Distributed Feedback Control for Redundant Manipulators based on Virtual Arms

*Toshio TSUJI **Seiya NAKAYAMA and *** Koji ITO

*Faculty of Engineering, Hiroshima University
1-4-1, Kagamiyama, Higashi-Hiroshima, 724 Japan
**Nippon Denso Co.
1-1, Showa-cho, Kariya, 448 Japan
***Toyohashi University of Technology
1-1, Hibarigaoka, Tenpaku-cho, Toyohashi, 441 Japan

Abstract—*The present paper proposes a distributed feedback control for redundant manipulators based on virtual arms. The virtual arm has the same kinematic structure as the manipulator except that its end-point is located on the joint or link of the manipulator. Therefore, the configuration of the redundant manipulator can be represented as a set of virtual arms. Firstly, each subsystem of the distributed control system is designed corresponding to the virtual arm. Dynamic behavior of each subsystem is based on an end-point feedback control of the virtual arm on the operational space. Then, it is shown that the position control of the manipulator utilizing kinematic redundancy can be realized through cooperative and competitive interactions between subsystems, and that the dynamic behavior of the whole system is always stable. The advantages of the method are summarized as follows: 1) each subsystem can work fully autonomously and the joint control torque of the redundant manipulator can be calculated in a parallel and distributed way, 2) the kinematic redundancy of the manipulator can be utilized positively using virtual arms, and 3) some subtasks can be executed by introducing a potential function on the joint space of the manipulator.*

Key Words: Autonomous decentralized system, Redundant manipulator, Distributed control, Trajectory generation, Virtual arm

1. Introduction

A redundant manipulator has more joint degrees of freedom than the one required for a given task. Therefore, it can offer significant advantages, for instance, avoiding obstacles or singular configurations in performing a given task. Several investigators have proposed the inverse kinematic solutions utilizing the manipulator redundancy [*Liegeois, 1977: Hollerbach and Suh, 1984: Vukobratovic and Kircanski, 1984: Yoshikawa, 1984*]. Especially, *Tsutsumi and Matsumoto(1987)* and *Lee and Kil(1990)* proposed trajectory generation methods utilizing the manipulator redundancy using neural networks. The control of the manipulator using the methods described above, however, needs the computation of the joint torques from the derived inverse kinematic solutions.

On the other hand, the manipulator control system usually uses only one computer and controls each joint by time sharing.The larger the number of joint degrees of freedom the manipulator has, the following problems will arise:

1) Flexibility: since the control software must become large and complex, the expansion, revision and maintenance of the control system will become difficult,
2) Reliability: the partial failure of the software or hardware may cause a crucial system breakdown, and
3) Real time computation: it is very difficult to perform highly advanced information processing required for an intelligent robot control by a single computer.

One of the possible approaches to overcome such problems is to construct an autonomous decentralized control system which is consisted of a set of autonomous subsystems. In order to realize such a system for robot control, firstly, we should consider how to design each subsystem which can behave autonomously and cooperate each other.

The present paper proposes a concept of virtual arm as a subsystem. The virtual arm has the same kinematic structure as the controlled manipulator except that its end-effector is located on the joint or link of the manipulator. Providing that the appropriate number of the virtual arms are used, the configuration of the manipulator can be represented by a set of the end-effectors of the

virtual arms. We have already proposed a trajectory generation method using the virtual arms, which is based on a distributed representation of the manipulator's kinematics on the neural networks[*Tsuji, Nakayama and Ito, 1991*]. This method, however, is based on only the kinematics, not the dynamics of the manipulator. In contrast to it, the method presented here can directly compute the joint control torques from the desired position of the virtual end-effectors. Each subsystem can work fully autonomously independent of the others and compute the joint control torques in a parallel and distributed way.

Firstly, we formalize the kinematics of the virtual arms, and then give a detail of the distributed feedback control method. Finally, the effectiveness of the method is verified by computer simulations using a planar manipulator with redundant joint degrees of freedom.

2. Virtual Arm and its Kinematics

We consider a redundant manipulator having m joints (hereafter referred to as an actual arm). Then virtual arm is defined which has an end-effector on a joint or a link of the actual arm. **Figure 1** shows an example of setting virtual arms for a 5-joint planar manipulator. The parameters of the virtual arm such as the base position, joint angle, and link length, are the same as those of the actual arm. Here, (n-1) virtual arms are generally to be set, and the actual arm is regarded as the n-th virtual arm. This allows the configuration of the actual arm to be represented as a set of virtual end-effectors in the task space.

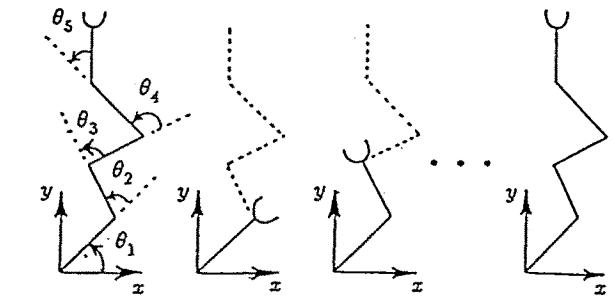
In order to control the redundant manipulator using the virtual arms, the following problems must be solved:

1). planning the desired end-effector position of each virtual arm in the corresponding subsystem independent of the others, and

2). computing the joint control torques of the actual arm from the desired position of the virtual end-effectors.

With respect to the former problem, we have proposed a trajectory planning algorithm for obstacle avoidance and winding control of hyperredundant manipulators [*Tsuji, Nakayama and Ito, 1990: Tsuji, Kaneta and Ito: 1990*]. This paper discuss the latter problem, that is, how to compute the joint control torques corresponding to the planned desired position of the virtual end-effectors.

Let the end-effector position vector of the i-th arm be denoted as $X_v^i \in R^l$ where l is thedimension of the task coordinate system. Let also the joint angle vector of the actual arm be denoted as $\theta \in R^m$. For redundant manipulators, m is larger than l. The forward kinematics of the i-th arm is given by



(a)Actual arm (b)Virtual arms

Fig.1 Examples of virtual arms for a five-joint planar manipulator

$$X_v^i = f^i(\theta), \tag{1}$$

where $f^i(\theta)$ is a nonlinear function. Concatenating (1) for all virtual arms, we can obtain

$$X_v = f(\theta), \tag{2}$$

where $X_v = [X_v^1, X_v^2, \cdots, X_v^n]^T \in R^{ln}$

and $f(\theta) = [f^1(\theta), f^2(\theta), \cdots, f^n(\theta)]^T \in R^{ln}$. (3)

When the desired virtual end-effector position, X_v^*, is given, the inverse kinematic problem of equation (2) can be categorized as the follows [*Tsuji, Nakayama and Ito, 1990*]:

1) redundant case: the desired position vector can be achieved for all virtual end-effectors, and the manipulator still has redundant joint degrees of freedom,

2) over-constrained case: the desired position vector cannot be achieved, since too many virtual arms are used comparing to the joint degrees of freedom of the actual arm, and

3) singular case: the desired position vector cannot be achieved because of the location of the virtual end-effectors. When there is no joint between the virtual end-effectors, for example, the virtual end-effectors cannot be positioned arbitrarily even if the joint degrees of freedom of the actual arm is much larger than the number of the virtual arms. In order to design the control system using the virtual arms, the above three cases should be taken into the consideration.

3. Distributed Feedback Control Based on Virtual Arms

3.1 Control Law

A motion equation of the manipulator is generally given by

$$M(\theta)\ddot{\theta} + h(\theta,\dot{\theta}) + g(\theta) = \tau, \tag{4}$$

where $M(\theta)\in R^{m\times m}$ is the non-singular inertia matrix, $h(\theta,\dot{\theta})\in R^{m}$ is Coriolis and centrifugal term, $g(\theta)\in R^{m}$ is the gravity term and $\tau\in R^{m}$ is joint control torque. In the present paper, we propose the following control law:

$$\tau = \sum_{i=1}^{n} w^i\tilde{\tau}^i, \tag{5}$$

$$\tilde{\tau}^i = \tilde{\delta}^i - \tilde{\lambda}^i\frac{\partial Q(\theta)}{\partial\theta} - \tilde{B}^i\dot{\theta} + \tilde{g}^i(\theta), \tag{6}$$

where w^i is a positive weighting coefficient representing an order of priority for the i-th arm, $\tilde{\tau}^i\in R^m$ is control torque for the i-th arm, $\tilde{\lambda}^i\in R^m$ is a Lagrangian multiplier satisfying

$$w^1\tilde{\lambda}^1 = w^2\tilde{\lambda}^2 = \cdots = w^n\tilde{\lambda}^n \geq 0,\ Q(\theta) > 0$$

is a differentiable potential function, $\tilde{B}^i\in R^{m\times m}$ is a positive definite velocity feedback gain matrix, and $\tilde{g}^i(\theta)$ is joint torque for gravity compensation which satisfies

$$g(\theta) = \sum_{i=1}^{n} w^i\tilde{g}^i(\theta)$$

In this paper, it is assumed that the gravity compensation torque, $\tilde{g}^i(\theta)$, can be computed in each subsystem.

Also, $\tilde{\delta}^i\in R^m$ is defined as

$$\tilde{\delta}^i = (J_v^i)^T(X_v^{i*} - X_v^i), \tag{7}$$

where $X_v^{i*}\in R^l$ is the desired position of the i-th virtual end-effector, and $J_v^i = \partial X_v^i/\partial\theta \in R^{l\times m}$ is the Jacobian matrix of the i-th virtual arm. **Figure** 2 shows a block diagram of the control system. The weighted sum τ of joint control torque for each virtual arm $\tilde{\tau}^i$ is the control torque for the actual arm.

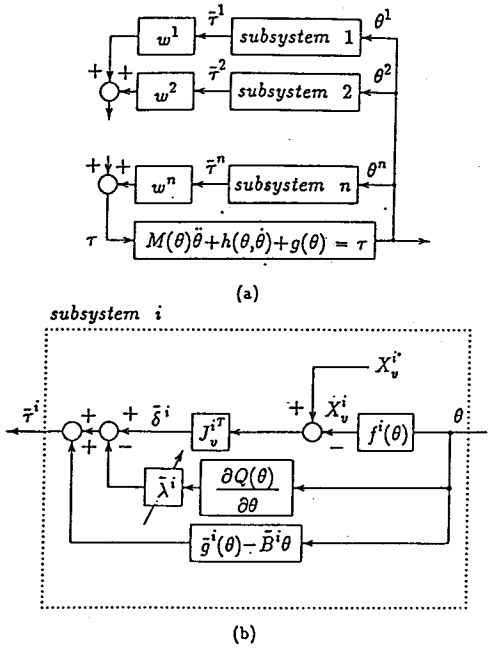


Fig.2 Distributed feedback control based on virtual arms

Next, we consider to set an order of priority for each degree of freedom of the virtual end-effector. Introducing another weighting matrix $W^i\in R^{l\times l}$ for the i-th virtual end-effector, the control law is revised as the follows,

$$\tau = \sum_{i=1}^{n} \tau^i, \tag{8}$$

$$\tau^i = \delta^i - \lambda\frac{\partial Q(\theta)}{\partial\theta} - B^i\dot{\theta} + g^i(\theta), \tag{9}$$

$$\delta^i = (J_v^i)^T W^i(X_v^{i*} - X_v^i), \tag{10}$$

where we define

$\lambda \equiv w^i\tilde{\lambda}^i, i=1,2,\cdots,n, g^i(\theta) \equiv w^i\tilde{g}^i(\theta), B^i \equiv w^i\tilde{B}^i$

and $W^i = \mathrm{diag.}[w_1^i, w_2^i, \cdots, w_l^i]$. w_j^i is a positive weighting coefficient for the j-th element of the i-th virtual arm.

3.2 Stability

Now, the following energy function is defined.

$$H = V + E + n\lambda Q(\theta) \tag{11}$$

$$V = \frac{1}{2}\dot{\theta}^T M(\theta)\dot{\theta} \tag{12}$$

$$E = \frac{1}{2}\sum_{k=1}^{n} (X_v^{k*} - X_v^k)^T W^k(X_v^{k*} - X_v^k) \tag{13}$$

V is the kinetic energy of the manipulator, E is the squared position error between the desired and the virtual end-effector, and Q(θ) is a differentiable potential function. The time derivative of the energy function is given by

$$\dot{H} = \frac{1}{2}\dot{\theta}^T \dot{M}(\theta)\dot{\theta} + \dot{\theta}^T M(\theta)\ddot{\theta} + \dot{\theta}^T \{-\frac{\partial E}{\partial \theta} + n\lambda\frac{\partial Q(\theta)}{\partial \theta}\} + nQ(\theta)\dot{\lambda}. \quad (14)$$

On the other hand, substituting equations (8) and (9) into (4), we can see

$$M(\theta)\ddot{\theta} = -h(\theta,\dot{\theta}) + \delta - n\lambda\frac{\partial Q(\theta)}{\partial \theta} - B\dot{\theta}, \quad (15)$$

where we define

$$\delta = \sum_{i=1}^{n} \delta^i \text{ and } B = \sum_{i=1}^{n} B^i$$

Substituting equation (15) into (14) and using the relation $\dot{\theta}^T \dot{M}(\theta)\dot{\theta} = 2\dot{\theta}^T h(\theta,\dot{\theta})$, we can obtain

$$\dot{H} = -\dot{\theta}^T B\dot{\theta} + nQ(\theta)\dot{\lambda}. \quad (16)$$

Then, we introduce a state variable s and define

$$\lambda = g(s), \quad (17)$$

where g(s) is a differentiable and monotonically increasing function, such as,

$$g(s) = \frac{g_{max}}{1+e^{-s}} \geq 0. \quad (18)$$

gmax is a positive constant. Also we define the time derivative of the state variable as

$$\dot{s} = -\alpha\frac{|\delta|}{Q(\theta)g'(s)} < 0, \quad (19)$$

where α is a positive constant, $|\delta|$ denotes the Euclidean norm of vector δ and g'(s)=dg(s)/ds<0 . From (16), (17) and (19), we have

$$\dot{H} = -\dot{\theta}^T B\dot{\theta} - n\alpha|\delta| \leq 0. \quad (20)$$

Since B is positive definite and nα > 0, we can see that the system is asymptotically stable and the energy function H decreases monotonically until $\dot{\theta} = 0$ and $\delta = 0$, that is the equilibrium point of the system.

3.3 Kinematic Meaning of Equilibrium Point

In this section, the kinematic meaning of the equilibrium point is analyzed. At the equilibrium point, error vector δ can be expressed as

$$\delta = J_v^T W(X_v^* - X_v(t_f)), \quad (21)$$

where $X_v(t_f) \in R^{ln}$ is the concatenated position vector of the virtual end-effectors at the equilibrium point,

$$J_v = [(J_v^1)^T,(J_v^2)^T,\cdots,(J_v^n)^T]^T \in R^{ln\times m}$$

is the concatenated Jacobian and

$$W = \text{block diag.}[W^1,W^2,\cdots,W^n] \in R^{ln\times ln}$$

is a block diagonal weighting matrix. Note that δ=0 at the equilibrium point. Using a displacement vector dX_v, we have

$$J_v^T W dX_v^* = J_v^T W dX_v, \quad (22)$$

where

$$dX_v^* = X_v^* - X_v(0), dX_v = X_v(t_f) - X_v(0) \in R^{ln}$$

and $X_v(0)$ denotes the initial position of the virtual end-effectors. Consequently, at the equilibrium point, the virtual end-effectors must exist at the points which are solutions of the simultaneous equation (22). The rank of the concatenated Jacobian matrix J_v dominates the simultaneous equation as the follows:

[**case 1**]: If rank(J_v)=ln, the rank of the matrix $J_v^T W$ is also ln, since the matrix W is positive definite. Therefore the unique solution of (22) is given by

$$dX_v = dX_v^*. \quad (23)$$

This guarantees the convergence of each virtual end-effector to the corresponding desired position as long as the entire arm is kinematically redundant. Moreover, it is expected that the final configuration of the manipulator minimizes the potential function Q(θ) by the effect of the third term of equation (14).

[**case 2**]: If rank(J_v)<ln,the simultaneous equation (22) becomes indefinite. The general solution of (22) is given by [*Tsuji, Nakayama and Ito, 1990*]

$$dX_v = (J_v^T W)^+ J_v^T W dX_v^* + [I_{ln} - (J_v^T W)^+(J_v^T W)]z_{ln}, \quad (24)$$

where $I_{ln} \in R^{ln\times ln}$ is a unit matrix, $z_{ln} \in R^{ln}$ is an arbitrary vector and $(\,)^+$ denotes the pseudo inverse matrix. This indicates that when the manipulator is over-constrained or singular, the virtual end-effectors converge to the position where the weighting sum of positioning error becomes minimum. Moreover, the final configuration of the manipulator minimizes the potential function Q(θ) among the general solutions of (24) by the effect of the third term of equation (14).

The distributed feedback control presented here can positively utilize the redundant joint degrees of freedom in the task space and compute the joint control torque in a parallel and distributed way. In the next section, the effectiveness of this method will be verified by computer simulations.

4. Simulation Experiments

4.1 Roles of Potential Function Q(θ)

Firstly, computer simulations were carried out for a three-joint planar manipulator without use of the virtual arm (l=2, m=3, n=1). Note that the manipulator is redundant. **Figure 3** shows examples of simulation results. The weighting matrix W^1 and the velocity feedback gain matrix B^1(see equation (10)) were set as

W^1=diag.[300.0, 100.0](N/m)

and

B^1=diag.[120.0, 160.0, 7.0](Nm/(rad/sec)),

respectively. The other parameters used in the simulations are shown in the figure. We used the Appel method for the manipulator dynamics [*Potknjak and Vukobratovic, 1978*] and the link parameters of the manipulator are shown in **Table 1.**

Table 1 Link parameters of a three-joint planar manipulator

	link1	*link2*	*link3*
length(m)	0.8	1.5	0.5
mass(kg)	1.0	1.2	0.5
center of mass(m)	0.4	0.75	0.25
moment of inertia($kg \cdot m^2$)	0.053333	0.225	0.0104166

In **Figure 3**, the initial and desired positions of the end-effector were set as $X_v(0) = [1.0, 0.0]^T$(m) and $X_v^* = [0.0, 1.5]^T$(m), respectively, and the potential functions were set as

$$Q_1(\theta) = \varepsilon_1, \tag{25}$$

in **Figure 3(a)**, and

$$Q_2(\theta) = \frac{1}{2}\{\theta(0) - \theta(t)\}^T\{\theta(0) - \theta(t)\} + \varepsilon_2, \tag{26}$$

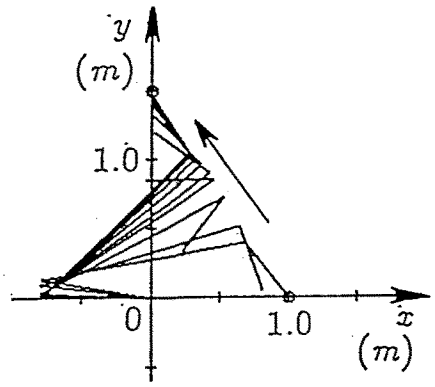
in **Figure 3(b)**, where $\varepsilon_1 = \varepsilon_2 = 1.0$.

Since $\partial Q_1(\theta)/\partial\theta = 0$, **Figure 3(a)** is equivalent to the feedback control method by *Takegaki and Arimoto(1981)*. On the other hand, **Figure 3(c)** shows a simulation result using the manipulability measure [*Yoshikawa, 1984*] as the potential function,

$$Q_3(\theta) = \sqrt{\det J_v J_v^T} + \varepsilon_3, \tag{27}$$

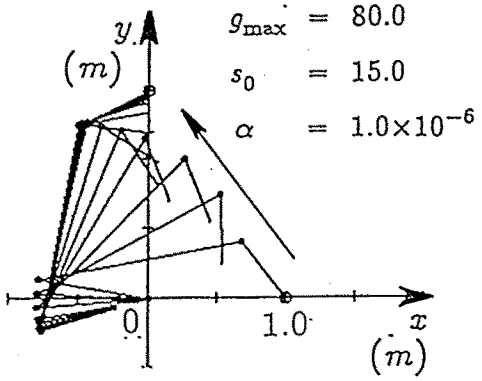
where ε_3=1.0. In this case, the potential function was maximized by changing the sign of the term, $\partial Q(\theta)/\partial\theta$, in equations (6) or (9).

In **Fig.3(a)**, the third joint rotated largely. As a result, the second and third links were overlapped. On the other hand, in **Fig.3(b)** and **(c)**, the end-effector reached to the goal point without such a overlapping, because the large rotation of some specific joints resulted in increasing the squared sum of the joint displacement of equation (26) and also the manipulability measure of equation (27) decreased near $\theta = -\pi$(rad), that is, at the po-



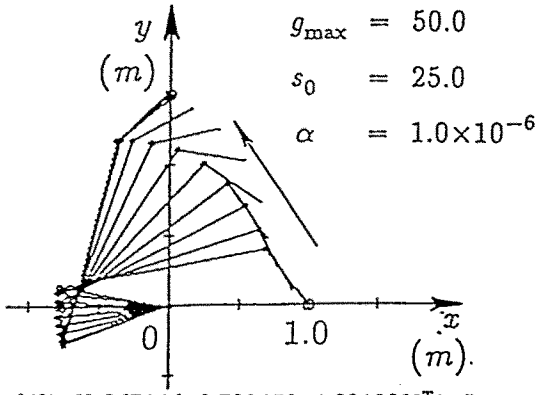
*initial posture*θ(0)=[2.967146,-2.792473,-1.091282]^T(rad)
final posture θ(t_f)=[3.163653,-2.307091,-4.988992]^T(rad)

(a)



*initial posture*θ(0)=[2.967146,-2.792473,-1.091282]^T(rad)
final posture θ(t_f)=[3.490397,-2.170737,-0.612733]^T(rad)

(b)

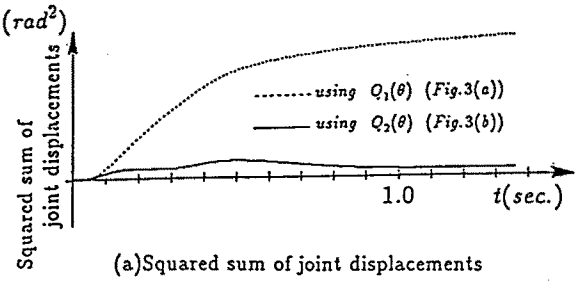


*initial posture*θ(0)=[2.967146,-2.792473,-1.091282]^T(rad)
final posture θ(t_f)=[3.406421,-2.058269,-0.852928]^T(rad)

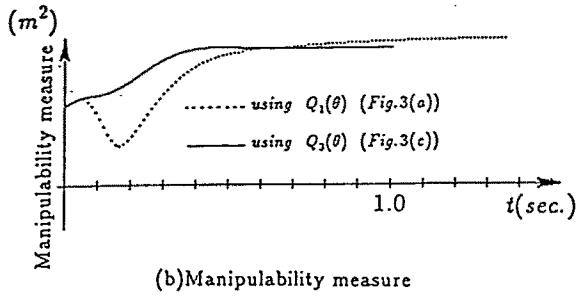
(c)

Fig.3 Roles of potential function Q(θ)

sition overlapping of the second and third links. **Figure 4** shows the time changes of the squared sum of the joint displacement and the manipulability measure (the first term on the right hand side of equations (26) and (27), respectively).



(a)Squared sum of joint displacements



(b)Manipulability measure

Fig.4 Time changes of squared sum of joint displacements and manipulability

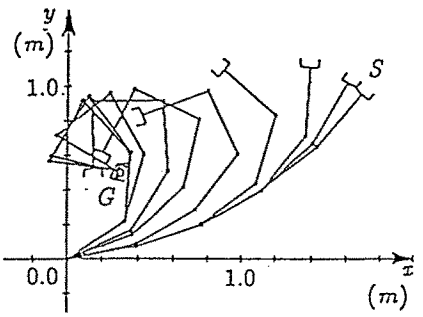
4.2 Role of Virtual Arms

Table 2 shows the link model of the manipulator used in simulations. The following parameters were used: g_{max}= 50 (see equation (8)), the initial value of the state variable s, s_0 = 0.1, $\alpha = 10^{-7}g'(s)$. **Figure 5(a)** shows a simulation result using a virtual arm (n=2), the end-effector of which was on the third joint of the manipulator. Both the desired positions of the actual and virtual end-effectors were given as the same goal as shown in the figure. From **Figure 5(a)**, the virtual end-effector as well as the actual end-effector reached to the goal point.

In **Figure 5(b)**, we used nine virtual arms (n=10), the end-effectors of which were on a middle point of each link and on each joint except the first joint. The desired position for the actual arm was given by the goal point shown in the figure, and each desired position for virtual end-effector was set to the corresponding initial position. Although the virtual end-effectors tried to keep their initial positions, the virtual end-effectors moved to the directions of the goal point, since they were pulled by the actual end-effector as seen in the figure.

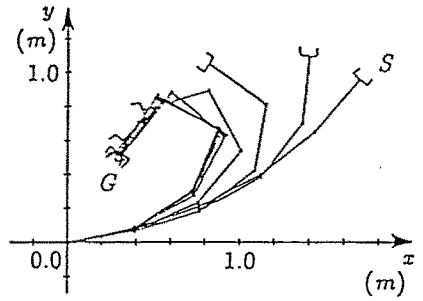
Table 2 Link parameters of a five-joint planar manipulator

	link i $(i=1,\cdots,5)$
length(m)	0.4
mass(kg)	1.0
center of mass(m)	0.2
moment of inertia$(kg\cdot m^2)$	0.013333



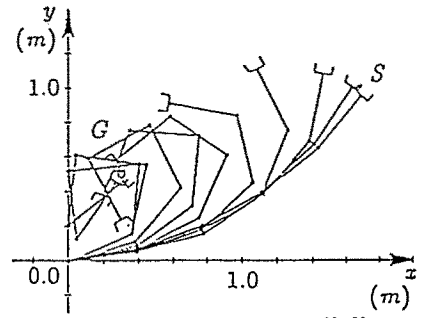
$W = diag.[20.0,20.0,20.0,20.0](N/m)$
$B^i = diag.[22.0,13.0,6.0,2.5,1.5](Nm/(rad/s))$

(a)



$W = diag.[1.0,\cdots,1.0,40.0,40.0] \in R^{20\times 20}(N/m)$
$B^i = diag.[5.0,3.0,1.4,0.5,0.3](Nm/(rad/s))$

(b)



$W = diag.[5.0,\cdots,5.0,40.0,40.0] \in R^{20\times 20}(N/m)$
$B^i = diag.[4.4,2.6,1.2,0.5,0.3](Nm/(rad/s))$

(c)

Fig.5 Simulation results of the distributed feedback control

On the other hand, in **Figure 5(c)**, the same virtual arms as **Figure 5(b)** except for their desired positions of the end-effector were used. The desired end-effector position of each virtual arm was given by the position of the first joint of the manipulator. While the actual end-effector reached to the goal point, the virtual end-effectors moved to the direction of the first joint of the manipulator. Consequently, it can be seen that the configuration as well as the end-effector of the actual arm can be controlled by planning the desired positions for the virtual end-effectors in the task space.

5. Concluding Remarks

In this paper, we proposed the distributed feedback control for the redundant manipulator based on virtual arms. The advantages of our method are summarized as follows:

1) each subsystem can work fully autonomously,
2) the joint control torque of the redundant manipulator can be calculated in a parallel and distributed way,
3) the kinematic redundancy of the manipulator can be utilized positively using virtual arms, and
4) some subtasks can be executed by introducing the potential function on the joint space of the manipulator utilizing the kinematic redundancy.

Further research will be directed to apply the control method presented here to the multiple manipulator systems.

Acknowledgment

The work presented in this paper was supported by the scientific research fund from the Ministry of Education, Science and Culture of Japan under Grant "Distributed Autonomous System" (#03234105).

References

[1] A.Liegeois, "Automatic Supervisory Control of Configuration and Behavior of Multibody Mechanisms," IEEE Trans. Systems, Man and Cybernetics, SMC7-12,861/871 (1977).

[2] J.M.Hollerbach and K.C.Suh, "Redundancy Resolution of Manipulators through Torque Optimization," IEEE J.of Robotics and Automation, RA3-4, 308/314(1984).

[3] M.Vukobratovic and M.Kircanski, "A Dynamic Approach to Normal Trajectory Synthesis for Redundant Manipulators," IEEE Trans. Systems, Man and Cybernetics, SMC14-4, 580/586(1984).

[4] T.Yoshikawa, "Analysis and Control of Robot Manipulators with Redundancy," Robotic Research, The First International Symposium, eds. M.Brady and R.Paul, MIT Press, Cambridge, Mass. 735/747 (1984).

[5] K.Tsutsumi and H.Matsumoto, "Neural Computation and Learning Strategy for Manipulator Position Control," Proc. of IEEE 1st ICNN, 525/534(1987).

[6] S.Lee and R.M.Kil, "Robot Kinematic Control Based on Bidirectional Mapping Neural Network," Proc. of IJCNN, 3, 327/335(1990).

[7] T.Tsuji, S.Nakayama and K.Ito, "Trajectory Generation for Redundant Manipulators Using Virtual Arms," Proc. of ICARCV, 554/558(1990).

[8] T.Tsuji, J.Kaneta and K.Ito, "A Hierarchical Collision-Free Path Planning for Redundant Manipulators Based on Virtual Arms," Proc. of IEEE International Workshop on Intelli-gent Motion Control, 1, 301/306(1990).

[9] M.Takegaki and S.Arimoto, "A New Feedback Method for Dynamic Control of Manipulators," Trans. of ASME J. of Dynamic Systems, Measurement and Control, 103, 119/125(1981).

[10] V.Potknjak and M.Vukobratovic, "Two new methods for Computer Forming of Dynamic Equation of Active Mechanisms," Mechanism and Machine Theory, 14-3, 189-200(1987).