplier method (**7**). Although the latter

The MS was received on 10 August 1995 and was accepted for publication on 5 November 1996.

method provides a direct mapping from the task space to the joint space, it requires lengthy computations and therefore is unsuitable for on-line applications. The velocity-based method is computationally efficient and thus is more suitable for real-time obstacle avoidance applications.

Obstacles are objects in the workspace of a manipulator and avoiding contact with them is essential for satisfactory task completion. The obstacle avoidance problem for redundant manipulators is generally approached in the context of secondary task descriptions, the primary task being to track an end effector trajectory (**2**, **8**–**12**, **17**). This is essentially an optimization problem where the primary and secondary tasks must be satisfied while physical limitations of the manipulator are respected.

There have been several approaches to redundant manipulator obstacle avoidance. Khatib and Le Maitre (**10**) used artificial potential and dissipative functions to determine the joint torques required to push the manipulator away from the obstacles. A drawback of this method is that it becomes inefficient when a local minima is reached. Yoshikawa (**2**) developed a kinematic method that makes the manipulator links approach closely a given reference configuration, which is determined by considering the obstacle locations in the workspace, so that the manipulator arrives at its final location without colliding with the obstacles. This method is invalid in dynamic environments. Maciejewski and Klein (**11**) proposed an obstacle avoidance method based on the closest point on a manipulator to an obstacle. They defined a secondary task by assigning to the point a velocity component directed away from the obstacle. The implementation of this method is not straightforward, due to the mathematical complexities in the task description and the addition of the velocity component. Similarly, Sciavicco and Siciliano (**12**) gave a dynamic solution to avoid a single obstacle in the workspace. De Luca *et al.* (**8**) and Glass *et al.* (**9**) proposed two different techniques to avoid collisions in real-time based on adaptive control schemes.

This paper presents two collision avoidance strategies for serial redundant manipulators having revolute joints and tracking predetermined end effector paths. The first approach deals with a workspace with a single obstacle and maximizes the sum of the instantaneous distances between the CCP on the manipulator and the obstacle vertex that is closest to each CCP. Thus the CCP are repelled from the obstacle and the links are configured away from it. The second approach deals with a workspace with multiple obstacles and a manipulator required to go between them. Voronoi boundaries (**6**), which give the locus of points equidistant from the obstacles, are used to achieve this purpose. The sum of distances between selected points on the Voronoi boundary and the CCP on the manipulator is minimized. Thus the links are configured close to the Voronoi boundary and away from the obstacles.

The methods are simple to implement and offer flexibility when executing tasks in environments cluttered with obstacles. They are also applicable to dynamic environments where the obstacles are no longer static. In the following sections, the two criteria are developed and tested in computer simulations.

2 REDUNDANCY RESOLUTION

Let the end effector position $\boldsymbol{X} \in \Re^m$ of a robot manipulator be related to the vector of joint coordinates $\theta \in \Re^n (n > m)$ by the forward kinematic equation:

$$\boldsymbol{X} = f(\theta) \tag{1}$$

where m is the dimension of the workspace and n is the number of degrees of freedom (DOF) of the manipulator.

If $n > m$ the manipulator is kinematically redundant, where $n - m$ represents the degree of redundancy, indicating the minimum number of constraints needed to fully specify the inverse kinematics problem. By differentiating equation (1) with respect to time the relationship between the joint coordinate velocity vector, $\dot{\theta} \in \Re^n$, and the end effector velocity vector, $\dot{\boldsymbol{X}} \in \Re^m$, is obtained:

$$\dot{\boldsymbol{X}} = \mathbf{J}(\theta)\dot{\theta} \tag{2}$$

where $\mathbf{J}(\theta) = \partial f / \partial \theta \in \Re^{m \times n}$ is the manipulator Jacobian matrix. For a redundant manipulator the Jacobian matrix is rectangular, and its inverse does not exist. However, generalized inverses of $\mathbf{J}(\theta)$ do exist, and of these the pseudo-inverse is most commonly used. Because the pseudo-inverse has the minimum norm, it yields the least-squares solution of the minimum norm. This is a highly desirable property and can be utilized for power minimization and singularity avoidance. Thus equation (2) can be inverted to give the so-called Jacobian control method (**13**):

$$\dot{\theta} = \mathbf{J}^+ \dot{\boldsymbol{X}} \tag{3}$$

where $\mathbf{J}^+ = \mathbf{J}^{\mathrm{T}}(\mathbf{J}\mathbf{J}^{\mathrm{T}})^{-1}$ is the $n \times m$ pseudo-inverse of $\mathbf{J}(\theta)$ and $\mathbf{J}^{\mathrm{T}}$ denotes the transpose of $\mathbf{J}$. Assuming the Jacobian is of full rank, that is rank $(\mathbf{J}) = m < n$, then equation (3) gives a particular solution for $\dot{\theta}$. It is noted that the pseudo-inverse solution is only valid for cases of serial manipulators where all the joints are of the same type.

Liégeois (**3**) introduced a method that allows the gradient of an arbitrary function to be incorporated. This is done by adding a homogeneous solution to equation (3):

$$\dot{\theta} = \mathbf{J}^+ \dot{\boldsymbol{X}} + \lambda(\mathbf{I} - \mathbf{J}^+ \mathbf{J}) \nabla H(\theta) \tag{4}$$

where $(\mathbf{I} - \mathbf{J}^+ \mathbf{J})$ is the projection operator on the

null space of $\mathbf{J}(\theta)$ and $\mathbf{I}$ is an $n \times n$ identity matrix. $H(\theta)$ is an objective function used to optimize a desired performance, its gradient being given by

$$\nabla H(\theta) = \left(\frac{\partial H}{\partial \theta_1} \frac{\partial H}{\partial \theta_2} \cdots \frac{\partial H}{\partial \theta_n} \right)^{\mathrm{T}} \tag{5}$$

λ is a gain constant governing the joint angle velocities of the links; the higher the value of λ the faster are the joint angular velocities. Negative values of λ minimize $H(\theta)$ and positive values maximize $H(\theta)$.

The homogeneous solution affects only the joint angular motion of the manipulator and does not influence the end effector motion. For a given end effector trajectory, the homogeneous solution can be used to improve a desired performance characteristic of the manipulator, represented by $H(\theta)$, which is a scalar function of the joint angles.

Yoshikawa (**5**) used $H(\theta)$ to avoid singularities and Maciejewski and Klein (**11**) to avoid obstacles. Many other researchers have used $H(\theta)$ to improve performance (**4**, **7**, **10**, **14**, **15**). In the present work, two objective functions are defined to avoid single and multiple obstacles in the workspace. Given a predetermined end effector path, the objective is to optimize the joint angular variables so as to configure the links away from the obstacles.

The method described above, equations (3) and (4), is called the resolved motion control method and maps the end effector velocity space to the joint velocity space. This paper considers the joint angle paths that are obtained through numerical integration. Although the resolved motion method is computationally efficient it has several drawbacks. For example, Clein and Chirco (**14**) showed that it may result in end effector tracking errors when a secondary criterion is imposed. Further, it does not account for infinitesimal changes in manipulator geometry and may result in drift, when cyclic operations in the joint space are performed. The drawback of instability, because of the nature of the pseudo-inverse (**16**), is also pointed out in references (**4**) and (**7**). The direct solution of the joint variables would eliminate the above difficulties. Such a method based on Lagrangian multipliers is introduced by Chang (**7**), where an equation is given containing a set of kinematic constraints in addition to the forward kinematic equations:

$$Z \nabla H(\theta) = 0 \tag{6}$$

where

$$Z = [(\mathbf{J}^{n-m})^{\mathrm{T}} (\mathbf{J}^{m})^{-1} \quad -\mathbf{I}] \in \Re^{(n-m) \times n} \tag{7}$$

$\mathbf{J}^{n-m} \in \Re^{m \times (n-m)}$ and $\mathbf{J}^{m} \in \Re^{m \times m}$ are matrices that are parts of the manipulator Jacobian matrix $\mathbf{J}^{m \times n}$, $\mathbf{I} \in \Re^{(n-m) \times (n-m)}$ is an identity matrix and $\nabla H(\theta)$ is given by equation (5). To find the joint variables directly equations (1) and (6) are iteratively solved using a numerical method. However, this requires significantly longer computation times than the velocity-based method and is in general not suitable for real-time applications.

It is noted that both the resolved motion method and the Lagrange multiplier method are alternative methods for solving the same problem. Thus for identical input conditions both methods will give the same joint configurations, provided that integration errors associated with the resolved motion method are minimal.

3 THE OBSTACLE AVOIDANCE CRITERIA

In this section, two objective functions for collision avoidance of planar serial redundant manipulators having revolute joints are presented and implemented using the redundancy resolution techniques outlined in Section 2. The two objective functions relate to environments with one obstacle and multiple obstacles. Both the objective functions are based on the concept of configuration control points (CCP), which are suitably placed along the kinematic structure of the manipulator. The location and number of CCP selected are of vital importance and these issues will be addressed later.

Once the CCP are placed along the manipulator kinematic structure, their coordinates can be found by using the forward kinematic relationships. The coordinate of the ith CCP can be represented by

$$\boldsymbol{X}_i(x_i(t), y_i(t), z_i(t)) \in \Re^{1 \times 3} = f_i(\theta, t) \tag{8}$$

where $\theta \in \Re^{n \times 1}$ is the vector of manipulator joint coordinates, $t \in [0, \tau]$, τ is the specified task time and $f_i \in \Re^3$ is the forward kinematic relationship relating to the ith CCP.

Consider the workspace of the manipulator to have polygonal obstacles which are defined by the coordinates of their vertices (Fig. 1). Let the jth obstacle have δ vertices given by the matrix

$$\boldsymbol{\kappa}_j(t) \in \Re^{\delta \times 3} = \begin{pmatrix} X_{1j}(t) & Y_{1j}(t) & Z_{1j}(t) \\ X_{2j}(t) & Y_{2j}(t) & Z_{2j}(t) \\ \vdots & \vdots & \vdots \\ X_{\delta j}(t) & Y_{\delta j}(t) & Z_{\delta j}(t) \end{pmatrix} \tag{9}$$

and $\boldsymbol{\kappa}_j^{\beta} \in \Re^{1 \times 3} = (X_{\beta j}(t) \quad Y_{\beta j}(t) \quad Z_{\beta j}(t))$ represents the βth vertex of the jth obstacle.

3.1 Workspaces with a single obstacle

The workspace has within it a single obstacle, whose vertices are defined by $\boldsymbol{\kappa}_1 \in \Re^{\delta \times 3}$. A collision is likely to occur if a link of the manipulator comes too close to the obstacle.

Let the ith CCP be $f_i(\theta, t)$ and κ_1^i be the obstacle vertex that is closest to the ith CCP. This vertex, together with the CCP, is used to define the objective function

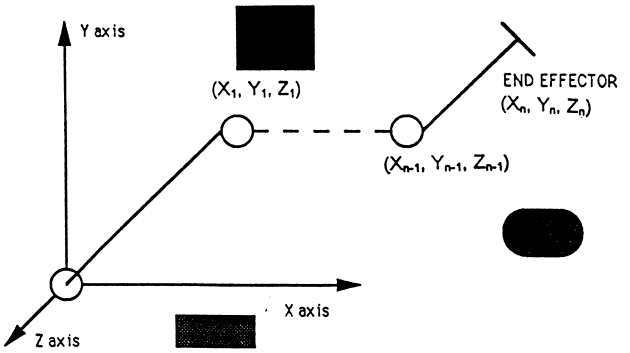

Fig. 1 An n DOF manipulator in an environment cluttered with obstacles

for collision avoidance:

$$\mathrm{DI}(\theta, t) = \sum_{i=1}^{k} \| f_i(\theta, t) - \kappa_1^i(t) \| \tag{10}$$

where k represents the number of CCP on the manipulator kinematic structure. $\mathrm{DI}(\theta, t)$ is a scalar function that is the sum of the minimum distances between the CCP and obstacle vertices. The maximization of DI, as a secondary task, will cause the links to configure away from the obstacle. DI can be implemented with the redundancy resolution schemes outlined in Section 2, equations (4) and (6), in order to safely configure the redundant manipulator.

3.2 Workspaces with multiple obstacles

When there is more than one obstacle and it is necessary to go between them, the above approach will not work. The technique adopted here uses the Voronoi boundary (VB), which represents the locus of equidistant points from the obstacles. The optimization scheme adopted attempts to keep the manipulator as close as possible to the VB, while the end effector is tracking the predetermined path.

In order to implement this scheme control points are selected on the VB, and $V_i \in \Re^{1\times 3}$ taken to represent the coordinates of the control point nearest to the ith CCP, $f_i(\theta, t)$. The objective function is then formed by the sum of the Euclidian norms between the CCP and their nearest control points on the VB:

$$\mathrm{DM} = \sum_{i=1}^{k} \| f_i(\theta, t) - V_i(t) \| \tag{11}$$

The minimization of this norm will keep the links of the manipulator in the neighbourhood of the VB and away from the obstacles.

As pointed out in reference (**3**), it is generally not feasible to employ a single criterion to satisfy all the constraints of a specific task, and the use of multiple criteria could greatly increase the versatility of redundant manipulators. This can be implemented in the framework of an algorithm that switches from an obstacle avoidance criterion to other specified criteria and vice versa, based on a preset threshold distance.

If there are more than two obstacles, both DI and DM can be used in combination to avoid these obstacles. For instance, when the end effector is approaching a gap formed by two obstacles to reach a location beyond the gap, initially it may be necessary to maximize distances between the links and the two obstacles using DI. Once the end effector has entered the gap, the minimization of the distances between the CCP and the selected points on the Voronoi boundary of the two obstacles will ensure safe task completion.

Since locations of the obstacles are updated at every sampling instant, the algorithms can also be used in dynamically changing environments.

4 COMPUTER SIMULATION RESULTS

4.1 Equation formulation

The redundant arm selected for the simulation study has 3 DOFs and performs in two-dimensional space (Fig. 2). For simplicity, it is assumed that the joint ranges are unconstrained. The forward kinematic equations of the manipulator depicted in Fig. 2 are given by

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} l_1 \sin\theta_1 + l_2 \sin(\theta_1+\theta_2) + l_3 \sin(\theta_1+\theta_2+\theta_3) \\ l_0 + l_1 \cos\theta_1 + l_2 \cos(\theta_1+\theta_2) + l_3 \cos(\theta_1+\theta_2+\theta_3) \end{pmatrix} \tag{12}$$

where l_i represents the link lengths ($l_1 = l_2 = l_3 = 1$ unit) and l_0 is the y coordinate of the fixed joint (Fig. 2).

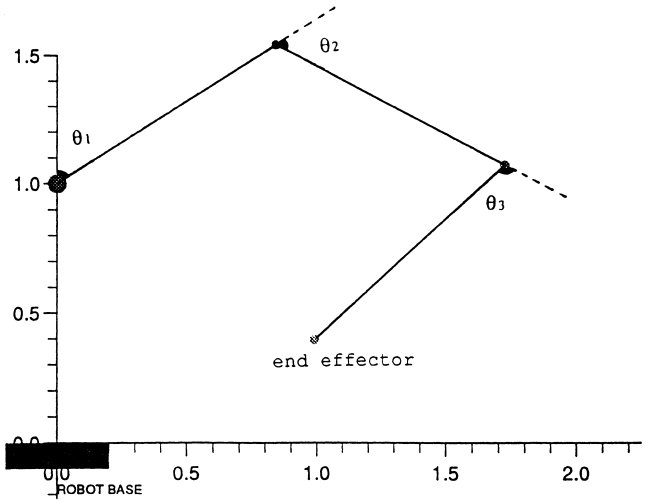

Fig. 2 The schematic diagram of the planar redundant manipulator to be simulated

Using the definition given in equation (2), the Jacobian matrix is derived as

$$\mathbf{J} = \begin{pmatrix} \sum_{i=1}^{3} l_i \cos\theta_i & \sum_{i=2}^{3} l_i \cos\theta_i & \sum_{i=3}^{3} l_i \cos\theta_i \\ -\sum_{i=1}^{3} l_i \sin\theta_i & -\sum_{i=2}^{3} l_i \sin\theta_i & -\sum_{i=3}^{3} l_i \sin\theta_i \end{pmatrix} \tag{13}$$

The pseudo-inverse of the Jacobian can be obtained from equation (3). Once the CCP are selected on the manipulator, their Cartesian coordinates can be found by adapting equation (12).

The problem to be solved is to find the optimal safe joint configurations for a given end effector path and a set of obstacles in the workspace. To implement the DI and DM criteria, equations (10) and (11), CCP are selected on the manipulator and expressed in terms of joint variables. The control points for a single obstacle are given by the obstacle vertices. For multiple obstacles, the control points are placed on the Voronoi boundary. The optimized link configurations are determined by solving equation (4) using the fourth-order Runge–Kutta numerical integration algorithm. The initial conditions are obtained by using the Lagrangian multiplier method at the first time step.

4.2 Test results

The obstacle avoidance criteria developed are tested using computer simulations for a variety of test situations. In particular, results are presented for:

(a) varying the number and location of the CCP;
(b) varying the number and size/shape of the obstacles;
(c) static and dynamic obstacles.

Example 1: single obstacle avoidance criterion—DI

The task is to move the manipulator from the initial point [0.5, 0.5] to the final point [2.5, −0.2] along a straight line in an environment containing a single obstacle. The initial joint state is calculated as

$$\theta(0) = [32.6^\circ,\ 101.3^\circ,\ 95.60^\circ]^{\mathrm{T}}$$

Two obstacle shapes are considered:

1. Obstacle 1 is a rectangle with coordinates

$$\boldsymbol{\kappa}_1 = \begin{pmatrix} 1.8 & 0.1 \\ 1.8 & 1.2 \\ 2.4 & 1.2 \\ 2.4 & 0.1 \end{pmatrix}$$

2. Obstacle 2 is an L-shaped object with coordinates

$$\boldsymbol{\kappa}_2 = \begin{pmatrix} 2.0 & 0.0 \\ 2.0 & 0.5 \\ 1.5 & 0.5 \\ 1.5 & 1.1 \\ 2.6 & 1.1 \\ 2.6 & 0.0 \end{pmatrix}$$

The task is executed using the proposed obstacle avoidance criterion for one CCP at different locations along the manipulator links. Figure 3a shows configurations found by selecting a CCP at the second joint failing to reach the final location, whereas configurations found by selecting a CCP at the third joint (Fig. 3b) and by selecting a CCP on the mid-point of the third link (Fig. 3c) avoid the obstacle and successfully reach the goal location. The joint solutions are listed in Table 1.

It can be seen from this example that the location of the CCP is critically important. A single CCP is sufficient to obtain a safe path, provided that this CCP is sufficiently close to the region of potential collision (Figs. 3b and c). However, if the CCP is away from the region of potential collision, then the criterion fails to find a safe path (Fig. 3a).

The same task with two CCP, (a) at the middle of links 1 and 2 and (b) at the middle of links 2 and 3, is carried out for obstacle 2. The joint configurations are illustrated in Fig. 4. Figure 4a (CCP at the middle of links 1 and 2) shows configurations failing to accomplish the task, whereas Fig. 4b (CCP at the middle of links 2 and 3) shows a successful execution of the task. The reason for failure in Fig. 4a is that the CCP are remote from the end effector and the obstacle. Thus they do not have the same degree of control over the region of collision as those in Fig. 4b.

Reviewing these results indicates that the dominance of the CCP increases with its nearness to the obstacle and the region of potential collision. For instance, compare Fig. 3c with Fig. 4b: despite there being one CCP in Fig. 3c, the configurations are similar to that of Fig. 4b where there are two CCP. The common CCP is at the middle of the third link, and is clearly more dominant than the CCP on the second link in terms of collision avoidance.

Increasing the number of CCP to three and four leads to successful completion of the task, as shown in Figs. 5a and b. Results presented in Fig. 5 for an increased number of CCP prompts the conclusion that selecting more points will not make a significant difference. The CCP near to the obstacle and end effector will be dominant, regardless of the number of CCP on the rest of the kinematic structure.

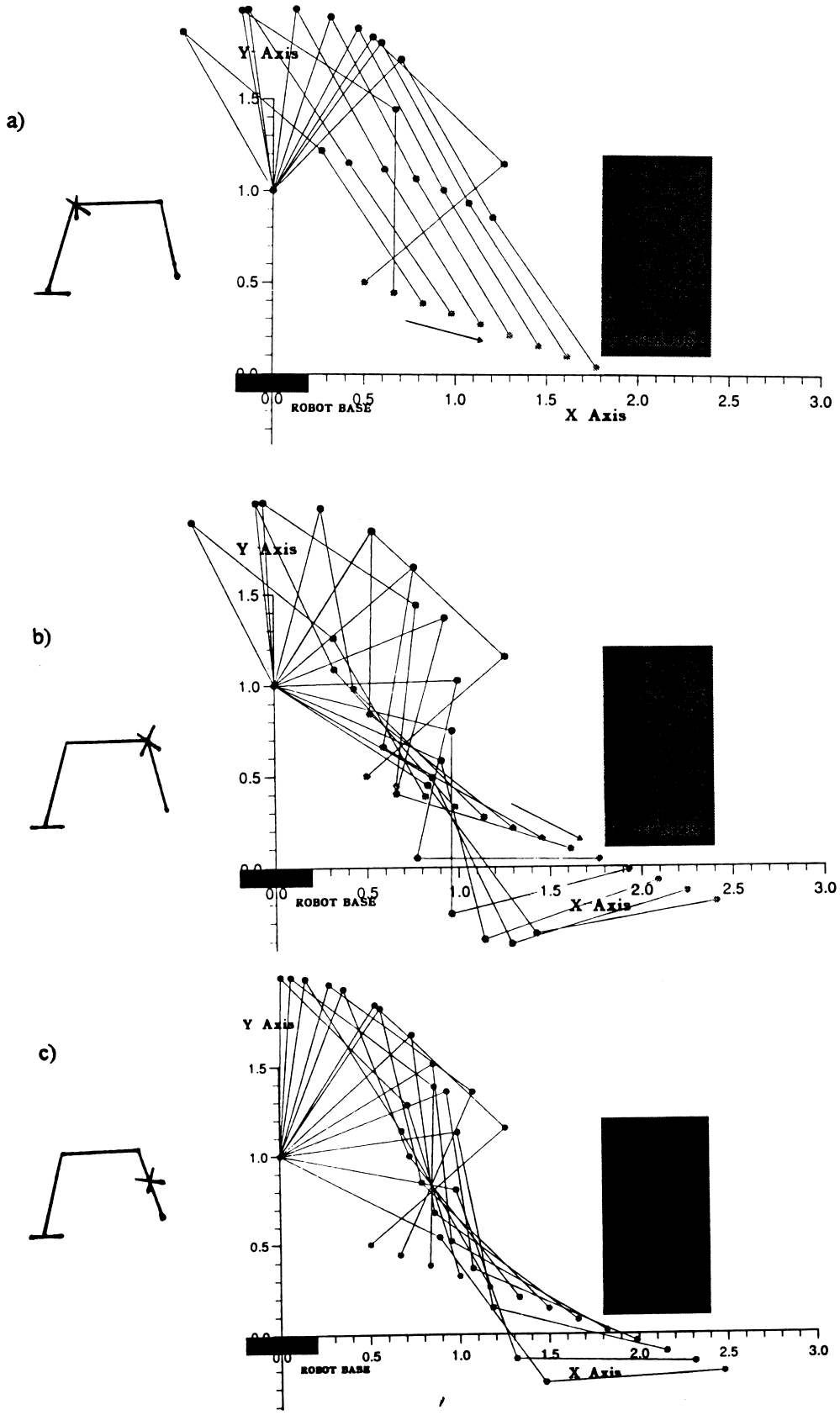

Fig. 3 Adjacent configurations for one CCP: (a) at joint 2 (collision), (b) at joint 3 (no collision), (c) at the mid-point of link 3 (no collision). × marks the location of the CCP

Table 1 Configurations for Fig. 3 (DEG)—Example 1

One CCP at joint 2

Knot point number	Joint 1	Joint 2	Joint 3
1	9.61	116.70	76.77
2	−23.89	146.54	39.68
3	−19.47	159.41	3.96
4	−1.11	149.43	−1.93
5	11.35	139.38	−2.98
6	21.38	129.86	−3.47
7	30.18	120.53	−3.87
8	38.35	111.14	−4.28
9	46.23	101.48	−4.74

One CCP at joint 3

1	14.82	112.70	80.38
2	−3.90	127.61	62.18
3	−18.98	142.38	43.21
4	−19.58	162.31	−1.74
5	4.11	158.30	−25.69
6	22.79	152.45	−43.71
7	40.27	145.31	−61.59
8	58.94	134.98	−80.94
9	81.04	115.77	−101.26
10	102.51	82.66	−110.34
11	108.51	65.53	−102.07
12	111.75	52.39	−90.23
13	116.18	37.02	−76.22
14	119.48	22.32	−58.49

One CCP at the middle of link 3

1	32.55	133.82	0.16
2	16.54	110.77	77.03
3	4.18	123.75	53.64
4	0.94	134.91	27.47
5	8.80	139.46	2.62
6	21.45	137.69	−16.69
7	34.58	132.68	−32.41
8	47.82	125.35	−46.53
9	61.70	115.08	−59.74
10	76.73	100.10	−71.13
11	92.02	78.83	−76.44
12	104.35	55.10	−70.91
13	112.13	33.99	−55.51

When several CCP are used, the optimization scheme attempts to maximize the sum of the norms between the CCP and the obstacle vertices. In practice, it is possible for one of the norms to be very small or zero, while still maximizing the sum. Thus, the choice of the number and location of the CCP needs careful consideration, within the mechanism constraint that the end effector path is predetermined.

In particular, the CCP should be placed in regions where collisions are most likely and not in regions where collisions are not possible, as this could lead to less safe results. In general, the proper location of a limited number of CCP will give better results than the use of a relatively large number of CCP.

Example 2: multiple obstacle avoidance criterion with a moving obstacle

This example demonstrates the criterion for avoiding multiple obstacles [equation (11)] for a case where one of the obstacles is moving while the other is stationary. The task is to move the end effector from [2.0 −0.2] to [2.0 1.4] along a straight line. The initial joint state is found as

$$\theta(0) = [77.59^\circ,\ 37.35^\circ,\ 58.36^\circ]^{\mathrm{T}}$$

The upper obstacle (Fig. 6) is moving horizontally at a rate of $[0.1\ \ 0.0]^{\mathrm{T}}$ unit per sampling period. This obstacle is triangular shaped and the vertex coordinates of its initial position are given by

$$\boldsymbol{\kappa}_1 = \begin{pmatrix} 1.4 & 1.4 \\ 0.9 & 1.9 \\ 1.6 & 2.0 \end{pmatrix}$$

The lower obstacle is rectangular and remains stationary. Its vertex coordinates are given by

$$\boldsymbol{\kappa}_2 = \begin{pmatrix} 0.9 & 0.3 \\ 0.9 & 0.8 \\ 1.4 & 0.8 \\ 1.4 & 0.3 \end{pmatrix}$$

The manipulator is represented by two CCP at joints 2 and 3. The control point on the Voronoi boundary is the mid-point between the two obstacle vertices forming the smallest gap between the obstacles, and its coordinates are updated at each sampling instant. Adjacent configurations depicted in Fig. 6 show that both of the obstacles are avoided as the end effector reaches its goal location.

Example 3: hybrid use of criteria

In this example it is demonstrated that by use of the proposed criteria, DI and DM, a redundant manipulator can reach a location beyond two obstacles through a narrow gap. The gap is formed by two rectangular obstacles in the workspace, whose vertex coordinates are given by

$$\boldsymbol{\kappa}_1 = \begin{pmatrix} 1.4 & 0.0 \\ 1.4 & 0.8 \\ 1.6 & 0.8 \\ 1.6 & 0.0 \end{pmatrix}$$

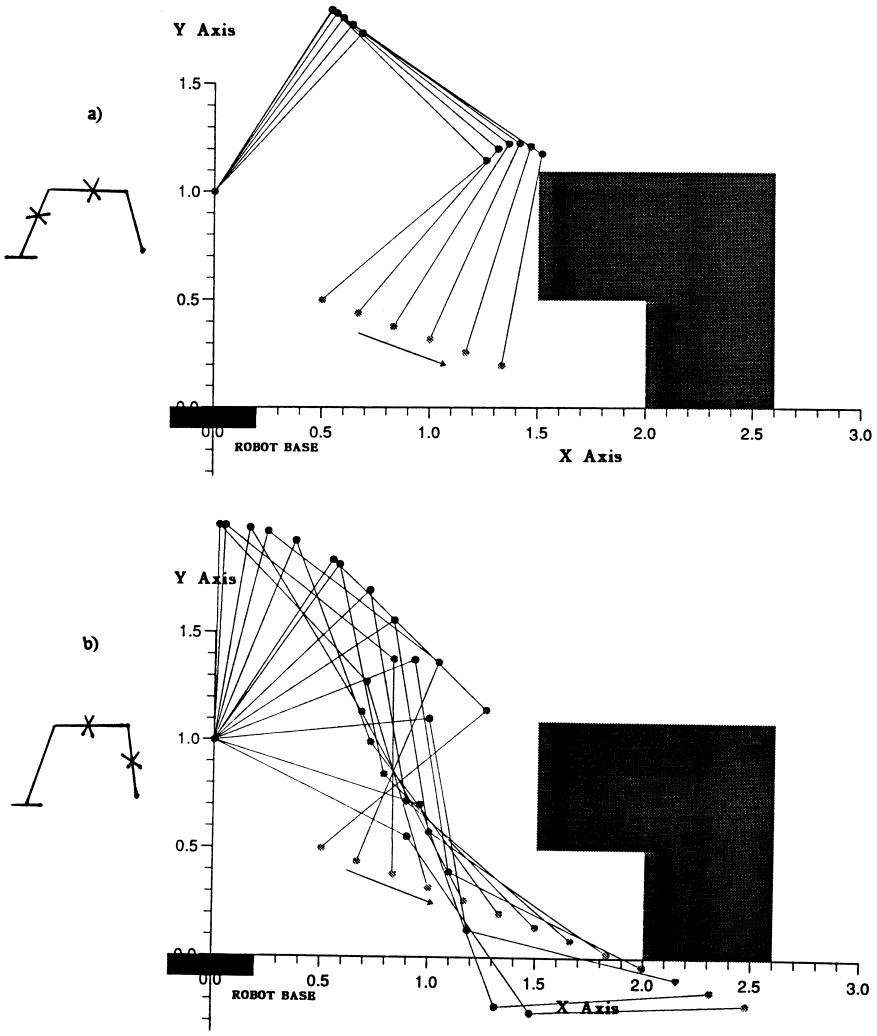

Fig. 4 Adjacent configurations for two CCP: (a) two CCP at the middle of links 1 and 2 (collision), (b) two CCP at the middle of links 2 and 3 (no collision). × marks the location of the CCP

and

$$\kappa_2 = \begin{pmatrix} 1.4 & 1.2 \\ 1.4 & 2.0 \\ 1.6 & 2.0 \\ 1.6 & 1.2 \end{pmatrix}$$

The task is to move the manipulator end effector from [0.5, 0.5] to [2.8, 0.4], via points [1.2, 1.0] and [2.7, 0.8]. The initial joint state is found as

$$\theta(0) = [32.6^\circ,\ 101.3^\circ,\ 95.60^\circ]^{\mathrm{T}}$$

For this example, two CCP are selected at joints 2 and 3, and both criteria [equations (10) and (11)] are used in combination. First, DI is used to make the links align with the gap and, when the end effector has reached the gap, DM is used to keep the links as near as possible to the control points on the Voronoi boundary (VB). The Voronoi boundary, in this case, is a straight line passing between the obstacles. Two control points are selected on the VB with coordinates of (1.4, 1.0) and (1.6, 1.0). The joint configurations at these key points and at some intermediate points are depicted in Fig. 7, showing the successful completion of the task.

Example 4: restricted environment

The proposed obstacle avoidance criteria can also provide solutions when a redundant manipulator is required to work in a restricted environment, e.g. next to a wall. Two examples are given in Fig. 8. The criterion [equation (10)] is formed using selected control points on the wall (the end and middle points) and two CCP at joints 2 and 3. It is seen in both examples that when the end effector is approaching the wall or moving along it the links are configured away from the wall.

4.3 Computational considerations

Two different optimization methods are used to calculate the optimized joint configurations in the simulations: namely the resolved motion method and the Lagrangian

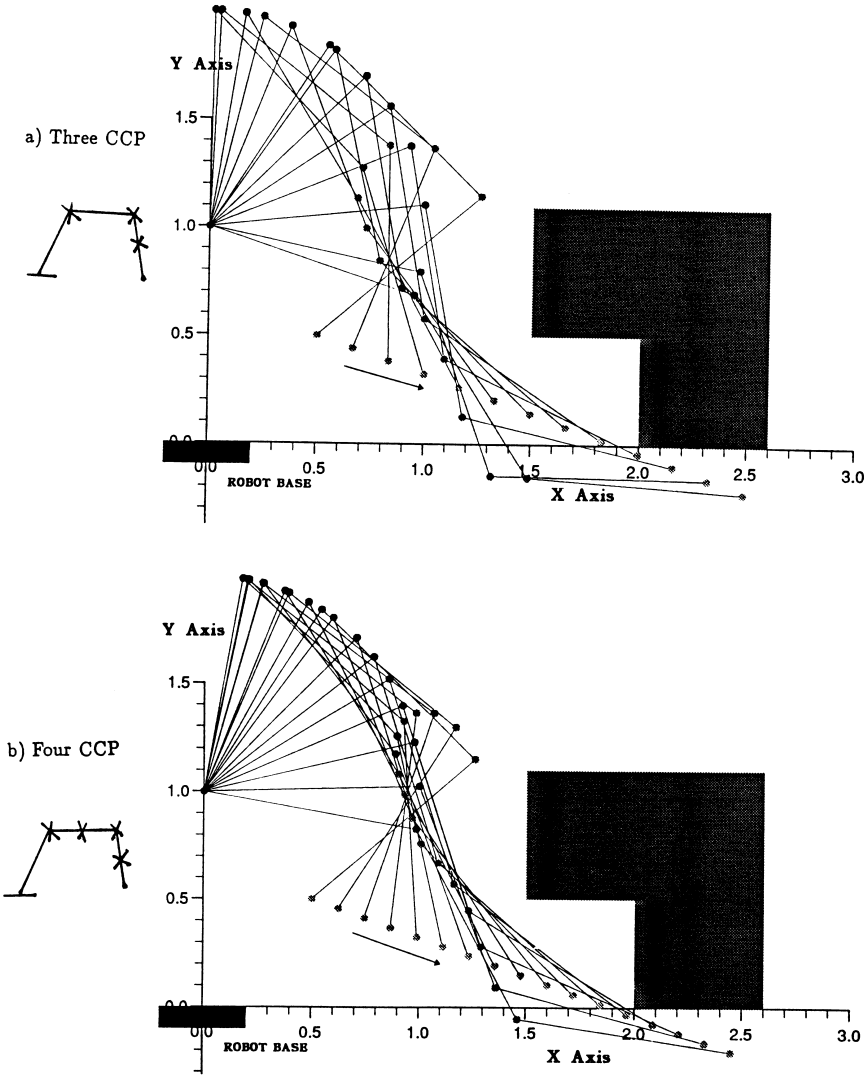

Fig. 5 Adjacent configurations for three and four CCP: (a) three CCP, two at joints 2 and 3 and one at the middle of link 3 (no collision), (b) four CCP, two at joints 2 and 3 and the other two at the middle of links 2 and 3 (no collision). × marks the location of the CCP

multiplier method. The CPU (central processing unit) times taken on a VAX4000-100a computer to calculate the joint variables of the manipulator, based on the criterion given by equation (10), are shown in Table 2 for one sampling distance (0.02 units). For the sake of comparison different numbers of CCP are selected on the manipulator.

It is seen that the resolved motion method costs considerably less than the Lagrangian multiplier method in computational terms. Further, as is expected, the greater the number of CCP, the greater the computation time. Thus, if a task is required to be executed in real-time the number of CCP should be kept to a sufficient minimum.

5 CONCLUSIONS

Two objective functions, DI for a single obstacle and DM for multiple obstacles, have been developed to avoid stationary and moving obstacles in the workspace of a planar redundant manipulator, given a predetermined end effector path. The functions compensate for the underdefined inverse kinematics and make beneficial use of kinematic redundancy.

The computer simulation studies demonstrate the validity of DI and DM in various environments. Two techniques, the resolved motion method and the Lagrangian multiplier method are used for finding the optimized joint configurations. The former is shown to be computationally more efficient.

In the formation of both criteria, the manipulator is represented by configuration control points (CCP) and the obstacles by their vertex coordinates. In the formation of DI and DM it is important to represent the manipulator sufficiently by the CCP so that it performs a safe operation. If the manipulator is not represented in this way some of its links will not be adequately controlled by the objective function and these links may

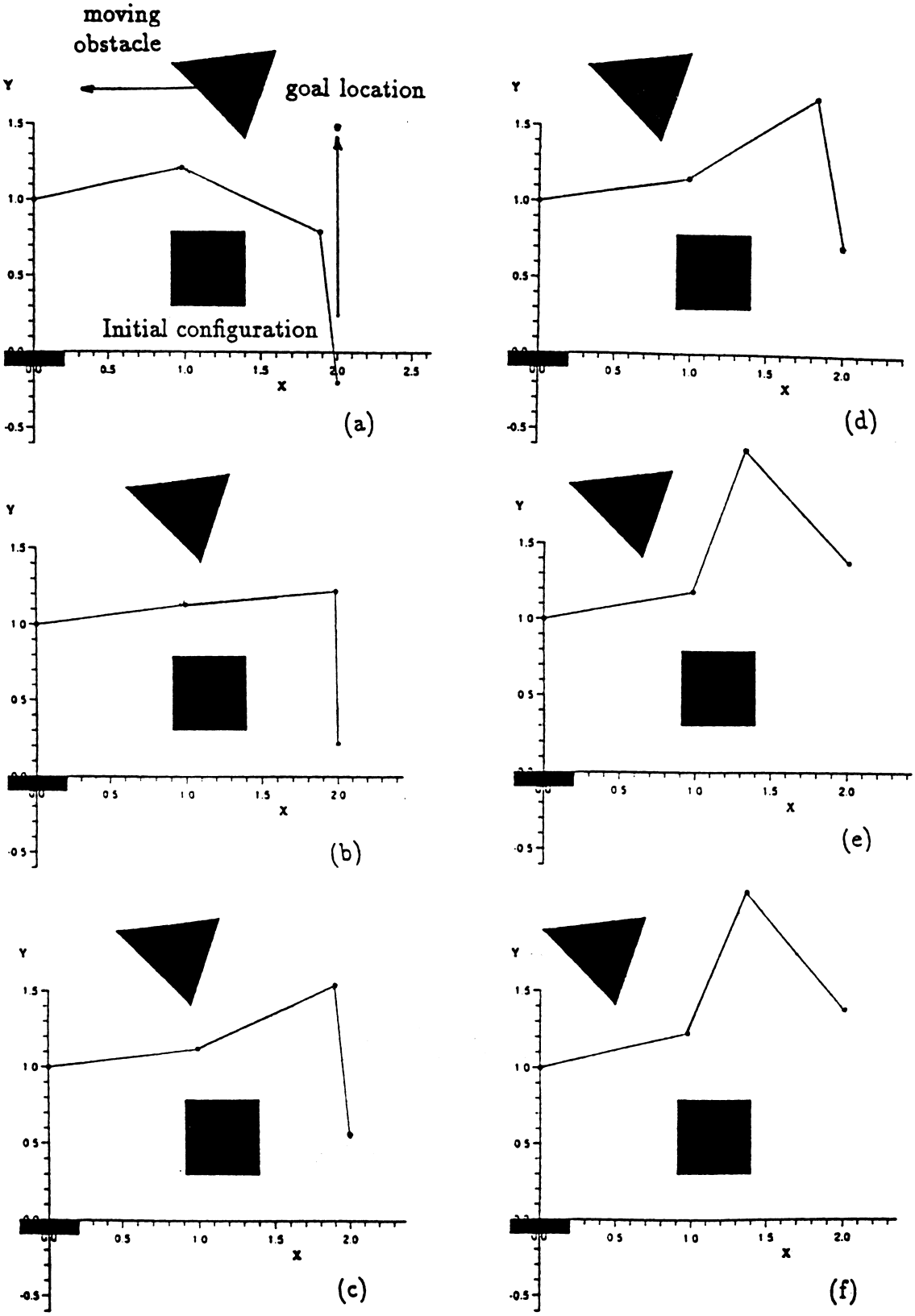

Fig. 6 The triangular obstacle moves horizontally as the end effector moves upwards

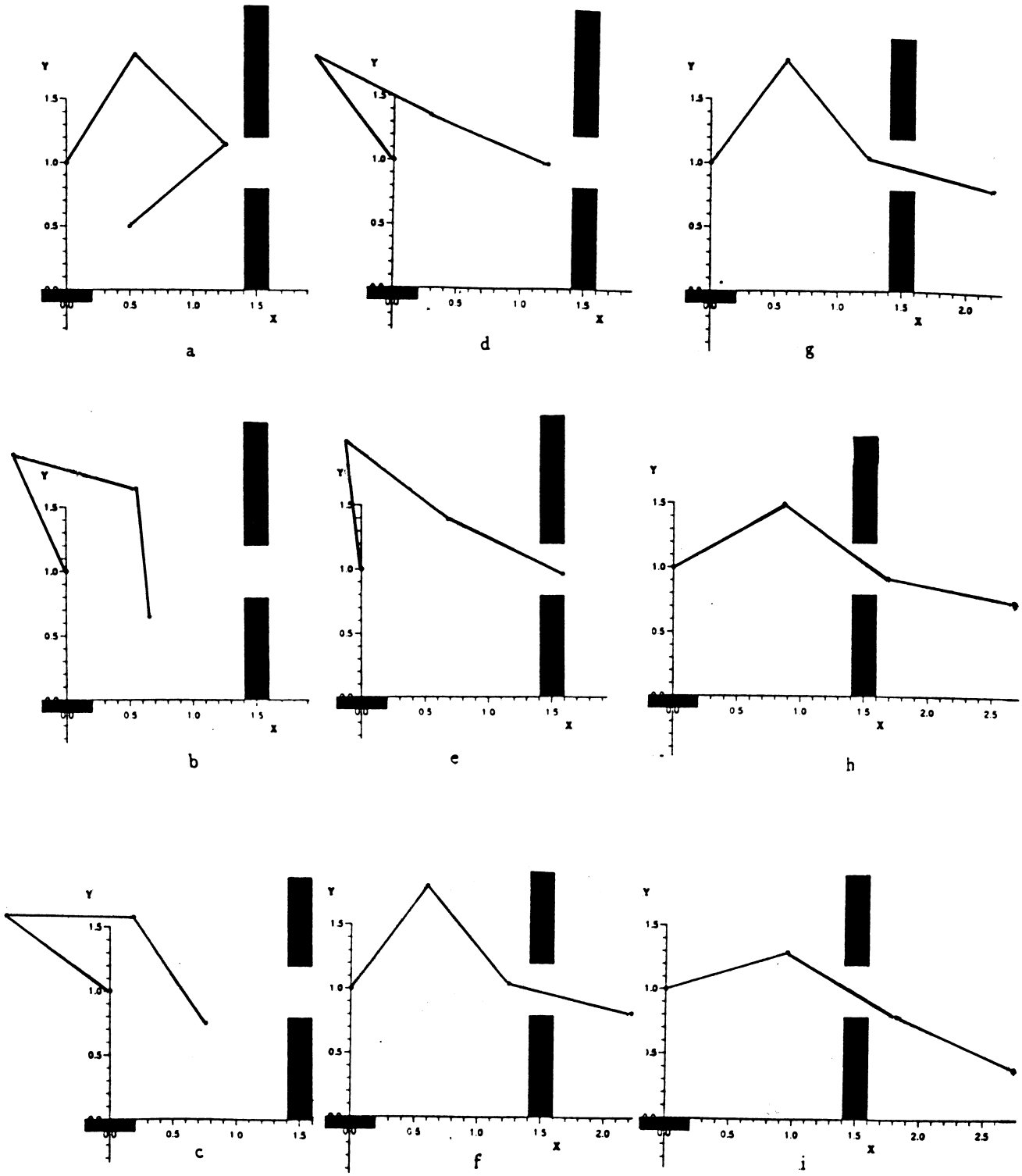

Fig. 7 Adjacent configurations showing a redundant manipulator reaching a point through a narrow gap

collide with the obstacles. In the simulation study it is found that selecting two CCP at joints 2 and 3, for a 3 DOF planar manipulator, provides safe operation. It is also found that the location of the CCP is an important consideration, and is generally more important than the number of CCP. From a practical viewpoint the implementation of the proposed criteria to spatial linkage mechanisms needs to be investigated, together with finding the *optimal* number of CCP for a given mechanism to obtain configurations that ensure safe

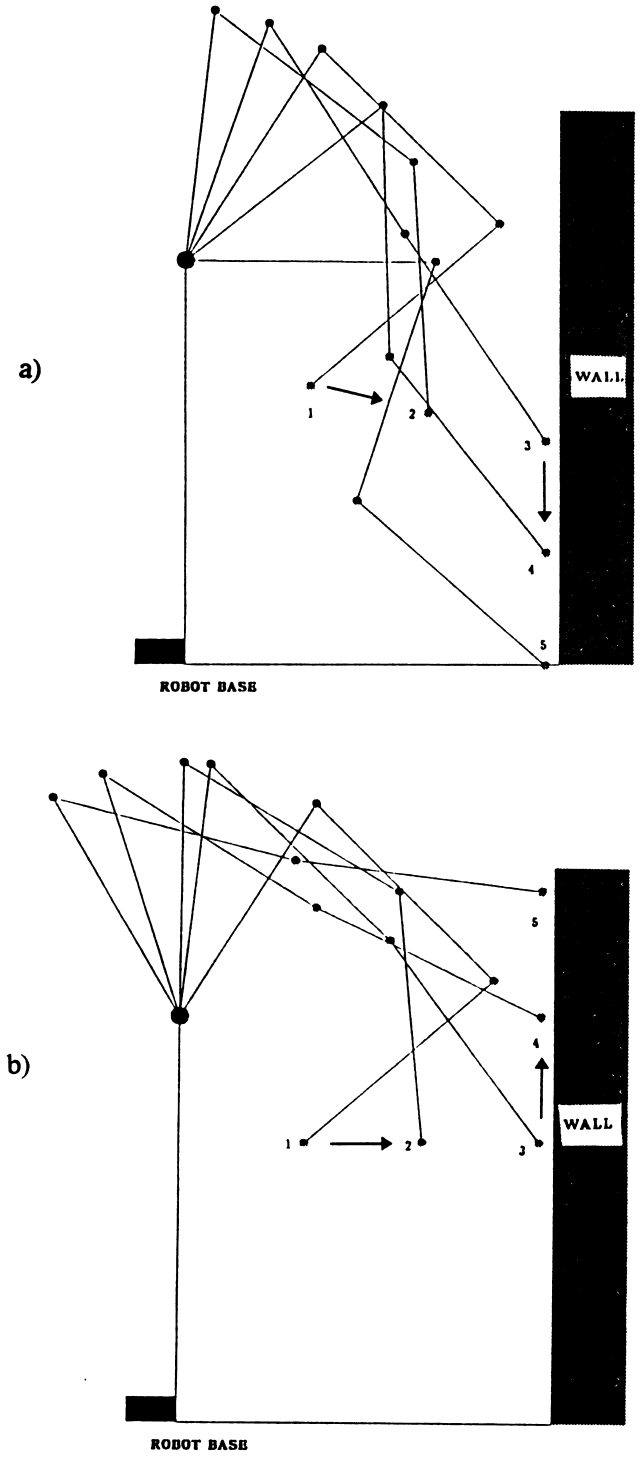

Fig. 8 A single redundant planar arm operating in a restricted workspace: (a) the end effector goes to the bottom of the wall; (b) the end effector goes to the top of the wall

Table 2 CPU times in milliseconds

Number of CCP	Resolved motion	Lagrangian multiplier
One	6	48
Two	7	59
Three	8	64
Four	9	71
Five	11	87

operation. The use of the criteria for manipulators with kinematic constraints (e.g. joint angle limits) also needs investigation.

REFERENCES

1 **Cleary, K.** and **Tesar, D.** Incorporating multiple criteria in the operation of redundant manipulators. In Proceedings of IEEE International Conference on *Robotics and Automation*, Cincinnati, Ohio, 1990, Vol. 1, pp. 618–623.

2 **Yoshikawa, T.** Manipulability of robotics mechanisms. *Int. J. Robotics Res.*, 1985, **4**(2), 3–9.

3 **Liégeois, A.** Automatic supervisory control of configuration and behaviour of multibody mechanisms. *IEEE Trans. Systems, Man, and Cybernetics*, 1977, **SMC-7**(12), 868–871.

4 **Hollerbach, J. M.** and **Suh, K. C.** Redundancy resolution of manipulators through torque optimization. *IEEE Trans. Robotics and Automn*, 1988, **Ra-3**(4), 308–316.

5 **Yoshikawa, T.** Analysis and control of robot manipulators with kinematic redundancy. In Proceedings of IEEE International Conference on *Robotics and Automation*, USA, 1985, pp. 1004–1010.

6 **Latombe, J. C.** *Robot Motion Planning*, 1991 (Kluwer Academic Publishers, Dordrecht).

7 **Chang, P. H.** A closed-form solution for the control of manipulators with kinematic redundancy. In Proceedings of IEEE International Conference on *Robotics and Automation*, San Francisco, California, 1986, pp. 9–14.

8 **De Luca, A.**, **Lanari, L.** and **Oriolo, G.** On-line collision avoidance for redundant manipulators. In Proceedings of IEEE International Conference on *Robotics and Automation*, Nice, France, 1992, pp. 500–506.

9 **Glass, K.**, **Colbough, R.**, **Lim, D.** and **Seraji, H.** On-line collision avoidance for redundant manipulators. In Proceedings of IEEE International Conference on *Robotics and Automation*, Atlanta, Georgia, 1993, pp. 36–43.

10 **Khatib, O.** and **Le Maitre, J. F.** Dynamic control of manipulators operating in complex environments. In Third Symposium on *Theory Practice of Robot Manipulators*, Italy, 1980, pp. 267–282.

11 **Maciejewski, A. A.** and **Klein, C. A.** Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *Int. J. Robotics Res.*, 1985, **4**(3), 109–117.

12 **Sciavicco, L.** and **Siciliano, B.** A solution algorithm to the inverse kinematic problem for redundant manipulators. *IEEE Trans. Robotics and Automn*, 1988, **4**(4), 403–410.

13 **Whitney, D. E.** The mathematics of coordinated control of prosthetic arms and manipulators. *Trans. ASME,*

J. Dynamic Systems, Measmt and Control, 1972, **94**, 303–309.

14 Clein, C. A. and **Chirco, A. I.** Dynamic simulation of a kinematically redundant manipulator system. *J. Robotic Systems*, 1987, **4**(1), 5–23.

15 Sezgin, U. Utilization of manipulator redundancies for collision avoidance strategies. PhD thesis, University of London Library, 1995, pp. 123–170.

16 Klein, C. A. and **Huang, C. H.** Review of pseudoinverse control for use with kinematically redundant manipulators. *IEEE Trans. Systems, Man, and Cybernetics*, 1983, **SMC-13**(3).

17 Baillieul, J. Avoiding obstacles and resolving kinematic redundancy. In Proceedings of IEEE International Conference on *Robotics and Automation*, San Francisco, California, 1986, pp. 1698–1704.