

진화전략 기반 경로탐색 알고리즘을 활용한 선박경제운항시스템

방 세 환*, 권 영 근^o

Economic Ship Routing System by a Path Search Algorithm Based on an Evolutionary Strategy

Se-hwan Bang*, Yung-keun Kwon^o

요 약

선박경제운항이란 기상예측정보를 활용하여 연료소모량을 최소화하도록 선박을 운항하는 것으로서 최근 다양한 방법론이 연구되고 있다. 성공적인 경제운항시스템을 구현하기 위해서는 기상을 고려하여 적절하게 엔진 출력을 조절하거나 지리적 운항 경로를 바꾸는 방법이 필요하다. 그러나 지리적 운항 경로의 결정은 항해 시각에 따라 연료소모량이 변하는 동적 비용 문제임을 고려할 때 최적의 해를 찾기가 어렵다. 이에 이 논문에서는 매우 많은 지리적 후보 경로들 중에서 우수한 품질의 해를 효과적으로 탐색하기 위한 진화전략 기반 경로탐색 알고리즘을 제안하였다. 제안된 방법을 7개 노선에 대해 실험한 결과 최단거리 운항 방법에 비해 운항소요시간은 거의 차이가 없으면서도 연료소모량을 평균 1.82%, 최대 2.49% 개선시킬 수 있었다. 특히 사례 분석을 통해 제안된 방법이 악천후를 회피할 수 있는 지리적 경로를 탐색할 수 있음을 확인하였다.

Key Words : Economic Ship Routing, Weather Routing, Evolutionary Strategy, Fuel Consumption

ABSTRACT

An economic ship routing means to sail a ship with a goal of minimizing the fuel consumption by utilizing weather forecast information, and there have been various systems which have been recently studied. For a successful economic ship routing system, it is needed to properly control an engine power or change a geographical path considering weather forecast. An optimal geographical path is difficult to be determined, though, because it is a minimal dynamic-cost path search problem where the actual fuel consumption is dynamically variable by the weather condition when the ship will pass the area. In this paper, we propose an geographical path-search algorithm based on evolutionary strategy to efficiently search a good quality solution out of tremendous candidate solutions. We tested our approach with the shortest path-based sailing method over seven testing routes and observed that the former reduced the estimated fuel consumption than the latter by 1.82% on average and the maximum 2.49% with little difference of estimated time of arrival. In particular, we observed that our method can find a path to avoid bad weather through a case analysis.

※ 이 논문은 현대중공업 지원에 의한 2014년 울산대학교 전기공학부 일류화 연구비에 의하여 연구되었음

• First Author : University of Ulsan Department of IT Convergence, paumqkd@mail.ulsan.ac.kr, 학생회원

o Corresponding Author : University of Ulsan Department of IT Convergence, kwonyk@ulsan.ac.kr, 정회원

논문번호 : KICS2014-04-150, Received April 30, 2014; Revised July 24, 2014; Accepted August 18, 2014

I. 서론

선박경제운항이란 연료소모량을 최소화하도록 선박을 경제적으로 운항하는 것을 말하는데, 국제적인 에너지 절약 필요성과 환경보호 필요성에 의해 그 중요성이 더욱 부각되고 있다. 가장 보편적인 경제 운항 방법으로는 선박 엔진의 출력을 적절하게 감속하여 연료소모량 대비 엔진 출력의 효율을 높이는 감속 운항 방법이 있다. 그러나 감속 운항 방법은 연료소모량을 줄이기는 하지만 결국 운항 시간이 늘어나므로 다른 비용을 증가시키게 된다. 감속 운항 방법 대신 최근에는 기상예측정보와 선박제원정보를 활용하여 운항 소요 시간이 비슷하면서도 연료소모량을 줄일 수 있는 방법에 대해 다양한 연구가 시도되고 있다¹⁻⁴⁾. 예를 들면, 선박이 실제로 지나가는 경로의 기상 상황을 고려하여 엔진출력을 세부구간별로 최적화 하는 방법이 제안되었는데 36가지 항로에 대해서 모의실험한 결과 고정출력 운항에 비해 평균 1.3% 연료소모량 감소된 결과를 보여주었다¹⁾. 하지만 이 시스템은 지리적으로 최소거리 항로를 계산하여 그 항로에 대해서만 최적의 세부구간별 엔진출력을 탐색한다는 한계점을 가지고 있다. 따라서 더 향상된 선박경제운항 시스템을 위해서는 다양한 지리적인 최적 항로를 찾는 방법이 필요하다.

이에 이 논문에서는 지리적인 최적화를 위해 진화전략(Evolutionary Strategy; ES) 기반 탐색 알고리즘을 제안하고 이를 바탕으로 개선된 선박경제운항시스템을 개발하고자 한다. 무분별한 탐색을 제한하고 효율적인 탐색을 위해 A* 알고리즘을 사용하여 지리적 최단경로를 찾고 이 최단경로를 초기해로 사용하여 진화전략기반 경로탐색을 할 수 있도록 구현하였다. 이 논문에서는 7개의 실험 노선에서 각각 12회 운항 시간표(즉, 총 84개 스케줄)에 대해 이전 연구¹⁾에서 제시된 운항방법과 이 논문에서 제안하는 시스템의 성능을 비교하였다. 그 결과 기존 시스템에 비해 이 시스템의 경제운항방법은 운항소요시간에 큰 차이가 없으면서도 평균 1.82%, 최대 2.49%의 연료소모량이 감소함을 관찰할 수 있었다.

이 논문의 전체 구성은 다음과 같다. 2장에서는 선박경제운항시스템에 대한 배경지식을 설명하고, 3장에서는 이 논문에서 제안하는 시스템에 대한 내용을 다룬다. 그리고 4장에서는 모의실험 결과에 대해 성능 분석을 하고, 5장에서 결론을 맺는다.

II. 배경 지식

이 장에서는 경제운항시스템을 구현하는데 필요한 배경 지식을 설명한다. 2.1절에서는 경제운항 시스템 구축에 필요한 기상예보데이터와 지리 정보데이터에 대해서 설명한다. 2.2절에서는 이 연구에서 사용하는 연료소모량 추정식에 대해서 살펴본다.

2.1 지리정보 및 기상예보 데이터

경제운항 시스템을 구축하기 위해서는 유효한 해상 경로를 찾는 과정에서 전 세계 지리정보 데이터가, 연료소모량을 추정하는 과정에서는 기상정보 데이터가 각각 필요하다. 이를 위해, 지리정보 데이터는 미국국립지리원(<http://www.ngdc.gov>)로부터 획득한 1,938 개의 벡터데이터를 바탕으로 7200 * 3600 픽셀의 비트맵 정보를 구축하여(그림 1) 육지와 해상을 구분하였다. 한편, 기상정보 데이터는 유럽중기예보센터(<http://www.ecmwf.int/>)에서 2011년 1월 1일부터 2011년 12월 31일까지 1년 동안의 세계 해양을 경위도 각각 1.5도 단위로 나눈 격자점에 대해서 6시간(0시, 6시, 12시, 18시)마다 이뤄진 예보 데이터이며 이 시스템에서는 많은 기상정보들 중에서 연료소모량추정식에 필요한 파향, 파고, 풍향, 풍속 등 네 가지 정보를 획득하였다¹²⁾.

2.2 연료소모량추정식

파도와 바람은 대표적인 기상 정보이다. 이들 기상정보의 크기야 방향성은 선박운항에 많은 변수를 만들게 된다. 일반적으로 선박의 엔진은 출력에 따른 속도와 기대치가 있는데, 이러한 기상환경들은 기대속도와 실제속도의 차이를 만드는 원인이 된다. 즉, 기상환경에 따라 선체에 부가저항이 생기게 되면, 부가저항은 선속저하로 이어지게 되고, 여기서 떨어진 선속을 기대치까지 보정하기 위해서는 추가적인 엔진의 마력 증가가 필요하고, 이는 결과적으로 연료소모량의

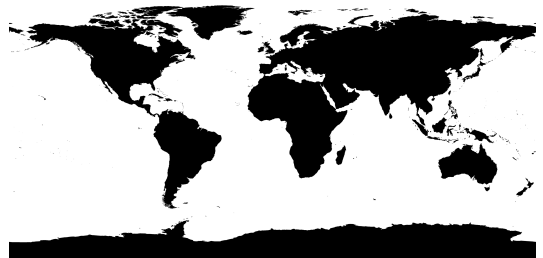


그림 1. 비트맵 기반 지리정보 데이터
Fig. 1. Bitmap-based geographical data

증가를 불러오게 된다.

이 논문에서는 기존의 해상 상태에 따른 선속 저하량의 계산 방법 중 Townsin-Kwon 연료추정식^[5]계산식을 활용하여 연료소모량을 추정하였다. 이 식을 적용하기 위해서는 선박의 길이, 너비, 깊이 등의 9가지 선박제원 정보(4.1절에서 설명)와 파고, 파향, 풍속, 풍향과 같은 4가지 기상정보, 그리고 선박, 운항방향, 운항속도, 운항거리와 같은 3가지 운항정보가 입력으로서 필요하다.

III. 제안된 선박경제운항시스템

3.1 시스템 개념

이 논문에서 제안하는 선박경제운항시스템의 전체적인 구성은 그림 2와 같이 초기항로 탐색, 최적 항로 탐색, 최적 출력 탐색의 3단계로 구성된다. 먼저, 첫 번째 단계에서 출항지와 입항지의 경위도가 주어지면 2.1절에서 설명한 세계 지리정보 데이터를 바탕으로 A* 알고리즘을 통해 지리적 최단 경로(P)를 찾아 다음 단계의 최적 항로 탐색에서 초기 항로로서 활용된다. 두 번째 단계에서는 여러 개의 세부 구간으로 구성된 초기 항로 P에 진화전략 알고리즘을 활용하여 지리적 최적 경로(P0)를 구한다. 마지막 단계에서는 여러 개의 세부 구간들로 구성된 지리적 최적 경로 P0에 대해 각 세부구간에서의 최적엔진출력을 탐색하기 위해 이전 연구^[1]에서 제안된 진화전략기반 출력탐색 알고리즘을 적용한다.

3.2 A* 알고리즘을 이용한 초기 항로 탐색

이 논문에서는 운항 경로라는 해의 표현을 위해 이전 연구^[1,2]에서 사용된 격자좌표체계와 같은 형태를 사용한다. 즉, 기상예보 데이터가 경위도 1.5도 단위로 주어짐에 따라 전 세계의 해상 지리를 역시 1.5도 단위로 나누어 전 세계의 지리를 총 240x120=28,800개의 블록으로 구분하고 각 블록에 8개의 정점을 배치함으로써 전 세계 지리를 그래프 G(V,E)로 표현한다. 그리고 그림 3과 같은 복잡도가 O(|V|2)인 A* 알고리즘을 적용하여 지리적 최단 경로 P를 구한다^[1,2].

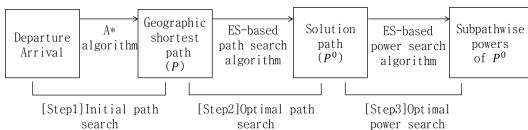


그림 2. 이 논문에서 제안하는 선박경제운항시스템의 구성
Fig. 2. An economic sailing system proposed in this paper

```

A*(G(V,E), v_s, v_g) {
    // G(V,E) : A graph of world map.
    // v_s ∈ V, v_g ∈ V : Departure and arrival vertices.
    P(v_s) ← null; // P(v): A string of the shortest path
    from v_s to v.
    // h(v,w) : The distance between the vertex v and w.
    g(v_s) ← 0, f(v_s) ← g(v_s) + h(v_s, v_g);
    g(v) ← ∞, f(v) ← ∞ ∀ v ∈ V \ {v_s};
    C ← ∅;
    while (TRUE) {
        w ← argmin_{v ∈ V \ C} f(v);
        if (w = v_g) break;
        for (∀ v ∈ {v | v ∈ V \ C and (w,v) ∈ E}) {
            // If (w, v) collides the land, collision_check
            (w, v) is TRUE.
            if (collision_check(w, v)) continue;
            l ← g(w) + h(w, v);
            if (l < g(v)) {
                g(v) ← l;
                f(v) ← g(v) + h(v, v_g);
                // ⊕
                P(v) ← P(w) ⊕ v; (string concatenation)
            }
        }
        C ← C ∪ {w};
    }
    return P(v_g); // The result of A* Algorithm.
}
    
```

그림 3. 지리적 최단경로를 찾기 위한 A* 알고리즘
Fig. 3. A* Algorithm' for the geographical shortest path

지리적 최단 경로 P는 모든 해상 기상 정보가 동일할 경우 최적 항로가 될 수 있지만, 기상이 다를 경우 최적 항로의 탐색이 필요하다. 이를 위해 이 논문에서는 진화전략 기반 경로탐색 알고리즘을 제안하는데, 이 때 경로 P는 탐색 알고리즘의 초기 항로로 사용된다.

3.3 진화전략 기반 경로탐색 알고리즘을 사용한 지리적 최적경로 탐색

이 경제운항시스템에서 제안하는 진화전략 기반 경로탐색 알고리즘은 A* 알고리즘에 의해 찾은 지리적 최단 경로를 초기해로 하여 진화전략을 이용한 탐색 알고리즘이다. 그림 4는 이 연구에서 제안하는 진화전략 기반 경로탐색 알고리즘의 의사코드로서 해 x^t는 출발지에서 도착지까지의 경위도를 가진 노드 좌표들의 리스트로 표현된다. 각 세대에서 부모해 x^t는 변이 연산자를 통해 자식해 y를 생성하며 y의 품질이 x^t의 품질보다 우수할 경우 다음 세대의 부모해로 대체된

```

ES-PS( $V_s, V_g$ ) {
     $t \leftarrow 0$  // Initialize  $t$  at the first generation.
     $x^t \leftarrow A^*(V_s, V_g)$  // Initialize first parent solution.
    while ( $t < T$ ) { // Repeat the loop until  $t = T$ .
         $y \leftarrow mutate(x^t)$  // Node mutation.
        //  $\gamma$  : Sailing time,  $\delta$  : Acceptance rate of
        // sailing time,  $\zeta$  : Fuel consumption estimation
        if ( $\gamma(y^t) - \gamma(x^t) \leq \delta$  and  $\zeta(y^t) \leq \zeta(x^t)$ )
            then  $x^{t+1} \leftarrow y^t$  // Select the child solution for the next
            generaion.
        else  $x^{t+1} \leftarrow x^t$  // Select the parent solution.
         $t \leftarrow t + 1$ 
    }
    return  $x^T$ ; // The result of ES-based path-search
    algorithm.
}
    
```

그림 4. 진화전략 기반 경로탐색 알고리즘
 Fig. 4. ES-based path-search algorithm

다. 이 때, 해의 품질은 선박의 출력을 일정한 값으로 고정하여 평가한 예상 연료소모량으로 정의된다. 한편, 변이연산자는 경로 내 임의의 두 좌표를 정규 확률변수 값만큼 변화시키는 형태로 구현되었다(그림 5). 그림 6은 이 논문에서 정의한 변이연산자의 예를 나타낸다.

3.4 진화전략 기반 최적 출력 탐색

3.3절에서 진화전략 기반 경로탐색 알고리즘에 의해 얻은 지리적 최적 경로는 여러 개의 세부 구간들로 구성된다. 이 지리적 최적 경로는 기본적으로 선박의 출력을 일정한 값으로 고정하여 찾아낸 경로이므로 각 세부구간별로 출력 값을 변화 시키면(즉, 선박의 속도를 조절하는 효과) 연료소모량을 더욱 줄일 수 있

```

mutate( $x$ ) {
    //  $x := x_0 x_1 \dots x_n$  is a list of coordinates( $x_i$  is
    // ancoordinate represented by (latitude, longitude)).

    // Select  $x_i$  and  $x_j$  out of  $\{x_1, x_2, \dots, x_{l-1}\}$  at
    // random.
     $x_i(lat_i, lon_i), x_j(lat_j, lon_j) \leftarrow random\_select(\{x_1, x_2, \dots, x_{l-1}\})$ 
    //  $N(0, \sigma)$  denotes a random variable by a normal
    // distribution with mean 0 and standard deviation  $\sigma$ .
     $x_i' \leftarrow (lat_i + N(0, \sigma), lon_i + N(0, \sigma))$ 
     $x_j' \leftarrow (lat_j + N(0, \sigma), lon_j + N(0, \sigma))$ 
    return
     $y := x_0 x_1 \dots x_{i-1} x_i' x_{i+1} \dots x_{j-1} x_j' x_{j+1} \dots x_l$ 
}
    
```

그림 5. 변이함수 의사코드
 Fig. 5. Pseudo-code of a mutation

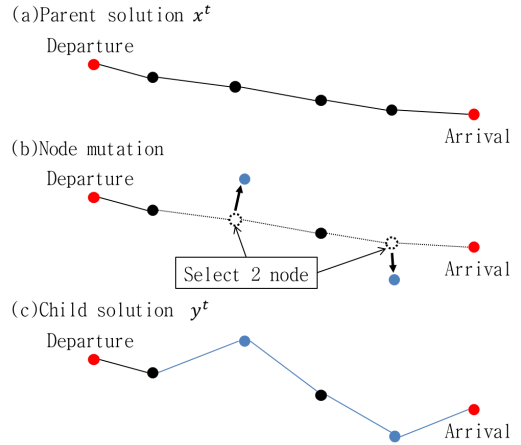


그림 6. 변이연산 예제
 Fig. 6. An illustrative example of a mutation operation

는 항로를 찾아 낼 수 있다. 이를 위해 이 논문에서는 기존 연구에서 활용되었던 진화 전략 기반의 출력 최적화 알고리즘¹¹⁾을 이용하여 세부구간별 최적 출력을 구한다. 또한, 적절한 범위 내의 출력 탐색이 되도록 하기 위해 ES기반 출력탐색 운항법에 의한 도착시간이 고정출력 운항법에 의한 도착시간에 비해 0.3% 이내의 오차범위(항해 시간 허용 비율)에서 출력탐색을 하도록 하였다.

IV. 모의실험 결과

4.1 매개변수 설정

표 1은 이 논문의 모의실험에서 사용된 선박제원에 관한 매개변수이다. 여기에는 선박의 길이, 너비, 깊이가

표 1. 모의실험에 사용된 선박제원 매개변수
 Table 1. Ship information parameters for simulation

	Item	Unit
Ship information parameters	Length	315.0(m)
	Breadth	57.2(m)
	Depth	30.4(m)
	Draft	20.8(m)
	Block Coefficient	0.8
	Longitudinal Center of Buoyancy, Forward Direction in m	10.17(m)
	propeller diameter	9.1(m)
	propeller efficiency	0.6
	Specific Fuel Oil Consumption (SFOC)	125.0(g/hr)

등 9가지의 매개변수가 포함되는데 이러한 정보들은 2.2절에서 설명한 연료소모량 추정식에서 사용된다. 이 논문에서는 기존 논문¹¹⁾에서 사용된 재화 중량 300,000톤급 VLCC(Very Large Crude Carrier)의 제원 정보를 활용하였다.

4.2 경로 및 운항방법별 성능 분석

이 논문에서 제안한 시스템의 성능을 검증하기 위하여 7개의 실험 노선에 대하여 2011년 1월부터 12월 까지 1개월 간격으로 출발 날짜를 변경하며 모의실험을 실행하였다(즉, 총 84가지의 운항 시간표). 운항 방법으로는 경로탐색 방법과 출력탐색 방법에 따라 “A*-FIX,” “A*-ES,” “ES-FIX,” “ES-ES” 등 4가지 조합이 가능하다(표 2). “A*-FIX”와 “A*-ES”는 A* 알고리즘을 이용하여 최단경로를 구한 후 세부 구간에 대하여 고정 출력으로 운항하거나 3.4절에서 설명한 진화전략 기반 출력 최적화 알고리즘을 적용한 방법을 각각 의미한다.

특히, “AF-ES”는 기존 연구¹¹⁾에서 제안된 방법을 의미한다. 또한, “ES-FIX”와 “ES-ES”는 3.3절에서 설명한 진화전략 기반 경로탐색 알고리즘을 이용하여 최적 경로를 탐색한 후 앞과 마찬가지로 고정 출력 혹은 진화전략기반 출력최적화 알고리즘을 적용한 방법을 의미한다. 한편, 공정한 비교를 위하여 모든 운항

표 2. 경로탐색 방법과 출력탐색 방법에 따른 4가지 조합
Table 2. Four approaches by combining path-search and power-search methods

	Power-search methods	Power search	ES searching
path-search methods			
Shortest path		A*-FIX	A*-ES
ES searching		ES-FIX	ES-ES

방법에 대해 동일한 도착시각에 관한 최종기한을 설정 하였다.

표 3은 운항 방법에 따른 7개 운항 시간표에 대한 모의실험 결과를 보여준다. 그 표에서 “A*-FIX”의 “평균 연료소모량(편차)”는 해당 실험 노선에 대해 12회(매월 1회) 운항에 따른 연료소모량의 평균 및 표준 편차이며 “A*-FIX대비 개선율”은“A*-FIX”운항 방법의 평균 연료소모량에 비해 해당 운항 방법들의 평균 연료소모량과 평균 운항시간이 몇 % 줄었는지를 나타낸다(즉, 개선율이 양수이면 개선 되었음을 나타낸다). 실험결과 세 운항 방법의 전체 운항시간 평균 개선율은 각각 -0.39%, 0.65%, 0.27%으로 “A*-FIX”의 운항시간과 거의 차이가 없었으며 전체 연료소모량 평균 개선율은 각각 1.17%, 0.93%, 1.82%으로 모두 “A*-FIX”의 성능을 개선시켰다. “A*-ES”는 운항 시간은 늘었지만 연료소모량을 줄일 수 있었고 “ES-FIX”와 “ES-ES”는 운항시간을 줄이면서 동시에

표 3. Performance comparison between A* path searching and the ES path searching
Table 3. A* 경로탐색과 ES 경로탐색의 성능비교

Simulation schedule	A*-FIX		A*-ES		ES-FIX		ES-ES	
	Average fuel consumption (Standard deviation)	Average sailing time (Standard deviation)	Improvement rate compared to A*-FIX		Improvement rate compared to A*-FIX		Improvement rate compared to A*-FIX	
			Fuel consumption	Sailing time	Fuel consumption	Sailing time	Fuel consumption	Sailing time
Durban → Santos	1,039.65(ton) (10.57)	270.56(hour) (2.49)	1.69%	-0.38%	1.26%	1.39%	2.49%	0.99%
Busan → Oakland	1,234.56(ton) (11.75)	321.29(hour) (2.51)	1.75%	-0.39%	1.38%	1.29%	2.27%	1.01%
Boston → Toulon	732.22(ton) (5.46)	192.97(hour) (0.59)	0.94%	-0.44%	0.79%	0.40%	1.85%	-0.17%
Busan → Sydney	1,097.93(ton) (7.31)	285.73(hour) (1.26)	0.87%	-0.37%	0.90%	0.89%	1.65%	0.47%
Acapolo → Chennai	2,582.60(ton) (17.33)	681.43(hour) (2.11)	1.07%	-0.38%	0.83%	0.47%	1.64%	0.21%
Montevideo → Toulon	1,385.82(ton) (8.32)	371.20(hour) (0.93)	0.84%	-0.37%	0.72%	0.11%	1.49%	-0.28%
Tampico → Lagos	1,438.93(ton) (8.29)	385.57(°) (0.97)	1.04%	-0.39%	0.64%	0.01%	1.35%	-0.35%
			Average 1.17%	Average -0.39%	Average 0.93%	Average 0.65%	Average 1.82%	Average 0.27%

연료소모량까지 줄었다. 또한 “ES - FIX”는 6개 실험 노선에 대하여 “A*-ES”보다 좋지 못한 성능을 보였는데 이는 ES에 의한 지리적 경로 탐색 효과가 출력 탐색에 의한 성능 개선보다 작음을 뜻한다. 하지만 “ES-ES”는 모든 경로에 대해서 가장 좋은 성능을 보이며 이는 ES에 의한 지리적 경로 탐색과 출력탐색이 결합되면 가장 큰 연료 소모량 감소를 기대할 수 있음을 의미한다.

다음으로 시기에 따른 성능 개선의 특성이 있는지를 살펴보기 위하여 “A*-ES,” “ES-FIX,” “ES-ES” 운항방법이 “A*-FIX” 대비 월별 평균 연료소모량을 얼마나 개선시키는 지를 비교 하였으며, 그림 7은 그 결과를 나타낸다. 그 그림에서 각 점은 해당 월의 7개 실험 노선에 대한 평균 결과이다. 그 그림에서 보이듯이, 성능개선의 폭이 월별로 변화가 있지만 표 3과 마찬가지로 “ES-ES”가 전체적으로 가장 좋은 성능을 보인다. 특히 6월에 가능 높은 성능 개선을 보였으며 7월에는 가장 낮은 성능개선을 보인다. 또한, 10월의 경우에는 “A*-ES”가 “ES-ES”보다 더 좋은 성능을 보여주는데, 이는 진화전략 기반 경로탐색이 효과적이지 못한 경우도 있음을 뜻한다.

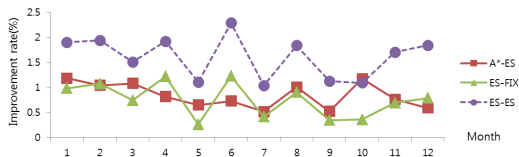


그림 7. 월별 연료소모량 평균
Fig. 7. Monthly average result of the routing methods

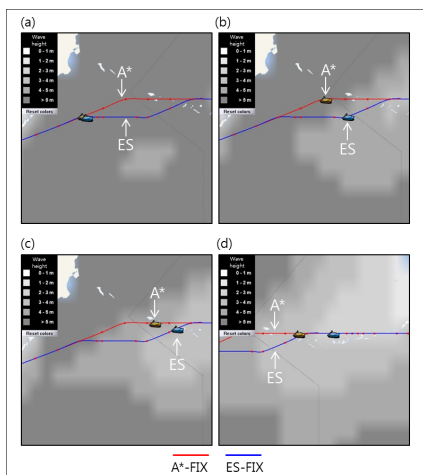


그림 8. ES-FIX 조합에 의한 악천후 회피
Fig. 8. Weather avoidance example by ES-FIX method

그림 8은 부산에서 출항하여 오클랜드로 입항하는 노선에 대해서 A* 알고리즘으로 얻은 지리적 최단 경로와 3.3절에서 설명한 진화전략 기반 경로탐색 알고리즘으로 얻은 경로가 지리적으로 얼마나 다른지를 보이기 위해 별도로 제작한 시각화 프로그램을 통해 그 일부를 보인 것이다. 이 때, 엔진의 출력은 두 경로 모두 같은 값으로 고정되어 있으며 해상 블록의 색상이 밝을수록 파고가 낮음을 의미한다. 그 그림에서 a, b, c, d는 시간 순으로 선박의 운항 위치와 파고를 나타낸다. 그 그림에서 진화전략 기반 경로탐색 알고리즘으로 얻은 경로가 A* 알고리즘으로 얻은 경로에 비해 높은 파고 지역을 효과적으로 피해가고 있음을 알 수 있다. 그림 8a에서 같은 시각에 비슷한 위치까지 항해를 하였지만 그림 8b에서 진화전략 기반 경로탐색 알고리즘으로 얻은 경로에서 파고가 더 낮은 곳을 향해 항해를 하는 것을 볼 수 있으며 그림 8c와 그림 8d에서 계속해서 파고가 더 낮은 곳으로 향해 앞서서 항해하는 것을 볼 수 있다. 이러한 악천후 회피 경로가 연료소모량을 감소시킬 수 있음을 설명한다.

실제로 이 예제에서 진화전략 기반 경로탐색 알고리즘에서 얻은 경로의 거리는 9,256.4192km로서 A* 알고리즘으로 얻은 경로의 길이 9,206.3295km보다 50.0897km 더 멀었지만, 도착시각은 전자의 경로가 후자의 경로보다 약 14시간 빨리 도착하였을 뿐만 아니라 연료소모량도 1,326.45톤에서 1,259.98톤으로 5.05% 감소시켰다. 이처럼 이 논문에서 제안한 진화전략 기반 경로탐색 알고리즘은 지리적 최적 경로를 탐색하는데 상당히 효과적임을 알 수 있다.

V. 결 론

기존의 진화전략 기법을 통해 항로를 세부구간별로 나누어 엔진출력을 최적화하는 방법은 지리적으로 최적화된 항로를 찾을 수 없다는 한계점을 가지고 있다. 이에 이 논문에서는 A* 알고리즘에 의해 찾은 항로를 초기해로 하여 진화전략을 통해 지리적으로 최적화된 항로를 찾을 수 있도록 제안하였다. 또한, 기존의 세부구간별 출력최적화 알고리즘과 자연스럽게 결합하여 시너지 효과를 낼 수 있도록 하였다. 실제로 7개 노선에서 12개월 운항 시간표(총 84가지)에 대해서 이 논문에서 제안된 시스템의 성능을 최단거리 기반 운항방법과 비교한 결과, 운항소요시간이 차이가 없으면서도 연료소모량을 평균 1.82%, 최대 2.49% 개선시킬 수 있음을 확인하였다.

이 연구 결과물을 향후 더욱 개선하기 위해서는 다

음과 같은 연구들이 필요할 것이다. 첫째, 이 논문에서는 지리적 경로최적화와 세부구간 출력최적화가 단계적으로 실행됨으로써 실험 노선에서 따라 두 최적화가 항상 시너지 효과를 발휘한다고 보장 할 수 없으므로 경로탐색과 출력탐색을 동시에 할 수 있는 적절한 탐색 알고리즘 개발이 필요할 것이다. 둘째, 진화전략 기반 경로탐색 알고리즘은 총 진화횟수와 변이시킬 노드의 개수에 따라 실행시간이 결정 되므로 실행시간을 줄이고 효율적인 탐색을 하기 위해서는 진화연산을 병렬화하는 방법을 찾아야 할 것이다. 셋째, 이 논문에서 제안하는 진화전략 기반 경로탐색 알고리즘은 변이할 노드의 개수와 기존 노드에서 변화되는 폭, 진화 횟수를 어떻게 설정하느냐에 따라서 개선율이 좌우되므로 충분한 모의실험과 통계에 기초한 적절한 값의 설정이 필요하다. 넷째, 항해사들에게는 디지털 시스템보다 예로부터 내려오는 항해 방법과 노하우에 대한 믿음이 커서, 이와 같은 경제운항시스템의 수용에 어려움이 있는 실정이다⁶⁾. 따라서 일방적인 시스템의 권고보다는 하이브리드 형태의 상호대화형 시스템의 개발을 하여 항해 보조 시스템으로서의 가치를 높이는 것이 필요하다. 다섯째, 경제운항시스템에서는 연료소모량 추정 계산과 기상 예보의 정확도에 따른 성능 의존도가 높다. 특히 기상 예보의 경우, 언제나 오차가 존재할 수 있으므로, 이를 동적 보정하는 기법을 도입하여 성능의 강건함을 유지할 수 있도록 해야 할 것이다.

References

[1] H. S. Jang and Y. K. Kwon, "An economic ship routing system by optimizing outputs of engine-power based on an evolutionary strategy," *J. KICS*, vol. 36, no. 4, pp. 412-421, Apr. 2011.

[2] S. Y. Joo, T. J. Cho, J. M. Cha, J. H. Yang, and Y. K. Kwon, "An economic ship routing system based on a minimal dynamic-cost path search algorithm," *J. KIPS*, vol. 1, no. 2, pp. 79-86, Jan. 2011.

[3] K. S. Choi, M. K. Park, J. H. Lee and, G. I. Park, "A study on the optimum navigation route safety assessment system using real time weather forecasting," *J. Korean Soc. Marine Environment and Safety*, vol. 13, no. 2, pp. 133-140, Jun. 2007.

[4] C. P. Padhy, D. Sen, and P. K Bhaskaran "Application of wave model for weather routing of ships in the North Indian Ocean," *Nat. Hazards*, vol. 44, no. 3, pp. 373-385, May 2008.

[5] J. K. Panigrahi, J. K. Tripathy, and P. A. Umesh, "Optimum tracking of ship routes in 3g-WAM simulated rough weather using IRS-P4 (MSMR) analysed wind fields," *J. Indian Soc. Remote Sensing*, vol. 36, no. 2, pp. 149-158, Jun. 2008.

[6] J. I. U. Aretxabala, S. I. Baniela and E. M. Rodriguez, "The optimum track using the classic method with the help of a personal computer," *J. Maritime Research*, vol. 4. no. 1, pp. 63-76, Jan. 2007.

방 세 환 (Se-hwan Bang)



2012년 8월 : 울산대학교 컴퓨터정보통신공학부 졸업
 2014년 3월~현재 : 울산대학교 전기전자컴퓨터공학과 석사과정
 <관심분야> 빅데이터, 대용량 데이터베이스, 인공지능, 최적화 알고리즘

권 영 근 (Yung-Keun Kwon)



1999년 2월 : 서울대학교 전산과학과 졸업(학사)
 2001년 2월 : 서울대학교 대학원 컴퓨터공학부(공학석사)
 2006년 2월 : 서울대학교 대학원 컴퓨터공학부(공학박사)
 2008년 3월~현재 : 울산대학교

전기공학부 IT융합전공 부교수
 <관심분야> 최적화 이론 및 실제, 복잡계산시스템, 시스템생물학, 소셜네트워크 등