

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Tracking Control for Multiple Trailer Systems by Adaptive Algorithmic Control

Tomoaki Kobayashi, Toru Yoshida, Junichi Maenishi,
Joe Imae and Guisheng Zhai
*Osaka Prefecture University
Japan*

1. Introduction

In recent years, a truck-trailer system is the most useful physical distribution system. The truck-trailer systems have more convenience than coastal services or freight trains. Meanwhile, problems of the traffic jam and the air pollution in an urban area have become serious, year after year. Therefore improvement and rationalization of the transport efficiency are social needs. There are many papers suggesting a platoon system of several trucks as a part of development of ITS (Intelligent Transport System). These platoon systems consist of several unmanned trucks automatically following a truck driven by a conductor, and it is commonly believed that it brings improvements of energy efficiency along with alleviation of the traffic jam. Moreover, there is a purpose of covering insufficient workforce of truck drivers who have to do severe labors, too. In the platoon, trucks are not physically connected to each other, and thus there is much flexibility. On the other hand, even if each vehicle is physically connected by mechanical linkage, this is not important restrictions, for transport robots which are operated in the factory, because moving range and action plan are limited. Moreover, the multiple trailer system is safer than platoon system, because if each vehicle is physically connected, there is no danger of collision among trailers. In this paper, we deal with a control method for a physically connected multiple trailer robot, which is a transport system in factories.

The control method of connected vehicle has been studied for a long time (Laumond, 1986). In particular, there are many papers which studied controlling its backward motion with guaranteed stability (Sampei & Kobayashi, 1994). Moreover, kinematical model of a multiple trailer system is described by a nonholonomic system, and it is a controllable nonlinear system (Hermann & Krener, 1977). In theoretical field, it has been a hot subject of research, because asymptotic stabilization is impossible using one continuous time-invariant since the nonholonomic system does not satisfy the Brockett's necessary condition for stabilizability (Brockett, 1983). Therefore, the control problem of nonholonomic system is a theoretically difficult problem, thereupon various researches such as time-variant controller (M'Closkey & Murray, 1993) or hybrid control techniques (Matsune et al., 2005) are performed. We look at this issue from more practical point of view, then investigate a real-time control algorithm, which is based on the so called algorithmic control (Kobayashi et al., 2005a), (Imae et al., 2005) with a similar formulation of the model predictive control (MPC)

technique for nonlinear continuous time system. Our algorithmic design approach is a technique for ensuring robustness by adopting a numeric solution called Riccati Equation Based (REB) algorithm using quasi linearization that includes feedback solution. Moreover, though details are described later, the control technique by algorithmic design which we proposed is an effective method for nonholonomic systems because our method is switching and applying the control strategy on a short control interval and thus the controller is discontinuous time variant, which does not violate Brockett's theorem. We showed the effectiveness of proposed method applicable to nonholonomic systems through some simulations and an experiment with a differential-driven unicycle vehicle model (Kobayashi et al., 2005b). Then, we extend our design method by incorporating numerical robustness for disturbances and parameter uncertainties and, by focusing on the switching interval of control strategy on iterative process of algorithmic design (Kobayashi et al., 2006). We discussed about effectiveness of our approach for an unstable motion control of high order nonlinear system, in this paper. In the most of conventional research, the direct-hooked type model (Lee et al., 2001) is treated. The direct-hooked model can be transformed to a canonical form called chained form (Murray & Sastry, 1993). Then, control problem for the direct-hooked model can be reduced to a canonical problem. However, the direct-hooked model has a tracking error of follow-on trailers (Fig.1). Therefore, there are many suggestions for eliminating the tracking error by model constructions or mechanical linkage design. We pick up a off-hooked model (Lee et al., 2004) which has a most simple structure and cannot be converted to canonical form (Ishikawa, 1993). Therefore, proposed algorithmic design is considered as an effective strategy for the off-hooked trailer system, because our approach can treat the general nonlinear systems. The effectiveness is discussed through a numerical simulation result.

The outline of this paper is as follows. In section 2, we describe the nonlinear optimal control problems and the Riccati Equation Based algorithm. In section 3, the algorithmic design method is described in detail. Also, we make an extension of our design method for robustness. The backward motion control problem of multiple trailer systems is formulated in section 4. In section 5, we show some simulation results in order to demonstrate the effectiveness of adaptive algorithmic design. Section 6 concludes the paper.

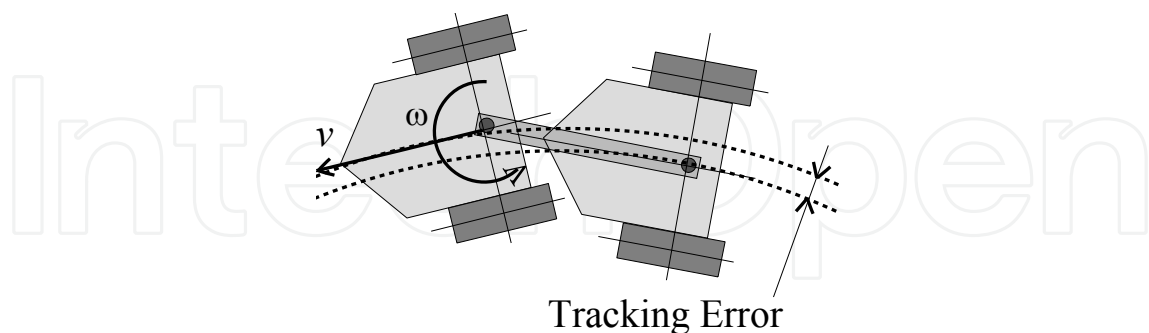


Fig. 1 Tracking error of the direct-hooked trailer system

2. Optimal control problem

2.1 Formulation

We deal with the following general nonlinear system

$$\dot{x}(t) = f(t, x(t), u(t)) \quad (1)$$

$$x(t_0) = x_0 \in \mathfrak{R}^n \quad (2)$$

where t_0 is initial time, x_0 is initial state given. Here, we denote the state variable by $x(t) = [x_1(t), \dots, x_n(t)]^T \in \mathfrak{R}^n$, and the input variable by $u(t) = [u_1(t), \dots, u_r(t)]^T \in \mathfrak{R}^r$. Then, the purpose is to find the controller which minimizes a performance index J over a time interval $[t_0, t_1]$.

$$J = G(x(t_1)) + \int_{t_0}^{t_1} L(t, x(t), u(t)) dt \quad (3)$$

Based on the problem formulation (1) to (2), we describe our on-line computational design method, that is to say, algorithmic design method (Kobayashi et al., 2005a).

It is known that whether or not the algorithmic design method succeeds depends on how effective the algorithm is to iteratively search the numerical solutions of optimal control problems. In this paper, we adopt one of the so-called Riccati-equation based algorithms (REB algorithms (Imae & Torisu, 1998)), which is known to be reliable and effective in searching numerical solutions. Details are given later.

2.2 Riccati-equation based algorithm

Under the problem formulation (1) to (3), we describe an iterative algorithm for the numerical solutions of optimal control problems, based on Riccati differential equations. In this respect, the algorithm falls in the category of optimal control algorithms, as presented in (Nedeljkovic, 1981), (Imae et al., 1992), and so on.

[Assumptions]

Let $x: [t_0, t_1] \rightarrow \mathfrak{R}^n$ be an absolutely continuous function, and $u: [t_0, t_1] \rightarrow \mathfrak{R}^r$ be an essentially bounded measurable function. For each positive integer j , let us denote by AC^j all absolutely continuous functions: $[t_0, t_1] \rightarrow \mathfrak{R}^j$, and by L_∞^j all essentially bounded measurable functions: $[t_0, t_1] \rightarrow \mathfrak{R}^j$. Moreover, we define the following norms on AC^j and L_∞^j respectively:

$$\begin{aligned} \|x\| &= \max |x(t)| \text{ for } x \in AC^j, t \in [t_0, t_1] \\ \|y\| &= \text{ess sup } |y(t)| \text{ for } y \in L_\infty^j, t \in [t_0, t_1] \end{aligned}$$

where the vertical bars are used to denote Euclidean norms for vectors.

Now, we make some assumptions.

- i. $G: \mathfrak{R}^n \rightarrow \mathfrak{R}^1$, $f: \mathfrak{R}^1 \times \mathfrak{R}^n \times \mathfrak{R}^r \rightarrow \mathfrak{R}^n$, $L: \mathfrak{R}^1 \times \mathfrak{R}^n \times \mathfrak{R}^r \rightarrow \mathfrak{R}^1$ are continuous in all their arguments, and their partial derivatives $G_x(x)$, $f_x(t, x, u)$, $f_u(t, x, u)$, $L_x(t, x, u)$ and $L_u(t, x, u)$ exist and are continuous in all their arguments.
- ii. For each compact set $U \subset \mathfrak{R}^r$ there exists some $M_1 \in (0, \infty)$ such that

$$|f(t, x, u)| \leq M_1(|x| + 1) \quad (4)$$

for all $t \in \mathfrak{R}^1$, $x \in \mathfrak{R}^n$ and $u \in U$.

[Algorithm]

STEP A0 Let $\beta \in (0,1)$ and $M_2 \in (0,1)$. Select arbitrarily an initial input $u^0 \in L_\infty^r$.

STEP A1 $i = 0$.

STEP A2 Calculate $x^i(t)$ with $u^i(t)$ from the equation (1).

STEP A3 Select $A^i \in \mathfrak{R}^{n \times n}$, $B_{11}^i \in L_\infty^{n \times n}$, $B_{12}^i \in L_\infty^{n \times r}$ and $B_{22}^i \in L_\infty^{r \times r}$ so that Kalman's sufficient conditions for the boundedness of Riccati solutions (Jacobson & Mayne, 1970) hold, that is, for almost all $t \in [t_0, t_1]$,

$$\begin{aligned} A^i(t) &\geq 0 \\ B_{22}^i(t) &> 0 \\ B_{11}^i(t) - B_{12}^i(t)B_{22}^i(t)^{-1}B_{12}^i(t)^T &\geq 0 \end{aligned} \quad (5)$$

where A^i, B_{11}^i and B_{22}^i are symmetric and $(\cdot)^T$ means the transpose of vectors and matrices. We solve (6), (7), and (8) with respect to δx , K , r and denote the solutions as $\delta x^i(t)$, $K^i(t)$, $r^i(t)$.

$$\begin{aligned} \delta \dot{x}(t) &= \{f_x(t, x^i, u^i) + f_u(t, x^i, u^i)B_{22}^i{}^{-1}(f_u^T(t, x^i, u^i)K(t) - B_{12}^i{}^T)\} \delta x(t) \\ &\quad + f_u(t, x^i, u^i)B_{22}^i{}^{-1}(f_u^T(t, x^i, u^i)r(t) - L_u^T(t, x^i, u^i)), \\ \delta x(t_0) &= 0, \end{aligned} \quad (6)$$

$$\begin{aligned} \dot{K}(t) &= -K(t)f_x(t, x^i, u^i) - f_x^T(t, x^i, u^i)K(t) + B_{11}^i \\ &\quad + (K(t)f_u(t, x^i, u^i) - B_{12}^i)B_{22}^i{}^{-1}(B_{12}^i{}^T - f_u^T(t, x^i, u^i)K(t)), \\ K(t_1) &= -A^i, \end{aligned} \quad (7)$$

$$\begin{aligned} \dot{r}(t) &= -f_x^T(t, x^i, u^i)r(t) + L_x^T(t, x^i, u^i) \\ &\quad + \{B_{12}^i - K(t)f_u^T(t, x^i, u^i)\}B_{22}^i{}^{-1}(-L_u^T(t, x^i, u^i) + f_u^T(t, x^i, u^i)r(t)), \\ r(t_1) &= -G(x(t_1)), \end{aligned} \quad (8)$$

and determine δu^i as follows.

$$\begin{aligned} \delta u^i(t) &= B_{22}^i{}^{-1}\{(f_u^T(t, x^i, u^i)K^i(t) - B_{12}^i{}^T)\delta x^i \\ &\quad + f_u^T(t, x^i, u^i)r^i(t) - L_u^T(t, x^i, u^i)\}. \end{aligned} \quad (9)$$

STEP A4 Determine $(\tilde{x}^i, \tilde{u}^i)$ satisfying

$$\begin{aligned} \dot{x}(t) &= f(t, x(t), u(t)) \\ x(t_0) &= x_0 \in \mathfrak{R}^n \\ H^i(t, (x - x^i), (u - u^i), p^i) &= \max_{v \in \mathfrak{R}^r} H^i(t, (x - x^i), (u - u^i), p^i) \end{aligned}$$

where

$$\begin{aligned} H^i(t, \delta x, \delta u, p) = & -\{L_x(t, x^i, u^i)\delta x + L_u(t, x^i, u^i)\delta u \\ & + \frac{1}{2}(\delta x^T B_{11}^i \delta x + 2\delta x^T B_{12}^i \delta u + \delta u^T B_{22}^i \delta u)\} \\ & + p^T (f_x(t, x^i, u^i)\delta x + f_u(t, x^i, u^i)\delta u) \end{aligned}$$

and p^i is the solution of the following equation.

$$\begin{aligned} \dot{p}(t) = & -f_x^T(t, x^i, u^i)p(t) + L_x^T(t, x^i, u^i) \\ p(t_1) = & -G_x^T(x(t_1)) \end{aligned}$$

STEP A5 $\alpha_i = 1$.

STEP A6 Set $u^{i+1}(t) = u^i(t) + \alpha_i \delta u^i(t) + \alpha_i^2 (\tilde{u}^i(t) - u^i(t) - \delta u^i(t))$.

if (10) holds, go to Step A7. Otherwise, set $\alpha_i = \beta \alpha_i$ and repeat Step A6.

$$\begin{aligned} J(u^{i+1}) - J(u^i) \leq & \alpha_i M_2 \{G(x_i(t_1))\delta x(t_1) \\ & + \int_{t_0}^{t_1} (L_x(t, x^i, u^i)\delta x^i + L_u(t, x^i, u^i)\delta u^i) dt\} \end{aligned} \quad (10)$$

STEP A7 Set $i = i + 1$, and go to Step A2. Repeat Step A2 to Step A7 until the performance index J converges. Here, the integer i represents the number of iterations.

3. Algorithmic design

3.1 Real time control technique

In this section, we describe the outline of the algorithmic design for real time control of nonlinear system. See (Imae et al., 2005), (Kobayashi et al., 2005a) for more details. The basic idea of this real-time control design is the control strategy u^N is executed one by one through N iterations of the above-mentioned REB algorithm from Step A2 to Step A7. In this design method, the controller is not needed in an explicit expression, and the control strategy is decided repeatedly by the REB algorithm. After the actual states are observed, the states of the next ΔT seconds from now are predicted by the state equation (1). Then, with the predicted states set as initial states, we obtain the next control strategy u^N by N iterations of the REB algorithm from Step A2 to Step A7. Through sufficiently large number of iterations N , it could be expected to eventually reach the possible optimal solutions. However, the value of N should be decided for the iterative processing to end in the ΔT [sec]. We here describe how the algorithmic controller works. See also figure 1. Here, the feedback structure of the solution in (Imae et al., 2005) and (Kobayashi et al., 2005a) is not adopted for simplification of computation.

[Real Time Algorithm]

STEP B1 Let $k = 0$. Select arbitrarily an initial input u_k^N .

STEP B2 Measure the actual state x_{ak} , and apply the input u_k^N to the plant over the interval of the unit time of calculation ΔT . During this time interval, we proceed with two kinds of calculations: One is to predict the one-unit-time-ahead state $x_{p(k+1)}$ through the system equation (1) with the initial state x_{ak} , and the other is to calculate

the N -iteration-ahead solution with the updated initial state $x_{p(k+1)}$. Then, we obtain the next control strategy u_{k+1}^N . If the rate of the value of performance index is less than a sufficiently small value γ , that is if following inequalities are satisfied, stop the iteration because it seems that the optimal solution was obtained.

$$\left| \frac{J(u^{i+1}) - J(u^i)}{J(u^i)} \right| < \gamma \text{ or } |J(u^i)| < \gamma \quad (11)$$

STEP B3 Set $k = k + 1$, and go to Step B2.

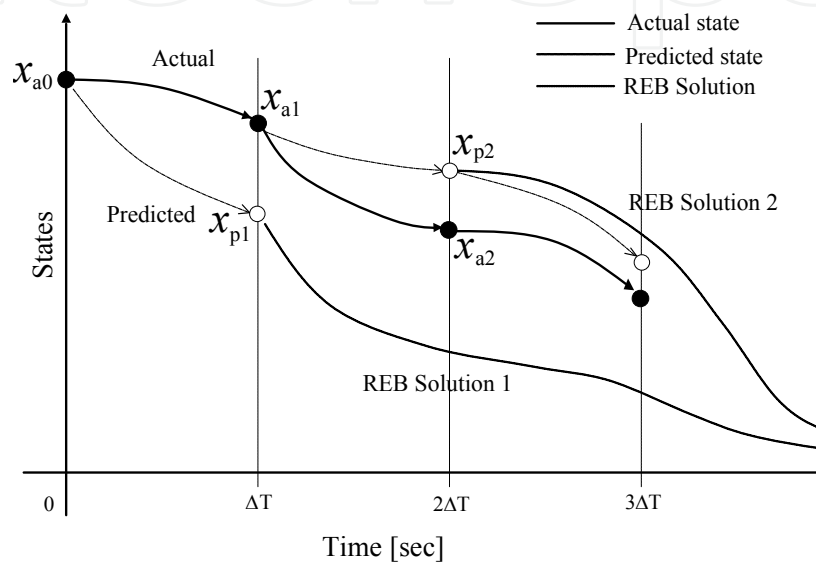


Fig. 2 Optimal / actual trajectory.

In our previous works, we verified the effectiveness of our algorithmic approach by applying to various nonlinear systems. For example, we tried a swing-up problem of inverted pendulum, or the obstacle avoidance problem for a unicycle robot. As a result, our approach gave the effective solution for these problems. The backward motion control problem for the multiple trailer system that we treat in this paper is a more difficult problem, because the system is a higher order nonlinear system. In spite of these difficulties, we confirmed the effectiveness of our algorithmic approach for such a complex problem through some numerical simulations. However, it is necessary to select carefully ΔT and N that are the design parameters of this algorithm. In the case of including disturbance, the feasibility of the algorithm depends on the combination of ΔT and N . For reducing the complexity of the method of deciding these design parameters, a simple way of computational artifice is shown in the next section. The simulation result is described in section 6.

3.2 Algorithmic design incorporating computational time

In this section, a simple computational artifice of the above-mentioned algorithmic design is pointed out. First, we describe the key notes here. In the above-mentioned algorithm, the interval of time ΔT to apply one control strategy u_k^N is called "switching time". And the maximum number of the iteration executed in a switching time N is called "maximum

iteration". When the state was predicted, the obtained state trajectory is called "predictive trajectory" and actual trajectory is called "trajectory".

In our algorithmic design, the computation of maximum iteration should be done in switching interval. The search process of the optimal solution is executed in this algorithm, and the required computation time depends on the state. Therefore, it was necessary to give some margin to the switching interval. If the maximum iteration is sufficiently large, it may obtain an optimal solution in each switching interval. However, the switching interval has to set to large, because long computation time is required. Because the feedback effect is obtained by observing each switching interval, it seems that if the switching interval is as short as possible, the performance of robustness is better. The key idea of the algorithm which we propose here is to treat the switching interval as varying. It increases the maximum iteration when time is required for searching the optimal solution, and the switching interval is increased along with it. On the other hand, when long time is not required to find the optimal solution, reduce the maximum number of iteration and the switching interval for improving the robustness. The maximum iteration is decided based on Fig.2 and the computation time which was required to execute the algorithm. The maximum allowed computation time is set to τ_{max} , and the total time interval $[0, \tau_{max}]$ is divided into five sections as

$$[0, \tau_{max}] = [0, \tau_1] \cup [\tau_1, \tau_2] \cup [\tau_2, \tau_3] \cup [\tau_3, \tau_4] \cup [\tau_4, \tau_5]$$

where $t_5 = \tau_{max}$. For simplicity, let $\tau_i = \alpha i$ ($i = 1, 2, \dots, 5$). Moreover, the maximum iteration N and the switching interval ΔT^N are determined as follows.

$$\Delta T^N = \beta N \quad (12)$$

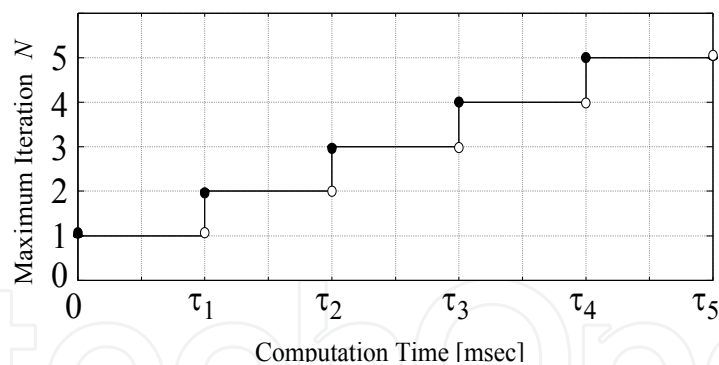


Fig. 3 Maximum iteration.

When actual calculation time is τ , the maximum iteration N is decided from Fig.2 and switching interval ΔT^N is obtained from expression (12). However, note that the present switching interval and the present maximum iteration are used in the next step. Here, based on the average computation time for one-iteration, the constants α and β are set to $\alpha = 0.02$ [sec] and $\beta = 0.03$ [sec]. In general, it is possible to decide N and ΔT^N such as $N_\sigma = g(\sigma)$ and $\Delta T_\sigma^N = h(\sigma)$ using a certain switching parameter σ .

[Robust Algorithm]

STEP C1 Let $k = 0$. Select arbitrarily initial input u_k^N and maximum iteration N_k . Then, ΔT^{N_k} is decided.

STEP C2 Measure the actual state x_{ak} , and apply the input u_k^N to the system over the interval of the unit time of calculation ΔT^{N_k} . During this time interval, we proceed with two kinds of calculations: One is to predict the one-unit-time-ahead state $x_{p(k+1)}$ through the system equation (1) with the initial state x_{ak} , and the other is to calculate from Step A3 to Step A7 with the updated initial state $x_{p(k+1)}$.

STEP C3 The maximum iteration is N_k , and calculate the rate of the value of performance index in each iteration, similarly as the computation from Step A3 to Step A7 ($i = 1, 2, \dots, N_k$).

STEP C4 If the rate of the value of performance index is larger than a sufficiently small value γ , that is if following inequalities are satisfied, it seems that the optimal solution was not obtained.

$$\left| \frac{J(u^{i+1}) - J(u^i)}{J(u^i)} \right| \geq \gamma \quad \text{and} \quad |J(u^i)| \geq \gamma \quad (13)$$

where $\gamma > 0$. Then, let $i = i + 1$, and execute the computation from Step A3 to Step A7. Execute these iterative computations till maximum $i = N_k$.

If following inequalities are satisfied, discontinue the iteration because it seems that the optimal solution was obtained.

$$\left| \frac{J(u^{i+1}) - J(u^i)}{J(u^i)} \right| < \gamma \quad \text{or} \quad |J(u^i)| < \gamma \quad (14)$$

The computation time which was required to the above-mentioned computation is set to τ_k .

Then, we obtain the next control strategy u_{k+1}^N .

STEP C5 The maximum iteration N_{k+1} and the switching interval $\Delta T^{N_{k+1}}$ for the next interval are decided based on the computation time which was required for current interval, equation (12) and Fig. 2.

STEP C6 Set $k = k + 1$, and go to Step C2.

4. Modeling

The kinematical model of the multiple trailer system which we treat is shown in Fig.4. The meaning of next equation (15) is the state equation of the first vehicle (autotruck) which is driven pulling the follow-on passive trailers.

$$\begin{bmatrix} \dot{x}_0 \\ \dot{y}_0 \\ \dot{\theta}_0 \end{bmatrix} = \begin{bmatrix} \cos \theta_0 \\ \sin \theta_0 \\ 0 \end{bmatrix} v_0 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega \quad (15)$$

The control input vector of this system is denoted by $u = [v_0 \quad \omega]^T$. Here, v_0 and ω denotes the velocity and angular velocity of the first vehicle respectively. This model is a differential-driven vehicle model which has nonholonomic constraint, and is regarded as one of the most typical nonholonomic systems. It is known that although this model has

controllability, it can not be asymptotically stabilizable by any continuous time-invariant controller (Brockett, 1993). For this reason, there have been many references dealing with the stabilization problem for this model using various kinds of controllers. One successful approach is to convert it into the so-called chained form and then establish a time-varying controller. Although such an approach leads to asymptotical stabilization, it is applicable only for the case where the system's dimension is low (less than four). Since we deal with a multiple trailer system, whose dimension is obviously much larger than four, the approach of utilizing chained form with a time-varying controller can not be applied here, and more practical strategy is desirable.

The most of conventional research have treated the direct-hooked type trailer model. This model is obtained by $D_0, D_1 = 0$ in Fig.4, and the kinematics of the i^{th} trailer is as follows.

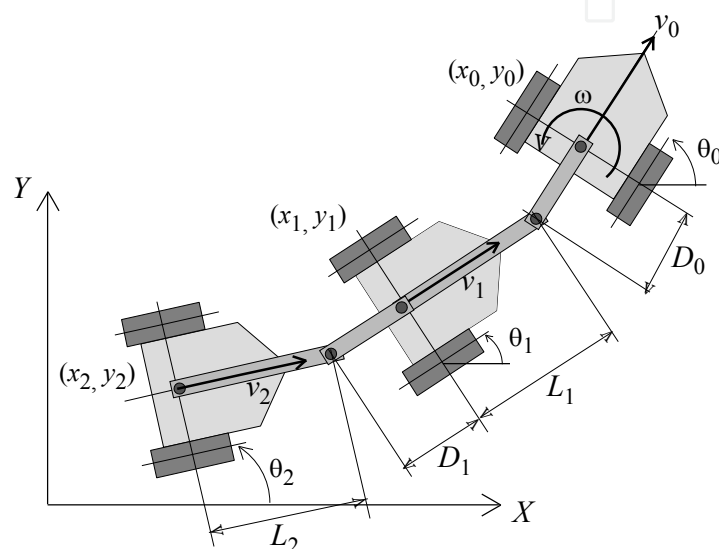


Fig. 4 Mechanical linkage design of multiple trailer system.

Only the first vehicle (truck) is driven and the following vehicles (trailers) are passively pulled by the truck.

$$\begin{aligned} \dot{\theta}_i &= \frac{\sin(\theta_{i-1} - \theta_i)v_{i-1}}{L_i} \\ v_i &= \cos(\theta_{i-1} - \theta_i)v_{i-1} \end{aligned} \quad (16)$$

where, θ_i denotes the attitude angle of the i^{th} trailer, and L_i is the length of the i^{th} linkage. v_i and $\dot{\theta}_i$ denote the velocity and angular velocity of i^{th} trailer respectively.

The direct-hooked model can be transformed to a chained form. However, this model has a tracking error of follow-on trailers. Therefore, we deal with the off-hooked model ($L_i = D_{i-1} \neq 0$) which can eliminate the tracking error (Fig. 5). However, the model of off-hooked trailer system cannot be transformed to canonical form. Fig. 4 shows a off-hooked model, and the following equation denotes the i^{th} trailer's kinematics.

$$\begin{aligned} \dot{\theta}_i &= \frac{(\sin(\theta_{i-1} - \theta_i)v_{i-1} - \cos(\theta_{i-1} - \theta_i)\dot{\theta}_{i-1}D_{i-1})}{L_i} \\ v_i &= \cos(\theta_{i-1} - \theta_i)v_{i-1} + \sin(\theta_{i-1} - \theta_i)\dot{\theta}_{i-1}D_{i-1} \end{aligned}$$

5. Problem formulation

Tracking control problem of the multiple trailer system is formulated as a nonlinear optimal control problem in this section. For simplicity of notation, we consider one truck and two trailers. Even if the number of the trailer increases, our control design can be extended very easily. In that case, increase of the computational cost is inevitable.

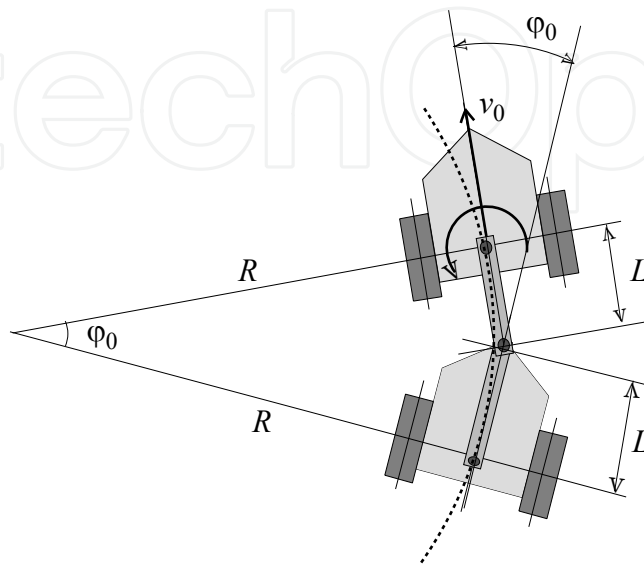


Fig. 5 Tracking path of the off-hooked trailer system.

The state equation of the 1-truck and 2-trailers model is given by

$$\begin{aligned} \dot{\xi} &= A(\xi)u \\ \xi &= [x_0 \quad y_0 \quad \theta_0 \quad \theta_1 \quad \theta_2]^T, \quad u = [v_0 \quad \omega]^T \end{aligned} \quad (17)$$

$$A(\xi) = \begin{bmatrix} \cos \theta_0 & 0 \\ \sin \theta_0 & 0 \\ 0 & 1 \\ \sin(\theta_0 - \theta_1)/L & -\cos(\theta_0 - \theta_1) \\ \sin(2\theta_1 - \theta_0 - \theta_2)/L & -\cos(2\theta_1 - \theta_0 - \theta_2) \end{bmatrix}$$

The performance index is given by

$$\begin{aligned} J &= (\xi(t) - \xi_f(t))^T P (\xi(t) - \xi_f(t)) \Big|_{t=t_1} \\ &+ \frac{1}{2} \int_{t_0}^{t_1} ((\xi(t) - \xi_f(t))^T Q (\xi(t) - \xi_f(t)) + u(t)^T R u(t)) dt \end{aligned}$$

where the state vector and input vector are denoted by ξ and u respectively. P , Q , R denote the weighting matrices. We set $P = 0.5I$, $Q = \text{diag}[0.2 \quad 0.2 \quad 0.001 \quad 0.001 \quad 0.001]$, $R = \text{diag}[0.05 \quad 0.01]$. $\xi_f(t)$ is the target state, and it is the circle of radius 0.5[m] with constant velocity. Furthermore, we treat the state constraints and input constraints by introducing the penalty term.

$$\bar{J} = J + \int_{t_0}^{t_1} \left(\sum_{i=1}^2 \frac{r_i}{\theta_{i\text{lim}}^2 - (\theta_{i-1} - \theta_i)^2} \right) dt + \int_{t_0}^{t_1} \frac{r_v}{v_{\text{lim}}^2 - v^2} + \frac{r_\omega}{\omega_{\text{lim}}^2 - \omega^2} dt \quad (18)$$

where, $\theta_{i\text{lim}}(i=1,2)$ is an absolute value of limitation of the relative angle, and v_{lim} and ω_{lim} are the absolute value of the limitation of the control input.

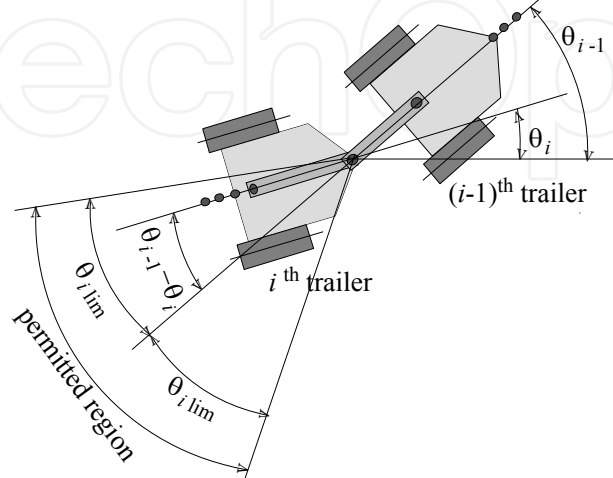


Fig. 6 Permitted region of i^{th} trailer.

We chose $|\theta_i| < \theta_{i\text{lim}} = 0.5$ [rad] ($i = 1, 2$), $|v| < v_{\text{lim}} = 1.0$ [m/sec], $|\omega| < \omega_{\text{lim}} = 4.848$ [rad/sec].

Fig. 6 shows the permitted region of follow-on trailers. The weight parameters are set to $r_i = 0.001$ ($i = 1, 2$), $r_v = 0.0001$, $r_\omega = 0.0001$.

6. Numerical simulation

The control strategy of our approach is obtained by processing the iterative calculation of the REB algorithm in each ΔT . Through sufficiently large number of iterations N , it could be expected to eventually reach the possible optimal solutions. Through some simulation results we can obtain the effective solution with roughly $\Delta T=100$ [msec] by the PC which we use. However, it is not necessarily the case that the effective solution is obtained, especially in the case of including a disturbance. The simple computational artifice described in section 3.2. partially reduces such a problem. The example of the simulation result of applying the algorithm to the case of including a disturbance is shown in the following.

Fig. 7 shows the simulation result of the computation time of each ΔT with fixed number of iterations N and switching time $\Delta T=100$ [msec]. Simulated time is 30 [sec], then average and minimum/maximum value of the computation time is shown. The solid lines are $\Delta T_k^N = \beta N_k$ with $\beta = 0.02$ [sec] and $\beta = 0.03$ [sec] respectively. According to Fig. 7, proposed algorithm is almost executable in real time with $\beta = 0.03$ [sec]. Therefore, we simply choose as $\alpha = 0.02$ [sec], $\beta = 0.03$ [sec]. However, real time feasibility is not guaranteed by these parameters, because the computation time varies according to running condition.

Fig. 8 - Fig. 11 show the simulation result with the initial state $\xi_0 = [0 \ 0 \ -\pi/2 \ -\pi/2 \ -\pi/2]^T$. Impulsive disturbances on θ_1 and θ_2 have been added in this simulation at 5, 10, 15 and 20[sec], whose magnitude is 0.5[rad].

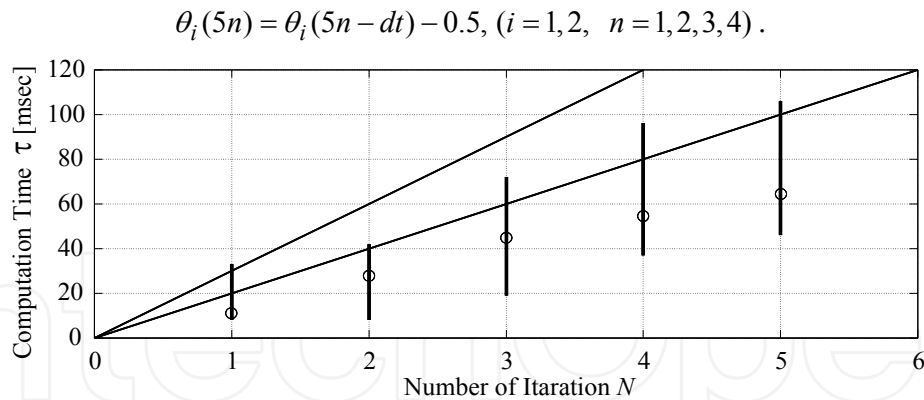


Fig. 7. Computational time. o: average of the computation time, with the maximum and minimum computation time, solid line: $\Delta T_k^N = \beta N_k$ with $\beta = 0.02[\text{sec}]$ and $\beta = 0.03[\text{sec}]$ respectively.

The lower part of Fig.9 shows the computation time of each switching time and its upper bound. ΔT has changed corresponding to disturbances. Also, this figure shows that this algorithm is feasible in real time, because the computation time is less than switching time ΔT .

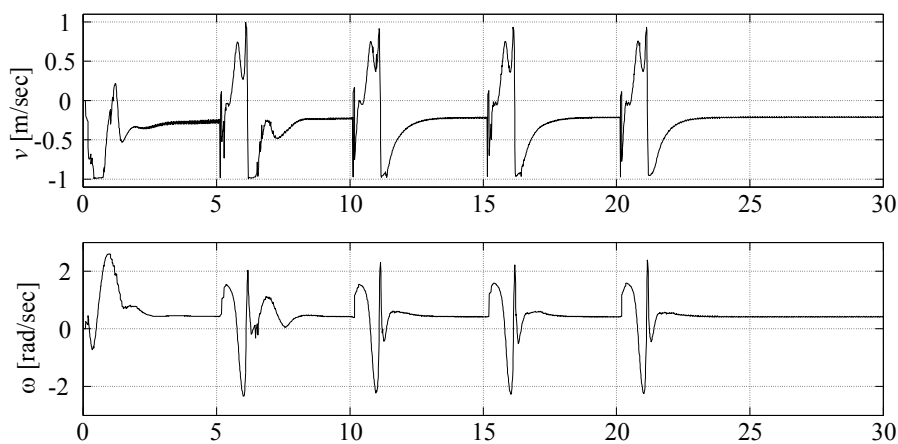


Fig. 8 Simulation results: control inputs.

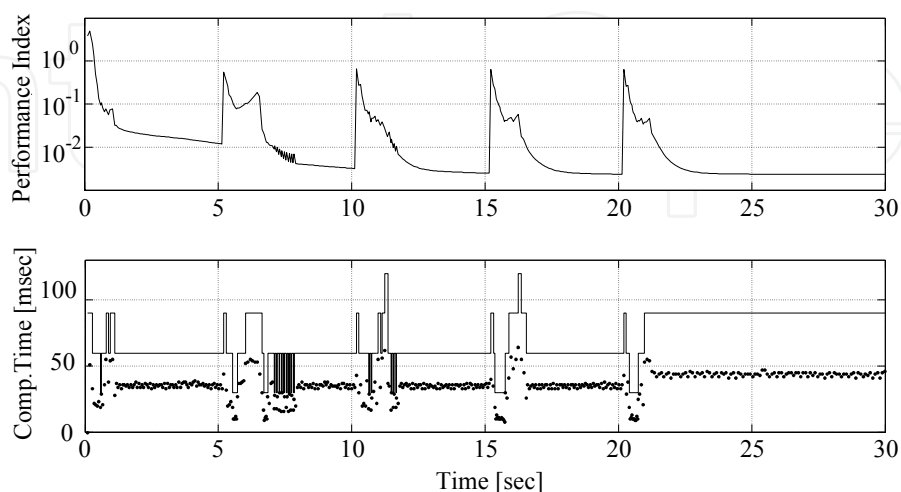


Fig. 9 Simulation results: value of performance index (upper stand). Computation time of each ΔT and its bound (lower part).

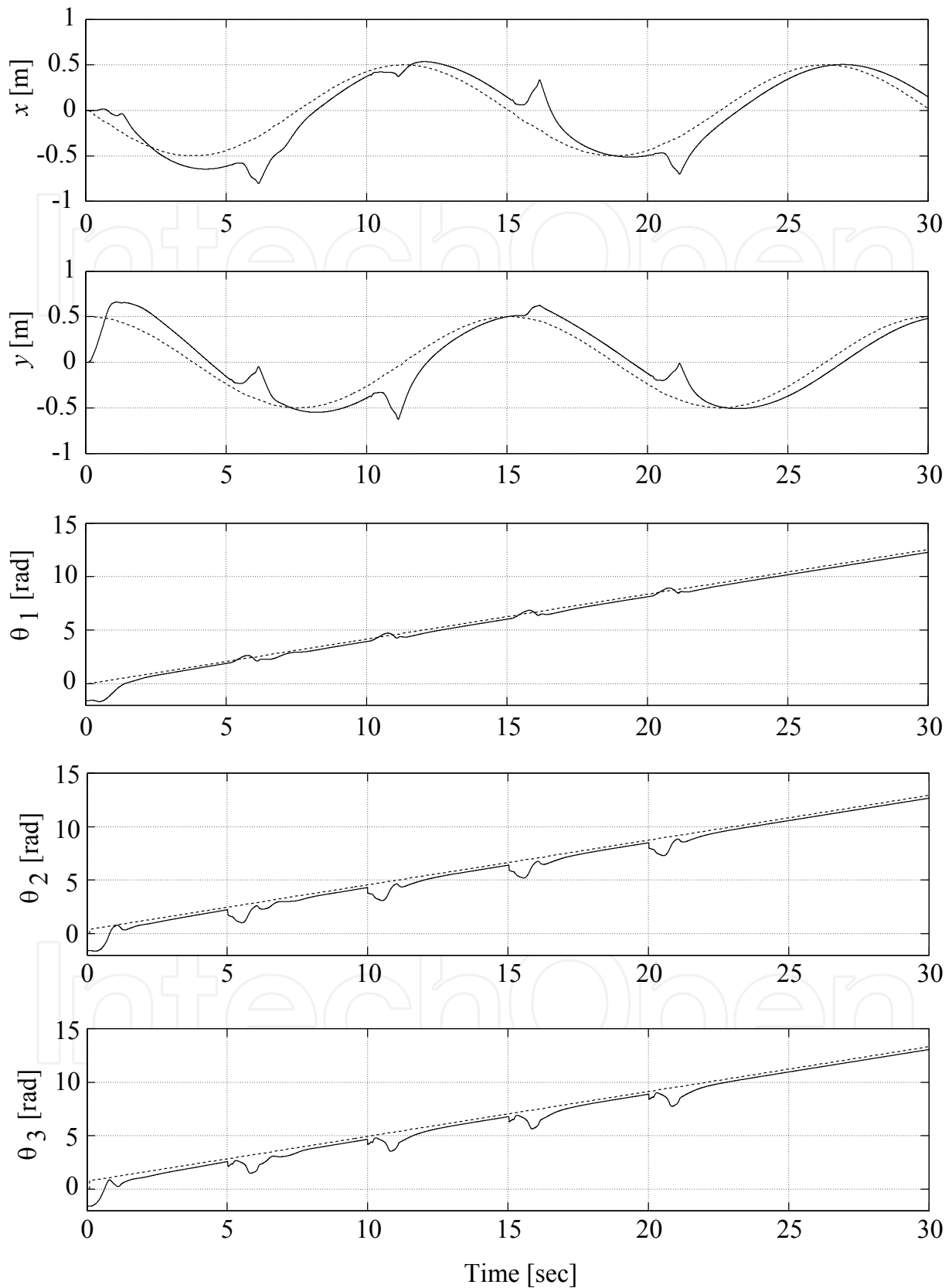


Fig. 10 Simulation results: state trajectories. (solid line: actual states, dotted line: target states).

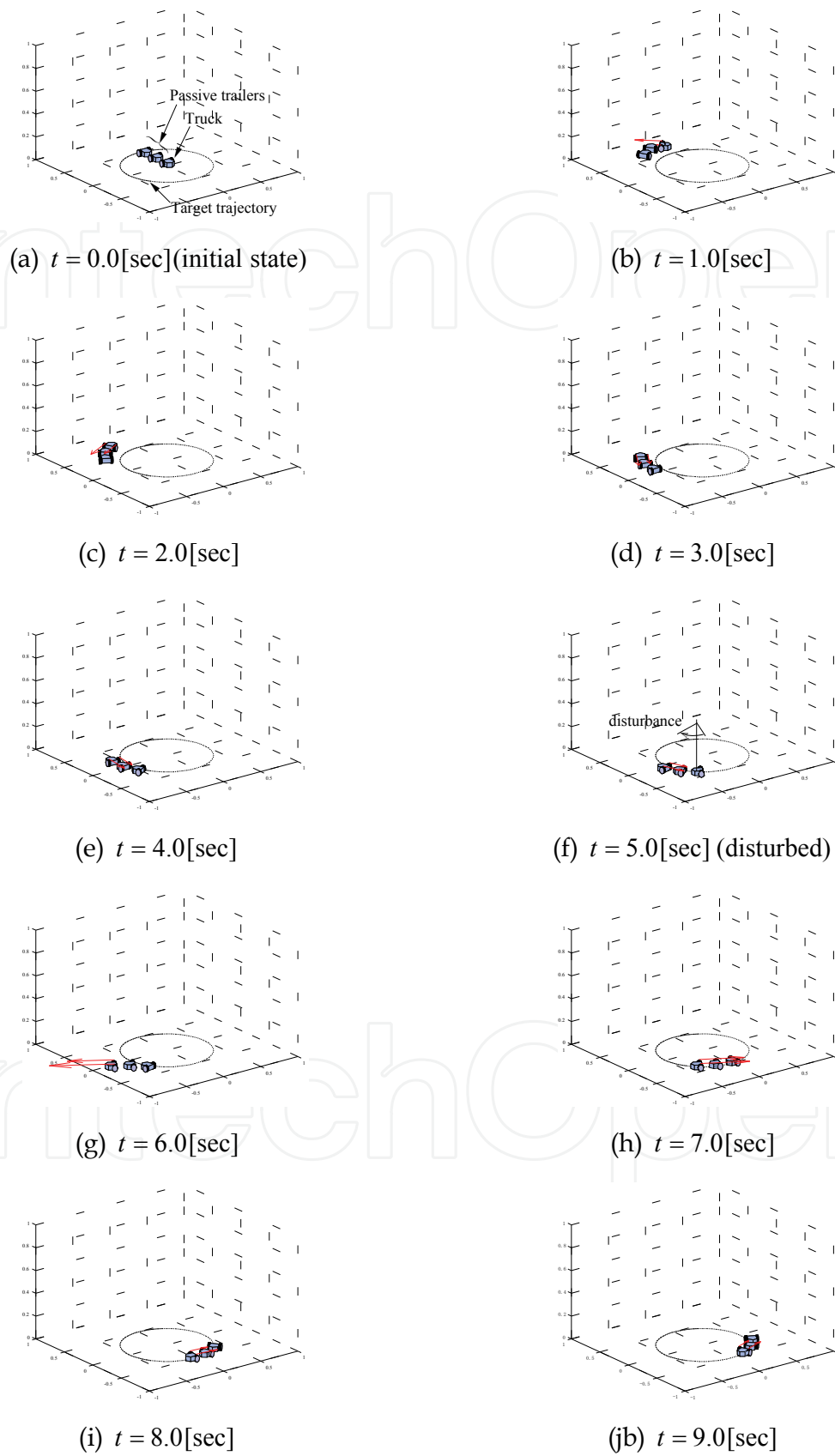


Fig. 11 Simulation results.

7. Conclusions

We discussed the real time control algorithm using the numerical solution called algorithmic control. Then, we improved the conventional algorithmic design for the numerical robustness via incorporating computation time. The key idea is to adjust the maximum number of iteration with the computational time. This approach was applied to a tracking control problem of the multiple trailer system. We showed through a numerical simulation that the proposed algorithm is executable in real time, and it has robustness against disturbances.

8. Acknowledgment

This research has been supported in part by the Japan Ministry of Education, Sciences and Culture under Grants-in-Aid for Scientific Research (B) 18760326.

9. References

- Brockett, R. W. (1983). Asymptotic Stability and Feedback Stabilization, *Differential Geometric Control Theory*, Vol. 27, pp. 181-191.
- Hermann, R. & Krener, A. J. (1977). Nonlinear Controllability and Observability, *IEEE Transactions on Automatic Control*, Vol. 22, No. 5, pp. 728-740.
- Imae, J.; Irlicht, L., Obinata, G. & Moore, J. B. (1992). Enhancing Optimal Controllers via Techniques from Robust and Adaptive Control, *International Journal of Adaptive Control and Signal Processing*, Vol. 6, pp. 413-429.
- Imae, J. & Torisu, R. (1998). A Riccati-Equation Based Algorithm for Nonlinear Optimal Control Problems, *Proceedings of the 37th Conference on Decision and Control*, pp. 4422-4427.
- Imae, J.; Yoshimizu, K., Kobayashi, T. & Zhai, G. (2005). Algorithmic Control for Real-time Optimization of Nonlinear Systems: Simulations and Experiments, *Proceedings of the 44th Conference on Decision and Control*, pp. 3729-3734.
- Ishikawa, M. (1993). Control of Nonholonomic Systems having Complicated Controllability Structure, *Journal of The Society of Instrument and Control Engineers*, Vol. 42, No. 10, pp. 841-846.
- Jacobson, D. H. & Mayne, D. Q. (1970). *Differential Dynamic Programming*, ISBN. 978-0444000705, Elsevier.
- Kobayashi, T.; Magono, M., Imae, J., Yoshimizu, K. & Zhai, G. (2005a). Real-time Optimization for Nonlinear Systems Using Algorithmic Control, *Preprints of IFAC '05*, Prague, Czech Republic.
- Kobayashi, T.; Maenishi, J., Imae, J. & Zhai, G. (2005b). Real-Time Optimization for Mobile Robot Based on Algorithmic Control, *Proceedings of 2005 International Conference on Control, Automation and Systems*, pp. 2102-2107.
- Kobayashi, T.; Maenishi, J., Imae, J. & Zhai, G. (2006). Real Time Control for 4-wheeled Vehicles via Algorithmic Control Incorporating Computation Time, *Proceedings of 9th International Conference on Control, Automation, Robotics and Vision*, pp. 1353-1358.
- Laumond, J.-P. (1986). Feasible Trajectories for Mobile Robots with Kinematic and Environment Constraints, *Proceedings of International Conference on Intelligent Autonomous Systems*, pp. 346-354.

- Lee, J.; Chung, W., Kim, M., Lee, C. & Song, J. (2001). A Passive Multiple Trailer System for Indoor Service Robots, *Proceedings of 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 827-832.
- Lee, J.; Chung, W., Kim, M., Lee, C. & Song, J. (2004). A Passive Multiple Trailer System with Off-axle Hitching, *International Journal of Control, Automation, and Systems*, Vol. 2, No. 3, pp. 289-297.
- Matsune, I.; Zhai, G., Kobayashi, T., & Imae, J. (2005) A Study on Hybrid Control of Nonholonomic Systems, *Proceedings of The Society of Instrument and Control Engineers Annual Conference*, pp. 211-214.
- M'Closkey, R. T. & Murray, R. M. (1993). Nonholonomic Systems and Exponential Convergence: Some Analysis Tools, *Proceedings of the 32nd Conference on Decision and Control*, pp. 943-948.
- Murray, R. M. & Sastry, S. S. (1993). Nonholonomic Motion Planning: Steering Using Sinusoids, *IEEE Transactions on Automatic Control*, Vol. 38, No. 5, pp. 700-716.
- Nedeljkovic, N. B. (1981). New Algorithms for Unconstrained Nonlinear Optimal Control Problems, *IEEE Transactions on Automatic Control*, Vol. 26, pp. 868-884.
- Sampei, M. & Kobayashi, T. (1994). Path Tracking Control of Car-Caravan Type Articulated Vehicles Using Nonlinear Control Theory, *Transactions of the Society of Instrument and Control Engineers (in Japanese)*, Vol. 30, No. 4, pp. 427-434.

IntechOpen



Automation and Robotics

Edited by Juan Manuel Ramos Arreguin

ISBN 978-3-902613-41-7

Hard cover, 388 pages

Publisher I-Tech Education and Publishing

Published online 01, May, 2008

Published in print edition May, 2008

In this book, a set of relevant, updated and selected papers in the field of automation and robotics are presented. These papers describe projects where topics of artificial intelligence, modeling and simulation process, target tracking algorithms, kinematic constraints of the closed loops, non-linear control, are used in advanced and recent research.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Tomoaki Kobayashi, Toru Yoshida, Junichi Maenishi, Joe Imae and Guisheng Zhai (2008). Tracking Control for Multiple Trailer Systems by Adaptive Algorithmic Control, Automation and Robotics, Juan Manuel Ramos Arreguin (Ed.), ISBN: 978-3-902613-41-7, InTech, Available from:
http://www.intechopen.com/books/automation_and_robotics/tracking_control_for_multiple_trailer_systems_by_adaptive_algorithmic_control

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2008 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](#), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen