

조이스틱을 이용한 선박의 입출항 및 접이안 시스템의 제어 알고리즘 개발

홍성국* · 정윤하* · † 원문철* · 김신영**

*충남대학교 BK21 메카트로닉스 고급인력양성 산학공동사업단, **한국해양연구원 해상교통안전연구사업단

Development of Control Algorithm for Ship Berthing and Unberthing Systems Using a Joystick

Seong-Kuk Hong* · Yun-Ha Jung* · † Moon-Cheol Won* · Sun-Young Kim**

*Dept. of Mechatronics of BK 21 Mechatronics Group, Chungnam National University, Daejeon 305-764, Korea

**Maritime and Ocean Engineering Research Institute/KORDI, Daejeon 305-343, Korea

요약 : 본 연구에서는 조이스틱을 이용하여 프로펠러와 타, 선수/선미 쓰러스터를 갖는 선박의 접이안을 위한 제어 알고리즘을 개발하였다. 조이스틱으로부터 전진 방향 및 회전 방향의 속도명령을 받아 전진 방향 및 회전 방향의 속도를 제어하는 MIMO(Multi-Input Multi-Output) 비선형 제어 알고리즘을 개발하기 위해 저속 조종수학모형을 사용하였다. 또한, 본 연구에서는 비선형 및 PID 제어기의 성능을 검증하기 위해 선박 접이안 가상 HILS(Hardware in the Loop Simulation) 프로그램을 구현하였다. HILS 프로그램은 LabWindow/CVI를 이용하여 개발하였으며, 사용자는 선박의 현재 위치와 원하는 궤적을 모니터를 통해 본 후 조이스틱을 이용하여 선박의 전진 방향 및 회전방향 속도를 제어함으로써 선박을 조종한다. 시뮬레이션 결과를 보면 비선형 제어기와 PID 제어기는 개루프 조이스틱 제어기보다 타와 쓰러스터의 입력 크기뿐 아니라 선박의 위치오차 면에서도 우수한 성능을 보였다.

핵심용어 : 조이스틱, 접이안 시스템, 다중입력다중출력, HILS(hardware in the loop simulation)시스템, 쓰러스터, 비선형 제어, PID 제어

Abstract : This study develops a control algorithm on berthing/unberthing systems using a joystick for ships with thrusters and a rudder. A nonlinear mathematical model for low speed maneuvering of typical container ships is used to develop a MIMO(multi-input multi-output) nonlinear control algorithm for velocity feedback joystick control. Also a virtual HILS(hardware in the loop simulation) software program for berthing/unberthing is developed to test the performance of the nonlinear and a PID control algorithm. The program is developed using LabWindow/CVI, and a user can see current position and desired trajectory of ship in a monitor, then he can control forward and yaw velocities of a ship using a joystick. The simulation results show that the nonlinear and the PID controller have superior performance over a simple open loop joystick control algorithm.

Key words : Joystick, Berthing and unberthing system, MIMO(multi-input multi-output), HILS(hardware in the loop simulation) system, Thruster, Non-linear control, PID control

1. 서론

최근 급격히 늘어나는 해상 물동량으로 인해 제한된 항만 내에서 많은 선박들을 빠르고 안전하게 접이안해야 할 필요성이 증가하고 있다. 선박의 접이안은 전체 항해 중 난이도가 가장 높은 항해로 사고위험도 또한 가장 높다. 따라서 개항에서의 선박 접이안은 항만의 바다 속 지형, 바람, 조류 등에 익숙한 도선사에 의해 이루어지고 있다. 하지만 이러한 접이안 시스템은 많은 비용과 시간이 든다는 단점이 있다. 최근 이러한 문제에 대한 대책으로 기존의 도선사 및 예인선에 의한 접이안 대신 항

만에 설치되어있는 선박과 부두와의 거리정보센서 등의 항만 인프라와 쓰러스터를 이용한 자동 접이안(Yamato and Koyama, 1992; Hasegawa and Kitera, 1993; Hasegawa, 2002)시스템이 연구되고 있다. 또한, 상기의 항만 인프라와 쓰러스터와 같은 보조 조종장치를 이용하지만, 완전 자동 접이안이 아닌 조이스틱 접이안 시스템도 연구 및 실용화(HarbourMarine, 2006; MARIMATECH, 2006) 되고 있다.

보조 조종 장치를 갖춘 선박의 접이안 제어 시스템은 MIMO(Multi-Input Multi-Output)시스템이다. 즉, 다수의 선박운동 정보를 입력 받아 타, 프로펠러, 쓰러스터 혹은 다른

† 교신저자 : 원문철(정회원), mcwon@cnu.ac.kr 042)821-6875
* annyunghsk@nate.com, 042)821-7784
* uunnaa@nate.com 042)821-7784
** sykim@moeri.re.kr 042)868-7113

보조 조선장치에 대한 다수의 명령을 출력하는 제어 시스템이다. 효율적인 선박 접이안 시스템의 개발을 위해서는 선박의 특성에 맞는 최적 MIMO 제어기 설계 기법(Fossen, 2002)을 정립할 필요가 있다.

본 연구에서는 선박의 비선형 저속 운동 방정식(Fossen, 1994; 원, 2005)을 이용한 비선형 MIMO 제어기와 조이스틱을 이용한 접이안 시스템을 설계하고 이를 가상 접이안 HILS(Hardware in the Loop Simulation)와 선박 조종 시뮬레이터를 통해 검증하는 것을 목표로 한다.

본 논문의 구성은 다음과 같다. 2장에서는 조이스틱을 이용한 접이안 가상 HILS(Hardware in the Loop Simulation) 프로그램과 제어 알고리즘 설계를 위한 컨테이너 선박의 비선형 저속 운항 동역학 모델을 설명하였다. 또한, 3장에서는 조이스틱을 이용한 접이안 가상 HILS 프로그램과 제어 알고리즘 설계과정을 설명하였으며, 수학적 시뮬레이션 결과 및 실험 결과는 4장과 결론에서 언급하였다.

2. 비선형 저속 운항 동역학 모델

저속 운항 선박의 동역학은 복잡한 비선형 미분 방정식으로 표현할 수 있다. 본 연구에서는, 전형적인 컨테이너 선박의 비선형 모델을 접이안 제어 알고리즘 설계 및 가상 접이안 HILS에 이용하였다.

선박의 운동방정식 기술을 위한 좌표계는 Fig. 1과 같이 정의한다. 선박은 일반적으로 6자유도 운동을 하지만 접이안 문제에서는 간단하게 3자유도 운동으로 선박의 Surge, Sway, 그리고 Yaw 운동만을 고려하기로 한다. 이 경우, 선박 운동방정식은 다음 식 (1)~(3)과 같이 나타낼 수 있다.

$$\dot{x} = u \cos(\psi) - v \sin(\psi) \quad (1)$$

$$\dot{y} = u \sin(\psi) + v \cos(\psi) \quad (2)$$

$$\dot{\psi} = r \quad (3)$$

여기서, (x, y, ψ) 는 Surge, Sway와 Yaw방향의 공간고정 변위 좌표계이다. 또한 (u, v, r) 는 선체고정 속도 좌표계이다. 위와 같은 좌표계 정의에 의하여, 선박의 동역학식은 다음 식 (4)~(6)과 같다.

$$m(\dot{u} - vr - x_G r^2) = X_H + X_R + X_P \quad (4)$$

$$m(\dot{v} + ur + x_G r) = Y_H + Y_R + Y_T \quad (5)$$

$$I_{zz} \dot{r} + m x_G (\dot{v} + ur) = N_H + N_R + N_T \quad (6)$$

식 (4)에서 x_G 는 선박의 무게중심, I_{zz} 는 무게중심에 대한

질량 관성 모우멘트이다.

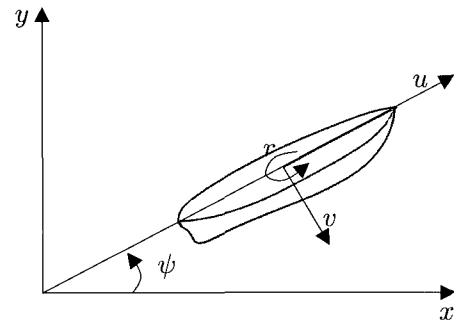


Fig. 1 Coordinate definition for describing dynamics of low speed maneuvering of ship

여기서 선체 주위의 유체에 의해 선체에 작용하는 유체력 (X_H, Y_H, N_H) 은 다음 식 (7)~(9)로 주어진다(원, 2005).

$$X_H = X_u \dot{u} + X_{|u|} u |u| + X_{vr} vr + X_\alpha \alpha + X_{|\alpha|} |\alpha| \quad (7)$$

$$Y_H = Y_v \dot{v} + Y_r \dot{r} + Y_{vr} vr + Y_\beta \beta + Y_{|\beta|} |\beta| |\beta| - \frac{\rho}{2} LC_{Dv} \left\{ \int_{-L/2}^{L/2} v + C_{Yr} r x [(v + C_{Yr} r x) dx - L |v| |v| \right\} \quad (8)$$

$$N_H = N_v \dot{v} + N_r \dot{r} + N_{|u|r} |u| r + N_\alpha \alpha + N_{|\alpha|} |\alpha| - \frac{\rho}{2} LC_{Dv} \int_{-L/2}^{L/2} v + C_{Nr} r x [(v + C_{Nr} r x) x dx \quad (9)$$

식 (8)에서, 선박의 사향각(Drift angle) β 는 식 (10)과 같이 정의된다.

$$\beta = \tan^{-1} \left(\frac{-v}{u} \right) \quad (10)$$

식 (7)에서 α 는 사향각 β 를 후진시에도 일반화하여 사용할 수 있는 수정 사향각으로 식 (11)로 정의된다. 식 (11)에서 α_0 는 좌우동요방향 운동시 선체에 작용하는 유체력에 의한 선수 동요 모우멘트가 0이 되는 사향각이다(원, 2005).

$$\alpha = \beta - \text{sgn}(\beta) \alpha_0 \quad (11)$$

타각(δ)의 변화로 인해 발생하는 유체력 (X_R, Y_R, N_R) 은 다음 식 (12)~(14)와 같이 표현할 수 있다.

$$X_R = -(1 - t_R) F_N \sin \delta \quad (12)$$

$$Y_R = -(1 + a_H) F_N \cos \delta \quad (13)$$

$$N_R = -(x_R + a_H x'_H L) F_N \cos \delta \quad (14)$$

여기서, 타직압력인 F_N 은 식 (15)로 정의 한다.

$$F_N = \frac{\rho}{2} A_R f_\alpha U_R^2 \sin \alpha_R \quad (15)$$

식 (14)에서 t_{Rb}, a_H, \dot{x}_H 는 선체와 타 간의 간섭계수이다.

식 (15)에서, 타직압력 계수 f_α 와 타로 유입되는 유체의 속도인 U_R 는 $f_\alpha = \frac{6.13\lambda}{\lambda + 2.25}$ 와 $U_R = \sqrt{u_R^2 + v_R^2}$ 으로 주어진다. 여기서 u_R 과 v_R 는 각각 타로 유입되는 유체의 전진방향 및 횡방향 유체의 속도이다.

또한 유입류의 유효영각 α_R 은 $\alpha_R = \delta + \frac{v_R}{u_R}$ 로 산정한다. 주 추진 프로펠러에 의한 힘 X_p 는 식 (16)과 같이 모델 될 수 있다.

$$X_p = (1 - t_p) \rho n |n| D_p^4 K_T \quad (16)$$

여기서 n 은 프로펠러의 회전수(rps), K_T 는 프로펠러 추력계수로, 식 (17)과 같이 전진비(J_p)의 함수로 나타낼 수 있다.

$$K_T = a_1 + a_2 J_p + a_3 J_p^2 \quad (17)$$

쓰러스터에 의한 힘(Y_T, N_T)는 선속에 따라 크게 달라지나 선속이 충분히 작은 경우 다음 식 (18)~(19)와 같이 쓰러스터 회전수만의 함수로 간단하게 모델 될 수 있다.

$$Y_T = Y_{T_b} + Y_{T_s} \quad (18)$$

$$N_T = x_{T_b} Y_{T_b} + x_{T_s} Y_{T_s} \quad (19)$$

여기서 Y_{T_b} 와 Y_{T_s} 는 식 (20)~(21)과 같이 모델 하였는데 b_S 및 b_B 는 각각 쓰러스터 추력 실험에 의해 정해지는 상수이며, n_B 와 n_S 는 각각 선수 및 선미 쓰러스터의 회전수(rps)이다.

$$Y_{T_b} = b_B n_B |n_B| \quad (20)$$

$$Y_{T_s} = b_S n_S |n_S| \quad (21)$$

각 액추에이터들의 동적 모델은 아래 식 (22)~(25)와 같이 모두 1차 지연요소의 선형 시스템으로 간단하게 모델 될 수 있다.

$$\dot{n} = \frac{1}{\tau_P} (n_C - n) \quad (22)$$

$$\dot{\delta} = \frac{1}{\tau_\delta} (\delta_C - \delta) \quad (23)$$

$$\dot{n}_B = \frac{1}{\tau_{tB}} (n_{BC} - n_B) \quad (24)$$

$$\dot{n}_S = \frac{1}{\tau_{tS}} (n_{SC} - n_S) \quad (25)$$

여기서 n_C, δ_C, n_{SC} 와 n_{BC} 는 각 액추에이터들의 명령 값이며 $\tau_P, \tau_\delta, \tau_{tB}$, 및 τ_{tS} 는 각 액추에이터들의 시간 상수이다.

3. 조이스틱을 이용한 선박 접이안 시스템

본 연구에서는 조이스틱을 이용한 선박 접이안 시스템을 개발하기 위하여 사용자가 GUI(Graphic User Interface)를 통해 선박 운항에 관계된 모든 정보를 받아 선박을 조종할 수 있도록 선박 접이안 시스템을 설계하였다.

3.1 조이스틱을 이용한 가상 선박 접이안 HILS 설계

Fig. 2는 조이스틱을 이용한 선박 접이안 시스템의 개념도로서 사용자는 GUI에 나타난 선박의 현재 궤적 및 정해진 항로를 보면서 조이스틱을 조종하여 선박의 위치를 제어한다. 이때 조이스틱 위치에 따라 전진 방향 속도와 회전 방향 속도 명령이 제어 알고리즘으로 입력되고, 이 입력된 속도 명령과 실제 선박의 속도를 이용한 페루프 제어 알고리즘에 의해 프로펠러의 회전수와 타각, 선수/선미 쓰러스터의 회전수 명령을 출력한다.

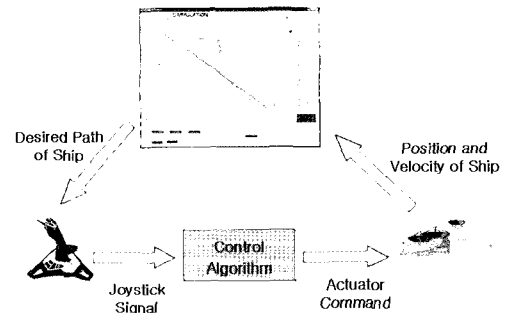


Fig. 2 Schematic diagram of berthing and unberthing using a joystick

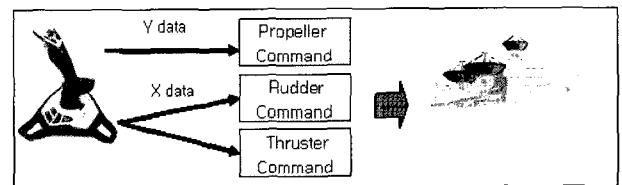


Fig. 3 Data flow of berthing and unberthing using a joystick(open-loop)

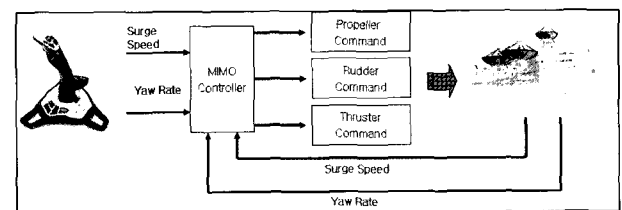


Fig. 4 Data flow of berthing and unberthing using a joystick(closed-loop)

한편 개루프 제어에서는 조이스틱의 X, Y 위치가 주 추진 프로펠러 및 쓰러스터 명령으로 바로 변환되는 데 Fig. 3과 같은 그 개념이 도시될 수 있다.

Fig. 4는 본 연구에서 설계한 비선형 제어기와 PID 제어기 즉 페루프 제어기의 데이터 흐름을 나타내는 그림이다. 사용자가 조이스틱을 조종하면 조이스틱의 X축, Y축 데이터는 각각 선박의 회전방향 속도와 전진 방향 속도 명령이 되어 MIMO 제어기에 입력된다. MIMO제어기는 입력된 속도 명령과 현재 선박의 속도를 기반으로 각 액츄에이터 명령을 출력한다. 본 연구에서 개발한 HILS에서는 사용자가 조작하는 조이스틱이 하드웨어가 되고, 실제 선박의 거동은 시뮬레이션 모델로 대체된다. 즉, 2장에서 설명한 선박의 동역학 모델이 수치적으로 적분되어 선박의 위치 속도 등을 구하게 된다. 이때 구해진 전진 방향 속도 및 회전 방향 속도는 MIMO 제어기로 피드백되어 되어 사용자가 원하는 속도명령과 비교되어 선박의 제어에 이용된다.

아래 Fig. 5가 본 연구에서 설계한 HILS 프로그램의 GUI(Graphic User Interface)의 모습이다. GUI에는 선박의 현 위치, 항구의 위치, 각 액츄에이터들의 현재 속도 및 각도, 선박의 항로 등이 표현되어 있다. 사용자는 GUI를 통해 여러 정보를 얻어 조이스틱을 조종함으로써 선박을 항구에 접이안시킬 수 있다.

HILS 프로그램 개발에는 NI사의 LabWindows/CVI를 사용하였으며, 시리얼 통신을 이용하여 조이스틱 입력을 받아들였다.

이때 조이스틱 입력 신호의 노이즈를 줄이기 위하여 식 (26)과 같은 점화식 형태의 필터를 이용하여 신호처리를 수행하였다.

$$\hat{x}_k = \hat{x}_{k-1} + \frac{1}{k}(z_k - \hat{x}_{k-1}) \quad (26)$$

여기서 \hat{x}_k 는 현재까지 조이스틱의 입력 추정치이며, \hat{x}_{k-1} 는 이전 스텝까지의 필터링 된 조이스틱 입력, z_k 는 현재의 조이스틱 입력이다. 이를 이용하여 매 100msec 마다 100개의 조이스틱 신호를 빠르게 받아들여 평균을 내 사용함으로써 비교적 노이즈가 적은 조이스틱 신호를 받아들일 수 있다.

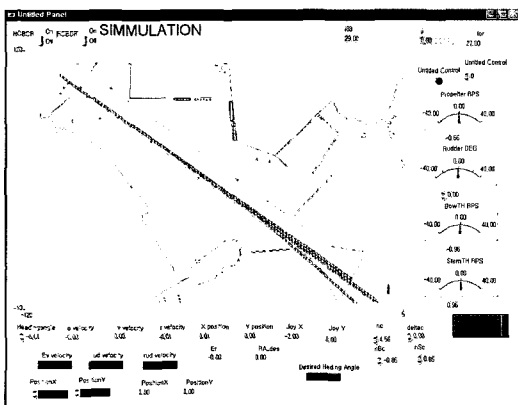


Fig. 5 HILS GUI of real-time berthing simulation using joystick

3.2 조이스틱 비선형 제어알고리즘 설계

비선형 제어기 설계(Slotine and Li, 2000)에서는 선박의 동역학식인 식 (4), (5), (6)만을 사용하였다. 이들 식에서 부가 질량 항에 관련된 항을 좌변으로 이동시키면 아래 식 (27), (28)을 얻는다.

$$(m - X_u)\dot{u} = X_h(u, v, r) + X_R(u, v, n, \delta) + X_P(u, n) \quad (27)$$

$$\begin{bmatrix} (m - Y_v), & (mX_G - Y_r) \\ (mX_G - N_v), & (I_{ZZ} - N_r) \end{bmatrix} \begin{bmatrix} \dot{v} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} Y_h(u, v, r) + Y_R(u, v, n, \delta) + Y_T(n_B, n_S) \\ N_h(u, v, r) + N_R(u, v, n, \delta) + N_T(n_B, n_S) \end{bmatrix} \quad (28)$$

여기서 X_h , Y_h , 및 N_h 는 식(7), (8), (9)에 있는 N_H , Y_H , 및 N_H 에서 부가질량과 관련이 있는 $X_u\dot{u}$, $Y_v\dot{v}$ 및 $N_v\dot{v}$ 항만을 각각 제외한 것들이다.

다음에서는 제어 알고리즘 유도과정을 설명하는데, 먼저 전진 방향 속도 제어를 위한 알고리즘을 도출한다.

전진방향 속도오차 e_u 를 식 (29)와 같이 정의한다. 여기서 $u_d(s)$ 는 선박의 원하는 전진방향속도이다.

$$e_u = u - u_d(s) \quad (29)$$

원하는 페루프 속도제어 동역학은 식 (30)과 같이 설정한다.

$$\dot{e}_u + \lambda_1 e_u = 0 \quad (30)$$

여기서 λ_1 은 양의 상수이다. 이 방정식의 근은 $e_u(t) = e_u(0)exp(-\lambda_1 t)$ 로 되어 시간 t 가 증가하면 $e_u(t)$ 가 0으로 수렴하게 된다.

식 (30)과 식 (27)을 이용하면 원하는 프로펠러 추력 X_{Pd} 는 식 (31)과 같이 표현할 수 있다.

$$X_{Pd} = m_0(\dot{u}_d(s) - \lambda_1 e_u) - X_h(u, v, r) - X_R(u, v, n, \delta) \quad (31)$$

여기서 $m_0 = (m - X_u)$ 이다.

프로펠러 속도명령 n_0 는 X_{Pd} 를 이용하여 식 (16)에서 구할 수 있다.

다음은 선수요 방향, 즉, Yaw Rate 제어알고리즘 설계과정이다. 먼저 Yaw Rate 제어 오차 e_r 을 식 (32)와 같이 정의한다.

$$e_r = r - r_d(s) \quad (32)$$

여기서 $r_d(s)$ 는 선박의 원하는 Yaw Rate이다.

또한 원하는 페루프 동역학을 식 (33)과 같이 표현한다.

$$\dot{e}_r + \lambda_2 e_r = 0 \quad (33)$$

여기서 λ_2 는 양의 상수이다. 식 (28)과 식 (33)을 이용하면 아래 식 (34)를 얻는다. 식 (34)에서 u_{rd} 는 타 및 쓰러스터가 발생시켜야 할 선수요방향 유체 모우멘트라 할 수 있다.

$$c(Y_R + Y_T) + d(N_R + N_T) = -cY_h - dN_h - \lambda_2 e_r + \dot{r}_d(s) = u_{rd} \quad (34)$$

여기서 c 와 d 는 각각 아래 M_1 행렬의 역행렬의 (2,1) 및 (2,2) 항이다.

$$M_1 = \begin{bmatrix} (m - Y_v) & (mx_G - Y_r) \\ (mx_G - N_v) & (I_{ZZ} - N_r) \end{bmatrix} \quad (35)$$

제어기는 타와 쓰러스터가 식 (34)를 만족시키는 유체력 모우멘트를 발생시켜야 한다. 이 경우 타가 없어도 선수/선미 쓰러스터만으로 식 (34)를 만족하게 할 수 있으므로 역시 과도 입력 제어문제가 된다. 문제를 비교적 간단하게 하기위해 선수 쓰러스터 회전수와 선미 쓰러스터 회전수가 서로 절대값은 같고 부호만 다르게 설계 할 수 있다. 이 경우 쓰러스터 시스템이 근사적으로 선수요 모우멘트만 발생시키고 횡방향 힘은 발생시키지 않는다. 이 경우 $Y_T = 0$ 가 되므로 식 (34)는 식 (36)이 된다.

$$cY_R + d(N_R + N_T) = u_{rd} \quad (36)$$

과도입력 문제를 풀기위해 최적화 기법을 이용한 제어입력 분배를 적용 할 수 있다. 먼저 식 (36)은 식 (9)~(14) 및 (18)~(21)의 타력 및 쓰러스터가 발생시키는 유체력 표현식 으로부터 식 (37)과 같이 표현 될 수 있다.

$$u_{rd} = 2dx_{TB}b_B v_B n_B - \{c(1 + a_H) + d(x_R + a_H \dot{x}_H L)\} \frac{\rho}{2} A_R f_a U_R^2 \sin(\delta + \frac{v_R}{u_R}) \cos \delta \quad (37)$$

식 (37)을 유도하는 데에는 쓰러스터에 관한 계수들의 관계를 $n_S = -n_B$ 및 $b_S = b_B$, $x_{TS} = x_{TB}$ 로 근사적으로 가정하였다.

최적화 문제는 선속이 예를 들어 $0.4m/sec$ 이상인 경우에 대하여 성능지수 $J = n_B^2$ 를 최소화하고 식 (37)을 만족하는 δ 를 구하는 것으로 설정할 수 있다. 이 경우 선속이 $0.4m/sec$ 이상인 경우 가급적 타를 이용하는 것이 된다.

한편 J 는 식 (37)을 이용하면 식 (38)과 같이 표현된다.

$$J = n_B^2 = \frac{|u_{rd} + (c(1 + a_H) + d(x_R + a_H \dot{x}_H L)) \frac{\rho}{2} A_R f_a v_R^2 \sin(\delta + \frac{v_R}{u_R}) \cos \delta|}{2dx_{TB}b_B} \quad (38)$$

식 (38)에서 수치적으로 δ 를 변화시켜가며 계산하여 최소의 J 를 갖는 δ 를 찾을 수 있다. 선속이 $0.4m/sec$ 이하인 경우 $J = \delta^2$ 으로 놓고 상기 문제를 풀면 가급적 타의 사용을 최소화 하는 제어입력 분배가 된다. 본 연구의 시뮬레이션에서는 선속이 $0.4m/sec$ 이하인 경우 쓰러스터만을 사용하였다.

3.3 조이스틱 PID 제어알고리즘 설계

PID(Proportional, Integral, Derivative)제어기는 제어 대상 시스템의 모델 또는 동역학식을 이용하지 않고 제어 알고리즘을 설계 가능하다는 장점이 있는 제어기이다. 본 연구에서는 저속 운항 동역학 모델을 이용한 비선형 제어기와의 성능을 비교하고자 PID제어기를 적용하였다.

진진 방향 속도 및 회전 방향 속도의 오차를 역시 식 (29), (32)와 같이 정의할 경우, 각 액추에이터들의 속도 명령은 아래 식 (39)~(42)와 같이 설계될 수 있다.

$$n_C = -k_{P,I} e_u - k_{P,I} \int_0^t e_u(t) dt - k_{P,D} \frac{de_u(t)}{dt} \quad (39)$$

$$\delta_C = -k_{r,P} e_r - k_{r,I} \int_0^t e_r(t) dt - k_{r,D} \frac{de_r(t)}{dt} \quad (40)$$

$$n_{BC} = -k_{t,I} e_r - k_{t,I} \int_0^t e_r(t) dt - k_{t,D} \frac{de_r(t)}{dt} \quad (41)$$

$$n_{SC} = -n_{BC} \quad (42)$$

여기서, $k_{P,P}, k_{P,I}, k_{P,D}$ 와 $k_{r,P}, k_{r,I}, k_{r,D}, k_{t,P}, k_{t,I}, k_{t,D}$ 는 각각 Proportional, Integral, Derivative 제어 정수이다.

4. 시뮬레이션 및 실험

본 연구에서 개발한 제어기의 성능을 검증하기 위해 선박이 Fig. 6과 같은 Z형의 좁은 수로를 통과하는 경우에서의 운항 난이도를 평가하기 위해 다음 식 (43)과 같은 액추에이터 사용 지수(Index)(IMO 선박 조종성 기준 검토 위원회, 2002)를 도입하였다.

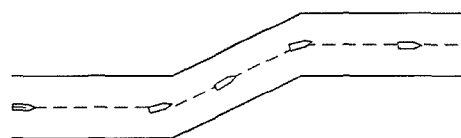


Fig. 6 Type of assumed water way(30bend water way)

$$Actuator\ Index = \frac{1}{T_f - T_0} \int_{T_0}^{T_f} |actuator\ value| dt \quad (43)$$

여기서, T_f 는 수로 통과를 위해 액츄에이터의 사용이 끝나 는 시각이며, T_0 는 수로 통과를 위해 액츄에이터의 사용이 시 작되는 시각이다.

각 제어기들의 검증에서 사용된 궤적 추종오차는 Fig. 7의 빗금 친 면적으로 정해진 항로와 실제 선박의 궤적간의 차이 를 나타내며 식 (44)과 같이 계산하였다.

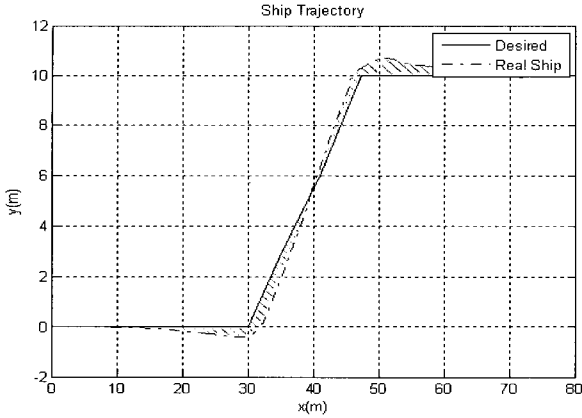


Fig. 7 Error Index area for a typical closed loop control

$$Error\ Index = \int_{X_i}^{X_f} |Desired\ Y - Real\ Y| dx \quad (44)$$

여기서, X_i 는 선박 초기위치의 X좌표이며 X_f 는 선박 최종 위치의 X좌표이다.

위의 실험에서 선박의 평균 속도는 1.3knot, 최고 속도는 3.34knot이며 3knot를 넘는 구간이 전체의 6.28%만을 차지하기 때문에 쓰러스터의 영향이 충분하다고 판단하였다.

각 제어기의 수로통과 HILS 실험을 5회 수행하여 궤적 추종 오차와, 각 액츄에이터의 사용지수의 평균값을 통해 비교 분석하였다. 또한 한국 해양연구원의 선박 운항 시뮬레이터에 설계한 제어기를 적용하여 그 성능을 검증하였다.

4.1 비선형 및 PID 제어기의 시뮬레이션 검증

아래 Fig. 8은 다섯 번에 걸친 실험에서의 궤적 추종 오차를 구하고 그 값의 평균을 그래프로 표현한 그림으로서 개루프에서의 궤적 추종 오차가 PID제어기나 비선형 제어기와 같은 폐루프 제어기에서의 그것 보다 상당히 크다는 것을 알 수 있다. 이를 통해 사용자가 원하는 궤적을 추종함에 있어서 폐루프 제어기가 좋은 성능을 가진다는 것을 확인 할 수 있다.

Fig. 8~Fig. 10에서의 결과를 보면 비선형 제어기와 PID제어기와의 성능의 차이가 없는데, 조류 및 풍력과 같은 외란이 있는 경우 비선형 제어기가 PID 제어기보다 우수한 성능을 나타낼 수 있다.

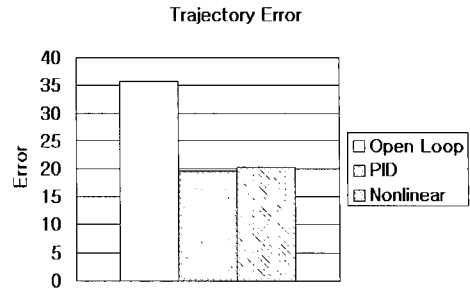


Fig. 8 Trajectory following error areas for open loop, PID and nonlinear controllers

아래 Fig. 9과 Fig. 10는 각각 타와 선수 쓰러스터의 사용지수 평균값을 나타낸 그림이다. 그림을 보면 폐루프에서 보다 개루프에서의 액츄에이터 사용지수(식(43)에 정의)가 현격히 더 크다는 것을 알 수 있다. 이를 통해 폐루프가 개루프보다 선박의 조종 편이성 측면에서 크게 유리하다는 것을 알 수 있다.

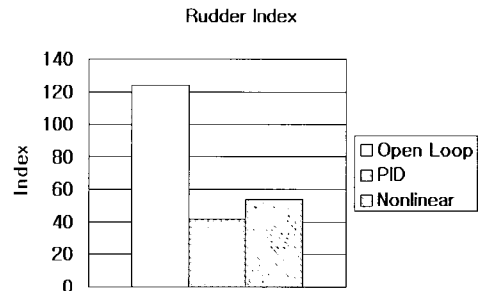


Fig. 9 The average values of rudder indices for open loop, PID and nonlinear controllers

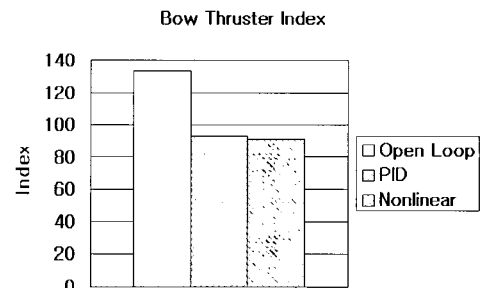


Fig. 10 The average values of bow thruster indices for open loop, PID and nonlinear controllers

4.2 PID 제어기의 항해시뮬레이터 적용 검증

선박 조종 시뮬레이터에서의 실험을 통한 검증은 개발한 제어 알고리즘을 실선에 적용하기 전에 제어 알고리즘의 효율성을 검증하고, 개선점을 찾는 데 목적이 있다.

본 연구에서는 주 추진 프로펠러와 타 이외에 선수/선미 쓰러스터를 구동기로 갖는 해양경찰청의 5,000톤급 경비정(L : 141m, B : 16.5m)의 특성 및 외부 조건을 사실적으로 구현한 한국 해양연구원의 선박 조종 시뮬레이터를 이용하여 제어 알고리즘 검증을 수행하였다.

본 실험은 선박 조종자가 선박의 현재 위치와 외부 조건(다른 선박)을 파악하여 부두로 선박을 접안하는 실험, 선박이 부두에서 이안하는 실험 등 두 가지 실험을 수행하였다.

먼저 Fig. 11이 선박의 출발지에서 접안까지의 궤적과, 타 선박의 궤적, 항구의 위치, 부표와 방파제 등의 장애물들이 표현된 그림이다. 그림을 보면 선박이 항로를 유지하면서 타 선박과의 충돌을 피하고, 기존에 부두 내에 정박해 있던 선박들을 피해 정확히 접안한 것을 확인 할 수 있다.

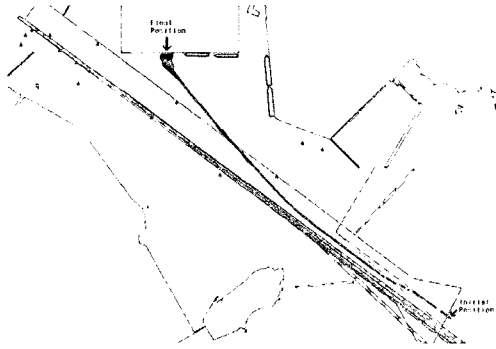


Fig. 11 Ship trajectory of berthing experiment on ship maneuvering simulator using PID controller

를 보면 원하는 속도 명령과 약간의 오차가 있지만 대체적으로 잘 추종하는 것을 볼 수 있다. 다만 원하는 속도가 급격하게 변하는 구간에서 실제 속도가 원하는 속도를 추종하지 못하고 있다.

다음으로 Fig. 13는 접안 실험과 반대로 선박이 부두에 정박해 있다가, 항만 밖으로 나가기 위해 이안한 실험에서의 선박의 궤적과 타 선박의 궤적, 항구위치, 부표, 방파제등을 나타내는 그림이다.

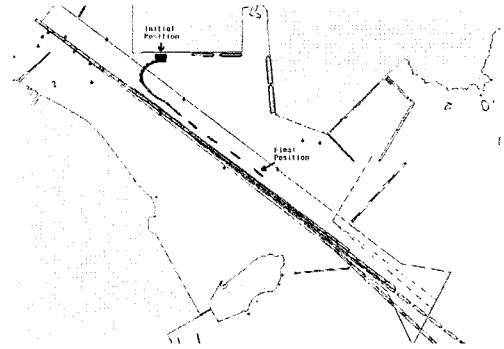


Fig. 13 Ship trajectory of unberthing experiment on ship maneuvering simulator using PID controller

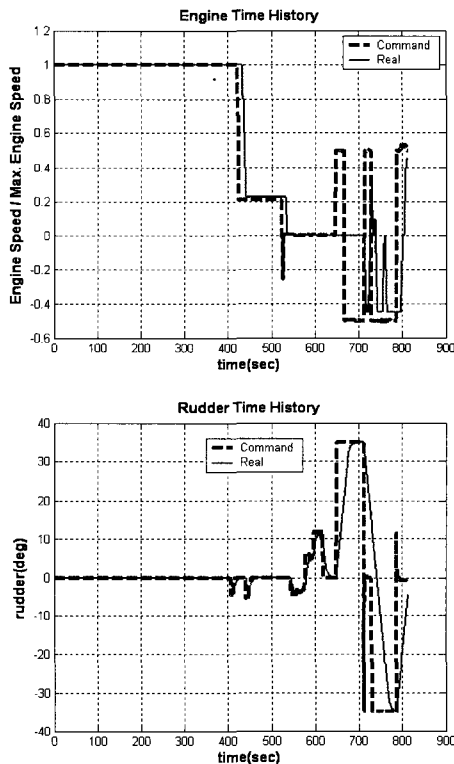


Fig. 12 Engine and rudder time history of berthing experiment on ship maneuvering simulator using PID controller

아래 Fig. 12은 이 실험에서의 주 엔진과 타의 시간이력이 원하는 속도 명령을 얼마나 잘 따라 가는지 보여준다. 결과

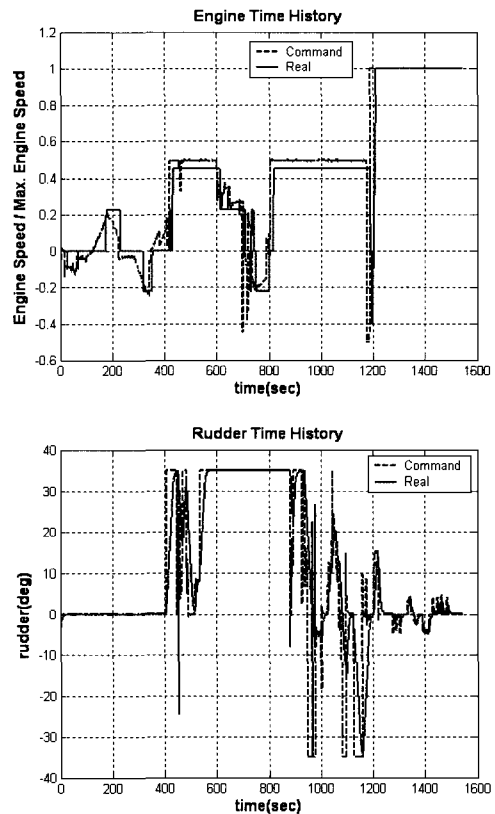


Fig. 14 Engine and rudder Time history of unberthing experiment on ship maneuvering simulator using PID controller

Fig. 14은 이안 실험에서의 주 엔진과 타의 시간이력이다.

그림을 살펴보면 속도 명령이 급격하게 변하는 부분에서는 약간의 시간 지연을 가지면서 액츄에이터의 실제 속도가 액츄에이터 속도 명령을 추종하는 것을 볼 수 있다. 그러나 숙련된 선박 조종자일수록 이러한 급격한 제어 명령 변화가 줄어들므로 이러한 문제는 해결될 수 있을 것이다.

5. 결 론

본 연구에서는 도선사와 예인선에 의지하지 않고 선박 운항 경험이 없는 초보자도 쉽게 조종할 수 있는 조이스틱에 의한 선박 접이안 시스템의 개발을 위한 연구를 수행하였다. 이는 기존의 조타 휠보다 사용자에게 친숙한 조이스틱을 활용함으로써 사용자에게 편안함을 주고 정확한 속도명령을 줌으로서 선박을 정확하게 제어하기 위함이다.

시스템을 개발하기 위해 주 추진 프로펠러와 타 의외에 선수/선미 쓰러스터를 갖는 선박을 대상으로 설정하여 비선형 제어이론을 이용한 제어 알고리즘 및 PID 제어 알고리즘을 설계하였다. 또한, 실시간으로 조이스틱의 입력과 저속 운항 동역학모델을 이용한 선박 접이안 HILS 프로그램을 LabWindows/CVI를 이용하여 구현하였다. 또한 제어기의 성능을 검증하기 위하여 한국 해양연구원의 선박 운항 시뮬레이터에 개발한 PID 제어기를 적용하여 실험 검증하였다.

본 연구를 통해 개발된 페루프 조이스틱 제어기가 여러 종류의 시뮬레이션과 항해 시뮬레이터에 적용한 실험을 통해 개루프 제어보다 선박의 조종성이 뛰어난 것을 알 수 있었다. 또한, PID 제어기와 비선형 제어기의 성능을 비교하면 시스템에 파도나, 바람, 조류와 같은 외란이 없는 시뮬레이션 환경에서는 두 제어기의 성능차이는 크게 없음을 알 수 있었다.

후 기

본 논문은 한국해양연구원에서 수행하고 있는 기본사업 “네트워크 기반 항만관제 및 항법체계 기술 개발”의 지원으로 수행되었습니다.

참 고 문 헌

- [1] IMO 선박 조종성 기준 검토 위원회(2002) “IMO 조종성 기준 개정을 위한 연구”, 대한 조선학회, pp.110-226.
- [2] 원문철(2005) “선박 자동 접이안용 MIMO 제어기 설계 및 검증 연구 결과 보고서”
- [3] 윤현규, 김선영(2005), “저속시 선체에 작용하는 유체력 수학적모형 정립을 위한 모형실험 방안 연구”, 대한조선학회 논문집 제 42권 제 2호, pp.98-104.
- [4] Fossen, T.(1994), “Guidance and Control of Ocean Vehicles.” pp. 307-321.
- [5] Fossen, T.(2002), “Marine control Systems”, Marine Cybernetics
- [6] HarbourMarine(2006), SmartDock® Docking & Piloting Systems <http://www.harbourmarine.com/>
- [7] Hasegawa, K., Kitera, K.(1993), “Automatic Berthing Control System Using Network and Knowledge-base.”, Journal of Kansai Society of Naval Architecture of Japan, pp.144-153.
- [8] Im, N., Hasegawa, K. (2002) “Motion Identification using Neural Networks and Its Application to Automatic Ship”, Journal of Ship and Ocean Technology, pp.16-26.
- [9] MARIMATECH(2006), Portable Piloting System, <http://www.marimatech.com/>
- [10] Slotine, J. J. and Li, W.(1991), “Applied Nonlinear Control”, Prentice Hall.
- [11] Yamato, H., Koyama, T.(1992), “Automatic berthing using the expert system”, control application in marine systems, pp. 173-184.

원고접수일 : 2007년 3월 29일

원고채택일 : 2007년 6월 27일