《研究論文》

Journal of the Human Engineering Society of Korea
Vol.7 No. 2, Dec. 1988

지적 원격조작시스템의 수동모드 개선을 위한 기하학적 해석에 관한 연구

(A Study on a Geometrical Analysis for the Manual Mode of an Advanced Teleoperator System)

이순요* 김창대* 박세권*

Abstract

If an error occurs in the automatic mode when the advanced teleoperator system performs a task in hostile environment, then the automatic mode changes into the manual mode. The operation by the control program and the operation by a human recover the error in the manual mode. The system resumes the automatic mode and continues the given task. It is necessary to improve the manual mode in order to make the best use of a man-robot system, as a part of the human interface technique.

Therefore, the error recovery task is performed by combining the operation by the control program representing autonomy of a robot and the operation by a human representing versatility of a human operator effectively in the view point of human factors engineering. The geometric inverse kinematics is used for the calculation of the robot joint values in the operation by the control program.

The singularity operation error and the parameter operation error often occur in this procedure. These two operation errors increase the movement time of the robot and the coordinate reading time, during the error recovery task. A singularity algorithm, parameter algorithm and fuzzy control are studied so as to remove the disadvantages of geometric inverse kinematics. And the geometric straight line motion is studied so as to improve the disadvantages of the operation by a human.

1. 서 론

극한작업용의 인간-로보트 시스템에 있어서 로보트의 환경적응능력을 최대한으로 살리면서 안전한 원격지점으로부터 인간조작자의 거시적인 판단과 조작에 의해 인간과 로보트가 단독으로서는 할 수 없는 작업을 협력하여 효율적으로 실시하는 원격조작기술(teleoperation)이 제안되었다〔1〕. 더우기 극한작업용의 인간-로보트시스템을 실현하기 위한 human in-

*고려대학교 산업공학과

terface 기술에서, 지적 원격조작기술은 원격조작작업에 컴퓨터를 도입하여 인간의 부담을 경감시키고 작업의 고도화를 실현하는 방법과 전자동로보트시스템에 인간의 원조를 도입해 행동감시와 에러회복을 실시하는 방법으로 기존의 연구에서 제안되었다[2,3].

본 연구에서는 극한 작업환경이나 불확실한 상황하에서의 작업을 보다 지능적으로 수행하기 위하여 자동모드에 의한 작업의 수행도중 에러가 발생했을 때 자동모드에서 수동모드로 변환하여 에러를 회복하는 인간-로보트시스템의 유용성을 향상시키기 위하여 수동모드를 개선하고자 한다.

수동모드에서의 에러회복작업은 프로그램에 의한 코맨드조작과 인간조작자에 의한 직접조작에 의해 수행되어진다. 코맨드조작의 위치제어시 로보트의 관절값계산은 입력된 작업물체의 위치정보에 의해 geometric inverse kinematics로 계산한다. 이 때 발생하는 singularity 조작에러와 parameter 조작에러에 대해 algorithm을 설정하여 두 조작에러를 해결한다. 또한 자세제어에 fuzzy control을 적용하여 자세판독시 인간의 판단에 내재되는 모호성을 해결함으로써 자세판독시간을 감소시키며 X, Y 좌표에 의한 방향과 자세를 함께 고려하여 코맨드조작의 성능향상을 도모한다.

직접조작은 end point로부터 인간조작자의 직접조작에 의해 최종적으로 에러를 회복하는 것이다. 직접조작의 위치제어는 end point로부터 작업물체까지의 직선운동을 가능케 하는 geometric straight line motion을 이용하여 최소한의 직접조작으로 에러를 회복케 하고, 직접조작의 자세제어는 로보트 관절운동의 단위증분을 이용하여 사용한 UC(micro motion control)에 의하여 행하도록 한다[4, 5].

그림 1은 위와 같은 연구진행의 흐름을 나타내고 있다.

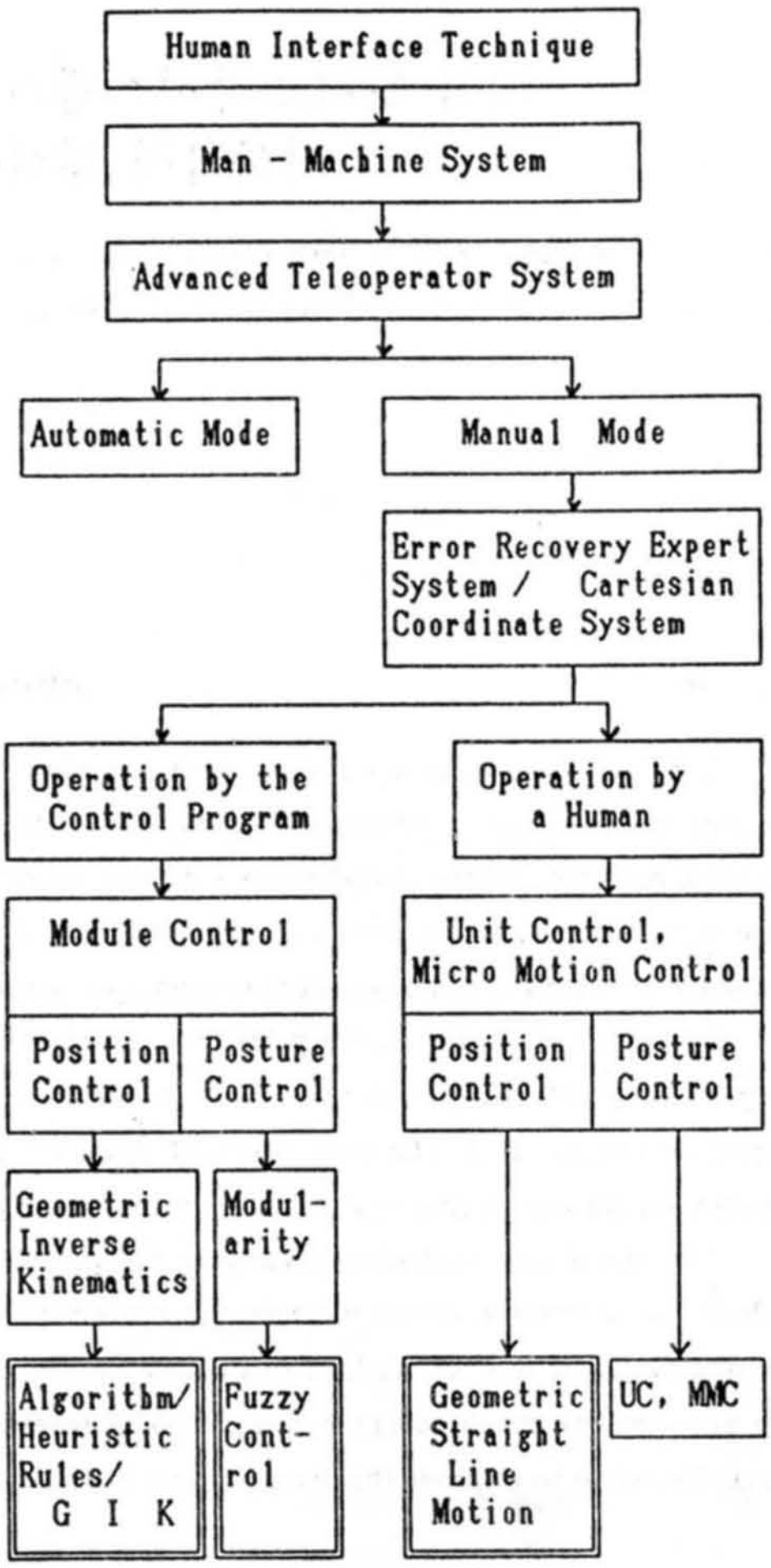


Fig 1. Schematic representation of this study

2. 수동모드의 개선

2.1. 수동모드 개선의 의의

지적 원격조작시스템의 자동모드에서 에러가

발생하면 수동모드로 넘어가, 코맨드조작에 의해 직교좌표계의 입력자료에 따라 관절값을 계산하여 end point로 로보트를 이동시킨다.

그림2와 같이 ERES/CCS는 로보트베이스를 기준으로 일반적으로 사용되고 있는 직교좌표계를 이용하여 직교좌표상의 위치에 따라 end point에 해당하는 좌표의 x, y, z값 및 원하는 pitch 값을 입력하면 그 입력자료를 이용하여 $\theta 1$을 구하며, 이는 그림2에서 end effector의 끝 점이 작업장 바닥면의 px, py일 때 $\theta 1$은 다음과 같이 구해진다. a2, a3, AC에 의해 결정되는 삼각형에 의해 $\theta 2$, $\theta 3$를 구하며 마지막으로 $\theta 4$를 구하게 된다.

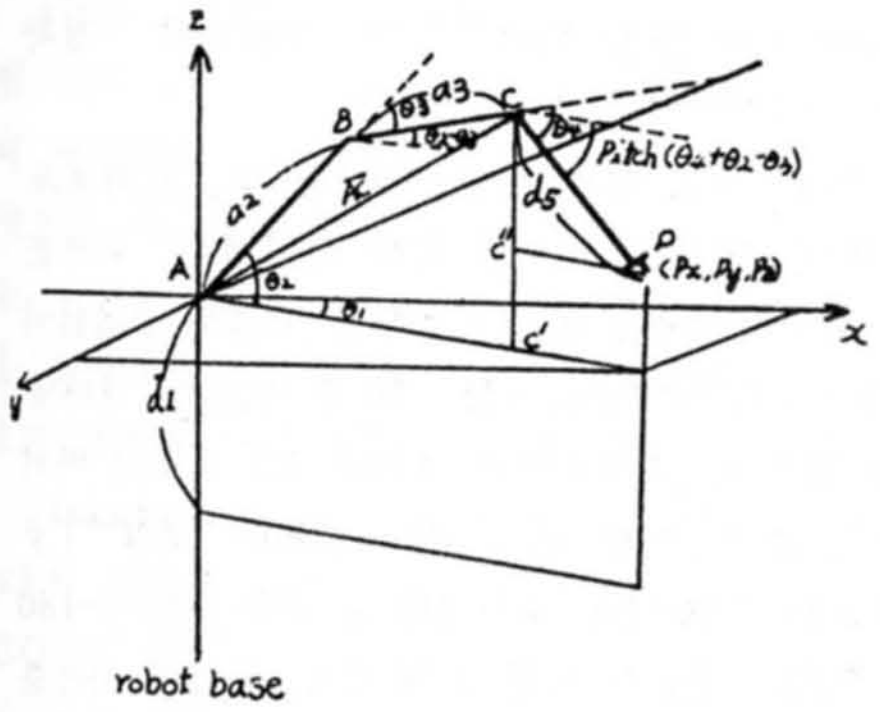


Fig 2. Cartesian Coordinate System of the ERES/CCS

코맨드조작의 위치제어에 있어 geometric inverse kinematics에 의해 계산되는 관절값 중 $\theta 2$와 $\theta 3$의 계산에서 arccosine이 이용되며, 따라서 COS(t)에서 t값이 1을 초과하지 않아야 한다. 만약 t값이 절대치 1을 초과하게 되면 계산상의 에러가 발생하여 좌표를 재판독하여 입력하여야 한다. 이 때 t값이 절대치 1이 되는 경우는 a2, a3, $\overline{AC}$가 형성하는 삼각형이 일직선이 되는 경우이다. 다시 말해서 두 변의 길이의 합이 한 변의 길이와 같아지는 경우에 singularity가 발생하며, a3와 end effecter가 일직선이 되도록 pitch 각을 주었을 때 발생하는 singularity를 workspace boundary singularity라 한다. Singularity에서 pitch 각에 따라 singular point는 boundary singularity 안쪽에 나타나며, 이와 같이 작업내부에 형성된 singularity를 workspace interior singularity라 한다〔5, 7〕.

Geometric inverse kinematics에 의해 계산된 로보트조인트들의 관절값이 로보트 각 관절에 주어져 있는 parameter 제한범위를 벗어나게 되면 에러가 발생하게 된다. 따라서 에러를 수정한 후 제한범위 속에 들어가는 입력자료를 가질 때까지 반복작업을 하게 된다〔8〕.

본 연구에서 사용하는 ROB-501 로보트에 대한 각 관절의 parameter 제한범위를 나타낸 것이 그림3 이다.

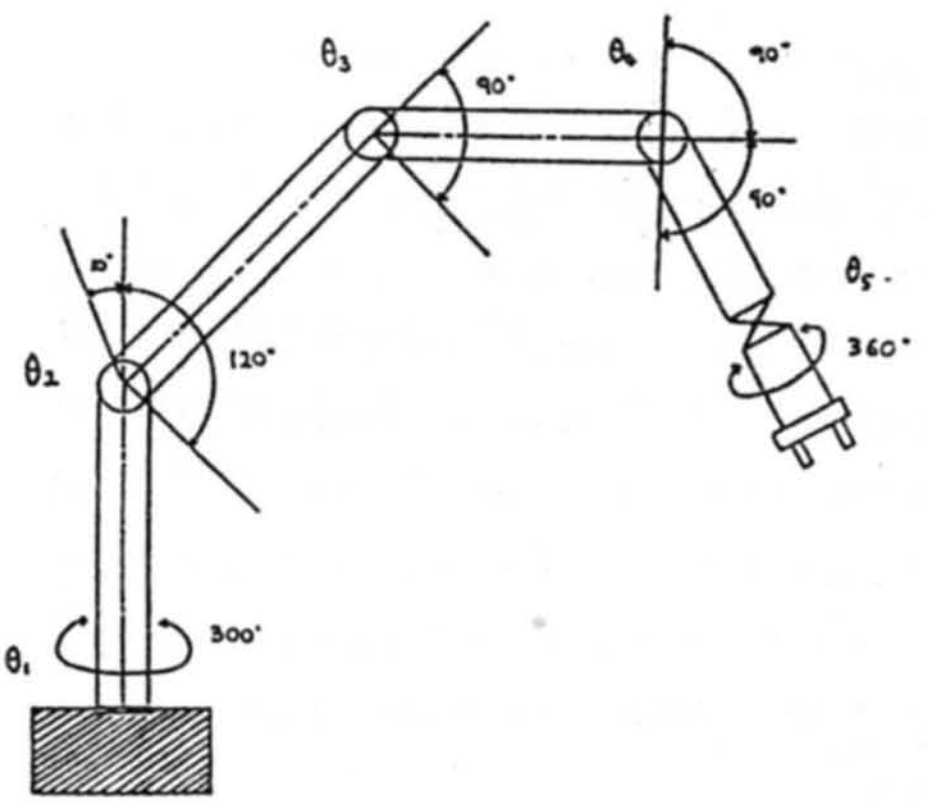
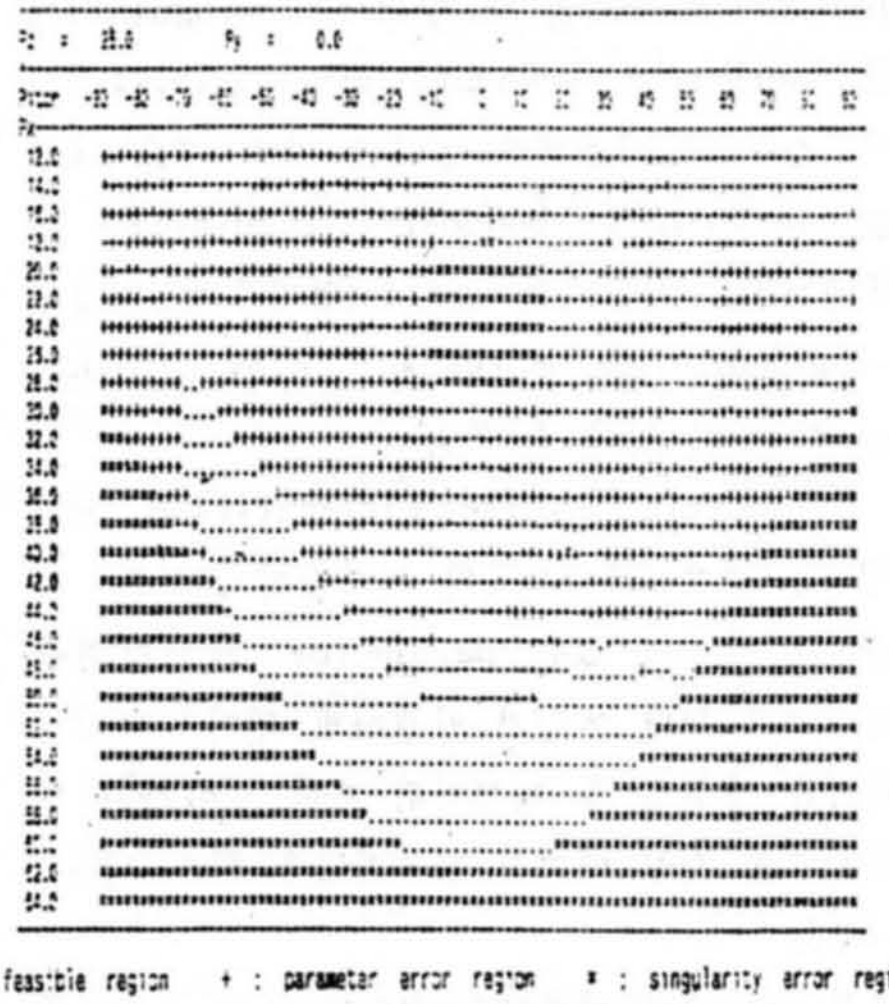


Fig 3. Parameter Limitation of the ROB-501 Robot Joint

이와 같은 singularity 조작에러와 para-

meter 조작에러를 고정된 y, z 값에 대해 x, pitch 각의 입력자료에 따라 살펴보면 그림 4와 같다.



. : feasible region + : parameter error region * : singularity error region

Fig 4. Singularity Operation Error and Parameter Operation Error Range

로보트가 도달할 수 없는 거리에 위치가 존재하는 singularity 조작에러와 로보트에 주어진 parameter 제한조건의 범위를 벗어나는 관절값에 대한 parameter 조작에러를 해결할 수 있는 algorithm을 기하학적 해석에 의해 설정하며 이를 heuristic rules에 보완한다. 기존의 heuristic rule의 보완에 의한 코맨드조작에 비해 algorithm의 보완에 의한 heuristic rules에 의한 코맨드조작은 입력된 위치와 로보트 자동사이의 정확도를 향상시킨다.

코맨드조작의 위치제어 개선과 직접조작의 위치제어 개선은 각 각의 제어에 대한 기하학적 해석에 의해 이루어지나 코맨드조작의 위치제어 개선은 fuzzy control 에 의한 개선으로 수동모드 개선의 의의에서 간략히 언급하고자 한다.

Geometric inverse kinematics에 직교좌표의 위치를 입력하기 위하여 모니터에 나타나는 극한작업장에서 로보트베이스로부터 작업물체에 대한 좌표(x, y, z) 및 자세를 판독하여야 한다. 그러나, 좌표판독은 공간상의 직교좌표계에서 기상의 X, Y, Z 축을 형성하여, 이 세 축으로부터 좌표를 판독하게 된다. 이 가상의 축으로부터 좌표를 판독하기란 용이하지 않으며, 따라서 좌표를 판독하는 데 많은 시간이 소요되며, 정확한 좌표를 추정하기도 곤란하다. 이러한 물체자세에 대한 판독의 어려움을 모호한 linguistic variable을 입력케 함으로써 자세판독의 어려움을 해소하며 로보트베이스에 대한 방향을 함께 고려하여 정확한 자세제어가 되도록 한다.

자세제어를 위한 방향($\theta 1$)을 x, y 좌표로 구하면, 먼저 x, y에 의해 형성되는 4사분면에서 1사분면과 2사분면은 부호에 상관없이 $\theta 1 = (TAN^{-1}(x/y))/30$의 식으로 나타낼 수 있으나, 3사분면과 4사분면은 부호가 바뀌므로 3사분면일 때는 $\theta 1 = (180 - TAN^{-1}(y/x))/30$이며 4사분면 일 때는 $\theta 1 = (-180 - TAN^{-1}(y/-x))/30$이다. 방향과 자세를 함께 고려함으로써 정확한 자세제어에 따라 직접조작도 줄여줄 수 있게 된다.

기존의 직접조작에 사용되는 UC와 MMC는 단위 증분식 제어방식으로 키보드상에 각각의 function key를 할당해서 어느 하나를 누름에 따라 로보트축을 일정한 양만큼씩 움직이게 할 수 있으며, 위치제어와 자세제어의 구분없이 사용하였다. 따라서 물체를 잡기 위해서는 로보트 각 축의 움직임과 그 상호관계를 적절히 조절하여 조작하여야 하므로 시간과 조작량이 많이 소요된다. 이러한 직접조작의 문제점을 해결하기 위하여 기존의 cartesian straight line motion을 기하학적으로 해석한 geomet-

ric straight line motion 을 직접조작의 위치제어에 적용함으로써 직접조작시간을 감소시킨다〔10〕.

2.2. 코맨드조작의 위치제어 개선

2.2.1. 조작에러의 기하학적 해석에 의한 algorithm 설정

singularity 조작에러와 parameter 조작에러는 기하학적인 해석에 의해 살펴볼 수 있으며 이를 이용한 algorithm을 heuristic rules에 보완함으로써 heuristic rules에 대한 이해와 RMS 오차를 줄이는 정확도를 향상시킨다.

(1) Singularity 조작에러에 대한 기하학적 해석에 의한 algorithm 설정

singularity 조작에러가 발생하는 것은 그림 2의 a2, a3, $\overline{AC}$의 세 가지 요인에 의해 결정되며 a2, a3 는 로보트에 주어진 특성으로 $\overline{AC}$ 만이 입력자료에 의해 바뀔 수 있는 변수로 singularity 에 대한 문제는 $\overline{AC}$를 고려하여 해결할 수 있으며, 정확한 입력자료의 조절에 따른 RMS 오차를 줄일 수 있다.

singularity 조작에러는 먼저, $\overline{AC} > a2 + a3$인 경우에 대해 살펴보면 $\theta 2$의 계산중, $COS^{-1}(\overline{AC}^2 + a2^2 - a3^2) / (2\overline{AC}\ a2)$ 에서 분자 $\overline{AC}^2 + a2^2 - a3^2$은 조건 $\overline{AC} > a2 + a3$을 제곱하여 이항하면 $\overline{AC}^2 + a2^2 - a3^2 > 2\overline{AC}\ a2$를 만족하며, 따라서 분자는 분모보다 큰 값으로 절대치 1을 초과하게 되어 에러가 발생한다. $a2 - a3 > \overline{AC}$ 에 대해서도 같은 방법으로 살펴볼 수 있다〔8〕.

위의 증명으로부터 singularity 조작에러가 발생하지 않기 위해서는 다음 두 조건 $\overline{AC} \leq a2 + a3$ 와 $a2 - a3 \leq \overline{AC}$를 만족하여야 하며 이는 다음 표1의 singularity test를 통해 살펴볼 수 있다.

singlarity 조작에러가 발생하지 않기 위해서는 주어진 입력자료가 윗 식을 만족하여야 하며 따라서 입력된 자료치를 이용하여 세 요인에 대한 나머지 한 요인의 제한범위를 산출해낼 수 있다.

먼저, 첫번째 식으로부터 px, py, pz 에 대한 각 각의 상한치를 산출해낼 수 있으며, 두번째 식으로부터 각 각의 하한치를 산출해낼수 있다. 이를 나타내면 표 2와 같다.

위치자료를 유지하며 pitch 값의 변화에 의해 singularity 조작에러를 해결할 수가 있으며, 이는 입력된 위치자료가 pitch값보다 더 중요한 요소로 작용할 경우에 해당한다.

$\overline{AC} \leq a2 + a3$ 경우와 $a2 - a3 \leq \overline{AC}$ 경우에 대해 singlarity 조작에러가 발생하지 않기 위해서는 주어진 pitch 값이 윗 식을 만족하여야 하며 따라서 입력된 자료치를 이용하여 pitch 값의 조절범위를 산출해낼 수 있다. 첫번째 식으로부터 pitch 값의 상한치를 산출해내며, 두 번째 식으로부터 하한치를 산출해낼 수 있다. 이를 나타내면 표 3과 같다. 단, 이는 작업물체의 고도가 로보트베이스와 일치할 때이다.

$\theta 1$의 경우, 초기자세를 기준으로 좌우 150° 회전이 가능하며, 따라서 입력자료인 x, y가 $x > 2y$ 나 $x > -2y$ 중 한 조건을 만족하여야 한다. 그렇지 못하면 계산된 관절값이 제한범위를 벗어나 에러가 발생하는 경우가 생기며, 이는 px의 값이 음수일 경우에만 해당된다.

(2) Parameter 조작에러의 기하학적 해석에 의한 algorithm 설정

로보트 각 관절에 주어진 parameter 제한범위에 대한 기하학적인 해석에 의해 parameter 조작에러를 해결해낼 수 있으며, 더우기 singularity 조작에러와 마찬가지로 pitch 값의 조절이 가능하여 제한범위까지의 이동뿐만 아니라 원하는 위치정보까지 움직이게 하여 RMS 오차를 줄인다.

그림 2에서와 같이 a2, a3, $\overline{AC}$ 에 의해

형성된 삼각형에 의해 $\theta 2$, $\theta 3$를 구할 수 있으며, 그 값이 그림 4의 제한범위 속에 포함되어야 한다. 만약, 그렇지 못할 경우 hard-ware 상의 에러가 발생하며, 새로운 입력자료를 제공하여야 한다.

$\theta 3 = \mathrm{COS}^{-1}(\overline{AC}^2 - a2^2 - a3^2) / (2\ a2\ a3)$로부터 구해진 $\theta 3$의 값은 ROB-501 로보트의 경우 link 3가 움직일 수 있는 상한치인 0°와 하한치인 −90°의 두 값의 범위 속에 포함되어야 에러가 발생하지 않는다.

먼저, $\theta 3$가 상한치인 0°를 벗어날 경우에 대한 기하학적 해석에서 이는 singularity 조작에러와 같은 의미로 앞에서 singularity 조작에러가 해결되어 들어오기 때문에 이에 대한

Table 1. Singularity Test

lower hound	measured value	upper hound
$(d5\ \mathrm{COS}(\mathrm{pitch}) + ((a2-a3)^2 - (d5\ \mathrm{SIN}(\mathrm{pitch}))^2)^{\frac{1}{2}}$	$(x^2 + y^2 + (z-25)^2)$	$(d5\ \mathrm{COS}(\mathrm{pitch}) + ((a2+a3)^2 - (d5\ \mathrm{SIN}(\mathrm{pitch}))^2)^{\frac{1}{2}}$

Table 2. Limitation range of the input data in order not to occur the singularity opertion error

limitation range of the input data in order not to occur the singularity operation error		
input data	lower bound	upper bound
px	$(((a2-a3)^2 - (pz - d5 \times \mathrm{SIN}(\mathrm{pitch}) - d1)^2)^{\frac{1}{2}} + d5 \times \mathrm{COS}(\mathrm{pitch}) - py^2)^{\frac{1}{2}}$	$(((a2+a3)^2 - (pz - d5 \times \mathrm{SIN}(\mathrm{pitch}) - d1)^2)^{\frac{1}{2}} + d5 \times \mathrm{COS}(\mathrm{pitch}) - py^2)^{\frac{1}{2}}$
py	$(((a2-a3)^2 - (pz - d5 \times \mathrm{SIN}(\mathrm{pitch}) - d1)^2)^{\frac{1}{2}} + d5 \times \mathrm{COS}(\mathrm{pitch}) - px^2)^{\frac{1}{2}}$	$(((a2+a3)^2 - (pz - d5 \times \mathrm{SIN}(\mathrm{pitch}) - d1)^2)^{\frac{1}{2}} + d5 \times \mathrm{COS}(\mathrm{pitch}) - px^2)^{\frac{1}{2}}$
pz	$((a2-a3)^2 - ((px^2+py^2)^{\frac{1}{2}} - d5 \times \mathrm{COS}(\mathrm{pitch}))^2)^{\frac{1}{2}} + d5 \times \mathrm{SIN}(\mathrm{pitch}) + d1)$	$((a2+a3)^2 - (px^2+py^2)^{\frac{1}{2}} - d5 \times \mathrm{COS}(\mathrm{pitch}))^2)^{\frac{1}{2}} + d5 \times \mathrm{SIN}(\mathrm{pitch}) + d1)$

Table 3. Limitation range of the pitch in order not to occur the singularity opertion error

limitation range of the pitch in order not to occur the singularity opertion error		
input data	lower bound	upper bound
pitch	$\mathrm{COS}^{-1}(px^2 + py^2 + d5^2 + (pz-d1)^2 - (a2-a3)^2) / 2\ d5(px+py)$	$\mathrm{COS}^{-1}(px^2 + py^2 + d5^2 + (pz-d1)^2 - (a2+a3)^2) / 2\ d5(px+py)$

Table 4. Limitation range of the input data and the robot joint not occuring parameter operation error

limitation range of the robot joint in order not to occur the parameter operation error					
θi	measured value	upper bound	over	lower bound	under
$\theta 1$	py / -px	0.5775	150°	−0.5775	−150°
$\theta 2$	$COS^{-1}(\overline{AC}^2 + a2^2 - a3^2) / 2\overline{AC}\ a3)$ $+ TAN^{-1}(Ze / Xe)$	100°	100°	− 30°	−30°
$\theta 3$	$\overline{AC}$	740	90°	1444	0°
$\theta 4$	pitch $-\theta 2 + \theta 3$	90°	90°	−90°	− 90°

에러발생은 일어나지 않는다. 먼저, parameter test 는 표 4와 같이 실시되어지며 pitch값의 조절에 의한 입력자료의 변화 없이 가능한 관절범위로의 이동을 나타내고 있다.

제한범위까지의 로보트 이동이 아닌 pitch 값의 변화에 의해 원하는 위치정보로의 이동에 대해 살펴보면 다음과 같다. $\theta 1$의 경우에 대한 parameter 조작에러는 방향에 대한 에러로 pitch 값의 변화에 의해 해결되지 않으며 단지 위에 주어진 표 4와 같이 제한범위까지의 이동에 의해 해결할 수 있다. 또한 $\theta 2$ 의 경우에 대해서는 로보트관절 특성상 Ze = d1 + a2COS(10) + a3 SIN(10) 이하 및 Xe = a3COS(10) − a2SIN(10) 이상의 경우에 대해서는 parameter 조작에러가 발생하지 않으며, 그 밖의 경우에 대해서는 제한범위까지의 이동에 의해 해결할 수 있다. 그러나 $\theta 3$의 경우에 있어서는 하한치인 −90°를 벗어날 때 pitch 를 $COS^{-1}(px^2 + py^2 + d5^2 + (pz-d1)^2 - a2^2 - a3^2) / 2\ d5(px + py)$의 값으로 조절하여 원하는 위치까지 이동할 수가 있다. $\theta 4$의 제한범위를 벗어날 경우에도 $90° + \theta 2 - \theta 3$ 와 $-90° + \theta 2 - \theta 3$로 상한과 하한 pitch 값을 선택하여 원하는 위치정보까지 로보트를 이동시킬 수 있다.

2.2.2. Heuristic rules의 보완에 의한 개선

(1) Singularity heuristic rules의 개선

수동모드에 대한 흐름도를 나타내면 그림 5와 같다. 기존의 singularity heuristic rules에 의해 singularity 조작에러를 해결할 수가 있으나, 이는 px, py, pz에 대한 제한범위 산출이 아닌 단지 pitch 값의 대략적인 조절로 타당한 이론적 배경을 갖지 못하며 정확한 pitch 값을 찾지 못하여 RMS (root mean square) 오차가 발생한다.

이와 같은 singularity heuristic rules의 문제점을 보완하기 위하여 algorithm에 따른 heuristic rules 를 구성하여 heuristic rule의 이론적 배경 및 RMS 오차를 줄인다.

보완된 singularity heuristic rules는 다음과 같은 과정을 거쳐 singualrity 조작에러를 해결한다. 첫째, 작업물체에 놓인 직교좌표상의 위치에 따라 end point에 해당하는 좌표값 및 원하는 pitch각과 조절할 수 있는 입력자료를 제시한다. 둘째, 입력자료에 따라 표

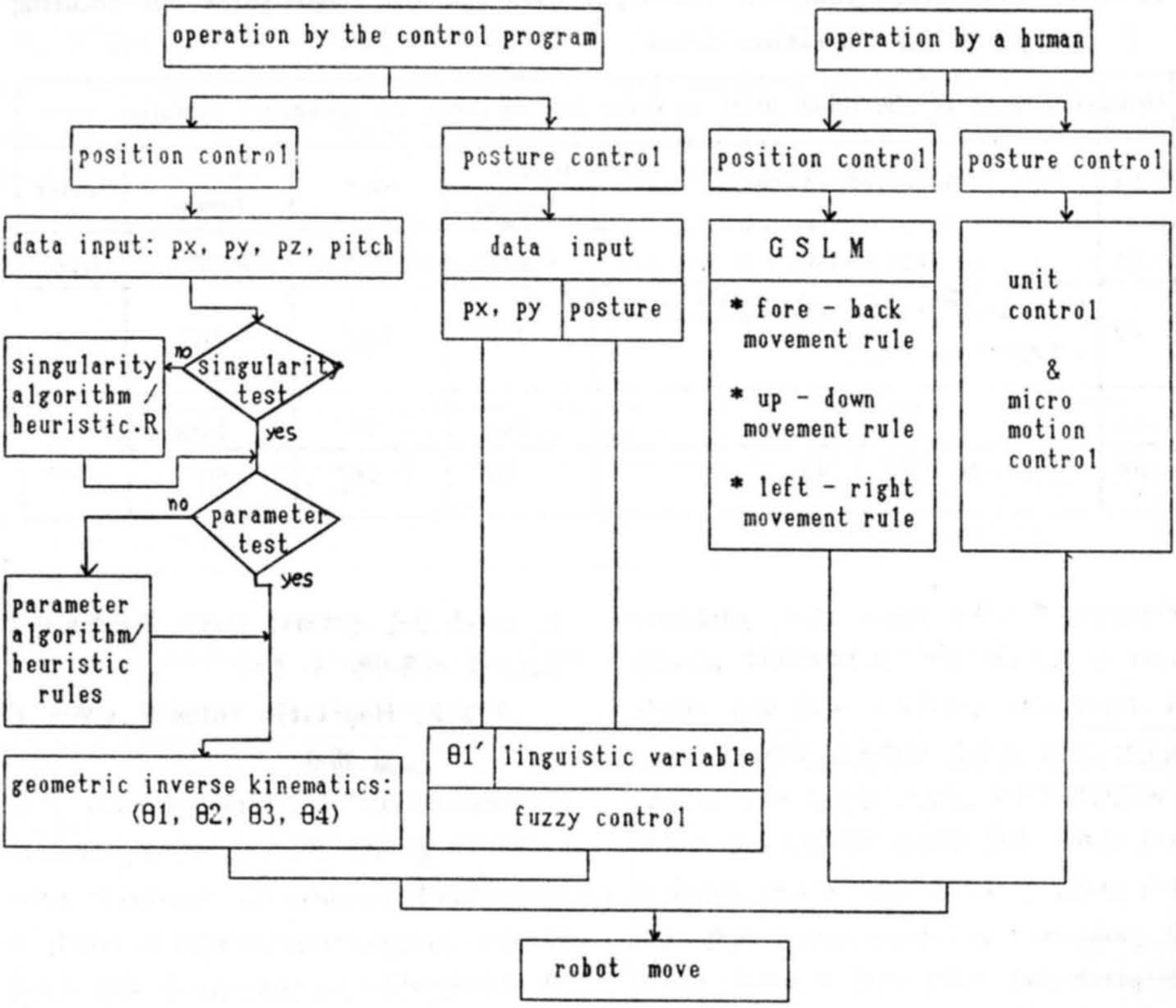


Fig 5. Flowchart for the position control of the manual mode

1과 같은 singularity test를 거쳐 singularity 조작에러가 발생하는가를 조사한다. singularity 조작에러가 발생하지 않는 값이면 그대로 관절값계산에 들어간다. singularity 조작에러가 발생하는 값이면 px, py, pz 값과 pitch값 중 조절하고자 하는 입력자료에 대해 singularity 조작에러를 유발하는 변수값의 제한범위를 sigularity algorithm을 통해 적절한 값을 재입력함으로써 singularity 조작에러를 해결할 수가 있다. singularity 조작에러를 발생하지 않는 입력자료로 조절된 좌표값을 이용하여 parameter test로 들어간다.

singularity algorithm에 따른 heuristic rules는 잘못된 입력자료에 의해 singularity test를 벗어날 경우 위와 같은 singularity heuristic rules을 거쳐 parameter test에 들어가게 된다. 보완된 singularity heuristic rules은 위치자료인 px, py, pz의 값과 pitch값 중 적절한 값을 조절하여 singularity 조작에러를 해결한다. 각 입력자료의 변화에 따른 singularity 조작에

러 지역을 살펴보면 px, py, pz에 의해서 결정되는 end point의 거리와 pitch에 의해 singularity 조작에러가 결정됨을 알 수 있다. 따라서 위치자료에 의해 구해진 end point의 거리에 따른 pitch의 범위와 주어진 pitch 사이의 관계를 탐색함으로써 singularity 조작에러를 해결할 수 있다.

⑵ Parameter heuristic rules의 개선

기존의 parameter heuristic rules에 의한 parameter 조작에러의 해결은 geometric inverse kinematics에 의한 관절값계산이 이루어지고 난 후 그 관절값이 제한범위를 벗어날 경우, 제한범위까지 움직이게 해주는 것으로 미리 parameter 조작에러의 여부를 판가름할 수 없으며, 따라서 원하는 위치정보까지의 이동이 불가능해지며, 따라서 RMS 오차가 발생한다.

이와 같은 parameter heuristic rules 의 문제점을 보완하기 위하여 parameter 조작에러의 기하학적 해석에 근거를 둔 heuristic rules를 구성하여 원하는 위치정보에 가까운 지점으로 로보트를 이동하게 한다.

보완된 parameter heuristic rules는 표 4와 같은 parameter test를 이용한다. 잘못된 입력자료에 의해 parameter test를 벗어날 경우 algorithm에 의한 parameter heuristic rules을 통한 관절값 계산을 거치게 되어 end point의 가까이에 이동하게 되어 RMS 오차를 줄인다.

parameter 조작에러에 대한 기하학적 해석에 의해 산출된 식을 이용하여 parameter 조작에러는 다음과 같은 과정을 거쳐 해결된다.

첫째, 작업물체의 좌표값과 pitch 값을 parameter test에 입력한다. 360° 회전이 가능한 $\theta 5$를 제외하고 나머지 관절각에 대한 parameter test를 실시한다.

둘째, parameter test 과정을 거쳐 주어진 입력자료가 parameter 조건에 위배되지 않을 경우 geometric inverse kinematics에 의한 관절값계산에 들어가며, 그렇지 못하고 입력자료가 parameter 조작에러를 유발하는 값일 경우 보완된 heuristic rules을 통해 각 입력자료에 대한 제한범위와 parameter 조작에러가 발생치 않는 pitch값을 선택하여 geometric inverse kinematics에 의한 관절값 계산과 로보트 이동을 하게 된다.

2.3 직접조작의 위치제어 개선

기존의 직접조작에 사용되는 UC와 MMC는 joint coordinate motion〔9〕을 나타내며 각 축이 joint coordinate에 따라 움직이므로 UC MMC를 사용하여, 물체를 잡을 때까지 형성된 경로는 지그재그의 모양으로 위치제어와 자세제어의 구분없이 사용하였다.

인간의 손동작처럼 로보트를 이용하여 물체를 향해 똑바로, 즉 직선운동을 할 수 있도록 한다면 UC MMC에서 처럼 각 축의 움직임과 그 상호관계를 생각할 필요없이 보다 효율적인 직접조작을 이룰 수 있다. 이에 대해 Taylor〔9〕과 Paul〔10〕 등의 연구가 있는데, 본 연구에서는 기존의 cartesian straight line motion을 대수학적이 아닌 기하학적인 방법으로 분석하여 직선운동에 필요한 각 축들의 값을 산출했으며 이를 geometric straight line motion(GSLM) 방식이라 부른다. 이 방식은 로보트의 베이스를 중심으로 fore-back, right-left, up-down의 여섯 개 동작에 해당하는 로보트 1, 2, 3 축의 상호관계를 기하학적인 방법으로 도출하여 rule로서 프로그램 내에 내장했다. 제어를 위한 입력 형식으로는 fore 방향으로 10 *cm* 이동시키려 할 때 " F 10 "으로 입력하면 된다.

Forward (Backward) 방향으로 제어할 때

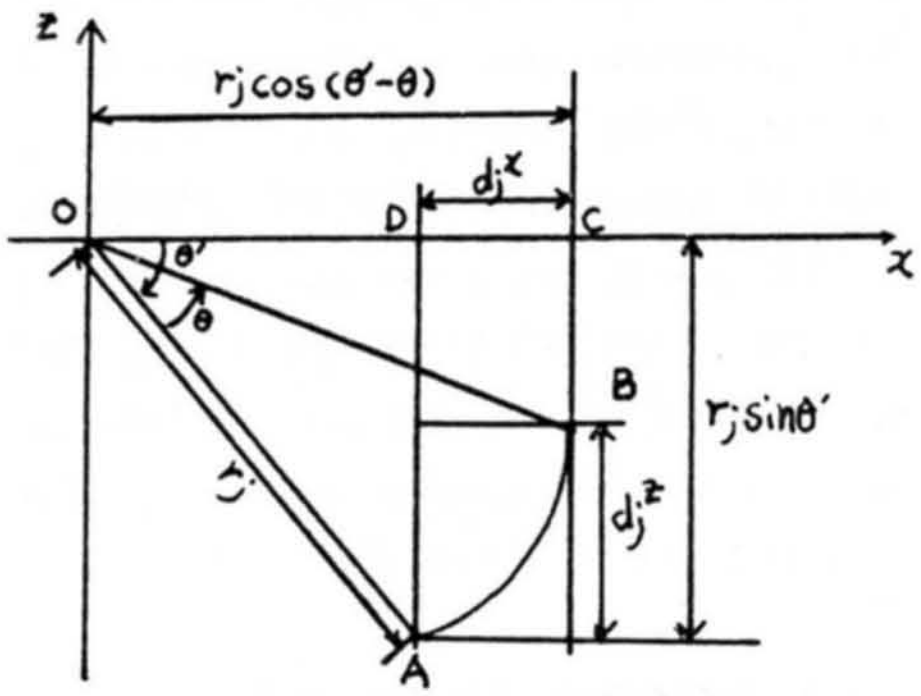


Fig 6. Geometrical analysis for the straight line motion

여기서, f : 제어하고자 하는 양(단위는 *cm*)
p : 로보트손의 pitch angle
max : 고도= h, pitch = p일 때 로보트손이 도달할 수 있는 최대거리
min : 고도= h, pitch일 때 현재 로보트손의 로보트 어깨축으로부터 거리
m : 고도= h, pitch일 때 로보트손이 이동할 수 있는 거리
a2, a3, a4 : 로보트어깨, 팔굽, 손목축이다.

위와 같이 세 개의 축으로 이루어진 manipulator의 경우, forward 방향으로 "f *cm*" 만큼 움직이고자 할 때 $\theta 2$, $\theta 3$로써 원하는 동작을 얻을 수 있으며 로보트축의 운동특성상 원운동을 하므로 어깨축을 $\theta 2$가 되도록 내려주고 그 움직임만큼 팔굽축을 $\theta 3$로 올려주면 된다.

pitch가 0°라면 팔굽축과 손목축이 마치 하나의 축처럼 다루어지므로 어깨축과 팔굽축(손목축 포함), 그리고 나머지 한 축으로 그 사이각들을 구할 수 있으므로 쉽게 해석이 되는데, 그림 6처럼 pitch가 0°가 아닐 경우에는 4개의 변으로 이루어진 사각형이 되므로 그 사이각들을 구하는 데는 문제가 있다.

그런데 여기에서의 pitch각은 팔굽축 연장선과의 각이므로 팔굽축과 손목축과의 사이각은 pitch의 보각으로 항상 일정하다. 이런 점에 착안을 해서 GSLM에서는 pitch가 0°가 아닐 경우에 팔굽축과 손목축을 따로 나누어 생각하지 않고 이들과 pitch로 이루어지는 삼각형에서의 다른 한 변을 가상의 축으로 해서 이 가상의 축과 어깨축으로 형성되는 삼각형에서 $\theta 2$를 구해낸다.

먼저 pitch가 0°가 아닐 때 형성되는 BCD에서 a34는 cosine 제2법칙에 의해,

$a34^2 + a4^2 - 2 \times a3 \times a4 \times COS(180 - |p|)$ 이므로 이 a34와 a2, 그리고 그림 5에서 (min + p)를 알면 $\theta 2$를 구할 수 있다.

min는 그림 6에서와 같이 고도의 h이고, pitch가 p일 때 로보트손이 당겨질 수 있는 최소의 거리로 ABD에서 역시 cosine 제2법칙으로 구할 수 있다.

이에 앞서 a2와 a34와의 사이각을 β라 했을 때 이 각은, min는 주로 a2와 a3 사이각이 직각이므로 α를 알면 얻을 수 있다.

α도 역시 cosine 제2법칙에 의해

$\alpha = COS^{-1}(a3^2 + a34^2 - a4^2 / 2 \times a3 \times a3)$ 이므로 $(\beta = 90° - \alpha)$로 구하면,

$min = a2^2 + a34^2 - 2 \times a2 \times a34 \times COS\beta$ 이 된다.

그러면 이 min와 원하는 제어량 f를 합한 선분 $\overline{AF}$와 a2, a34를 이용해서 consine 제2법칙을 적용하면 $\theta 2$는 다음과 같이 구한다.

$$\theta 2 = COS^{-1} \frac{a2^2 + \overline{AF}^2 - a34^2}{2 \times a2 \times \overline{AF}}$$

다음으로 $\theta 3$ 역시 같은 방법으로 구할 수 있으나 계산상의 효율성을 위해 δ각을 규정해서 그 관계를 살펴보자.

a2/a34 = λ로 놓으면, a2 × SINθ2 = a34 × SINδ이므로, δ = SIN(λ × SIN 2)이며, θ3는 ABD에서 θ3 = 180° − (θ2 + δ)로 구하며, 이에 α를 합한 θ3 = θ3 + α가 원하는 θ3값이 된다.

3. 실험 및 결과분석

3.1 지식습득 및 표현

ERES/CCS의 지식기반을 위한 지식습득은 크게 인간조작자의 지식, 작업환경모델과 기본이론, ROB-501 로보트교범으로부터 얻어진다.

지식베이스의 세 가지 objects인 parts, rules, goals로 ERES/CCS의 architecture를 표현하면 그림 7과 같다.

3.2 실험장치 및 방법

에러회복을 위한 인간-로보트 시스템의 실험장치도를 나타낸 것이 그림 8이다.

에러회복을 위한 중추적 역할을 수행하는 knowledge base 내의 inference engine과 제어를 담당하는 HIS (Human Interactive Subsystem) 컴퓨터로는 IBM-PC/XT가 사용되며, 이는 인간작업자의 command를 받아 작업을 수행하고 그 결과를 feed back 하는 interface 기능을 수행한다. TIS (Task Interactive Subsystem) 컴퓨터의 역할은 ROB-501 robot drive unit가 담당하며, 이는 HIS로부터 동작데이터를 받아 각 조인트의 이동량을 actuator signal로 전환시킨다.

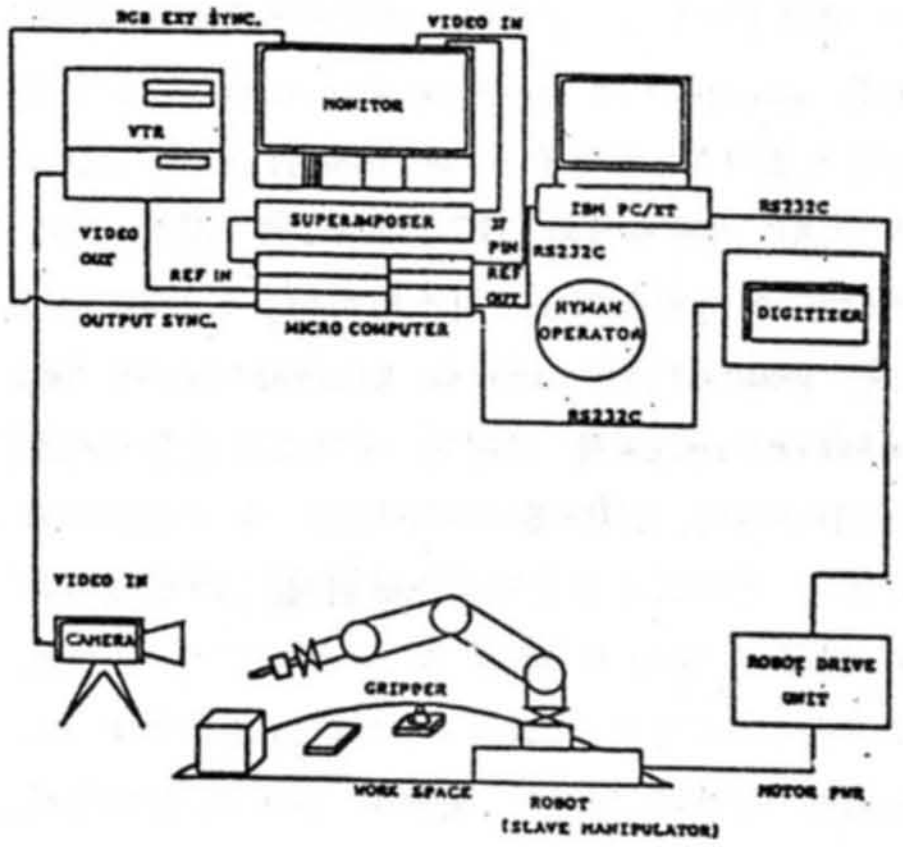


Fig 8. Experimental design of the ERES/CCS

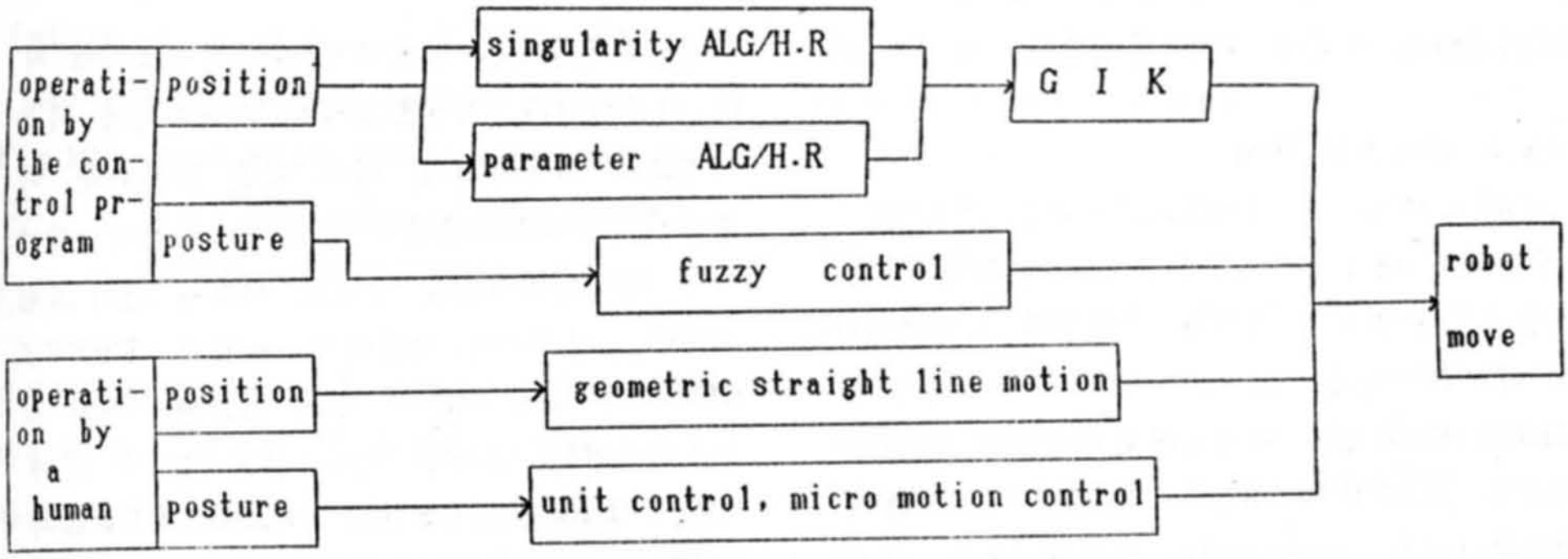


Fig 7. Knowledge representation of the ERES/CCS architecture

극한작업환경하의 작업을 수행하는 manipulator로는 ROB-501 robot를 사용하며, PTP 제어방식으로 구동된다. 또한 원거리조작의 수행을 위해 작업장에 sensor의 역할을 수행하는 T.V. camera가 설치되어 있으며, 인간조작자는 monitor에 의해 화상을 전달받는다.

ERES에 의한 에러회복실험은 ERES/CCS(1), ERES/CCS(2) 및 ERSE/CCS(3)에 대한 비교실험으로 행한다. 여기에서 ERES/CCS(1)은 geometric inverse kinematics에 의한 코맨트조작의 위치제어 및 modularity rules에 의한 자세제어와 UC, MMC에 의한 직접조작의 에러회복전문가시스템이며, ERES/CCS(2)는 geometric inverse kinematics에 heuristic rules을 보완한 에러회복전문가시스템[10]이며, ERES/CCS(3)는 본 논문에서 연구된 에러회복전문가시스템이다. 피실험자는 본 연구에 대한 이해가 깊은 대학원생 3명으로 한다. 수행도 척도로는 에러회복작업에 소요되는 시간과 입력된 위치와 로보트가 이동한 위치 사이의 오차이며, 비교대상의 세 시스템은 동일한 실험장치와 작업환경모델을 사용한다. 실험작업은 작업장 평면의 한 위치에 놓여있는 성냥봉을 잡고, 성냥통으로 이동하여 불을 켜는 lamp 점화작업 중에 발생하는 에러를 회복하는 실험에 대해 실시한다.

3.3 실험결과 분석

ERES/CCS(1), ERES/CCS(2), ERES/CCS(3)에 대해 세 시스템을 좌표판독과 이동시간의 코맨드 조작과 직접조작 각각에 대해 비교분석한다. 기존의 ERSE/CCS(1)과 ERES/CCS(2)의 평균에러회복시간이 같다는 가정과 ERES/CCS(2)와 ERES/CCS(3)의 평균에러회복시간이 같다는 가정에 대한 기각역 $F > F_{0.05}(v1, v2) = 3.65$에 분산분석 결과 산출된 F값이 포함되어 평균치가 같다는 귀무가설,

H_0 : u1 = u2, H1 : u1 ≠ u2

H_0' : u2 = u3, H1′ : u2 ≠ u3

여기서, u1 : ERES/CCS(1)의 평균에러회복시간

u2 : ERES/CCS(2)의 평균에러회복시간

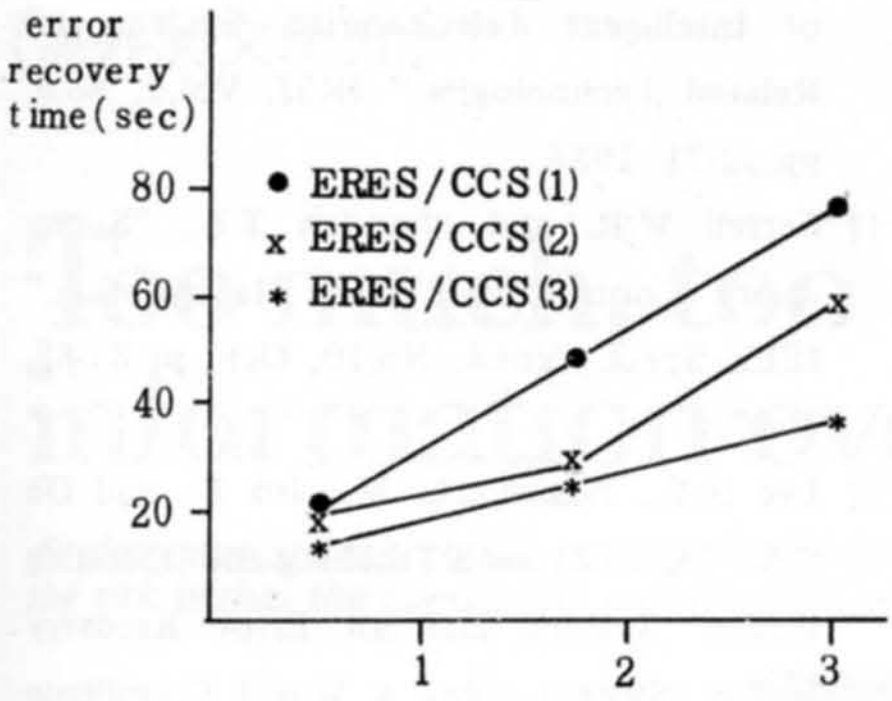
u3 : ERES/CCS(3)의 평균에러회복시간

에 위배되며 이는 다음의 표 5의 ANOVA분석으로 주어져 있다. 아래 그림 9에서 알 수 있는 바와 같이 ERES/CCS(1)에 비해 ERES/CCS(2)가 ERES/CCS(2)에 비해 ERES/CCS(3)의 전체적인 에러회복시간이 적게 소요됨을 검증할 수 있다.

Table 5. Experimental results of the ANOVA

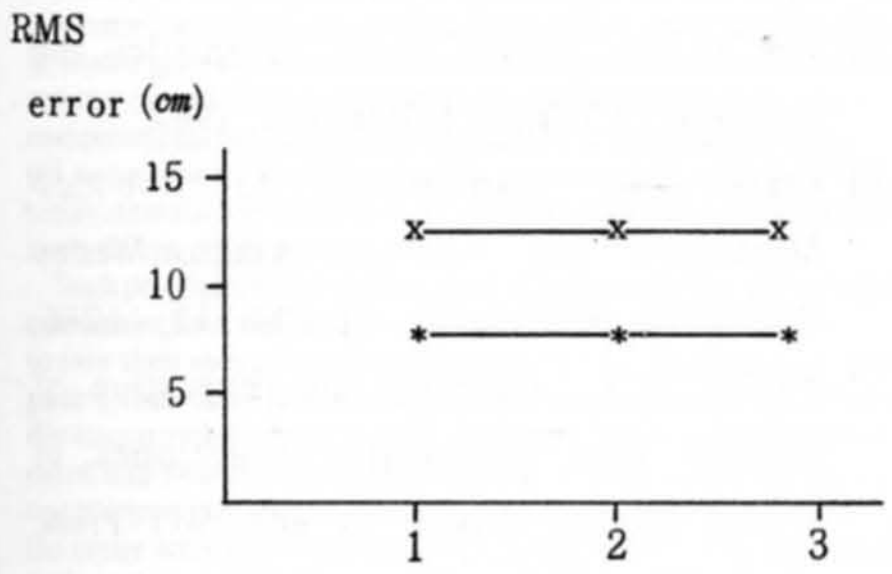
ANOVA			
No	characteristic	H_0	H_0'
1	significant level	5 %	5 %
2	acceptance region	[0,3.65]	[0,3.65]
3	F-value	160.6704	144.4000
4	coordinate reading	rejected	accepted
5	movement time	accepted	rejected
6	total	rejected	rejected

에러회복작업은 코맨드조작과 직접조작에 의해 이루어진다. 코맨드조작은 좌표판독과 이동시간으로 나누어진다. 이와 같은 좌표판독, 이동시간, 직접조작시간에 대해 그 작업이 차지하는 에러회복시간을 누계로 나타낸 것이 그림 9이며, 대부분의 작업에서 보완된 ERES/CCS(3)의 우월성을 볼 수가 있으며, 특히 직접조작시간의 감소가 두드러짐을 알 수 있다. 또한, ERES/CCS(2)와 ERES/CCS(3)의 RMS 오차를 나타낸 것이 그림 10으로 ERES/CCS(3)의 정확성을 살펴볼 수 있다. 이러한



(where, 1 : coordinate reading time
2 : movement time
3 : operation by a human time)

Fig 9. Error recovery time of the ERES/CCS(1), ERES/CCS (2), ERES/CCS(3)



(where, 1, 2, 3 are task difficulty for the attitude of task object)

Fig 10. Root mean square error by the task difficulty of the ERES/CCS(2), ERES/CCS(3)

실험결과의 분석에 의하면 ERES/CCS (3)는 다음과 같은 세 가지의 장점을 가진다. 첫째, 자세판독시의 어려움을 fuzzy set theory 의 적용에 의해 linguistic variable을 입력함으로써 좌표판독시간을 감소할 수 있다. 둘째, geometric inverse kinematics에 algorithm에 의한 heuristic rules을 부가하여 singularity 조작에러와 parameter 조작에러를 해결하는 algorithm의 보완에 의한 heuristic rules의 코맨드조작의 위치제어 개선은 기존의 heuristic rules에 비해 입력된 위치정보와 로보트 이동위치 사이의 RMS 오차가 적게 나타날 수 있다. 세째, ERES/CCS (3)에서는 직접 조작의 위치제어에 geometric inverse kinematics를 이용한 straight line motion을 적용하므로 직접조작 시간의 감소가 가능하다.

4. 결 론

human interface 기술의 일환인 지적 원격조작 기술의 발전을 위해 인간-로보트시스템의 유용성을 가져다 줄 수 있게 하는 효율적인 수동모드를 설계하였다.

프로그램에 의한 코맨드조작은 직교좌표를 이용한 입력자료에 대해 inverse kinematics의 단점을 보완할 수 있는 geometric inverse kinematics에 의해 로보트관절값을 계산하여 에러를 회복하는 시스템이다. 그러나, geometric inverse kinematics에 내포된 여러 문제점에 의해 많은 에러회복시간을 요구하였다. 따라서 이와 같은 좌표판독시의 어려*움, singularity 조작에러와 parameter 조*작에러를 해결하기 위해 다음과 같은 연구개선이 이루어졌다.

첫째, geometric inverse kinematics 에 의한 관절값계산시 발생하는 singularity 조*작에러와 parameter 조작에러를 기하학적 해*석에 의한 algorithm을 이용한 heuristic rules를 통해 해결할 수 있었다. Geometric inverse kinematics의 두 조작에러를 singularity test와 parameter test를 통해 에러 여부를 test하고, 에러발생시 algorithm에 의한 heuristic rules를 통한 새로

운 입력자료를 산출하여 이 값을 이용한 관절 값 계산으로 기존의 heuristic rule의 보완에 의한 ERES/CCS에 비해 RMS 오차를 줄일 수 있었다.

둘째, 좌표판독시 fuzzy control를 이용한 linguistic variable을 입력하여 자세판독의 모호성을 해결함으로써 자세제어 시간을 감소할 수 있었다.

세째, 로보트의 직선운동을 가능케 한 GSLM에 의한 빠른 위치제어를 실시하며, 인간조작자에 의한 직접조작시 로보트관절의 단위증가분을 이용한 UC, MMC에 의해 자세제어를 실시함으로써 직접조작시간의 감소가 가능해졌다.

앞으로의 과제로는 좌표판독의 정확성을 향상시키는 방법과 ASR(automatic speech recognition)을 적용한 직접조작의 개선이 따라야 할 것이다.

참 고 문 헌

[1] Sheridan T.B., "Supervisory Control of Remote Manipulators, Vehicles and Dynamic Process: Experiments in Command and Display Aiding," Advances in Man-Machine Systems Research, Vol.1, pp.49-137, 1984.

[2] Hirai S. and Sato T., "Language Aided Robotics Teleoperation System(LARTS) for Advanced Teleoperation," ICAR, 15th, pp.329-336, 1985.

[3] Takase K. and Wakamatsu S., "A Concept of Intelligent Teleoperation System and Related Technologies," JRSJ, Vol.2, No.6, pp.62-71, 1984.

[4] Ferrell W.R. and Sheridan T.B., "Supervisory Control of Remote Manipulation," IEEE Spect., Vol.4, No.10, Oct., pp.81-88, 1967.

[5] Lee S.Y., Nagamachi M., Ito K. and Oh C.S., "A Study on a Teaching and Operating Expert System and an Error Recovery Expert System using a World Coordinate System in the Advanced Teleoperator System," 10th Congress, International Ergonomics Association, Sydney, Austallia, pp. 645-647, 1988.

[6] Ahmad S. and Luo S., "Coordinated Motion Control of Multiple Robotics Devices for Welding," Applied Motion Control, Con., pp.117-124, 1984.

[7] Lee C.S.G., Robot Arm Kinematics and Dynamics, CICE/ECE/IOE 494, 1983.

[8] Craig J.J., Introduction to Robotics-Mechanics and Control, Addison-Wesley Publishing Company, pp.104-111, 1986.

[9] Taylor R.N., Planning and Execution of Straight Line Manipulator Trajectories, in Robot Motion Brady et all), MIT Press, pp.26-286, 1982.

[10] Paul R.P., Robot Manipulators: Mathematics, Programming, and Control, MIT Press, 1981.