2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Lyapunov-stable Position/Force Control Based on Dual Nature in Constraint Motion

Mamoru Minami[1], Fujia Yu[1] and Akira Yanou[1]

***Abstract*— In the model of the constrained dynamic system of a rigid robot contacting with rigid environment, constrained forces can be expressed as an algebraic function of states (instantaneous process) and a redundancy existing in constraint dynamics (constraint redundancy). Using these results, a force and position control law is proposed by taking the advantages of the redundancy of input generalized forces to the constrained forces and instantaneous process without involving any force sensor, using dual nature of constraint motion stated in this document. Then proof confirming by Lyapunov method that the exerting force equals instantaneously and constantly to desired one and that the motion of robot's hand in a movable space converges to desired hand's position. The effectiveness has been confirmed by a 2-link robot in simulations.**

I. Introduction

It is well known that robots, particularly articulated types, are very dexterous and have large operable space. Hence, they will have a promising future to introduce such kinds of robots more extensively into manufacturing. For example, the tasks of the grinding or cutting of deeply located surfaces within a cabinet might be too difficult to machine. Furthermore, for some auxiliary machining operations, it may cost too much for an expensive machining center to do. Therefore, employing robots in such areas will be a satisfactory alternative. On the other hand, comparing with a machine tool, the characteristics of robots on stiffness, damping and vibration-proofing are somewhat poor. In order to take the advantage of the dexterity of robots, much sophisticated design and control strategies have to be developed.

We think that a paper [1] had classified contacting tasks of robots practically. The following classification is along with the statements in [1]. Robot force control method can be largely classified into impedance control and hybrid control. In impedance control, a prescribed dynamic relation is sought to be maintained between the robot end-effector's force exerting to an object constraining the end-effector and position displacement toward the direction vertical to the object's surface [2]. In hybrid control, the end-effector's force is explicitly controlled in selected directions and the end-effector's position is controlled in the remaining (complementary) directions [3].

The hybrid control approaches can be further classified into three main categories [1]: (A) explicit (model based) hybrid control of rigid robots in elastic contact with a compliant environment, e.g. [4]-[5], in which the end-effector force is controlled by directly commanding the joint torques of the robot based on the sensed force error; (B) implicit (position/velocity based) hybrid control of rigid robots in elastic contact with a compliant environment, e.g. [6], in which the end-effector force is controlled indirectly by modifying the reference trajectory given into an inner loop joint position/velocity controller based on the sensed force error and (C) explicit (model based) hybrid control of rigid robots in hard contact with a rigid environment, e.g. [3], [7].

Many researches have discussed on the constraint-combined force/position hybrid control method. To ensure the stabilities of the constrained motion, those force and position control methods have utilized Lyapunov's stability analysis under the inverse dynamic compensation where force control strategies have been explained intelligibly in papers [1], [8], [9]. But these stability proofs are trying to divide the procedure into two different parts [10], [11]: force convergence $\lim_{t\to\infty}\boldsymbol{F}_n=\boldsymbol{F}_{nd}$ and position convergence $\lim_{t\to\infty}\boldsymbol{r}=\boldsymbol{r}_d$, here $\boldsymbol{F}_n$ and $\boldsymbol{F}_{nd}$ are the actual constraint force and the desired constraint force, while $\boldsymbol{r}$ and $\boldsymbol{r}_d$ are the actual hand position of the manipulator and the desired one.

In this research, the third category (C) of contacting situation that assumes rigid link manipulator and hard contacting with nonelastic environment. Given this prerequisite, Yamane and Nakamura have used the following matrix equation in [12], so both constraint condition and the dynamics can be represented simultaneously. This equation means that determining the constraint force F_n does not include time integration like $\ddot{\boldsymbol{q}}$. In this equation $\ddot{\boldsymbol{q}}$ expresses the angular acceleration, and F_n does the constraint force, the derivation will be given in section 2. Upper side represents equation of motion of robot, and the lower side does the constraint condition differentiated twice by time. Furthermore, they extended the dynamics representation method into the concept of dynamics filter [13], [14].

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{J}_c^T \\ \frac{\partial C}{\partial \boldsymbol{q}^T} & 0 \end{bmatrix}\begin{bmatrix} \ddot{\boldsymbol{q}} \\ F_n \end{bmatrix} = \begin{bmatrix} \boldsymbol{\tau}-\frac{1}{2}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}}-\boldsymbol{N}\dot{\boldsymbol{q}}-\boldsymbol{G}-\boldsymbol{J}_t^T F_t \\ -\dot{\boldsymbol{q}}^T[\frac{\partial C}{\partial \boldsymbol{q}}(\frac{\partial C}{\partial \boldsymbol{q}^T})]\dot{\boldsymbol{q}} \end{bmatrix} \quad (1)$$

In [15] it is written that "If contact is modeled by means of geometric constraints, then the contact forces cannot be expressed as algebraic functions of the state variables $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$." The $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ express the angle and angular velocity of the joints. We do not think it is right, because the contact force has been calculated in (2). (2) is a solution of (1), which has

[1]Mamoru Minami, Fujia Yu, and Akira Yanou are with Graduate School of Natural Science and Technology, Okayama University Tsushimanaka3-1-1 , Okayama, JAPAN. {minami, yufujia, yanou}@sys.okayama-u.ac.jp.

been pointed out by Hemami [16] in the analysis of biped walking robot, denotes clearly the algebraic relation between the input torque $\boldsymbol{\tau}$ of the robot and exerting force to the contacting object F_n have algebraic relation, when robot's end-effector being in touch with a surface in 3-D space:

$$F_n = a(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{A}(\boldsymbol{q})\boldsymbol{J}_r^T F_t - \boldsymbol{A}(\boldsymbol{q})\boldsymbol{\tau}. \tag{2}$$

Where $a(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is scalar function and $\boldsymbol{A}(\boldsymbol{q})$, $\boldsymbol{J}_r$ are vectors defined in following section. (2) exhibits vector $\boldsymbol{\tau}$ determining F_n has a redundancy against constraint force F_n since F_n is scalar. The other point is that the transmission of $\boldsymbol{\tau}$ to F_n is not time consuming process like joint angle $\boldsymbol{q}$ and velocity $\dot{\boldsymbol{q}}$. This means just that (2) is not time differential equation but algebraic equation, we have also reported these two characters existing in constraint motion by [17]-[20].

From (2) we can know that the force transmission process is an immediately finished process for a rigidly structured manipulator just as the acceleration being determined immediately by state variables and input generalized forces, this property of contacting motion is extended to a concept of "dual nature" in the following section. Exploiting (2), we design a new controller whose stability is guaranteed by Lyapunov method, which assures exerting force $F_n(t) = F_{nd}$ and $\lim_{t\to\infty} \boldsymbol{r} = \boldsymbol{r}_d$. This result that converge to the desired value instantaneously $F_n(t) = F_{nd}$—a part of this result has been presented by authors at a domestic meeting in Japan [21]—can be distinguished from the former consequences $\lim_{t\to\infty} F_n(t) = F_{nd}$ [10] [11]. The effectiveness of our proposed position/force control method has been confirmed by a 2-link grinding robot model in simulation.

II. Analysis of Grinding Task and Modeling

A. Contacting Force and Friction

The normal grinding force F_n is exerted in the perpendicular direction of the surface. It is a significant factor for grinding robots that affects ground accuracy and surface roughness of workpiece. The value of it is also related to the grinding power or directly to the tangential grinding force as

$$F_t = \mathrm{K_t} F_n, \tag{3}$$

where, $\mathrm{K_t}$ is an empirical coefficient, F_t is the tangential grinding force. This relation gives us an estimated value of F_t given that K_t and F_n are known.

B. Constrained Dynamics

Hemami and Wyman have addressed the issue of control of a moving robot according to constraint condition and examined the problem of the control of the biped locomotion constrained in the frontal plane. Their purpose was to control the position coordinates of the biped locomotion rather than generalized forces of constrained dynamic equation involved the item of generalized forces of constraints. And the constrained force is used as a determining condition to change the dynamic model from constrained motion to free motion of the manipulators. In this paper, the grinding manipulator, whose end-point is in contact with the constrained surface, is modeled according with Lagrangian equations of motion in term of the constraint forces, referring to what Hemami [16] and Arimoto [10] have done:

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\boldsymbol{q}}}\right) - \left(\frac{\partial L}{\partial \boldsymbol{q}}\right) \\
&= \boldsymbol{\tau} + \left(\frac{\partial C}{\partial \boldsymbol{q}^T}\right)^T / \parallel \frac{\partial C}{\partial \boldsymbol{r}} \parallel F_n - \left(\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{q}^T}\right)^T \dot{\boldsymbol{r}} / \parallel \boldsymbol{r} \parallel F_t \\
&= \boldsymbol{\tau} + \boldsymbol{J}_c^T(\boldsymbol{q}) F_n - \boldsymbol{J}_r^T(\boldsymbol{q}) F_t
\end{aligned} \tag{4}$$

where, $\boldsymbol{J}_c$ and $\boldsymbol{J}_r$ are defined as:

$$\boldsymbol{J}_c^T = \frac{\left(\dfrac{\partial C}{\partial \boldsymbol{q}^T}\right)^T}{\parallel \dfrac{\partial C}{\partial \boldsymbol{r}} \parallel} = \boldsymbol{J}^T \left(\frac{\partial C}{\partial \boldsymbol{r}^T}\right)^T \Big/ \parallel \frac{\partial C}{\partial \boldsymbol{r}} \parallel,$$

$$\boldsymbol{J} = \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{q}^T}, \qquad \boldsymbol{J}_r^T = \boldsymbol{J}^T \frac{\dot{\boldsymbol{r}}}{\parallel \dot{\boldsymbol{r}} \parallel},$$

$\boldsymbol{r}$ is the position vector of the hand and can be expressed as a kinematic equation,

$$\boldsymbol{r} = \boldsymbol{r}(\boldsymbol{q}). \tag{5}$$

$\boldsymbol{q}$ is $n(\geq 2)$ generalized coordinates. Then this manipulator does not have kinematic redundancy. In this research we only discuss the problem under only one constraint condition, so C is a scalar function of the constraint, and is expressed as an equation of constraints,

$$C(\boldsymbol{r}(\boldsymbol{q})) = 0, \tag{6}$$

F_n is the scalar express the value of the constrained force associated with C and F_t is the scalar express the value of tangential friction force.

In [10] (4) can be derived into :

$$\begin{aligned}
\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) \\
= \boldsymbol{\tau} + \boldsymbol{J}_c^T(\boldsymbol{q}) F_n - \boldsymbol{J}_r^T(\boldsymbol{q}) F_t,
\end{aligned} \tag{7}$$

here we express $\boldsymbol{M}(\boldsymbol{q})$ as $\boldsymbol{M}$ and $\boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ as $\boldsymbol{N}$ for short. $\boldsymbol{M}$ is an $n \times n$ matrix, $\boldsymbol{N}$ is a $n \times n$ skew-symmetrical matrix. $\boldsymbol{G}$ is a n row vectors. $\boldsymbol{\tau}$ is n inputs.

From the constraint condition (6) we can get

$$\frac{\partial C}{\partial \boldsymbol{q}^T} \ddot{\boldsymbol{q}} = -\dot{\boldsymbol{q}}^T \left[\frac{\partial C}{\partial \boldsymbol{q}}\left(\frac{\partial C}{\partial \boldsymbol{q}^T}\right)\right] \dot{\boldsymbol{q}}. \tag{8}$$

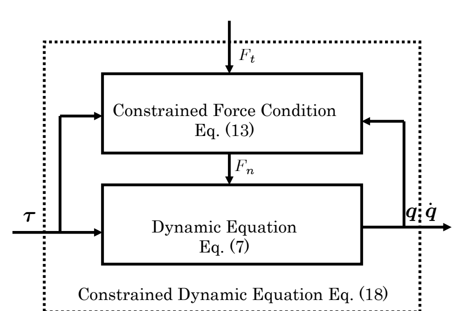


Fig. 1. Model of Constraint Dynamic System

The equation (7) and (8) can be combined as follows which is the same equation of (1):

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{J}_c^T \\ \frac{\partial C}{\partial \boldsymbol{q}^T} & 0 \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}} \\ F_n \end{bmatrix}$$

$$= \begin{bmatrix} \boldsymbol{\tau} - \frac{1}{2}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}} - \boldsymbol{N}\dot{\boldsymbol{q}} - \boldsymbol{G} - \boldsymbol{J}_t^T F_t \\ -\dot{\boldsymbol{q}}^T[\frac{\partial C}{\partial \boldsymbol{q}}(\frac{\partial C}{\partial \boldsymbol{q}^T})]\dot{\boldsymbol{q}} \end{bmatrix} \tag{9}$$

This equation is also used by Nakamura in [12], [13] and [14], it is easy to see that when the matrix on the left side is invertible, there exist a $\boldsymbol{\tau}$ which determines the $\ddot{\boldsymbol{q}}$ and F_n satisfying (7) and (8) respectively. And as below, the inertia matrix combined with constraint condition is guaranteed to be non-singular.

$$\begin{aligned} &\det \begin{bmatrix} \boldsymbol{M} & -\boldsymbol{J}_c^T \\ \frac{\partial C}{\partial \boldsymbol{q}^T} & 0 \end{bmatrix} \\ = &\det \boldsymbol{M} \cdot \det \Big(0 - \frac{\partial C}{\partial \boldsymbol{q}^T} \boldsymbol{M}^{-1}(-\boldsymbol{J}_c^T) \Big) \\ = &\frac{1}{\| \frac{\partial C}{\partial \boldsymbol{q}^T} \|} \det \boldsymbol{M} \cdot \det \Big(\frac{\partial C}{\partial \boldsymbol{q}^T} \boldsymbol{M}^{-1} (\frac{\partial C}{\partial \boldsymbol{q}^T})^T \Big) > 0 \end{aligned} \tag{10}$$

which means that the matrix is invertible since $\partial C(\boldsymbol{r}(\boldsymbol{q}))/\partial \boldsymbol{q}^T \neq \boldsymbol{0}$. We define:

$$m_c \triangleq (\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1} (\frac{\partial C}{\partial \boldsymbol{q}^T})^T. \tag{11}$$

The inverse matrix can be calculated as follow:

$$\begin{bmatrix} \boldsymbol{M} & -\boldsymbol{J}_c^T \\ \frac{\partial C}{\partial \boldsymbol{q}^T} & 0 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} \boldsymbol{M}^{-1}\{\boldsymbol{I} - \boldsymbol{J}_c^T m_c^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}} \| \frac{\partial C}{\partial \boldsymbol{q}^T} \boldsymbol{M}^{-1}\} & m_c^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}} \| \boldsymbol{M}^{-1} \boldsymbol{J}_c^T \\ -m_c^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}} \| \frac{\partial C}{\partial \boldsymbol{q}^T} \boldsymbol{M}^{-1} & m_c^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}} \| \end{bmatrix}. \tag{12}$$

From (9) and (12) the constraint force F_n being identical to (2) can be solved like:

$$\begin{aligned} F_n = &[(\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1} (\frac{\partial C}{\partial \boldsymbol{q}^T})^T]^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}} \| \{-[\frac{\partial}{\partial \boldsymbol{q}}(\frac{\partial C}{\partial \boldsymbol{q}^T})\dot{\boldsymbol{q}}]\dot{\boldsymbol{q}} \\ &+(\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1}(\frac{1}{2}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}} + \boldsymbol{N}\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) + J_r^T F_t)\} \\ &-[(\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1} (\frac{\partial C}{\partial \boldsymbol{q}^T})^T]^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}} \| \{(\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1}\}\boldsymbol{\tau} \\ &\triangleq a(\boldsymbol{q}, \dot{\boldsymbol{q}}) + \boldsymbol{A}(\boldsymbol{q}) \boldsymbol{J}_r^T F_t - \boldsymbol{A}(\boldsymbol{q})\boldsymbol{\tau}, \end{aligned} \tag{13}$$

where, $a(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is a scalar representing the first term in the expression of F_n, and $\boldsymbol{A}(\boldsymbol{q})$ is an n line vector. As shown clearly in (13) that dimension of $\boldsymbol{\tau}$ is larger than the dimension of F_n, and F_n can be realized in the range space of $\boldsymbol{A}(\boldsymbol{q})$. This means $\boldsymbol{\tau}$ has a kind of redundancy against F_n. We named this redundancy appearing always in constraint dynamics of manipulator as constraint redundancy. $a(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and $\boldsymbol{A}(\boldsymbol{q})$ are defined concretely as follow:

$$\begin{aligned} a(\boldsymbol{q}, \dot{\boldsymbol{q}}) \triangleq &{m_c}^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}^T} \| \{-[\frac{\partial}{\partial \boldsymbol{q}^T}(\frac{\partial C}{\partial \boldsymbol{q}})\dot{\boldsymbol{q}}]\dot{\boldsymbol{q}} \\ &+(\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1}(\frac{1}{2}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}} + \boldsymbol{N}\dot{\boldsymbol{q}} + \boldsymbol{G})\}, \end{aligned} \tag{14}$$

$$\boldsymbol{A}(\boldsymbol{q}) \triangleq {m_c}^{-1} \| \frac{\partial C}{\partial \boldsymbol{r}^T} \| \{(\frac{\partial C}{\partial \boldsymbol{q}^T}) \boldsymbol{M}^{-1}\}, \tag{15}$$

(13) is written as follow for short:

$$F_n = F_n(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{\tau}, F_t). \tag{16}$$

From (9) and (12), we can get that:

$$\ddot{\boldsymbol{q}} = \boldsymbol{M}^{-1}(\boldsymbol{\tau} - \frac{1}{2}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}} - \boldsymbol{N}\dot{\boldsymbol{q}} - \boldsymbol{G} - \boldsymbol{J}_t^T F_t + \boldsymbol{J}_c^T F_n). \tag{17}$$

Inserting F_n calculated in (13) into (17), the state equation of the system excluding the constrained force (as $F_n > 0$) can be rewritten as

$$\begin{aligned} &\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{N}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) \\ = &\boldsymbol{J}_c^T(\boldsymbol{q}) a(\boldsymbol{q}, \dot{\boldsymbol{q}}) + (\boldsymbol{I} - \boldsymbol{J}_c^T \boldsymbol{A})\boldsymbol{\tau} + (\boldsymbol{J}_c^T \boldsymbol{A} - \boldsymbol{I}) \boldsymbol{J}_r^T F_t \end{aligned} \tag{18}$$

which is denoted as a model of the constraint dynamic system in Fig. 1. We have notice the right hand side of (18) of $(\boldsymbol{I} - \boldsymbol{J}_c^T \boldsymbol{A})$ and $(\boldsymbol{J}_c^T \boldsymbol{A} - \boldsymbol{I})$ in [17] more than 15 years ago, having sterling up our quality of how to use the interesting similarity in them for a new controller. But we still have no idea.

Solutions of these dynamic equation always satisfy the constrained condition (6). The forward description of contacting dynamics has been represented by (7) and (6). The fact that the solution $\boldsymbol{q}$ of (7) have to satisfy (6) make us anticipate that F_d should be satisfied simultaneously and instantly regardless of the motion $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$, and any $\boldsymbol{\tau}$. The algebraic solution has been derived as (13). Then the dynamics of the manipulator whose solution $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ always satisfy the constraint condition (8) derived from (6) has been translated into (18). In the Fig. 2, the backward relation of (13) and (18) are described in the right hand half. (13) exhibits clearly the comment "If contact is modeled by means of geometric constraints, then the contact forces cannot be expressed as algebraic functions of the state variables $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$," in pp.55 [15] is not correct and contradicting to. The backward description of constraint dynamics (14) has been long ignored by robotic researchers, but we had proposed the force sensorless position/force control based on using this backward description directly [17]. Here two descriptions on left and right side of Fig. 2 are equivalent, then it can be called a "dual system." In this paper , we propose a new controller with Lyapunov -stability over non-constraint motion and with instantaneous achievement of desired contacting force.

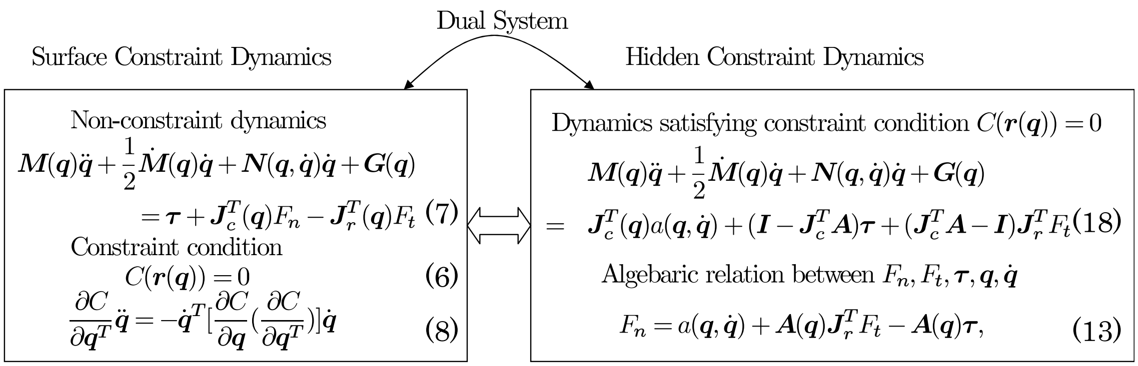


Fig. 2. Dual nature of manipulator under constraint condition.

III. Force and Position Controller

In this paper, we propose a controller whose convergence is guaranteed by Lyapunov method and introduce the calculate method of it.

A. Controller

Let $\boldsymbol{S}$ be a column full rank matrix spanning the null space of $\partial C/\partial \boldsymbol{q}$, we can get $\boldsymbol{S}^T\left(\partial C/\partial \boldsymbol{q}^T\right)^T=0$, i.e.

$$\boldsymbol{S}^T\boldsymbol{J}_c^T=\boldsymbol{0}. \tag{19}$$

It is possible to find an auxiliary vector $\boldsymbol{p}$ satisfies

$$\dot{\boldsymbol{q}}=\boldsymbol{S}\dot{\boldsymbol{p}}, \tag{20}$$

in the manipulator situation, $\boldsymbol{p}$ is the end-effector position except the constraint direction, and

$$\ddot{\boldsymbol{q}}=\dot{\boldsymbol{S}}\dot{\boldsymbol{p}}+\boldsymbol{S}\ddot{\boldsymbol{p}}. \tag{21}$$

From the definition of $\boldsymbol{A}$ in (15) and $\boldsymbol{S}$ in (19), we can get

$$\begin{aligned}\boldsymbol{S}^T\boldsymbol{A} &= {m_c}^{-1}\|\frac{\partial C}{\partial \boldsymbol{r}^T}\|\{\boldsymbol{S}^T(\frac{\partial C}{\partial \boldsymbol{q}^T})\boldsymbol{M}^{-1}\}\\ &= \boldsymbol{0}\end{aligned} \tag{22}$$

so $[\boldsymbol{A}^T,\boldsymbol{S}]$ is reversible, i.e. there certainly exist $\boldsymbol{B}$ ($n\times 1$ vector) and $\boldsymbol{D}$ ($n\times(n-1)$ matrix) satisfies

$$\begin{bmatrix}\boldsymbol{A}\\ \boldsymbol{S}^T\end{bmatrix}\boldsymbol{B}=\begin{bmatrix}1\\ \boldsymbol{0}\end{bmatrix} \tag{23}$$

$$\begin{bmatrix}\boldsymbol{A}\\ \boldsymbol{S}^T\end{bmatrix}\boldsymbol{D}=\begin{bmatrix}\boldsymbol{0}^T\\ \boldsymbol{I}_{n-1}\end{bmatrix} \tag{24}$$

respectively. Here $\boldsymbol{I}_{n-1}$ is a $(n-1)\times(n-1)$ identify matrix. $\boldsymbol{B}$ means selection matrix of range space of $\partial C/\partial \boldsymbol{q}$, which corresponds directly to the range space of $\boldsymbol{A}(\boldsymbol{q})$ as shown in (15) and null space of $\partial C/\partial \boldsymbol{q}^T$ being identical to the null space of $\boldsymbol{A}$. And $\boldsymbol{D}$ is vice versa.

Before proposing the controller we will put forward three assumptions:

(a)The constraint condition is known and expressed by $C(\boldsymbol{r}(\boldsymbol{q}))=0$.

(b)The tangential grinding force can be calculated by (3).

(c)The dynamic parameters of the system are known.

The following is a controller guaranteeing that the closed loop satisfies the exerted constrained force F_n be identical to the desired force F_{nd} regardless of time and the robot's motion along with the free motion directions.

$$\begin{aligned}\boldsymbol{\tau}=\boldsymbol{B}(F_{nd}-a)+\boldsymbol{D}\big[k_p(\boldsymbol{p}_d-\boldsymbol{p})+k_d(\dot{\boldsymbol{p}}_d-\dot{\boldsymbol{p}})\big]\\ +\boldsymbol{J}_r^TF_t\end{aligned} \tag{25}$$

Here on the right side, the first term is to realize the desired constrained force, the second term is to control the pose of the manipulator, while the third item is to compensate the friction, with an assumption of F_t being able to be gotten correctly. This assumption can be materialized by using F_n and (3). The block diagram of the system is given in Fig. 3.

Because (2) is a algebraic function of the input torque, when we substitute (25) into (13), we can get

$$\begin{aligned}F_n &= a(\boldsymbol{q},\dot{\boldsymbol{q}})+\boldsymbol{AB}(F_{nd}-a)\\ &+ \boldsymbol{AD}\big[k_p(\boldsymbol{p}_d-\boldsymbol{p})+k_d(\dot{\boldsymbol{p}}_d-\dot{\boldsymbol{p}})\big]\end{aligned} \tag{26}$$

from the definition we know that $\boldsymbol{AB}=1$ and $\boldsymbol{AD}=\boldsymbol{0}$ so $\boldsymbol{AB}(F_{nd}-a)=F_{nd}-a$ and $\boldsymbol{AD}\big[k_p(\boldsymbol{p}_d-\boldsymbol{p})+k_d(\dot{\boldsymbol{p}}_d-\dot{\boldsymbol{p}})\big]=0$, so

$$\begin{aligned}F_n &= a(\boldsymbol{q},\dot{\boldsymbol{q}})+F_{nd}-a(\boldsymbol{q},\dot{\boldsymbol{q}})\\ &= F_{nd},\end{aligned} \tag{27}$$

here (27) does not include the variable of time t as a time differential manner, meaning the output force always equals the desired one. When the $a(\boldsymbol{q},\dot{\boldsymbol{q}})$ calculated by (14) contains errors, denoted by $\hat{a}$, (27) will be

$$F_n=F_{nd}+a-\hat{a} \tag{28}$$

thus the exerted force would be suffered by the error stemming from measurements of $\boldsymbol{M}(q)$, $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$ and constraint condition $C=0$, and so on in (14). But the error would not be generated through dynamical behaviors of robot but geometric static algebraic relation depending on the results of dynamical behavior of motions $\boldsymbol{q}$, $\dot{\boldsymbol{q}}$.

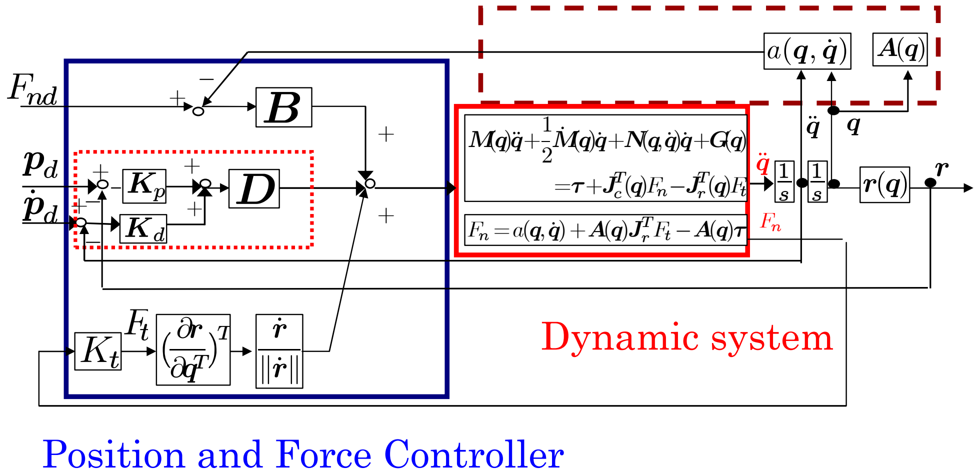


Fig. 3. Grinding position / force control system

B. Calculation of null space of $\boldsymbol{J}_c$, i.e., $\boldsymbol{S}$

From definition of Jacobian matrix $\boldsymbol{J}$ we can get

$$\dot{\boldsymbol{r}} = \boldsymbol{J}\dot{\boldsymbol{q}} \tag{29}$$

we can always find a matrix $\boldsymbol{P}$ whose row vectors are the orthogonal space of $\partial C/\partial \boldsymbol{r}^T$ since $\partial C/\partial \boldsymbol{r}^T$ cannot be a zero vector because of $C(\boldsymbol{r}) = 0$ is independant constraint condition, then ,

$$\boldsymbol{P} \perp \frac{\partial C}{\partial \boldsymbol{r}^T} \tag{30}$$

$$\dot{\boldsymbol{p}} = \boldsymbol{P}\boldsymbol{J}\dot{\boldsymbol{q}} = \tilde{\boldsymbol{J}}\dot{\boldsymbol{q}} \tag{31}$$

Here $\dot{\boldsymbol{p}}$ is a direction along the constraint condition. Define $\boldsymbol{P}\boldsymbol{J}$ as $\tilde{\boldsymbol{J}}$, taking (31) into (20) we can get

$$\tilde{\boldsymbol{J}}\boldsymbol{S}\dot{\boldsymbol{p}} = \dot{\boldsymbol{p}}. \tag{32}$$

$\tilde{\boldsymbol{J}}$ is a row full rank matrix of $(m-1)\times n$, so it is possible to find an $\boldsymbol{S} \in R^{n\times(m-1)}$ which satisfying $\tilde{\boldsymbol{J}}\boldsymbol{S} = \boldsymbol{I}_{n-1}$ ($\tilde{\boldsymbol{J}}\boldsymbol{S} \in R^{m-1\times(m-1)}$) is a solution of (32) $\tilde{\boldsymbol{J}}\boldsymbol{S} = \boldsymbol{I}_{n-1}$ and (19) $\boldsymbol{J}_c\boldsymbol{S} = 0$ can be combined into one matrix equation:

$$\begin{bmatrix} \tilde{\boldsymbol{J}} \\ \boldsymbol{J}_c \end{bmatrix} \boldsymbol{S} = \begin{bmatrix} \boldsymbol{I}_{n-1} \\ \boldsymbol{0} \end{bmatrix} \tag{33}$$

Here we define $\hat{\boldsymbol{J}} = [\tilde{\boldsymbol{J}}^T, \boldsymbol{J}_c^T]^T$. From the definition of $\boldsymbol{P}$ in (30) we know

$$\begin{bmatrix} \boldsymbol{P} \\ \frac{\partial C}{\partial \boldsymbol{r}^T}/\|\frac{\partial C}{\partial \boldsymbol{r}^T}\| \end{bmatrix} \in R^{m\times m} \tag{34}$$

is reversible. We assume the Jacobian matrix $\boldsymbol{J}$ is also row full rank matrix so

$$\begin{bmatrix} \boldsymbol{P} \\ \frac{\partial C}{\partial \boldsymbol{r}^T}/\|\frac{\partial C}{\partial \boldsymbol{r}^T}\| \end{bmatrix} \boldsymbol{J} = \begin{bmatrix} \tilde{\boldsymbol{J}} \\ \boldsymbol{J}_c \end{bmatrix} = \hat{\boldsymbol{J}} \tag{35}$$

$\hat{\boldsymbol{J}}$ is reversible, so $\boldsymbol{S}$ can be calculated as follow:

$$\boldsymbol{S} = \hat{\boldsymbol{J}}^{-1} \begin{bmatrix} \boldsymbol{I}_{n-1} \\ \boldsymbol{0} \end{bmatrix}. \tag{36}$$

By using $\boldsymbol{S}$, we can calculate $\boldsymbol{B}$ and $\boldsymbol{D}$ in (23) and (24), and calculate the input $\boldsymbol{\tau}$ in (25).

IV. Stability Analysis

Putting (20) into (7), premultiply $\boldsymbol{S}^T$ we can get

$$\begin{aligned} &\boldsymbol{S}^T\boldsymbol{M}\dot{\boldsymbol{S}}\dot{\boldsymbol{p}} + \boldsymbol{S}^T\boldsymbol{M}\boldsymbol{S}\ddot{\boldsymbol{p}} + \boldsymbol{S}^T(\frac{1}{2}\dot{\boldsymbol{M}} + \boldsymbol{N})\boldsymbol{S}\dot{\boldsymbol{p}} \\ &\quad = \boldsymbol{S}^T\boldsymbol{\tau} + \boldsymbol{S}^T\boldsymbol{J}_c^T F_n - \boldsymbol{S}^T\boldsymbol{J}_r^T F_t \end{aligned} \tag{37}$$

Substituting (25) into (37), from (23) we know $\boldsymbol{S}^T\boldsymbol{B} = \boldsymbol{0}$ we can get closed loop dynamics as,

$$\begin{aligned} &\boldsymbol{S}^T\boldsymbol{M}\dot{\boldsymbol{S}}\dot{\boldsymbol{p}} + \boldsymbol{S}^T\boldsymbol{M}\boldsymbol{S}\ddot{\boldsymbol{p}} + \boldsymbol{S}^T(\frac{1}{2}\dot{\boldsymbol{M}} + \boldsymbol{N})\boldsymbol{S}\dot{\boldsymbol{p}} \\ &\quad = k_p(\boldsymbol{p}_d - \boldsymbol{p}) + k_d(\dot{\boldsymbol{p}}_d - \dot{\boldsymbol{p}}) \end{aligned} \tag{38}$$

From the equation above we can see because $\boldsymbol{B}$ is defined in the null space of $\boldsymbol{S}$,which will not affect the motion of the end effector along the constraint surface, here we set the desired end-effector position is constant which means $\dot{\boldsymbol{p}}_d = 0$, so closed loop dynamics is

$$\begin{aligned} &\boldsymbol{S}^T\boldsymbol{M}\dot{\boldsymbol{S}}\dot{\boldsymbol{p}} + \boldsymbol{S}^T\boldsymbol{M}\boldsymbol{S}\ddot{\boldsymbol{p}} + \frac{1}{2}\boldsymbol{S}^T\dot{\boldsymbol{M}}\boldsymbol{S}\dot{\boldsymbol{p}} - k_p(\boldsymbol{p}_d - \boldsymbol{p}) \\ &\quad = -\boldsymbol{S}^T\boldsymbol{N}\boldsymbol{S}\dot{\boldsymbol{p}} - k_d\dot{\boldsymbol{p}}. \end{aligned} \tag{39}$$

Set Lyapunov argument as:

$$V = \frac{1}{2}\dot{\boldsymbol{p}}^T\boldsymbol{S}^T\boldsymbol{M}\boldsymbol{S}\dot{\boldsymbol{p}} + \frac{1}{2}(\boldsymbol{p}_d - \boldsymbol{p})^T k_p(\boldsymbol{p}_d - \boldsymbol{p}) \tag{40}$$

so

$$\begin{aligned} \dot{V} &= \dot{\boldsymbol{p}}^T\boldsymbol{S}^T\boldsymbol{M}\dot{\boldsymbol{S}}\dot{\boldsymbol{p}} + \dot{\boldsymbol{p}}^T\boldsymbol{S}^T\boldsymbol{M}\boldsymbol{S}\ddot{\boldsymbol{p}} \\ &+ \frac{1}{2}\dot{\boldsymbol{p}}^T\boldsymbol{S}^T\dot{\boldsymbol{M}}\boldsymbol{S}\dot{\boldsymbol{p}} - k_p\dot{\boldsymbol{p}}^T(\boldsymbol{p}_d - \boldsymbol{p}) \end{aligned} \tag{41}$$

From (39), (41) can be transformed to

$$\dot{V} = -\dot{\boldsymbol{p}}^T\boldsymbol{S}^T\boldsymbol{N}\boldsymbol{S}\dot{\boldsymbol{p}} - k_d\dot{\boldsymbol{p}}^T\dot{\boldsymbol{p}} \tag{42}$$

because $\boldsymbol{N}$ is a skew symmetrical matrix, $\dot{\boldsymbol{p}}^T\boldsymbol{S}^T\boldsymbol{N}\boldsymbol{S}\dot{\boldsymbol{p}} = 0$, so

$$\dot{V} = -k_d\dot{\boldsymbol{p}}^T\dot{\boldsymbol{p}} \tag{43}$$

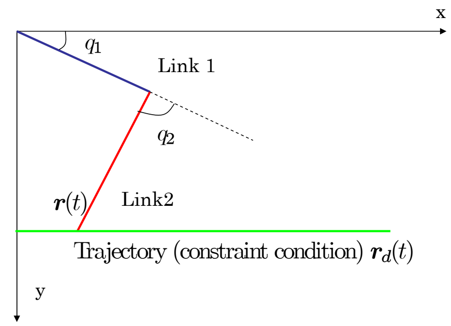


Fig. 4. 2-link grinding robot

because $\dot{V} \leq 0$ and $V \geq 0$, from (43) we can see if and only if $\dot{\boldsymbol{p}} = \boldsymbol{0}$, $\ddot{V} = \boldsymbol{0}$. Submitting $\dot{\boldsymbol{p}} = \boldsymbol{0}$ into (39) we can get $\boldsymbol{p}_d - \boldsymbol{p} = \boldsymbol{0}$, Lasalle theorem bring

$$\lim_{t\to\infty} \boldsymbol{p} = \boldsymbol{p}_d, \lim_{t\to\infty} \dot{\boldsymbol{p}} = \boldsymbol{0} \tag{44}$$

Because the robot's kinematics satisfy $\boldsymbol{r}_d = \boldsymbol{f}(\boldsymbol{q}_d)$, it leads to

$$\lim_{t\to\infty} \boldsymbol{r} = \boldsymbol{r}_d. \tag{45}$$

Once more we can see that there is no F_n or a in the process of the convergence proof from (38), which means that the system will converge to the desired pose even if there exist some error between F_n and F_{nd}.

V. Simulation

In this section we will introduce some simulations have been done to check the controller in 2-link condition as Fig. 4. To a 2-link manipulator the variables in (18) can be calculated as follow:

$$\boldsymbol{M} = \begin{bmatrix} J_1 + J_2 + 2\beta\cos q_2 & J_2 + 2\beta\cos q_2 \\ J_2 + 2\beta\cos q_2 & J_2 \end{bmatrix} \tag{46}$$

$$\frac{1}{2}\dot{\boldsymbol{M}}\dot{\boldsymbol{q}} + N\dot{\boldsymbol{q}} = \begin{bmatrix} -(2\dot{q}_1\dot{q}_2 + \dot{q}_1^2)\beta\sin q_2 \\ \dot{q}_1^2\beta\sin q_2 \end{bmatrix} \tag{47}$$

here $J_1 = I_1 + (m_1 + 4m_2)l_1^2$, $J_2 = I_2 + m_2 l_2^2$ and $\beta = 2m_2 l_1 l_2$ and I, m, l are the initial moment, mass and length of the links. Jacobian matrix $\boldsymbol{J}$ is.

$$\boldsymbol{J} = \begin{bmatrix} -\sin q_1 - \sin(q_1 + q_2) & -\sin(q_1 + q_2) \\ \cos q_1 + \cos(q_1 + q_2) & \cos(q_1 + q_2) \end{bmatrix} \tag{48}$$

In the simulation we set the constraint condition as:

$$C(\boldsymbol{r}(\boldsymbol{q})) = y - 0.6 = 0 \tag{49}$$

so

$$\frac{\partial C}{\partial \boldsymbol{r}} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad \dot{\boldsymbol{r}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{50}$$

so

$$\boldsymbol{J}_c = \begin{bmatrix} \cos(q_1 + q_2) \\ \cos q_1 + \cos(q_1 + q_2) \end{bmatrix} \tag{51}$$

$$\boldsymbol{J}_r = \begin{bmatrix} -\sin(q_1 + q_2) \\ -\sin q_1 - \sin(q_1 + q_2) \end{bmatrix} \tag{52}$$

from the variables above we can calculate a, $\boldsymbol{A}$ and m_c defined in (14) (15) and (11) and calculate F_n by (18).

For 2-link manipulator, $\boldsymbol{S}$ and $\partial C/\partial \boldsymbol{q}$ in (19) are both 2×1 vectors, we can get $\boldsymbol{S}^T\left(\partial C/\partial \boldsymbol{q}^T\right) = 0$, i.e.

$$\boldsymbol{S}^T \boldsymbol{J}_c^T = 0 \tag{53}$$

This $\boldsymbol{S}$ also satisfies the following equation,

$$\dot{\boldsymbol{q}} = \boldsymbol{S}\dot{p}_x \tag{54}$$

where $\dot{p}_x$ is the end-effector position on x-axis, here we define the two elements of $\boldsymbol{S}$ as $\boldsymbol{S} = [S_1, S_2]$ and $\boldsymbol{J}_c = [J_{c1}, J_{c2}]$, also the two angles of the joints are define as q_1 and q_2 respectively, so (54) can be written as

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} \dot{p}_x \tag{55}$$

from the definition of Jacobian matrix we can know

$$[J_{11}, J_{12}] \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = \dot{p}_x \tag{56}$$

here $[J_{11}, J_{12}]$ is the first row vector of Jacobian matrix and also the $\tilde{\boldsymbol{J}}$ in (31). To get $\boldsymbol{S}$ satisfies $\tilde{\boldsymbol{J}}\boldsymbol{S} = \boldsymbol{I}$, take (55) into (56), we can get

$$[J_{11}, J_{12}] \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = 1 \tag{57}$$

Combine (47) and (57), we have

$$\hat{\boldsymbol{J}}\boldsymbol{S} = \begin{bmatrix} J_{11} & J_{12} \\ J_{c1} & J_{c2} \end{bmatrix} \begin{bmatrix} S_1 \\ S_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{58}$$

which is corresponding to (33). Then we have, $\boldsymbol{S}$ as

$$\boldsymbol{S} = \hat{\boldsymbol{J}}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} \dfrac{J_{c2}}{J_{c2}J_{11} - J_{c1}J_{12}} \\ -\dfrac{J_{c1}}{J_{c2}J_{11} - J_{c1}J_{12}} \end{bmatrix}. \tag{59}$$

Because there are only 2 links, $\boldsymbol{B}$ and $\boldsymbol{D}$ in (25) can be determined as,

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{S}^T \end{bmatrix}^{-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{60}$$

$$\boldsymbol{D} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{S}^T \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{61}$$

The dual nature of manipulator under constraint condition in Fig. 2 for 2-link manipulator can be written as Fig. 5.

Then we did some simulations to check the controller, first we set the desired force as a step input, in this simulation $x_d = 0.5[m]$, $F_{nd} = 5[N]$ $k_p = 1000$ and $k_d = 300, 100, 30$ respectively, the result is shown in Fig. 6, the constraint force and y-position coincide and all x-position can converge to the desired position.

In the second simulation I will show the data of the simulation when $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 5[N]$. From Fig. 7 (a) we can see that the system will output the desired force. And from (b) and (c) we can see that the controller can control the end-effector move along the desired trajectory.

Surface Constraint Dynamics

Non-constraint dynamics

$$\begin{bmatrix} J_1+J_2+2\beta\cos q_2 & J_2+2\beta\cos q_2 \\ J_2+2\beta\cos q_2 & J_2 \end{bmatrix}\ddot{\boldsymbol{q}}+\begin{bmatrix} -(2\dot{q}_1\dot{q}_2+\dot{q}_1^2)\beta\sin q_2 \\ \dot{q}_1^2\beta\sin q_2 \end{bmatrix}=\boldsymbol{\tau}+\begin{bmatrix} \cos(q_1+q_2) \\ \cos q_1+\cos(q_1+q_2) \end{bmatrix}F_n+K_t\begin{bmatrix} -\sin(q_1+q_2) \\ -\sin q_1-\sin(q_1+q_2) \end{bmatrix}F_t$$

Constraint condition

$$y-0.6=0$$

$$\ddot{q}_2=-sin(q_1+q_2)\dot{q}_1^2-\big(sinq_1+sin(q_1+q_2)\big)\dot{q}_1\dot{q}_2-sin(q_1+q_2)\dot{q}_2^2$$

Hidden Constraint Dynamics

Dual System

Dynamics satisfying constraint condition $y-0.6=0$

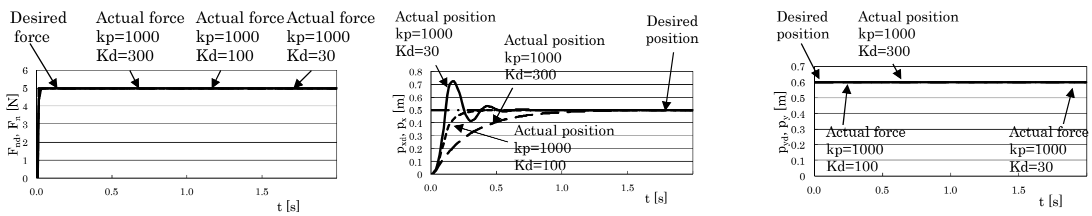
$$\begin{bmatrix} J_1+J_2+2\beta\cos q_2 & J_2+2\beta\cos q_2 \\ J_2+2\beta\cos q_2 & J_2 \end{bmatrix}\ddot{\boldsymbol{q}}+\begin{bmatrix} -(2\dot{q}_1\dot{q}_2+\dot{q}_1^2)\beta\sin q_2 \\ \dot{q}_1^2\beta\sin q_2 \end{bmatrix}$$

$$=\begin{bmatrix} \cos(q_1+q_2) \\ \cos q_1+\cos(q_1+q_2) \end{bmatrix}a(\boldsymbol{q},\dot{\boldsymbol{q}})+(\boldsymbol{I}-\boldsymbol{J}_c^T\boldsymbol{A})\boldsymbol{\tau}+(\boldsymbol{J}_c^T\boldsymbol{A}-\boldsymbol{I})\begin{bmatrix} -\sin(q_1+q_2) \\ -\sin q_1-\sin(q_1+q_2) \end{bmatrix}F_t$$

Algebaric relation between $F_n, F_t, \boldsymbol{\tau}, \boldsymbol{q}, \dot{\boldsymbol{q}}$

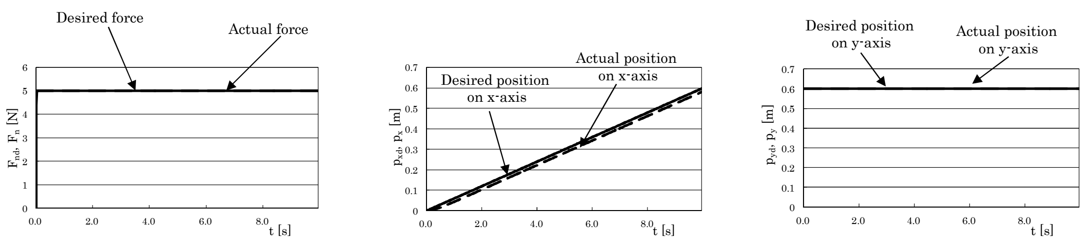
$$F_n=a(\boldsymbol{q},\dot{\boldsymbol{q}})+\boldsymbol{A}(\boldsymbol{q})\boldsymbol{J}_r^T F_t-\boldsymbol{A}(\boldsymbol{q})\boldsymbol{\tau},$$

Fig. 5. Dual nature of manipulator under constraint condition of two link grinding robot



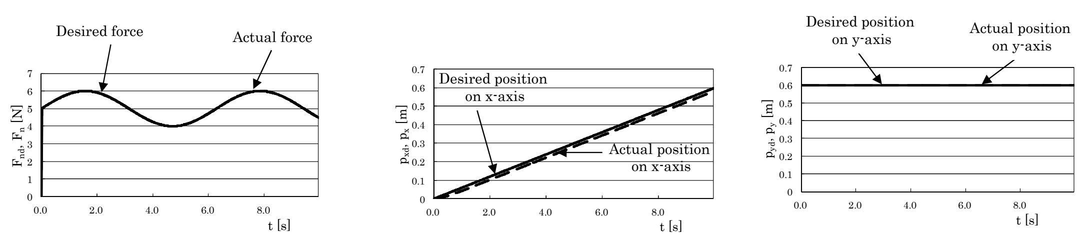
(a) Actual and desired constraint force (b) Actual and desired trajectory on x-axis (c) Actual and desired trajectory on y-axis

Fig. 6. Simulation result when $k_p = 1000$, $x_d = 0.5[m]$ and $F_{nd} = 5[N]$



(a) Actual and desired constraint force (b) Actual and desired trajectory on x-axis (c) Actual and desired trajectory on y-axis

Fig. 7. Simulation result when $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 5[N]$



(a) Actual and desired constraint force (b) Actual and desired trajectory on x-axis (c) Actual and desired trajectory on y-axis

Fig. 8. Simulation result when $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 5 + \sin t[N]$

In the third simulation we make the desired force as a function of time, in this simulation $k_p = 1000$, $k_d = 300$, $x_d = 0.06t[m]$ and $F_{nd} = 5+\sin t[N]$ the result is shown in Fig. 8, from the simulation result we can see that however the force changes, the end-effector tracks the desired position on $x-$axis while moves along the constraint line on y-axis. Just as it is explained, the surface constraint dynamics expresses the system by the equation of dynamics and constraint condition, but this is only a condition but not a limitation. In the hidden constraint dynamics, the constraint condition is combined into the dynamics equation, so no matter how much the input τ is, it just affect the constraint force, and the end-effector motion on the tangent direction, but the end-effector always moves along the constraint line of plane. This is the reason why we call the two dynamics dual system.

VI. Conclusions

In this paper we designed a constraint-combined force/position controller for the continuous shape-grinding system, and prove the convergence of the controller in a new way by Lyapunov method, the output force of the system can always equal to the desired one. At last we did some simulations to confirm the controller. In the future we will apply it into experiment.

References

[1] Jaydeep. Roy and Louis L. Whitcomb: Adaptive Force Control of Position/Velocity Controlled Robots: Theory and Experiment, IEEE Transactions on Robotics and Automation, Vol.18, No.2, April, 2002.

[2] N. Hogan: Impedance control: An approach to manipulation: Part 1-Theory, J.Dynamic Syst., Measurement Contr., Vol.107, pp.1-7, Mar, 1985.

[3] M.H.Raibert and J.J.Craig: Hybrid Position/Force Control of Manipulators, Trans. of the ASME, J. of Dynamic Systems, Measurement and Control, Vol.102, pp.126-133, June, 1981.

[4] B. Siciliano and L. Villani: A passivity-based approach to force regulation and motion control of robot manipulators, Automatica, Vol.32, No.3,pp.443-447, Mar, 1996.

[5] L. Villani, C.C. de Wit and B. Brogliato: An exponentially stable adaptive control for force and position tracking of robot manipulators, IEEE Trans. Automat. Contr., Vol44, pp.778-802, Apr, 1999.

[6] J. De Schutter and H.Van Brussel: Compliant robot motion 2. A control approach based on external control loops, Int. J. Robot. Res., Vol.7, No.4, pp.18-33, Aug, 1988.

[7] T. Yoshikawa: Dynamic Hybrid Position/Force control of Robot Manipulators — Description of Hand Constraints and Calculation of Joint Driving Force, IEEE J. on Robotics and Automation, Vol.RA-3, No.5, pp.386-392, 1987.

[8] Z. X. Peng and N. Adachi: Position and Force Control of Manipulators without Using Force Sensors(in Japanese), Trans. of JSME(C), Vol.57, No.537, pp.1625-1630, 1991.

[9] D.Wang and N.H. McClamroch: Position and Force Control for Constrained Manipulator Motion: Lyapunov's Direct Approach, IEEE Trans. Robotics and Automation, Vol.9, pp.308-313, 1993.

[10] S. Arimoto, Mechanics and Control of Robot, Asakura Publishing Co., Ltd., Tokyo, Japan, 1990.

[11] McClamroch, N.H., and Wang, Danwei, Feedback stabilization and tracking of constrained robots, IEEE Transaction on Automatic Control, Vol.33, No.5, May 1988, pp.419-426.

[12] Katsu Yamane, Yoshihiko Nakamura. "O(N) Forward Dynamics Computation of Open Kinematic Chains Based on the Principle of Virtual Work",pp2824–2831

[13] K.Yamane and Y.Nakamura. "Dynamics Filter - Concept and Implementation of On-Line Motion Generator for Human Figures" IEEE Transactions on Robotics and Automation, vol. 19, no. 3, pp. 421-432, 2003.

[14] Yamane,K.,Nakamura,Y. "Dynamics Filter - Concept and Implementation of On-Line Motion Generator for Human Figures",pp688-695 ICRA'00

[15] Miomir Vukobratovic, Veljko Potkonjak, Vladimir Matijevic "Dynamics of robots with contact tasks", pp.55, Springer, 2003

[16] Hooshang Hemami and Bostwick F. Wyman: Modeling and Control of Constrained Dynamic Systems with Application to Biped Locomotion in the Frontal Plane, IEEE Trans. on Automatic Control, Vol.AC-24, No.4, pp.526-535, 1979.

[17] M.Minami, T.Asakura, L.X.Dong, Y.M.Huang. "Position Control and Explicit Force Control of Constrained Manipulators for Accurate Grinding Tasks" Advanced Robotics, Int. J. of RSJ, Vol.11, No.3, 285-230(1997).

[18] Takeshi Ikeda, Mamoru Minami, Position/Force Control of a Manipulator by Using an Algebraic Relation and Evaluations by Experiments, The 2003 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics (AIM2003), Proceeding CD-ROM paper No.159, 503-508, 2003

[19] Takeshi Ikeda and Mamoru Minami: Research of Grinding Robot without Force Sensor by Using Algebraic Equation(in Japanese), Transactions of the Japan Society of Mechanical Engineers(C), Vol.71, No.702, pp.270-277, 2005.

[20] Mamoru Minami, Weiwei Xu, Shape-grinding by Direct Position / Force Control with On-line Constraint Estimation, 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Acropolis Convention Center, Nice, France, Sept, 22-26, 2008. pp. 943-948

[21] Fujia Yu, Mamoru Minami, Akira Yanou, Dual Nature of Constraint Motion and Lyapunov-stable Position/Force Control Method, 2nd SICE Symposium on Computational Intelligence (domestic symposium presented in Japanese), pp.65-72,2012