

# 선박과 화물에 대한 중개 시스템의 설계 및 구현

서상구\* · 윤경현\*\*

## Design and Implementation of A Brokering System for Ships and Cargos

Sang-Koo Seo\* · Kyung-Hyun Yoon\*\*

### Abstract

It is one of the crucial components of electronic logistics systems to manage logistics information of cargos and transportation companies and to mediate appropriate brokerage between them. Due to the advance of e-Commerce technologies many kinds of logistics transactions can be handled by means of EDI or XML/EDI applications, but the brokering processing relies mostly on the traditional processes and the research in this field is still at the initial stage. In this paper we study a logistics brokering system for ships and cargos and describe the design and implementation of the system. We analyze the brokering constraints for logistics of cargos and ships and construct an optimization model for their brokering. We also suggest a brokering procedure and a simple heuristic algorithm with respect to the proposed matching criteria. The experimental result shows that the proposed greedy-based heuristic algorithm performs very well. In its response time the proposed algorithm executed within a couple of seconds independently of the number of cargos and the container capacities of ships. The output of the algorithm is very close to that of the optimal solution, showing higher than 95% of approximation. The proposed system is implemented for the Web environment using JSP and PL/SQL.

Keywords : e-Logistics, Brokering System, Brokering Optimization

논문접수일 : 2004년 10월 7일      논문게재확정일 : 2004년 11월 20일

※ 본 연구는 2003년도 광운대학교 교내학술연구비 지원에 의하여 연구되었음.

\* 광운대학교 경영정보학과 교수

\*\* (주)윙콘

## 1. 서 론

최근 전자상거래의 발전은 광대역 인터넷의 대중화를 통하여 보다 가속화되고 있으며 이에 따라 전자상거래 기반기술 및 새로운 응용기술에 대한 연구가 활발히 진행되고 있다[한국전자거래진흥원, 2003]. 특히 기업간 혹은 기업과 개인간의 거래방식, 즉 전자상거래 프레임워크에 대한 연구가 국가기관과 민간기업들에 의하여 폭넓게 진행되어 왔다. 이러한 연구 중에서 UN/CEFACT와 OASIS가 공동으로 개발한 ebXML, Microsoft를 중심으로 하는 BizTalk.org의 BizTalk, RosettaNet의 RosettaNet 등이 XML 기반의 전자상거래 프레임워크를 형성하고 있다(아래 <표 1> 참조)[Hofreiter *et al.*, 2002 ; Lu *et al.*, 2001 ; 이수철 외, 2000].

이러한 전자상거래 프레임워크는 기업간 업무 프로세스를 통합하고 연계하여 거래업무가 보다 신속하고 효과적으로 이루어지게 하며, 기업 환경은 전통적인 상거래에서 이용되던 물적 유통방식에서 발전하여 거래 기업의 다양한 요구사항에 대응하여 상품을 보다 효율적으로 운송할 수 있는 새로운 물류 시스템을 필요로 하게 될 것이다. 새로운 물류 시스템 환경에서는 물류관리를 위한 정보의 흐름을 효율화함으로써 물류비용을 줄이는 동시에 물류 체계를 전자상거래의 환경에 맞는 형

태로 변환함으로써 실물의 흐름을 효율화하는 것이 필요하다[황재각 외, 2002 ; 전종미 외, 2002]. 물류와 관련한 최근 연구는 주로 물류 서비스 사용자에게 다양한 서비스를 제공하기 위한 비즈니스 프로세스 부분이거나 위치추적 혹은 정보 통합 등에 관련된 주제인데, 물류 시스템에서 화물과 운송업체를 중개하는 물류정보 중개에 대한 연구는 그 중요성에도 불구하고 연구결과가 미진한 상태이고, 대부분의 산업분야에서 중개 업무는 여전히 기존의 전문 업체들의 인적 네트워크에 의존하여 이루어지고 있는 실정이다. 또한 인터넷 전자 상거래에서 중개 에이전트에 관한 다양한 기술과 응용분야가 소개되고 왔으나 물류 운송에서 화물과 운송업체의 연결 작업에 대한 연구는 충분히 이루어지지 않고 있다. 본 연구에서는 선박과 화물의 운송 중개를 대상으로 하여 화주와 운송업체의 물류정보를 통합적으로 관리하고 양측의 요구사항을 고려하여 거래 대상을 중개하기 위한 물류정보 중개 시스템에 대하여 연구하였다. 중개 시스템 구성을 설계하고 화주와 선사의 중개 기준을 정의하였으며 이를 바탕으로 다차원 0-1 Knapsack 문제 형태의 최적 중개 모형을 설정하고, Greedy 기반의 휴리스틱 중개 알고리즘을 제안하고 성능평가 실험 결과를 제시하였다. 제안된 중개 시스템의 주요 기능들을 JSP와 Oracle PL/SQL을 이용하여 웹 환경에서 구현하였다.

<표 1> 전자상거래 프레임워크

구분	EDI	RosettaNet	eCo	BizTalk	ebXML
제정기구별분류	공적표준	사실표준	사실표준	사실표준	사실표준
단체명	UN/EDIFACT, XML/EDI 그룹	RosettaNet	ComerceNet	Microsoft 주도	UN/CEFACT, OASIS
적용 네트워크	전체	전자	전체	전체	전체
산업분야	VAN	Internet	Internet	Internet	Internet
전송 메시지	EDI 문서(XML)	XML 문서	XML 문서	BizTalk 문서	XML 문서
생명주기	쇠퇴기	성장기 -> 안정기	소멸기	탄생기 -> 성장기	탄생기

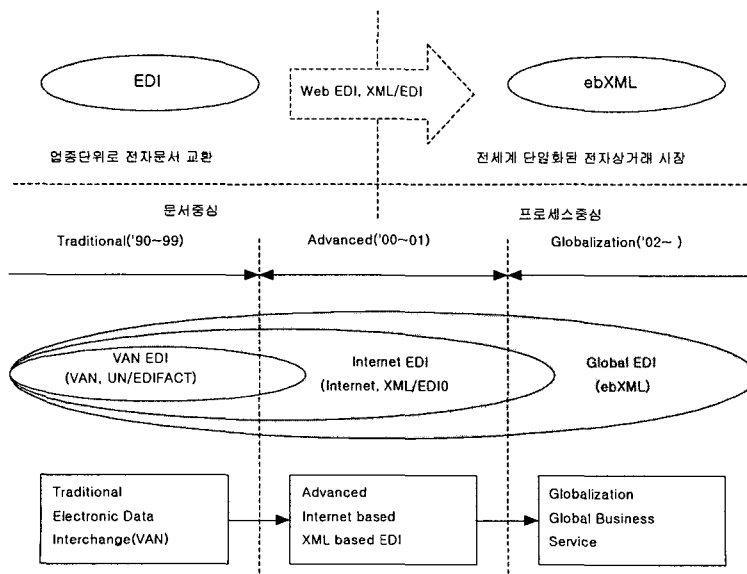
논문의 구성은 다음과 같다. 2장에서 관련 연구들을 살펴보고, 3장에서는 중개 시스템의 설계와 각 구성 모듈, 그리고 화물과 선박의 중개 기준과 최적 중개 모형, 그리고 중개 알고리즘 및 성능분석 결과를 설명한다. 4장에서는 물류정보 중개 시스템의 구현에 대해 기술하고, 끝으로 5장에서는 본 연구의 제약 사항과 향후 연구 과제를 언급하며 결론을 맺는다.

## 2. 관련연구

전자상거래는 1970년대 은행간 자금이체(EFT, Electronic Funds Transfer)가 출현한 이래로, 통제된 시스템 내에서 메시지 교환 방식을 준용하여 지정된 기업과 전자 문서를 교환하는 전자문서교환(EDI, Electronic Data Interchange)으로 발전하였다. 업종별 지정된 전자문서 표준에 대한 한계성, 새로운 거래 파트너의 검색에 대한 개방성의 제약, 그리고 고가의 구축비용으로 인한 이용 기업의 제한 등의 문제점을 개선하기 위하여 CALS, XML/EDI, ebXML 등으로 연구개발이 진행되고

있다[김동수, 2002 ; Lu *et al.*, 2001]. 현재 이러한 기술을 이용하여 기업은 시장조사 및 거래선 발굴에서부터 수출계약을 체결하고 통관에 이르는 많은 업무를 전자적인 방법으로 처리하고 있으며, 이로 인해 전자무역은 전체 무역에서 갈수록 그 비중이 증가하고 있는 추세이다. 전자 상거래에 대한 관심은 인터넷의 대중화를 통하여 더욱 가속화되고 있고, 국내외적으로 전자상거래 기반 기술 및 응용 기술에 대한 연구가 활발히 진행되고 있다. 이와 같은 전자상거래 발전과정은 아래 <그림 1>과 같이 도식화할 수 있다.

전자상거래의 발전과 확산에 의해 e-Logistics에 대한 관심과 연구가 많이 진행되고 있으나 e-Logistics의 도입을 통한 물류 혁신, 경쟁우위 등을 확보하기 위한 대응 및 물류정보화는 아직 미흡한 것으로 나타나고 있다. 현재 e-Logistics분야에서는 물류 기업간 정보 및 자료의 공유를 통한 다양한 서비스의 공급 망을 형성하기 위하여 ebXML 프레임워크를 활용하는 연구가 활발히 진행되어 왔다[오동근 외, 2003 ; 강태원 외, 2002, 김영일 외, 2003 ; Hofreiter *et al.*, 2002]. 하지만 이들



<그림 1> 전자상거래 발전과정(출처 : 한국 ebXML 포럼 <http://www.ebxmlkorea.org>)

연구는 물류 공급망 자체에 대한 연구로서, 본 연구의 주제인 화주와 운송 업체간의 운송 수요와 공급을 중개하는 시스템에 대한 주제는 다루고 있지 않다. 화물 운송과 관련하여, 컨테이너에 화물을 최적 적재하기 위한 연구가 시도된 바 있으나, 하나의 컨테이너를 대상으로 하였고, 화주와 운송 업체 간의 다양한 요구조건은 고려되지 않았다[권아름 외, 1995].

전자상거래에 대한 기대와 요구가 커지고 전체 상거래에서 차지하는 비중이 증가하면서 에이전트 기법을 이용한 매매 및 중개에 대한 연구가 활발하게 진행되었다. 전자상거래 에이전트는 사용자를 대신하여 상품이나 거래 상대를 검색하고 구매결정 등을 지원하는 것을 목적으로, 주로 제품 중개, 거래상 중개, 협상, 구매, 배송, 서비스 및 만족도 평가 등의 분야에 많은 연구가 진행되어왔다. 초기의 전자상거래 에이전트는 대부분 가격이라는 하나의 변수만을 고려하였으나, 최근 다수의 구매자와 다수의 판매자의 다양한 거래 요구사항을 고려한 다중 에이전트의 적용에 관한 연구가 진행되고 있다[최중민, 2002 ; Flores-Mendez, 2001 ; Ma, 1999]. [Freuder & Wallace, 1998]과 [Jung & Go, 2000]에서는 지능형 중개 에이전트를 제약만족문제(CSP, Constraint Satisfaction Problem) 모형으로 나타내고 최적해를 구하는 시도를 하였다. [Keskinocak *et al.*, 2001]은 구매자와 판매자를 다차원 속성 공간의 원소로 표현하고 원소간의 근접도 또는 거리를 중개 척도로 계산하여 거래를 중개하는 연구를 시도하였다. [Segev & Beam, 1999]는 여러 공급자와 여러 소비자 간의 가격중심의 온라인 상품 중개 전략을 연구하였고, [Lau & Goh, 2002]에서는 고객의 구매제안에 대하여 상품 공급자들의 에이전트가 입찰을 수행하고 e-Marketplace라는 중개 서버를 통하여 거래를 연결시키는 4자 물류의 중개 모형을 연구하였다. 이들 이전 연구들은 주로 상품의 구매와 공급에 대한 일반

적 프레임워크를 중점적으로 다루었다. 이에 비하여 본 연구에서는 화물 운송을 위한 선박과 화물의 요구사항을 구체적으로 고려할 뿐 아니라, 하나의 운송 스케줄에 화물의 요구사항을 만족하는 여러 화주가 중개되어야 하는 모형으로서, 하나의 고객에 하나의 상품을 최적 중개하는 일반적인 전자상거래 모형과는 차이가 있다. 또한 제안한 중개 모형을 웹 응용 시스템으로 구축한 사례를 소개한다는 점에서도 논문의 의미가 있다고 본다.

### 3. 물류정보 중개시스템

#### 3.1 시스템 구성

현재 전자무역은 거래기업을 찾고 수출계약과 대금 결제 과정까지의 업무와 화물이 선적되어 해당 지역에 도착한 후 통관업무가 진행되는 과정은 EDI 혹은 XML/EDI를 사용하여 상당부분 전자적으로 진행되고 있지만, 운송업체를 선택하는 과정은 아직까지도 전통적인 방법에 의해 진행되고 있는 실정이다. 본 연구에서는 화주와 운송업체의 거래 조건을 만족하는 대상 기업을 중개하기 위한 물류정보 중개시스템의 요구사항을 다음과 같이 정의하였다.

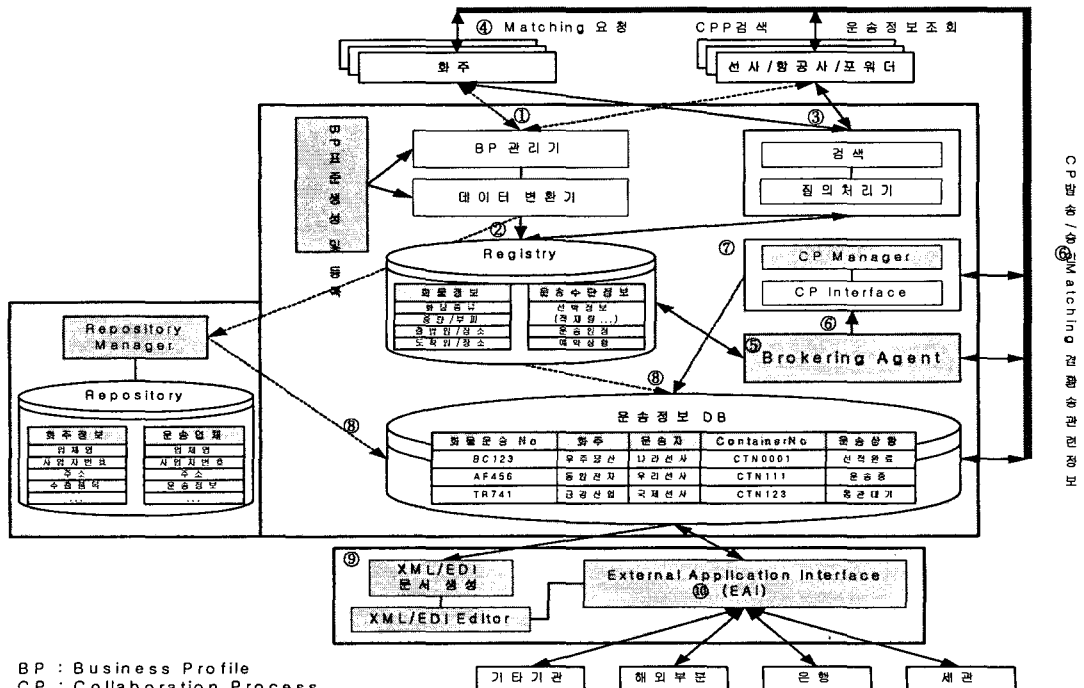
- 거래 기업의 일반정보, 화주의 화물 수배송 정보, 선사의 선박 운항정보와 선적 정보를 통합적으로 관리하고 검색할 수 있어야 한다.
- 화주의 수배송 일정에 적절하게 선박의 운송 스케줄이 할당되어야 한다.
- 선박의 컨테이너 선적을 가능한 최적화 할 수 있는 선적계획을 제공해야 한다.
- 선박의 운송 스케줄과 예약상황을 실시간으로 관리하고 확인할 수 있어야 한다.

이를 바탕으로, 본 논문에서 제안하는 물류정보 중개시스템은 크게 화주(예, 수출업체)와 운송업체가 정해진 규칙에 의해 정보를 등록/저장하는 부분과 등록된 정보를 이용하여 중개 에이전트가 적

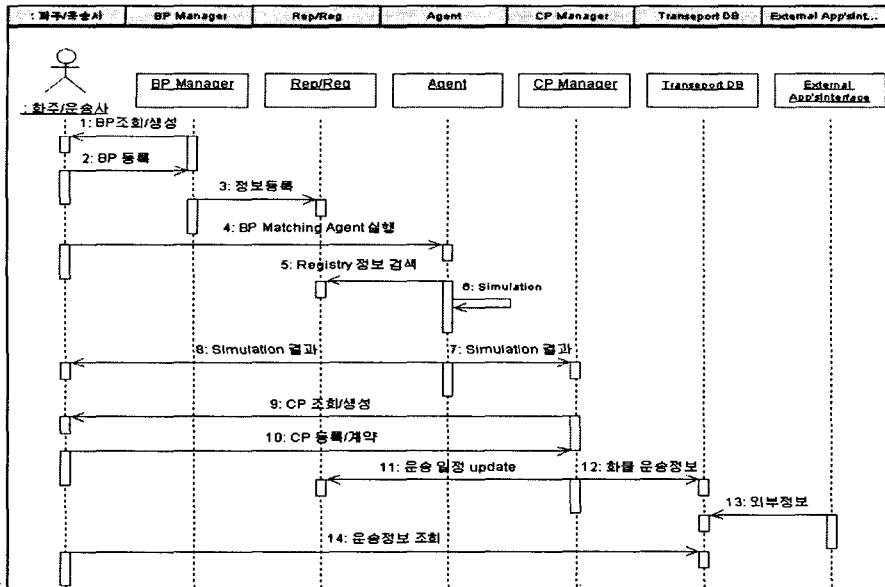
절한 거래 상대자를 검색해주는 부분으로 구성하였다. <그림 2>은 본 연구에서 제안하는 중개 시스템의 구성도로서, 메인 중개시스템, 화주/선주 정보를 공개하기 위한 리퍼지토리, 그리고 외부 인터페이스로 구분된 기능별 구성(질은 바탕색으로 표시된)과 세부 요소들을 보여주고 있다. 제안된 구성은 인터넷에서 운영되는 단일 웹 응용 시스템을 전제로 한 것이나, 화주와 선사의 등록, 검색, 중개 부분들의 상호 작용은 ebXML의 거래 모형과 유사한 면이 있다. 향후 물류운송 중개를 위한 ebXML 비즈니스 응용의 하나로 발전시키기에 용이한 형태라고 할 수 있다.

<그림 2>에서 점선은 데이터의 흐름을 나타내고 실선은 업무의 흐름을 나타낸다. 화주와 운송업체는 물류정보 중개시스템의 BP(Business Profile) 관리기를 통해 표준 입력정보를 확인하고 각각의 해당 정보를 등록한다(①). 등록된 정보는 데이터 변환기를 통해 데이터의 종류에 따라 화주

또는 운송업체의 고유한 정보는 리퍼지토리에, 화물의 운송 일정 및 선박의 개별 운항 일정 등은 레지스트리에 저장되며(②), 사용자는 검색 모듈을 통해 거래 대상을 검색할 수 있다(③). 사용자의 매칭 알고리즘 실행 요청(④) 또는 사전에 예정된 스케줄에 따라 Brokering Agent는 레지스트리에 저장된 정보를 기준으로 매칭 알고리즘을 실행하여 거래 가능 대상을 선별하며(⑤), 결과는 사용자와 협상관리 모듈(CP 매니저)에 통보된다(⑥). 화주와 운송업체는 협상관리 모듈을 통해 거래 대상과 계약 업무를 진행하며(⑦), 계약된 정보는 리퍼지토리에 입력되어 있는 해당 정보와 함께 운송정보 DB에 저장 된다(⑧). 이후 XML/EDI Editor를 통하여 무역 관련 서류를 생성하며(⑨), 은행이나 세관 및 기타 외부 기관과의 연계는 EAI를 통해 이루어진다(⑩). <그림 3>은 이상의 처리 과정을 순차 다이어그램(Sequence Diagram)으로 표현한 것이다.



<그림 2> 물류정보 중개시스템



〈그림 3〉 중개 시스템 업무 흐름도

시스템 각 구성 요소의 세부 기능/명세는 다음과 같다.

- 비즈니스 프로파일 및 정보 관리/등록 모듈  
물류정보 중개시스템에서 필요한 정보를 정의하고, 그 정보에 의해 각 업체가 비즈니스 프로파일을 등록 및 수정하게 하는 업무를 담당한다. 본 모듈을 통하여 중개 에이전트 및 문서생성에 필요한 정보를 표준화할 수 있으며, 무역 관련 서류의 서식이 변경되었을 경우 이 모듈을 통하여 새로운 표준을 적용시킬 수 있다. 등록된 정보는 데이터 변환기에 의해 레지스트리와 리파지토리로 전송되어 저장된다.
- 레지스트리/리파지토리  
레지스트리는 화주의 경우 화물종류, 중량 및 부피, 화물 출발일/도착일, 화물 출발장소/도착장소 등 화물과 관련된 상세 정보가 저장된다. 운송업체의 경우 적재 가능 총중량, 운송수단 출발장소, 경유지 및 도착 예정일 등 운송 스케줄에 관한 전반적인 정보 저장과 지속적인 변경을 담당한다. 레지스트리에 저장된 정보는 중개 에이전트 모듈이 대상업체를 선정하는데 사용

되며, 예약 상황이 실시간으로 갱신된다. 리파지토리는 영속적인 데이터를 저장하는데, 주로 기업에 관한 정보로 수출업체 및 운송업체의 상호, 사업자등록번호 및 운송수단에 관한 정보 등이다.

- 중개 에이전트 모듈  
중개 에이전트는 레지스트리에 등록된 정보를 검색하고 조건에 맞는 대상 업체를 선정하기 위하여 매칭 알고리즘을 실행한다. 또한 운송수단(선박, 항공)의 운항 스케줄이 등록된 시점이나, 화물 선적 후 일정 시점마다 레지스트리 정보를 이용하여 선적 계획을 실행한다, 실행 시 각 시점에 대한 최대 화물 선적율을 계산하여 화주와 운송업체를 중개한다.
- 협상 모듈  
대상 기업간 계약체결 업무를 지원하기 위한 모듈로서, 계약체결에 대한 업무 및 정보를 정의하는 Collaboration Process Manager와 각 업체간 실제 계약 업무를 진행하는 인터페이스 부분으로 구성되어 있고, 등록된 화물 정보와 운송 예약현황을 갱신한다.

- XML/EDI 문서 생성기 및 EAI 모듈  
선적이 결정된 화물은 선적, 통관, 기타 업무에 필요한 서류가 필요하다. XML/EDI문서 생성기는 Bill of Loading(B/L), Commercial Invoice, Packing List(P/L) 등의 통관 관련서류를 생성하기 위한 것이다. EAI 모듈은 외부 시스템과 연계하기 위한 것으로, 생성된 XML/EDI문서를 해당 관공서나 은행, 해외부분 등으로 전송한다. 또한 보험 등 추가적인 XML/EDI문서가 필요할 경우 XML/EDI Editor를 이용하여 해당 문서를 작성할 수 있다.

### 3.2 중개 에이전트

중개 에이전트는 다수의 화주에 대하여 다수의 운송업체가 보유하고 있는 선적공간이라는 상품을 주어진 제약조건에서 매핑하며, 화주와 운송업체의 요구사항을 가능한 만족시키는 거래 대상 집합을 구하는 것을 목표로 한다. 먼저 중개 결과를 고려한 전체적인 절차에 대하여 기술하고 그 다음으로 세부적인 화주와 운송업체의 중개 제약사항, 중개 기준, 그리고 이에 따른 최적화 모형을 제안한다.

#### 3.2.1 중개 절차

하나의 운송 스케줄에 포함될 수 있는 화물의 수가 제한되기 때문에 한 화물이 어떤 운송 스케줄에 포함되는지 여부는 다른 화주 화물이 그 운송 스케줄에 선적 가능한지에 영향을 줄 수 있다. 이러한 특성 때문에 중개 시스템의 결과물에 대한 정의가 먼저 필요하다. 중개 시스템의 결과물은 화물과 선박(또는 운항 스케줄)의 매핑 형태에 따라 다음과 같이 두 가지 방법을 생각할 수 있다.

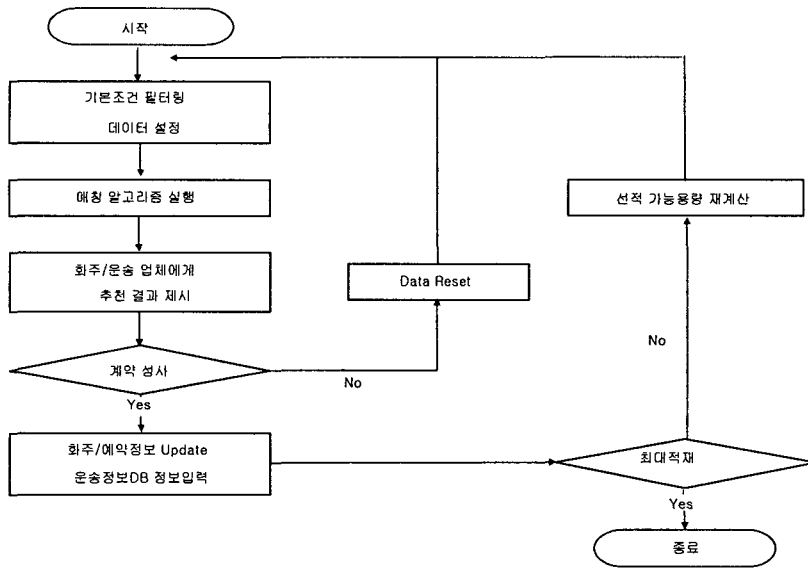
- 다대일(M : 1) 매핑 - 하나의 선박에 여러 화물이 배정되며 한 화물은 전체 운송 스케줄에 중복적으로 배정되지 않는 중개 결과이다. 이와

같은 결과는 전체 대상 화주와 선박들의 기대이익이 최대가 되도록 각 화물을 최대한 하나의 선박에 매핑한다는 것을 의미한다. 이를 위해서 각 화물이 임의의 운송 스케줄에 배정되었을 때 화물과 선박의 이득을 정량화하고 중개의 공평성을 유지할 수 있는 기준이 필요하다. 또한, 생성된 중개 결과에 따라 화주와 선주들이 실제 계약을 맺는다고 할 경우에 유효한 방안이다.

- 다대다(M : N) 매핑 - 다대다 매핑은 화물이 여러 운송 스케줄에 독립적으로 중복 고려되도록 중개 결과를 생성하는 것을 말한다. 달리 말하면 하나의 선박에 대하여 그 선박의 이득과 선적되는 화물의 이득의 합이 최대가 되는 화물을 선택하되, 각 선박마다 독립적으로 선정하도록 한다. 이 경우, 각 선주는 자신의 선박의 특성과 운송 스케줄에 가장 적합한 화물들을 알 수 있고, 또한 화주의 경우에도, 자신의 화물이 운송되기에 적합한 선박들을 알 수가 있다. 실제 계약은 이렇게 생성된 중개 가능 목록을 참조하여 별도의 협상을 통하여 이루어지게 될 것이다.

다대일의 매핑은 주어진 조건하에 가장 최적의 중개 매칭 결과를 제시하겠지만 현실적으로 실제 운송계약의 성사는 중개 시스템에서 고려되기 어려운 정성적 기준 (예, 기업관계, 협상 등)이 함께 고려되어 결정되는 것이 일반적이다. 따라서 본 연구에서의 중개 결과는 우선 협상 대상을 파악하는데 도움을 주기 위한 다대다 매핑을 생성하는 것으로 하고, 중개 절차와 중개 기준도 이에 따라 정의한다.

다음 <그림 4>는 하나의 선박을 대상으로 했을 때, 본 연구에서의 중개 처리 흐름을 보여주고 있다. 중개 알고리즘은 사용자의 실행 요구나 자동 스케줄링에 의해 실행된다. 주어진 거래 대상 선박과 화주의 목록에 대하여 각 선박별로 적재 대상



〈그림 4〉 중개 에이전트 처리 흐름

화물들을 선정한다. 이 때, 주어진 제약사항에 만족하지 않는 대상은 사전에 고려대상에서 배제하는 방법으로 탐색 공간을 줄여나간다. 알고리즘 실행 결과는 화주와 선주에게 제시되는 선적 추천목록의 역할을 한다. 계약 성사 또는 선적 부적합 등의 실제 당사자간의 결정이 내려지면, 중개 에이전트는 일정 시간이나 조건에 의해 반복적으로 알고리즘을 실행하는 방식으로 진행된다. 거래 대상 업체들은 본 시스템의 Collaboration 모듈을 이용하여 계약 업무를 진행하게 된다. 에이전트는 화주와 운송업체간 계약이 체결되면 예약관련 정보를 갱신하고 운송정보 DB에 해당 정보를 입력한 후, 운송 수단의 최대 적재용량과 현재 계약 완료된 정보를 비교하여 최대 적재가 이루어지지 않았을 경우 다시 정보를 수집하여 알고리즘을 수행한다. 이때 등록된 화주 중 이미 계약이 성사된 화주와 운송수단의 최대 적재용량 중 선적이 결정된 부분은 매칭 알고리즘의 대상에서 제외된다.

### 3.2.2 매칭 기준

중개 조건의 설정과 관련하여 몇 가지 가정을

둔다. 먼저, 화물과 선박의 선적 및 출항 장소는 동일하며 하나의 최종 목적지를 갖는다고 가정한다. 이 가정은 출발지가 여러 곳일 경우에는 출발지가 동일한 화물과 선박 집단들에 대하여 각각 중개하면 되므로 쉽게 확장할 수 있다. 또, 화물은 단일 크기(예, 20ft 또는 40ft)의 컨테이너 단위로 운송되고, 하나의 화물은 하나 이상의 컨테이너로 구성되며 각 컨테이너에는 한 화주의 화물이 적재(FCL: Full Container cargo Load)되는 것으로 가정한다. 이 가정은 해상 운송에서 일반적일 뿐 아니라 화물이 차지하는 컨테이너 수와 선박의 최대 허용 컨테이너 수와의 비교를 용이하게 하기 위한 것이다. 한 컨테이너에 여러 화주의 다양한 크기의 화물이 적재되는 LCL(Less than Container cargo Load)을 고려하려면, 한 컨테이너 분량 미만의 화물들이 선박에 매칭되어 소요되는 컨테이너의 수를 계산해야 하는데, 이 문제는 Bin Packing 문제의 한 형태로서 NP-Hard에 해당된다[Horowitz & Sahni 1978]. 이 경우, First Fit, Best Fit 등의 잘 알려진 휴리스틱 알고리즘을 적용하여 최소값에 근접한 컨테이너 수를 계산해 낼 수 있고, 동일



한 컨테이너에 화물을 적재한 여러 화주들에 대하여 대표화주를 선정함으로써 FCL 운송으로 간주할 수 있을 것이다. LCL 운송을 고려한 보다 상세한 중개 방안은 향후 연구로 미루기로 한다. 끝으로, 출발 및 도착 일자와 목적지가 동일한 한 화주의 화물은 하나의 선박 운항 스케줄에 포함되는 것으로 가정한다. 이는 운송관리 측면에서나 대부분의 일반적인 경우에 적용될 수 있는 가정이다.

선박과 화물의 적재와 운송과 관련하여 화물 규격, 운임지불 조건, 보험 및 통관 등 여러 정보가 있는데, 중개 매칭에 직접적으로 영향을 주는 요인으로는 날짜, 컨테이너 수, 그리고 중량이 가장 중요하다. 화물과 선박 각각에 대하여 다음과 같은 속성들을 매칭 변수로 고려하였다.

- 화물 : 화물의 선적 가능일, 목적지, 희망 도착일, 컨테이너 수 및 컨테이너 당 중량
- 선박 : 출발일, 목적지, 도착 예정일, 최대 허용 컨테이너 수, 최대 선적 중량

여기에 컨테이너의 종류(예, 건식 또는 습식) 등 직접적으로 선적 가능 여부를 제한하는 속성들은 중개 대상에서 쉽게 여과/분류될 수 있으므로 조건에 포함하지 않았다. 선박의 경우 컨테이너의 크기(20ft 또는 40ft) 별 최대 허용 중량이 설정되어 있고 최대 허용 컨테이너 수가 제한되며, 또 이와 별도로 총 허용중량이 지정되어 있다. 화물의 특성에 따라서 최대 허용 컨테이너 수 미만으로도 최대 허용 중량에 이를 수 있다. 또한 가격도 일반적으로 많이 고려되는 변수이기는 하지만 선박운송과 같이 큰 거래에서는 가격 규모가 크기 때문에 중개 대상자들 간의 협상 항목으로 두는 것이 적합할 것으로 판단하여 또한 포함하지 않았다. 이상의 가정과 매칭 변수들을 바탕으로 물류정보 중개 에이전트에 대한 중개 기준을 다음과 같이 정의하였다.

[기준 1] 화주가 요구하는 수배송 스케줄을 만족되는 선박과 중개되어야 한다. 즉, 화물이 선

적 가능한 시점(즉, 출고 시점) 이후에 출항하고 화물의 희망 도착 시점 이전에 목적지에 도착할 수 있는 선박 스케줄이어야 한다.

[기준 2] 화주의 경우, 출고 후 선적을 위한 대기시간이 가능한 짧은 운송 스케줄과 목적지에 도착 후 입고를 위한 대기시간이 가능한 짧은 운송 스케줄의 선박을 선호한다. 그러므로 가능한 화물의 배송 스케줄과 일치하는 선적 스케줄이 생성되어야 한다.

[기준 3] 선주의 입장에서는 선박의 총 적재중량과 최대 컨테이너 적재 한도 내에서 최대한 많은 수의 컨테이너를 적재하도록 하되, 동일한 컨테이너 개수의 화물일 경우 가능한 중량이 가벼운 화물이 우선 적재되도록 매칭이 이루어져야 한다.

위에서 [기준 1]은 각 선박에 대하여 적재 가능한 화물을 선택할 때 이용되는 제약조건으로서 선박의 출항 스케줄 이후에 출고되거나 선박의 목적지 예정 도착일 전에 배송되기를 희망하는 화물들을 해당 선박의 중개 대상에서 제외하는데 이용된다. [기준 2]와 [기준 3]은 각각 화주와 선주의 선호도를 표현한 것이다. 중개 알고리즘은 각 선박 별로 이들 선호도가 최대한 만족되는 선적 대상 화물을 선택해야 하는데, 화주와 선주의 선호도는 서로 상충될 수가 있다. 즉, 화주 A는 선박 S의 스케줄을 우선 선호하더라도 선박 S는 화주 A보다 화주 B의 화물적재를 선호할 수가 있다. 본 연구에서는, 화주와 선주의 이익을 동시에 고려하기 위하여 날짜의 근접성과 중량의 이득치에 대하여 각각 가중치를 부여한 다음에 두 값의 곱을 그 선박에 대한 특정 화물의 선적 적합도로 정의한다. 일반적으로 결혼 중개 에이전트에서와 같이 양자간의 선호를 통합적으로 고려하기 위해서는 합산보다 곱이 적절한 것으로 알려져 있다. 즉, 상대에 대한 선호가 각각 3과 7인 경우보다는 5와 5인 경우가 매치될

가능성이 높은 것으로 해석된다. 이상을 바탕으로 화물과 선박간의 선적 적합도를 다음과 같이 정의하였다.

• 화물  $i$ 와 선박  $j$ 의 선적 적합도 :  $P_{i,j}$

$$(w_1 \times (M_1 - DepartGap_{i,j}) + w_2 \times (M_2 - ArriveGap_{i,j})) \times (w_3 \times \frac{1}{ContWeight_i}) \quad (1)$$

여기서, 변수의 의미는 다음과 같다.

$w_1$  : 화물의 출발전 대기 가중치,

$M_1$  : 화물의 출발 대기 최대 한도값

$w_2$  : 도착후 대기 가중치,

$M_2$  : 화물의 도착 대기 최대 한도값

$DepartGap_{i,j}$  : 화물  $i$ 가 출고 후 선박  $j$ 에 선적되어 출발하기까지의 대기시간

$ArriveGap_{i,j}$  : 화물  $i$ 가 선박  $j$ 를 이용하여 목적지 도착 후 원래 도착예정일까지 대기시간

$w_3$  : 컨테이너 화물의 중량에 대한 가중치

$ContWeight_i$  : 화물  $i$ 의 컨테이너당 평균 중량

식 (1)의 선적 적합도  $P_{i,j}$ 는 화물과 선박의 일정 스케줄이 일치할수록, 컨테이너 단위별 중량이 가벼운 화물일수록 큰 값을 갖게 된다.  $w_1$ ,  $w_2$ ,  $w_3$ ,  $M_1$ ,  $M_2$  값은 현실성을 고려하여 적절한 결정하여야 할 것이다. 본 논문은 중개 시스템의 전체 프레임워크를 제시하고 그 틀에서의 구현 사례를 소개하는데 주요 목적이 있기 때문에, 선적 적합도가 현실 세계에서 어느 정도 타당한지와 적절한 가중치 값의 범위 등에 대해서는 향후 실증적 연구에서 다루어져야 할 것으로 본다.

이와 같이 계산된 각 화물과 선박에 대한 선적 적합도 값을 이용하여 중개 에이전트 모듈은 각 선박에 적재될 화물을 매핑한다. 각각의 선박에 대한 매핑 문제는 선박에 대하여 선적 적합도 합계가 최대가 되도록 하되, 컨테이너 총 적재 수량과 총

적재 중량에 대한 제약 조건이 만족되어야 하므로 다차원 Knapsack 문제[Xhafa, 2004; Martello & Toth, 1990]의 한 형태이며, 이를 적용하여 아래와 같이 모형화할 수 있다. 각 선박에 대하여,

$$\text{Maximize } \sum_{i=1}^n P_i x_i \quad (2)$$

$$\text{Subject to } \sum_{i=1}^n C_{k,i} x_i \leq M_k,$$

$$\text{for each } k(1 \leq k \leq 2) \quad (3)$$

식 (2)에서,  $P_i$ 는 선박에 대한 화물  $i$  ( $1 \leq i \leq n$ )의 적합도를 나타내고  $x_j$ 는 화물  $i$ 가 선택될 경우 1, 그렇지 않을 경우 0 값을 갖는 이진 결정변수이다. 식 (3)의 제약조건에서 결정변수의 계수인  $C$ 는 2행  $n$ 열의 행렬로서 첫째 행은 각 화물의 컨테이너 수를 갖고, 두 번째 행은 각 화물의 중량을 나타낸다. 예를 들어서  $C_{1,3}$ 은 세 번째 화물의 컨테이너 수를,  $C_{2,3}$ 은 세 번째 화물의 총 중량을 나타낸다.  $M$ 은 2행1열의 행렬로서 첫 행은 선박의 최대 적재가능 컨테이너수를, 둘째 행은 최대 적재 가능 중량을 나타낸다. 따라서 최적화 모형은 하나의 목표함수와 컨테이너 수와 총 중량 제한에 대한 두개의 제약조건으로 구성된다.

[문제 예] 선박과 화물의 정보 및 운송 조건에 대한 적합도가 다음과 같다고 하자.

선박의 최대 컨테이너 수10개

선박의 최대 적재 중량 : 1000t

화주1 : 컨테이너 4개, 중량 합계 500t, 적합도 7

화주2 : 컨테이너 7개, 중량 합계 400t, 적합도 9

화주3 : 컨테이너 5개, 중량 합계 300t, 적합도 5

이에 대한 목표함수와 제약조건은 아래와 같이 표현된다.

$$\text{Maximize } 7x_1 + 9x_2 + 5x_3$$

$$\text{Subject to } 4x_1 + 7x_2 + 5x_3 \leq 10 \quad (4)$$

$$500x_1 + 400x_2 + 300x_3 \leq 1000$$

여기서  $x_1, x_2, x_3$ 는 각각 화주1, 화주2, 화주3에 해당되는 이진 결정변수이다. 최적해는  $x_1 = x_3 = 1, x_2 = 0$ 로서 화주1과 화주3의 화물을 선택하여 적합도의 합은 12가 된다. □

본 논문에서 제시한 다대다 형식의 최적화 모형을 일반 상품 또는 서비스의 중개와 연관시켜볼 수 있다. 상품의 제공자는 하나의 물품만을 보유하고 이를 최종적으로 하나의 수요자에게만 공급 가능한 경우와 동일한 여러 물품을 보유하고 동시에 여러 수요자에게 공급할 수 있는 경우로 나누어 볼 수 있다. 이 때 상품 제공자 입장에서는 가격 혹은 수요자에 대한 선호도(예를 들어, 백화점이나 할인점 지명도 등)가 주요 기준이 될 수 있다. 본 연구에서 선주에 해당되는 상품의 수요자의 경우에도 하나 또는 여러 상품을 구매하는 것을 생각할 수 있고 물품가격, 평판 등의 선호도가 있을 수 있다. 이때 총 구매상품의 개수, 총 구매가격의 제한 등의 제약조건은 선주의 컨테이너 수의 제약 조건에 대응될 수 있고, 하나의 상품을 구매하는 개인 구매자의 경우 컨테이너 개수가 하나인 선주들로 간주할 수 있을 것이다. 상품 제공자의 상품의 개수가 하나 또는 제한되어 있는 경우에 상품의 중개 결과는 본 연구에서와 같이 다대다 중개 매핑을 기반으로 상품 제공자가 수요자를 결정하기 위한 대상 목록의 역할을 할 것이다. 이상과 같이 본 연구의 최적 중개 모형을 상품 중개 등의 분야에 응용하기 위해서는 상품 거래의 비즈니스 모델과 최적화 및 제약조건에 대한 세부적인 정의가 필요하며, 흥미로운 연구 주제가 될 것으로 생각한다.

### 3.3 매칭 알고리즘

#### 3.3.1 휴리스틱 알고리즘

제약조건이 하나인 단순 0-1 Knapsack 문제는 NP-Hard에 속하며 다차원 0-1 Knapsack의 특별

한 경우에 해당된다. 따라서 다차원 0-1 Knapsack 문제 또한 NP-Hard에 속하게 된다. NP-Hard에 속하는 문제는 문제 인스턴스의 크기(예, 입력 데이터의 수)가 커질수록 최적해 탐색시간이 기하급수적으로 증가하므로, 대부분의 응용에서 최적해보다는 최적해에 가능한 가까운 근접해를 이용하는 것이 현실적이다. 본 연구에서 매칭 알고리즘은 Greedy 탐색 방식의 간단한 휴리스틱을 적용하여 근접해를 구한다. Greedy의 선택 기준은 화물의 컨테이너 수에 대한 적합도의 비율 값에 대한 내림차순으로 적용한다. 이는 단순히 적합도 값만 고려하는 것 보다 컨테이너 당 적합도를 고려하는 것이 합리적이기 때문이다. 화주의 컨테이너 평균 중량에 대한 적합도 비율을 고려해볼 수 있으나, 컨테이너의 평균 중량은 이미 적합도에 반영되었을 뿐 아니라 선적에 있어서 일반적으로 컨테이너 수가 중량보다 적재여부에 결정적 역할을 하기 때문이다. 이와 같은 매칭 기준과 적재 선정 방법을 토대로, 매칭 프로그램은 각 선박에 대하여 선적 대상 화주들을 선택하며, 이를 pseudo 코드로 표현하면 다음과 같다.

프로그램은 각 선박에 대하여 화물의 적합도를 계산하고 Greedy 고려 기준으로 정렬한 다음 정렬된 순서에 따라 선박의 컨테이너 수와 중량 용량이 허용하는 범위 내에서 최대한 화물을 선적대상에 포함시킨다. 위 pseudo 프로그램에서(적합도 / 화물의컨테이너수)의 순서로 정렬하는 것을  $n \log(n)$ 로 보면, 알고리즘의 복잡도는  $O(m \times (O(n) + n \log(n) + O(n))) \approx O(mn \log(n))$  정도로 추정할 수 있다.

#### 3.3.2 알고리즘의 성능분석

제안된 휴리스틱의 성능을 알아보기 위하여 실험을 수행하였다. 실험 내용은 최적해 탐색과 비교하여 휴리스틱 알고리즘의 실행시간과 최적해와의 근접도를 비교 분석하는 것이다. 실험 환경은

### Procedure MatchingProc

Input : 출발지와 목적지, 컨테이너 타입이 동일한 화주의 화물과 선박들에 대하여

- 각 화물에 대한 출발 예정일, 도착 예정일, 컨테이너 수, 총 컨테이너 증량
- 각 선박의 출발일, 도착일, 최대 적재 컨테이너 수, 최대 적재 증량
- 화물 수  $n$ , 선박수  $m$

Output :

- $n \times m$  크기의 매칭 결과 테이블  $S[][]$   
(화물 $[i]$ 가 선박 $[j]$ 에 배정되면  $S[i][j] = 1$ , 그렇지 않으면  $S[i][j] = 0$ )

BEGIN

```

for (int j = 1 ; j ≤ m ; j++) {          /* 각 선박에 대하여 */
  선박[j].적재수 = 선박[j].적재량 = 0;
  for (int i = 1 ; i ≤ n ; i++) {
    if (화물[i]의 출발일이 선박[j]의 출발일 보다 나중이거나
        선박 도착일이 화물의 도착요구일보다 나중인 경우) continue ;
    else 선적 적합도  $P[i][j]$  를 계산 ;
  }
  선박 j에 대하여 (적합도/화물의컨테이너수) 값의 내림차순으로 화주 정렬 ;
  for (int i = 1 ; i ≤ n ; i++) { /* 각 화물에 대하여 Greedy 기준 선택 */
    if (선박[j].적재수 + 화물[i].컨테이너수 ≤ 선박[j].최대적재수 &&
        선박[j].적재량 + 화물[i].총증량 ≤ 선박[j].최대적재증량) {
      선박[j].적재수 += 화물[i].컨테이너수;
      선박[j].적재량 += 화물[i].평균증량 * 화물[i].컨테이너수 ;
       $S[i][j] = 1$  ;
    } else
       $S[i][j] = 0$  ;
  } /* end-for */
} /* end-for */
END

```

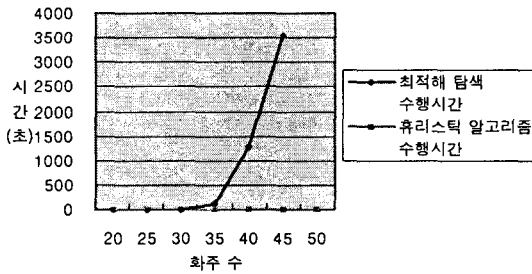
SunOS v5.6을 탑재한 Sun Enterprise 450 서버이며, 제안된 알고리즘은 GNU C/C++ Version 2.8.1로 구현하였고, 최적해 탐색은 선형/비선형 0-1 최적화 알고리즘으로 잘 알려져 있는 Davis-Putnam 기반의 OPBDP 최적화 프로그램을 이용하였다 [Barth, 1995].

실험은 하나의 선박을 대상으로, 화주(화물)의 수를 변화시키면서 알고리즘의 수행 성능을 알아보는 것과 주어진 화물에 대하여 선박의 용량(즉, 컨테이너 최대 적재 수 및 최대 적재 용량)을 변화시키면서 알고리즘의 수행 성능을 알아보는 두 부분으로 구성된다. 최적해 탐색에서 OPBDP의 의존성을 최소화하기 위하여 OPBDP의 자체 휴리스

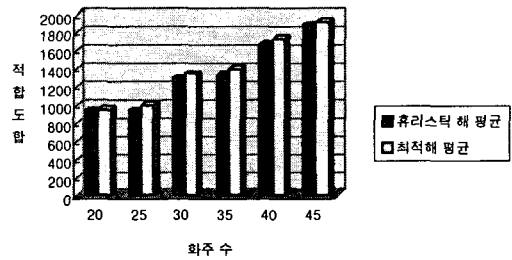
틱은 이용하지 않고 (-h0 옵션 적용)[Barth, 1995], Unix의 /bin/time 명령을 이용하여 제안된 휴리스틱과 최적해 탐색 프로그램의 수행시간(real time 값)을 측정하였고, 두 프로그램에서 구해진 해, 즉, 선적대상 화물들에 대한 적합도의 합계를 각각 구하였다. 측정 시간과 적합도 합계는 각 실험 경우에 대하여 세 차례씩 실시하여 평균값을 취하였다.

- 실험1 : 화주 수 변화에 대한 휴리스틱 알고리즘의 성능 분석

이 실험에서는 선박의 최대 컨테이너 수와 최대 적재용량을 고정된 채 화주의 수를 20부터 5씩 증가시키며 알고리즘의 실행시간과 실행 결과 값을



〈그림 5〉 실험 5 : 알고리즘 수행 시간 비교



〈그림 6〉 실험 6 : 실행 결과 적합도 값 비교

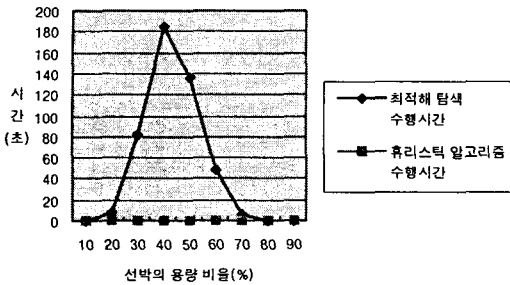
비교하였다. 이때 선박의 최대 컨테이너 수는 각 경우에 화주들의 화물의 총 컨테이너 개수 총합의 40%로 설정하였다. 최대 적재 용량도 마찬가지로 각 경우에 화주들의 화물의 총 컨테이너 중량 총합의 40%로 두었다. 화주 당 화물의 컨테이너 수는 1~4사이의 랜덤 값이 이용되었고, 컨테이너 당 평균 중량은 4~13 사이의 랜덤 값으로 하였다. 각 화주의 선박에 대한 적합도 값은 날짜 값을 랜덤 값으로 생성하여 적합도를 계산할 수도 있으나, 실험의 편의상 50~100 사이의 랜덤 값으로 부여하였다. 아래 그림은 실행 결과를 보여주고 있다.

프로그램 수행 시간에서 휴리스틱 알고리즘은 모든 경우에 대하여 0.02초 정도로 일관적이었던 반면, 최적 탐색 프로그램은 화주 수 35에서는 113초 정도였으나 45에서는 1시간에 이르렀고, 그 이상의 화주 수에 대해서는 3시간 이상을 실행하였으나 프로그램이 종료되지 않았다. <그림 6>의 실행 결과와의 비교에서 제안된 단순한 휴리스틱 알고리즘은 매우 우수한 성능을 나타내었다. 화물의 컨테이너 수는 다소 작은 편차 (1~4 범위)이지만 선박의 컨테이너 적재 중량 제약조건이 있음에도 휴리스틱 알고리즘의 성능이 우수하게 나온 것은 흥미로운 일이다. 즉, 화주당 컨테이너 중량 값의 범위는 (컨테이너 수 × 평균 중량)의 범위, 즉, 4~52로 적지 않은 편차를 갖고 있고, 선박의 최대 허용 중량은 이 실험에서 (화주수 × 화물의총중량

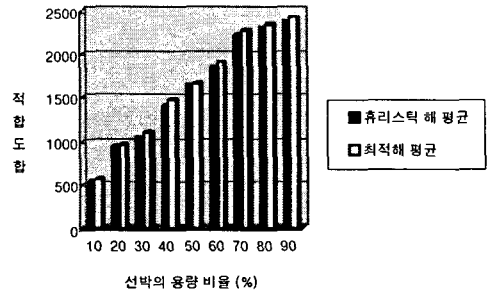
x40%)로서, 화주 수 35의 경우 선박의 중량 한도는 대략 280 정도이다.

● 실험2 : 선박의 용량 변화에 대한 휴리스틱 알고리즘의 성능 분석

이 실험에서는 화주의 수를 고정하고 선박의 컨테이너 최대 적재 수와 최대 적재 중량 값을 변화하여 휴리스틱 알고리즘과 최적해 탐색 알고리즘의 성능을 비교하였다. 화주의 수는 최적 탐색 프로그램의 수행시간이 측정하기 용이한 정도로 하기 위하여 35개로 하였다. 선박의 컨테이너 최대 적재 수와 최대 적재 중량 값은 총 화물의 컨테이너 수와 총 중량 값의 10%~90%까지 10%단위로 증가시키며 실험하였다. 각 측정은 마찬가지로 3회씩 실행한 평균값을 취하였고, 실험1에서 화주의 컨테이너 수가 다소 작고 그 편차도 작았던 점을 감안하여 이번 실험2에서는 그 수치를 늘여 보았다. 실험1에서 화주 수 35의 경우에, 선박의 최대 컨테이너 수는 35정도(화주수 × 2.5개 × 40%)인데 반하여 각 화주의 화물 컨테이너 수는 주로 1~3 사이의 값이 많았다. 실험2에서는 화주당 화물의 컨테이너 수를 5~25사이의 임의 값으로 부여하여 컨테이너수의 편차를 늘여서, 화물 컨테이너수의 편차가 휴리스틱 알고리즘의 성능에 영향을 미치는지의 여부도 알아보고자 하였다. 각 화주의 선박에 대한 적합도 값은 실험1과 마찬가지로 50~100 사이의 임의 값이다. 측정 결과는 아래 그림과 같이 정리되었다.



〈그림 7〉 실험2 : 알고리즘 수행 시간 비교



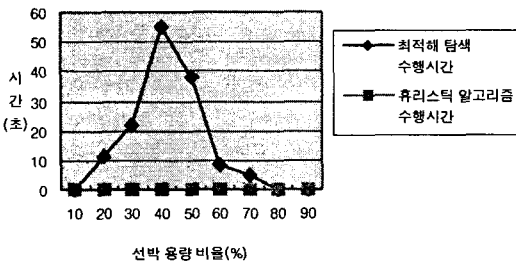
〈그림 8〉 실험2 : 실행 결과 적합도 값 비교

최적해 알고리즘 실행 시간은 대상 화물에 대한 선박의 용량, 즉, 허용 컨테이너 수와 최대 적재 중량에 따라 큰 차이를 보였다. 휴리스틱 알고리즘은 여전히 0.02초 정도의 균등한 실행 시간을 보인 반면, 최적해 프로그램은 선박의 용량인 총 화물의 30%~60% 사이에서 많은 실행 시간을 나타내었다. 즉, 선적 대상 화물이 적거나(선박 용량비율 20%미만), 선적 비대상 화물이 적은 경우(선박 용량비율 70%이상)에는 탐색노드수가 적었으나 그 이외의 경우에는, 특히, 비율이 40%~50% 에서 오랜 실행 시간을 보였다. 화주의 수를 40으로 두고 실험해 보았는데 40%, 50%에서 최적 탐색 프로그램의 측정 불가능할 정도의 장시간이 소요되었으나 그 소요시간 형태는 유사한 패턴을 보였다. 휴리스틱 해의 근접도 비교에서 선박의 컨테이너 수에 대한 제약조건 식의 계수(즉, 화주의 컨테이너 수)의 편차를 실험1

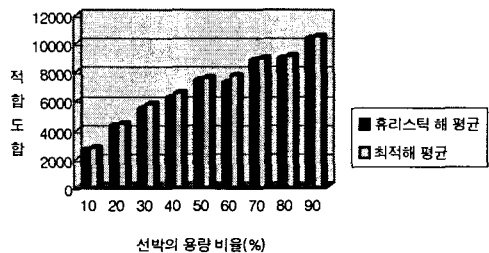
의 1~4범위보다 훨씬 큰 5~25로 두었고, 또한, 컨테이너 수의 범위가 크기 때문에 화주의 화물 중량 값의 범위도 실험1보다 더 크게 생성되었음에도 (20~300 정도) 휴리스틱 해는 최적해에 매우 근접한 결과를 나타내었다.

• 실험3 : 적합도 값의 변경에 대한 휴리스틱 알고리즘의 성능 분석

이 실험에서는 최적화 목표함수의 계수인 적합도 값 편차가 휴리스틱 해의 최적해 근접도에 영향을 주는지를 알아보려고 하였다. 실험1, 실험2에서는 적합도 값을 50~100 사이의 값으로 하였는데, 본 실험에서는 50~550 사이의 큰 범위 내에서 임의의 값으로 지정하였다. 나머지 조건은 실험 2에서와 동일하다. 즉, 화주의 화물 개수와 중량 값의 편차도 실험1보다 큰 범위로 적용되었다. 아래 그림은 실험 결과를 보여주고 있다.



〈그림 9〉 실험3 : 알고리즘 수행 시간 비교



〈그림 10〉 실험3 : 실행 결과 적합도 값 비교

그림에서 나타나듯이, 최적화 프로그램의 실행 시간은 실험2에서와 유사한 패턴을 보여주고 있다. 즉, 용량 비율이 50% 근처에서 상대적으로 최적 탐색시간이 오래 걸렸다. 최적해에 대한 휴리스틱해의 근접도는 <그림 10>에서 보인 바와 같이 적합도 합계는 증가하였으나 여전히 95%이상의 근접도를 나타내었다.

• 성능 분석 정리

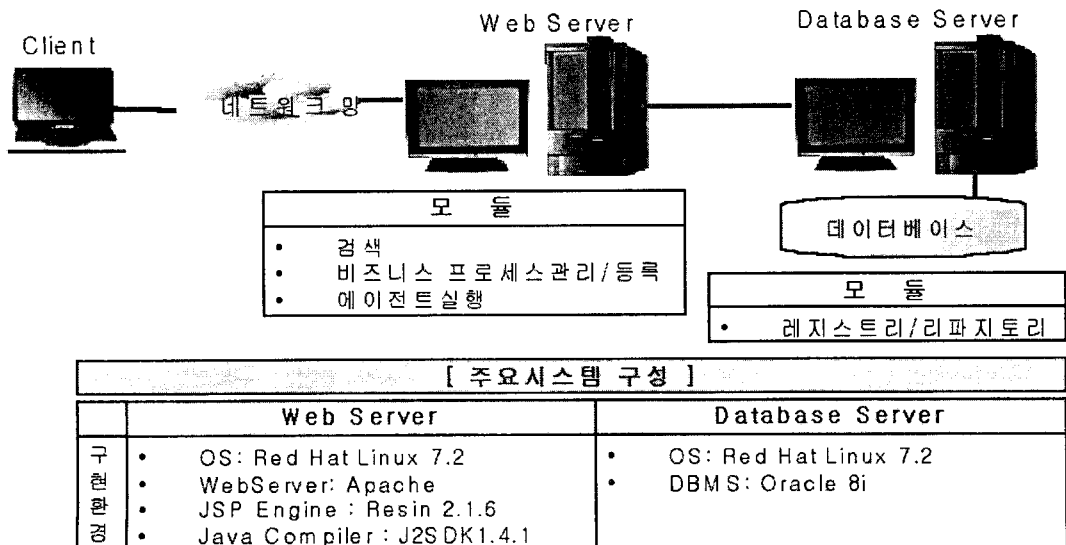
실험 결과 제안된 휴리스틱이 다양한 조건에서 실행 시간과 최적해와의 근접도 면에서 우수한 성능을 나타내었고, 이는 선박에 대한 화주의 중개 문제 모형에 잘 부합될 수 있다는 긍정적인 의미로 해석된다. 그러나 제안된 알고리즘은 절대적인 근접해 (Absolute Approximation)를 제공하지 않는 비교적 단순한 휴리스틱임에도 불구하고 실험에서 95% 이상의 근접도를 일관적으로 나타낸 점에 대하여는 보다 집중적인 연구가 필요하다고 판단된다. 본 연구에서는 전체적인 설계와 문제 모형화 및 실험적 구현에 초점을 두었기 때문에 휴리스틱 알고리즘의 이론적 분석과 데이터 구성과의 관련성에 대한 세부적인 탐구는 향후 연구 주제로 미루기로 한다.

### 4. 프로토타입 구현

#### 4.1 구현 범위 및 시스템 구성

제안한 물류정보 중개 시스템은 3.1절에서 설명한 바와 같이 여러 개의 모듈로 구성되어 있다. 본 연구에서는 그 중 화주와 운송업체로부터 정보를 등록하고 관리하는 비즈니스 프로파일 및 정보 등록/관리 모듈, 선사/화주 정보 검색 모듈, 입력된 정보를 분류하여 해당 테이블에 저장하고 관리하는 레지스트리/리파지토리, 그리고 화주와 운송업체의 요구사항 및 제약조건을 토대로 거래 대상을 생성하는 중개 에이전트 모듈에 대한 프로토타입 (Prototype)을 구현하였다. 본 연구의 구현에서는 중개 기능에 초점을 맞추었으며, <그림 2>에서 외부 시스템과의 연동과 관련한 부분인 XML/EDI 문서생성 및 EAI 부분은 구현 범위에 포함하지 않았다. 중개 시스템의 S/W 및 H/W의 구성은 <그림 11>과 같다.

중개 시스템은 그림과 같이 3-Tier로 구성된다. 화주와 운송업체는 웹 브라우저를 통하여 웹 서버



<그림 11> 중개 시스템 구현 구성도

에 접속하여 비즈니스 프로파일을 검색하거나 정보를 입력하며, 중개 에이전트의 실행을 통하여 거래 가능 대상 업체를 조회한다.

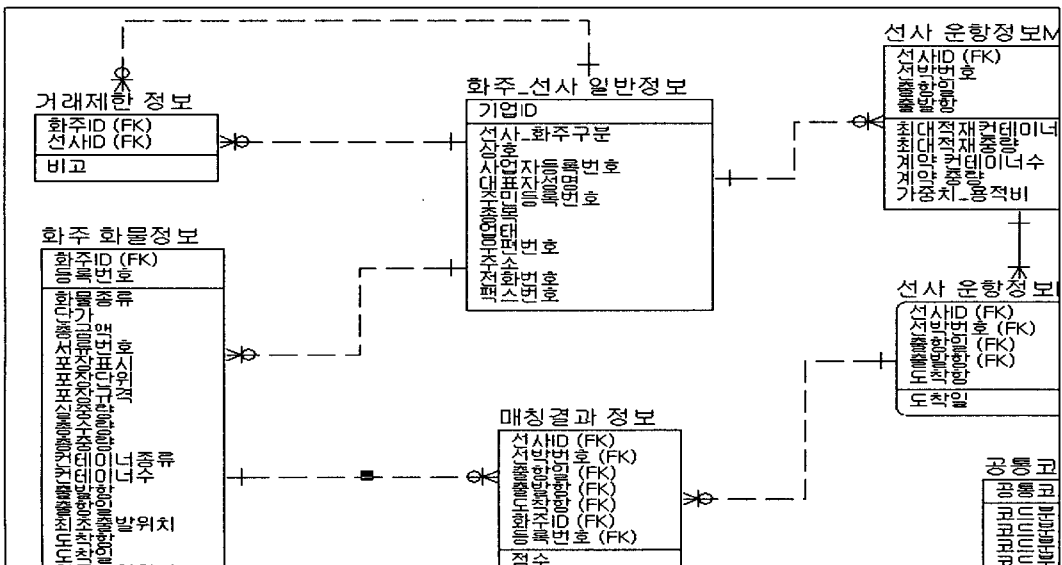
### 4.2 데이터베이스 설계

수출입과 관련된 기본적인 서류는 Commercial Invoice, Packing List(P/L), Bill of Loading(B/L) 등이 있다. Commercial Invoice는 해당 물품의 특성과 내용 명세로서 판매자, 주소, 도착일, 출발항, 도착항, 상품, 수량(중량), 은행, 송장번호, 구매자, 단위, 단가, 총금액 등의 정보를 갖는다. Packing List는 선적화물의 포장 및 포장 단위별 명세와 단

위별 중량, 총중량, 포장의 일련번호 등을 기재한 문서로서 판매자, 도착일, 출발항, 도착항, 상품, 수량(중량), 총중량, 부피, 은행, 송장번호, 구매자, 포장표시, 포장방식 등의 정보가 포함된다. Bill of Loading은 선주가 화주로부터 의뢰받은 운송화물을 선박에 적재 또는 적재를 위해 그 화물을 영수하였음을 증명하는 서류로서 판매자, 출발지, 출발항, 도착항, 도착지, 포장단위(컨테이너 종류, 포장조건), 상품, 총중량, 수량 등의 정보를 담고 있다. 본 연구에서는 위의 세 가지 서류에 기재된 정보를 참고하여 사용자(선사, 화주)가 중개 시스템에 제공해야하는 입력정보를 <표 2>와 같이 정의하고 이에 따라 데이터베이스를 설계하였다.

<표 2> 사용자 입력정보

화주입력정보	선사입력정보
화주기본정보(상호, 주소 등)	선사기본정보(상호, 주소 등)
상품정보	운송수단 보유현황
컨테이너 조건(FCL/LCL, 수량, PKG, 호물종류 등)	운항 Schedule
Port(출발항/도착항)	예상상황
From/To(출발장소/최종도착장소)	
출고(가능) 및 도착시점	



<그림 12> 개체관계도



<표 2>에서 제시된 내용은 본 연구에서 구현하고자 하는 프로토타입 시스템을 위한 기본적인 정보이다. 위의 표에서 제안된 정보를 관리하기 위하여 <그림 12>와 같이 개체관계도(ERD : Entity-Relationship Diagram)를 설계하였다.

<그림 12>의 ERD를 기초로 총 7개의 테이블을 생성하였는데, ERD에 대한 설명은 생성된 테이블의 설명으로 대신한다. 먼저, **화주\_선사\_일반정보** 테이블은 화주와 선사의 기업 ID, 상호, 주소 등 거래 참가 기업의 일반 정보를 저장하기 위한 테이블이다. **화주\_화물정보** 테이블은 화주의 수배송 일정, 화물종류, 포장방식 및 컨테이너 조건 등 화물의 정보를 저장하는 테이블이며 중개 에이전트는 이 테이블에 저장된 정보를 통해 화주의 요구사항을 확인하며 추후 매칭 기준으로 사용한다. **선사\_운항\_주요정보** 테이블은 선사의 전체 선박에 대하여 선박의 선적 가능 공간, 출발항, 출항일 등의 운항 스케줄을 저장하는 테이블로서, 중개 에이전트는 이 테이블에 저장된 정보를 통해 선박의 정보를 확인하고 매칭 기준으로 사용한다. **선사\_운항\_상세정보** 테이블은 하나의 선박이 여러 도착항(경유지)을 가지고 있을 때, 각각의 선박에 대하여 각 도착항과 도착일 정보를 저장하는 테이블이다. **거래\_제한\_정보** 테이블은 각 선사와 화주가 매칭 대상에서 의도적으로 제외시키기를 원하는 대상을 등록하는 테이블이며, 이 테이블에 등록된 선사(화주)는 자신을 등록한 화주(선사)의 매칭 알고리즘 실행시 거래 가능 대상에서 제외된다. **매칭\_결과\_정보** 테이블은 매칭 알고리즘이 실행된 결과를 저장하는 테이블로서, 각 선박의 운송일정을 기준으로 등록되었다가 사용자의 알고리즘 실행 요청이 있을 경우 선박ID, 선박번호, 출발항, 출발항을 기준으로 기존의 매칭 결과를 삭제하고 새로운 매칭 작업 결과를 저장하게 된다. 끝으로, **공통코드** 테이블은 출발 및 도착항, 컨테이너 종류, 포장 단위 등의 데이터를 위하여 편의상 공통코드를 저장하

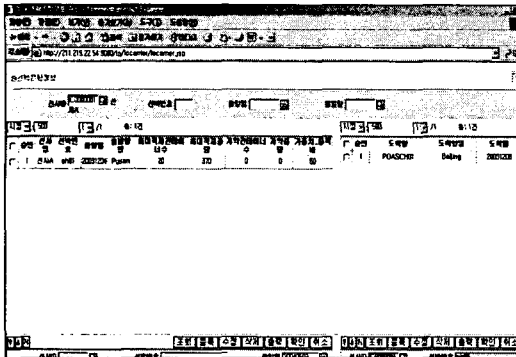
는 테이블이다.

### 4.3 사용자 화면

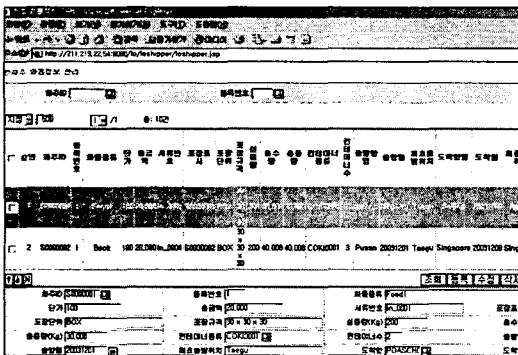
사용자 인터페이스는 JSP를 이용하여 구현하였고 사용자는 웹 브라우저를 통하여 화주와 선사의 정보 입력, 검색, 거래 가능 대상을 생성 등의 작업을 할 수 있다. 구현된 주요 기능들과 그에 대한 화면의 설명은 다음과 같다.

- **선사 및 화주 일반정보 등록/수정/조회 화면**  
선사/화주 일반정보 관리 화면은 중개에 참여하기를 원하는 기업의 기업ID, 상호, 주소, 전화번호 등 일반 정보를 관리하는 화면이다. 이 화면을 통해 입력된 정보는 화주 및 선사 일반정보 테이블에 저장된다. 선사 및 화주 일반정보 관리화면은 <그림 12>의 개체관계도로부터 쉽게 유추될 수 있으므로 생략한다.
- **선사 운항정보 관리 화면**  
선박 운항정보 관리 화면은 각 선사가 보유한 선박에 대한 출항일/출발항, 최대적재 중량/컨테이너수, 계약 중량/컨테이너수 등의 정보를 관리하는 주요정보 관리 화면(좌측)과 각 선박에 대한 여러 도착일/도착항(경유지)에 대한 일정을 관리하는 상세정보 관리 화면(우측)으로 구성된다. <그림 13>과 같이 선사는 운항 일정에 따라 각 선박의 정보를 주요정보 관리 화면에서 입력한 후, 상세정보 관리 화면에서 해당하는 선박에 대한 도착지(경유지) 정보를 입력하게 된다. 이 화면에서 입력된 정보는 각각 선사 운항 주요정보 테이블과 선사 운항 상세정보 테이블에 저장된다.
- **화주 화물정보 관리 화면**  
화주 화물정보 관리 화면은 화주의 화물 수배송과 관련하여 출발항/출발일, 도착항/도착일 등의

수배송 일정과, 화물 종류, 중량, 컨테이너수 등 화물과 관련된 정보를 관리하는 화면이다. 화주는 화물정보를 입력하기 전에 미리 일반정보를 입력해야 하며, 이 화면을 통해 입력된 정보는 화주 화물정보 테이블에 저장된다. <그림 14>는 화면 예를 보여주고 있다.



<그림 13> 선사 운항정보 관리 화면 예



<그림 14> 화주 화물정보 관리 화면 예

● 매칭 알고리즘 실행 및 매칭 결과 조회 화면  
 매칭 정보생성 및 조회 화면을 통해 사용자는 매칭 작업을 실행시키고 그 결과를 확인할 수 있다. 선사는 선사ID, 선박번호, 출항일, 출발항을 입력하고 “Matching” 버튼을 클릭하면 매칭 프로그램이 실행되어 결과를 보여준다. 매칭 프로그램은 선사를 기준으로 실행되는데, 화주의 경우 단순히 화주의 화물을 선적할 수 있는 선사를 찾는 것은 일반적인 조회 작업으로 가능하기 때문에 본 프로토타입

에서는 선사를 기준으로 매칭 프로그램을 설계하였다. 화주는 이 화면을 통해 화주는 매칭 프로그램 실행 결과를 확인할 수 있다. 매칭 프로그램 실행 결과는 매칭 결과정보 테이블에 저장된다.

선사ID	선박번호	출항일	출발항	도착항	화물번호	화물명	중량	컨테이너수	선적일	선적량
1	0000001	2023-04-01	Busan	Seoul	0000001	001	1	0000001	2023-04-01	1
2	0000002	2023-04-01	Busan	Seoul	0000002	002	1	0000002	2023-04-01	1
3	0000003	2023-04-01	Busan	Seoul	0000003	003	1	0000003	2023-04-01	1
4	0000004	2023-04-01	Busan	Seoul	0000004	004	1	0000004	2023-04-01	1
5	0000005	2023-04-01	Busan	Seoul	0000005	005	1	0000005	2023-04-01	1
6	0000006	2023-04-01	Busan	Seoul	0000006	006	1	0000006	2023-04-01	1
7	0000007	2023-04-01	Busan	Seoul	0000007	007	1	0000007	2023-04-01	1
8	0000008	2023-04-01	Busan	Seoul	0000008	008	1	0000008	2023-04-01	1
9	0000009	2023-04-01	Busan	Seoul	0000009	009	1	0000009	2023-04-01	1
10	0000010	2023-04-01	Busan	Seoul	0000010	010	1	0000010	2023-04-01	1

<그림 15> 매칭 정보 생성/조회 화면 예

## 5. 결 론

물류체제의 효율화는 전자상거래가 확산될수록 그 중요성이 부각되고 있다. 특히 다양한 분야에서 상품/화물의 운송이 요구되고 있으며, 많은 부분들이 전자화 또는 자동화되고 있다. 그러나 운송업체와 화주간의 중개는 아직도 전문 중개인을 통한 전통적 거래에 주로 의지하고 있다. 본 논문에서는 화주와 운송업체의 정보를 통합적으로 관리하고 거래 대상을 중개하는 물류정보 중개 시스템에 대하여 연구하였다. 구체적으로, 화주와 선박간의 중개를 대상으로 하여, 그 요구사항을 정의하고 중개 방법과 중개 기준을 제시하였으며, 이를 바탕으로 중개 최적화 모형을 설정하고 Greedy 탐색에 근거한 중개 알고리즘을 제시하였다. 제안된 알고리즘은 간단한 휴리스틱에 기반하고 있음에도 불구하고 성능분석 실험에서 실행시간 뿐 아니라 최적해와의 근접도에서도 우수한 결과를 보여주어, Greedy 휴리스틱이 화물과 선박의 중개에 대한 응용 데이터의 특성에 매우 적합하다는 것을 시사하였다. 실행 시간 면에서 화주 수와 선박의 용량 변화에 거의 영향을 받지 않고 수초이내의 매우 빠른

응답시간을 보였으며, 최적해와의 목표함수값을 비교한 결과 대부분 실험 환경에서 95%이상의 우수한 근접도를 보였다. 제안된 시스템을 웹 환경에서 웹 환경에서 JSP와 PL/SQL을 이용하여 구현하고 데이터베이스 설계와 주요 사용자 화면설계의 구현 예를 소개하였다.

본 연구에서는 선박의 운항 일정, 화물의 선적 방법, 화주와 선주의 계약 조건 등에 대하여 여러 가정을 두었다. 향후 연구 주제로서, LCL의 고려 등 이러한 가정들에 대한 완화와 함께, 본 논문에서 제시한 중개 요구조건과 중개 기준 및 가중치에 대한 다양한 운송 환경에서의 실증적 연구 및 보완, 화주와 선박의 중개 특성과 데이터 특성을 고려한 휴리스틱 고안 등에 대한 보다 집중적인 연구가 필요할 것이다.

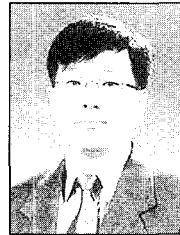
### 〈참 고 문 헌〉

- [1] 강태원, 전서현, "위치기반 유무선 통합 물류 시스템 설계 및 구현", *한국정보과학회, 추계 학술발표논문집*, Vol. 29, No. 2, 2002.
- [2] 권아름, 박운재, 김정중, "컨테이너 최적 적재를 위한 알고리즘", *한국정보처리학회, 춘계 학술발표대회*, 제2권 제1호, 1995, pp. 188-191.
- [3] 김동수, "차세대 e-비즈니스 표준화 추진전략", *전자상거래표준화통합포럼, 정보화 정책*, 제9권 제1호, 2002, pp. 61-80.
- [4] 김영일, 이용준, 황재각, "ebXML을 활용한 e-Logistics 통합플랫폼에서의 등록저장소의 설계", *한국정보과학회, 2003년 춘계학술대회*, Vol.30, No.1, 2003.
- [5] 오동근, 홍정선, 전종미, 김광훈, "ebXML 기반의 전자물류 비즈니스 프로세스 관리 시스템 엔진", *한국정보과학회, 2003년 춘계학술대회*, Vol.30, No.1, 2003.
- [6] 이수철, 변광준, 황인준, "XML 기반 B2B 전자상거래 솔루션 동향", *한국정보과학회지*, 제18권, 제7호, 2000, pp. 21-27.
- [7] 전종미 외, "ebXML 기반 e-Logistics 프로세스 모델링 시스템 아키텍처", *한국정보처리학회, 추계학술발표대회논문집*, 제9권 제2호, 2002.
- [8] 최중민, "에이전트기술 및 전자상거래", *공학기술학회지*, Vol. 7, No. 2, 2002, pp.37-40.
- [9] 한국전자거래진흥원, "e-비즈니스 기술로드맵", 산업자원부, 2003.
- [10] 황재각, 오세원, 이용준, "EbXML 표준에 기반한 다자간 물류 통합정보시스템", *한국정보과학회, 2002년 춘계학술대회*, Vol.29, No.2, 2002.
- [11] Barth, P., *A Davis-Putnam Based Enumeration Algorithm for Linear Pseudo-Boolean Optimization*, MPI-1-95-2-003, MPI Informatic, Germany, 1995 (<http://www.mpi-sb.mpg.de/units/ag2/software/opbdp>).
- [12] Flores-Mendez, R.A., *Towards a Standardization of Multi-Agent System Frameworks*, ACM Crossroads student magazine, 2001, pp.1-25.
- [13] Freuder, E. and Wallace, R., *Suggestion Strategies for Constraint-based Matchmaker Agents*, in Principles and Practice of Constraint Programming, CP'98, 1998.
- [14] Hofreiter, B., Huemer, C. and W. Klas, *ebXML : Status, Research Issues, and Obstacles*, IEEE RIDE'02, 2002.
- [15] Horowitz, E. and Sahni, S., *Fundamentals of Computer Algorithms*, Computer Science Press, 1978.
- [16] Jung, J. and Jo, G., "Brokerage between Buyer and Seller Agents using Constraint Satisfaction Problem Models", *Decision Support Systems*, Vol. 28, Issue 4, 2000.

- [17] Keskinocak, P., Godwin, R., Wu, F., Akkiraju, R., and Murthy, S., "Decision Support for Managing an Electronic Supply Chain," *Electronic Commerce Research*, Vol.1, 2001.
- [18] Lau, H.C. and Goh, Y.G., *An Intelligent Brokering System to Support Multi-Agent Web-based 4th-Party Logistics*, Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02), 2002.
- [19] Lu, E., Tsai, R.H and Chou, S., *An empirical study of XML/EDI*, The Journal of Systems and Software 58, 2001.
- [20] Ma, M., "Agents in E-Commerce", *Communications of the ACM*, Vol.42, No.3, 1999.
- [21] Martello, S. and Toth, P., *Knapsack Problems*, John Wiley & Sons, 1990.
- [22] Segev, A. and Beam, C., *Brokering Strategies in Electronic Commerce Markets*, Proceedings of 1st ACM Conference on Electronic Commerce, Nov., 1999.
- [23] Xhafa, F., *0-1 Multidimensional Knapsack*

*Repository*, TRACER Project, Dept. of Languages and Informatic Systems, Universitat Politècnica de Catalunya (Barcelona) SPAIN, 2004.

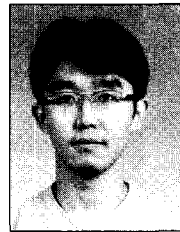
#### ■ 저자소개



#### 서 상 구

서울대학교 컴퓨터공학과를 졸업하고, KAIST 전산학과에서 석사, 박사학위를 취득하였으며 현대전자에서 데이터 웨어하우징 업무를 담당하였다.

현재 광운대학교 경영정보학과 교수로 재직하고 있으며 주요관심분야는 데이터베이스, 웹 서비스, 전자상거래 QoS 등이다.



#### 윤 경 현

순천향대학교 독어독문학과를 졸업하고, 광운대학교 경영정보학과에서 석사학위를 취득하였다. 현재 (주) 옴톤의 기술 지원팀에 재직하고 있으며 시

스템 분석 및 설계 업무를 담당하고 있다. 주요관심분야는 데이터 웨어하우징, ebXML, 등이다.