

<https://doi.org/10.7236/IIBC.2018.18.6.213>

IIBC 2018-6-29

독점시장에서 생존할 수 있는 신규 점포 위치 결정 알고리즘

Decision Algorithm for Survival New Establishment Stores Location in Monopoly Market

이상운*

Sang-Un, Lee*

요약 본 논문은 동종업계의 경쟁업체가 시장을 독점하고 있는 상황에서, 신규로 점포를 개설하여 경쟁업체 고객을 빼앗아 경쟁업체 점포를 문을 닫게 하여 생존할 수 있는 점포위치를 결정하는 생존 점포 개설 위치 선정 문제(SFLP)를 다룬다. 이 문제는 포화된 시장에서 문을 여는 점포보다 문을 닫는 점포가 훨씬 많아지고 있는 현실에서 점포를 신규로 개설하고자 할 때 부딪히는 난제이다. 이 문제에 대해 Serra et al.은 반복적 휴리스틱 집중 알고리즘을, Han et al.은 반복적으로 최대 고객 확보 위치를 찾는 알고리즘을 제안하였다. 그러나 이들 방법은 다수의 위치들에 대해 반복적으로 계산하는 어려움이 있다. 본 논문에서는 경쟁업체 인접 지점들만을 대상으로 경쟁업체 점포를 폐쇄시키면서 최대 고객을 확보할 수 있는 지점으로 해를 구하고, 최소 고객 확보 지점을 최대 고객을 확보한 다른 경쟁업체 점포를 폐쇄시킬 수 있는 지점으로 변경하는 과정을 수행하였다. 제안된 알고리즘은 단순하면서도 MS-Excel을 활용하여 해를 구할 수 있는 장점을 갖고 있다.

Abstract This paper deals with survival facility location problem(SFLP) that the store with less of demand threshold level is closed result from another new establishment of store in the same kind of comparative firms have a monopoly market. We will be faced with a difficult problem when a new establishment stores in market saturation that the closed stores more than opening stores. Serra et al. proposes recursive heuristic concentration algorithm, and Han et al. suggests maximum insurance of customer location. But the drawback of these algorithms is a recursively computation for many locations. This paper get the solution from only neighborhood search of comparative firm's stores that can be maximum customers and closed comparative firm's store, and the location with minimum customer exchange to the location that can be closed the comparative firm's store with maximum customer. The advantage of this algorithm is to get the solution using a MS-Excel.

Key Words : Monopoly market, Demand threshold level, Closed, Neighborhood search, Survival

1. 서론

최근 들어 베이비부머들이 은퇴 하면서, 너도나도 자

영업, 프랜차이즈 가맹점, 또는 편의점을 개설하고 있다. 그러나 기존 경쟁업체들이 시장을 독점하여 경쟁이 치열 하면서 시장이 포화된 상황에서, 동종업계로 신규로 시

*정회원, 강릉원주대학교 과학기술대학 멀티미디어공학과
접수일자: 2018년 5월 30일, 수정완료: 2018년 11월 1일
게재확정일자: 2018년 12월 7일

Received: 30 May, 2018 / Revised: 1 November, 2018 /

Accepted: 7 December, 2018

*Corresponding Author: sulee@gwnu.ac.kr

Dept. of Multimedia Eng., Gangneung-Wonju National University, Korea

장에 진출하여 성공하기는 매우 어렵다. 따라서 문을 여는 가격보다 문 닫는 가격이 보다 많은 게 현실로 커다란 사회문제가 되고 있다. 이러한 상황에서도 기존의 경쟁업체 점포 고객을 한계수준 이하로 하여 폐쇄(문을 닫게)시키고, 자신의 신설 점포가 성공하기 위해서는 어느 위치에 점포를 개설해야 하는가라는 문제는 점포를 개설하기 이전에 생존차원에서 필수적으로 해결해야만 하는 난제이다. 따라서 본 논문에서는 이 문제에 대해 다룬다.

p 명의 주민(population)이 거주하고 있는 n 개 노드(지점)이 복잡한 도로들로 연결된 하나의 도시에 동일 업종의 업체 F_A 가 k 개의 점포를 운영하여 전체 주민 p 명을 독점하고 있는 상황에서, 동일 업종, 동일 제품 인지도를 갖는 후발업체인 F_B 가 신규로 k 개의 점포를 개설하고자 한다. 여기서 고객 확보 수 한계치 T 가 되지 못하는 점포는 폐쇄된다고 가정한다. 이 경우 F_A 의 점포를 가능한 많이 폐쇄시키고, F_B 가 k 개의 점포를 운영하도록 F_B 의 신규 개설 점포 k 개를 어느 노드에 개설해야 하는가가 문제로 발생한다. 여기서, 고객은 동일 업종의 동일한 경쟁력과 제품 인지도를 갖고 있어 가장 가까운 점포에서 물건을 구입한다고 가정한다. 생존 점포 개설 위치 선정 문제(survivable facility location problem, SFLP)라 한다.^[1]

SFLP에 대해, Serra et al.^[1]은 F_A 업체 점포 위치를 제외한 $n-k$ 의 노드들을 대상으로 F_A 의 점포를 폐쇄시키면서 최대 고객을 확보할 수 있는 위치를 반복적으로 찾는 휴리스틱 집중 (Heuristic concentration, HC) 알고리즘을 제안하였다. 또한 Han et al.^[2]은 F_A 업체 점포 위치를 제외한 $n-k$ 의 노드들을 대상으로 F_A 가 선점한 노드들을 대상으로 F_A 의 점포를 폐쇄시키면서 최대 고객을 확보할 수 있는 위치를 반복적으로 찾는 최대 고객 확보 위치(maximum insurance of customer location, MICL) 알고리즘을 제안하였다. 그러나 이들 알고리즘은 $n-k$ 의 노드들을 대상으로 반복수행해야 하며, 확보 고객 수를 계산하기 위해 반복적으로 각 노드에서의 최단 경로를 계산해야 하는 어려움이 있다.

이와는 별개로, 경쟁업체 F_A 가 k 개 점포를 운영시 후발 업체 F_B 가 동일한 k 개 점포를 개설하는 경우 확보 고객이 적더라도 폐쇄되지 않는 경우 경쟁업체에 비해 보다 많은 고객을 확보하는 데만 초점을 맞춘 문제에 대해서는 Serra와 Reville^[3]의 $n-k$ 개 중 신규로 개설할

점포 수 k 개와 반복적으로 상호 교환하는 PREMAL(PRe-EMptive heuristic ALgorithm)과, Choi et al.^[4]의 망을 k 개 영역으로 분할하여 각 영역에서 최대 고객을 확보할 수 있는 지점을 선택하는 분할정복 알고리즘(divide-and-conquer algorithm, DCA)이 있다. 또한, 경쟁업체 F_A 가 k 개 점포를 운영시 F_B 가 1, 2, ..., k 개 점포를 개설하는 경우, 점포 폐쇄 없이 경쟁업체에 비해 보다 많은 고객을 확보하는 데만 초점(단지 경쟁우위만 추구)을 맞춘 문제에 대해 Lee et al.^[5]의 구성 알고리즘(constructive algorithm, CA)이 있다. 이들 알고리즘은 본 연구대상인 기존 업체의 점포를 폐쇄시켜 후발업체의 신설 점포들이 생존과 동시에 시장까지 독점하는데 초점을 맞춘 문제인 SFLP^[6]와 상이하여 직접적으로 적용할 수 없다. 따라서 이들 연구 결과는 이후로는 다루지 않는다.

본 논문에서는 Serra et al.^[1]의 HC나 Han et al.^[2]의 MICL에 비해 단순히 MS-Excel을 활용하여 빠르게 해를 구할 수 있는 알고리즘을 제안한다. 2장에서는 관련 연구와 문제점을 고찰한다. 3장에서는 경쟁업체 점포와 인접한 노드들만을 대상으로 신규 점포 위치를 결정하는 알고리즘을 제안한다. 4장에서는 HC와 MICL로 해를 얻은 실험 데이터에 대해 제안 알고리즘을 적용하여 적합성을 검증하여 본다.

II. 관련 연구와 문제점

각 점포는 확보한 고객 수가 요구 한계 수준(demand threshold level, T) 이상을 충족시켜야만 점포 운영비용을 제외하고 이득을 남길 수 있어 점포를 계속 운영할 수 있다. 그렇지 못하면 점포를 운영하면 할수록 매출급감에 따른 지속적인 적자로 인해 점포는 폐업한다(문을 닫는다)고 가정한다. Serra et al.^[1]는 식 (1)의 T 를 제시하였다.

$$T = 0.8 \frac{p}{k+k} \quad (1)$$

n 개 노드로 구성된 도시에 F_A 경쟁업체가 k 개 점포를 운영하여 도시 전체 주민 수 p 를 독점하고 있는 상황에서 동종업의 후발업체 F_B 가 신규로 점포를 F_A 와 동일한 개수인 k 개 개설할 경우, F_A 점포를 폐업시키면서 경쟁우위를 확보할 수 있는 k 개의 신규 개설 점포 위치를

결정하고자 한다.

Serra et al.^[1]은 이 문제를 해결하기 위해 다음 2가지 규칙을 적용하였다.

[규칙 1]

1. 새로 입점하는 점포는 항상 한계수준을 충족시키는 노드에 입점한다.
2. 만약, 2개 이상의 경쟁점포가 한계수준 이하의 고객을 확보하면 최소로 고객을 확보한 점포를 가장 먼저 폐쇄하고 나머지 점포들에 고객을 재배치한다.

[규칙 2]

1. 새로 입점하는 점포는 한계수준 충족 여부와 상관 없이 임의의 노드에 입점한다.
2. 일단 점포가 입점하면, 각 점포들이 가장 가까운 고객들을 확보한 것으로 확보고객수를 계산한다. 만약, 2개 이상의 경쟁점포와 새로 입점한 점포가 한계수준 이하의 고객을 확보하면 최소로 고객을 확보한 점포를 가장 먼저 폐쇄하고 나머지 점포들에 고객을 재배치한다. 이 절차는 한계 제약조건을 충족시키지 못하는 점포가 없을 때까지 반복 수행한다.

Serra et al.^[1]은 이 문제를 풀기 위해 휴리스틱 집중 (HC) 알고리즘을 제안하였다. Serra et al.^[1]의 HC는 $n-k$ 개 노드 중에서 F_B 의 신규 개설 점포 수 k 개를 [규칙 1]과 [규칙 2]를 반복적으로 적용하면서 찾는 방법으로 최악의 경우 ${}_{n-k}C_k$ 의 수행 횟수가 요구된다.

Han et al.^[2]의 MICL은 F_A 의 각 점포가 확보한 최단 거리 노드들을 구하여 이들 노드들 중에서 고객을 최대 로 확보할 수 있는 위치로 F_B 점포 후보로 선정하고, 폐쇄되지 않은 F_A 점포를 집중 공격할 수 있는 나머지 점포 위치를 찾는 방법이다. 이 방법 역시 $n-k$ 개 노드 중에서 F_B 의 신규 개설 점포 수 k 개를 찾기 위해 최대 고객을 확보할 수 있는 위치를 반복적으로 계산하여 $n-k$ 의 수행횟수가 요구된다.

그림 1은 Serra et al.^[1]과 Han et al.^[2]이 해를 얻는데 사용한 Swain-55 노드 망이다. 그림 3의 망에 대해, $F_A = \{1,16,29,41\}$ 인 경우 Serra et al.^[1]과 Han et al.^[2]이 얻은 결과인 F_B 는 표 1과 같다.

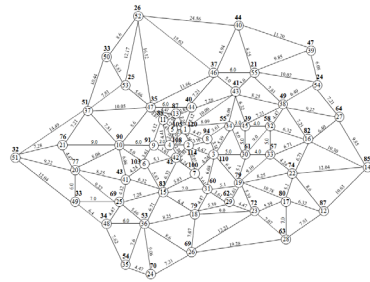


그림 1. Swain 도시 망
 Fig. 1. Swain-55 network

표 1. 알고리즘 수행 결과

Table 1. Result of algorithm

$F_A = \{1,16,29,41\}$		
Serra et al. ^[1]	$F_A = \{1\}$	$F_B = \{9,31,33,49\}$, 2610(73.9%)
HC 알고리즘	$F_A = \{41\}$	$F_B = \{2,13,23,32\}$, 2707(75.7%)
Han et al. ^[2]	$F_A = \{1\}$	$F_B = \{2,11,31,33\}$, 2828(79.1%)

표 1의 결과를 얻는 과정을 상세히 설명하면 다음과 같다. Serra et al.^[1]은 [규칙 1]을 적용한 HC 알고리즘으로 $F_B = \{9,31,33,49\}$, $F_A = \{1\}$ 로 $F_A = \{16,29,41\}$ 점포가 폐쇄되는 결과를 얻었으며, [규칙 2]를 적용한 HC 알고리즘은 $F_B = \{2,13,23,32\}$, $F_A = \{41\}$ 로 $F_A = \{1,16,29\}$ 점포가 폐쇄되는 결과를 얻었다.

규칙	업체	점포별 고객수				시장 점유율
1	F_B	9(528)	31(777)	33(807)	49(498)	2,610(73.0%)
	F_A	1(965)	16(closed)	29(closed)	41(closed)	965(27.0%)
2	F_B	2(714)	13(681)	23(651)	32(661)	2,707(75.7%)
	F_A	1(closed)	16(closed)	29(closed)	41(868)	868(24.3%)

Han et al.^[2]의 MICL 알고리즘을 적용하여 사전처리를 수행한 결과는 1(1,316), 16(666), 29(725), 41(868)이다.

업체	점포별 고객수				시장 점유율
F_B	-	-	-	-	0.0%
F_A	1(1,316)	16(666)	29(725)	41(868)	3,575(100%)

$n-k = 55 - 4 = 51$ 개 노드에 대해 Step 1 수행 결과 2(614), 33(315), 31(308), 49(376)을 얻는다. Step 2 수행 결과 $F_B = \{33\}$ 을 입점시키면 $F_A = \{16\}$ 을 폐쇄시킬 수 있다.

$$C(1) - T = 1316 - 357.5 = 958.5 > 2(614)$$

$$C(16) - T = 666 - 357.5 = 308.5 < 33(315)$$

$$C(29) - T = 725 - 357.5 = 367.5 > 31(308)$$

$$C(41) - T = 868 - 357.5 = 510.5 > 49(376)$$

업체	점포별 고객수			
F_B	2(614)	33(807)	31(308)	49(376)
F_A	1(702)	16(closed)	29(276)	41(492)

Step 2 수행 결과 $F_B = \{2, 33, 31, 49\}$ 와 $F_A = \{1, 29, 41\}$ 의 확보 고객수를 재할당하여 Step 3 수행 결과 $T = 357.5$ 이하의 확보 고객 수는 $F_A = \{29\}$ 가 존재하여 폐쇄된다. 남은 $F_B = \{2, 33, 31, 49\}$ 와 $F_A = \{1, 41\}$ 의 확보 고객수를 재할당한 결과 $T = 357.5$ 이하가 되는 점포가 없어 더 이상 폐쇄되는 노드가 없다. 따라서 F_B 의 마지막 후보 노드는 $F_A = \{41\}$ 에 대응하는 $F_B = \{49\}$ 를 삭제하고 최대 확보 고객수를 가진 $F_A = \{1\}$ 을 공격할 수 있는 점포 노드를 찾는다.

업체	점포별 고객수			
F_B	2(614)	33(807)	31(584)	49(376)
F_A	1(702)	16(closed)	29(closed)	41(492)

이 노드들은 702-357.5=344.5 이상인 {5,11,13}로 결정되며, {5,11,13} 중에서 $F_A = \{1\}$ 의 확보 고객수를 최소로 하는 노드는 $F_B = \{11\}$ 로 결정되었다. 최종적으로 얻은 해는 $F_B = \{2, 11, 31, 33\}$, $F_A = \{41\}$ 이다.

업체	점포별 고객수				
F_B	2(523)	11(541)	33(807)	31(654)	-
F_A	1(303)	-	-	-	41(747)

업체	점포별 고객수				
F_B	2(614)	13(261)	33(807)	31(654)	-
F_A	1(408)	-	-	-	41(798)

업체	점포별 고객수				
F_B	2(515)	5(499)	33(807)	31(654)	-
F_A	1(408)	16(closed)	29(closed)	41(798)	-

업체	점포별 고객수					시장 점유율
F_B	2(643)	11(724)	33(807)	31(654)	-	2,828(79.1%)
F_A	1(closed)	16(closed)	29(closed)	41(747)	-	747(20.9%)

Serra et al.^[1]의 HC 알고리즘에 비해 Han et al.^[2]의 MICL 알고리즘은 수행횟수를 1차 후보 l 개 선정시 $n-k$

회, 2차 후보 선정시 $n-k-l$ 회로 감소시켰으나 각 노드에 대한 고객 확보 수를 다시 계산해야 하는 복잡성을 갖고 있다. 따라서 컴퓨터 프로그램의 도움 없이 현장 실무자가 이들 알고리즘을 적용하여 해를 얻기에는 어려움이 따른다.

이러한 어려움이 있어 3장에서는 단순히 MS-Excel을 활용하여 간단히 해를 구할 수 있는 알고리즘을 제안하여 현장 실무자가 쉽게 활용할 수 있도록 하고자 한다.

III. 근방 탐색 알고리즘

본 장에서는 시장을 독점하고 있는 경쟁업체 F_A 점포의 고객을 한계수준 T 이하가 되도록 하여 폐쇄시킬 수 있도록 신규 시장 진입 F_B 업체의 개설 점포 위치를 쉽게 구할 수 있는 알고리즘을 제안한다. 제안된 알고리즘은 다음 규칙에 기반을 두고 있다.

[규칙 1] 신규 시장 진입 F_B 업체의 개설 점포 위치는 독점 업체 F_A 의 점포 위치와 동일하면 안된다. 왜냐하면, 동종업의 동일한 경쟁력을 가진 제품을 판매하기 때문에 해당 위치 점포의 확보 고객 수를 양분하여 최악의 경우 확보 고객 수가 T 이하로 부족으로 두 점포 모두 폐쇄되거나 두 점포 모두 운영되기 때문이다. 따라서 신규 개설 점포 위치는 $n-k$ 개 노드를 대상으로 한다.

[규칙 2] 한 도시의 최 외곽 노드들은 인접(이웃) 노드들이 적어 이 노드에 점포를 입점시키면 확보 고객 수가 T 이하로 폐쇄되기 때문에 이들 노드 m 개를 제외시킨다. 따라서 대상 노드 수는 $n-k-m$ 으로 감소한다.

[규칙 3] 독점 업체 F_A 의 점포 위치 u 에 대해, 이 점포의 고객을 최대한 빼앗아 생존하기 위해서는 u 와 멀리 떨어져 있으면 안되며, 가능한 가까운 노드에 입점시켜야만 한다. 이 규칙에 따라 F_B 업체의 개설 점포 위치는 기본적으로 $N_C(u)$ 로 한정시키며, 추가로 $N_C(u)$ 의 최대 이격거리 내에 있는 노드들도 $N_C(u)$ 로 가정한다. 단, [규칙 2]에 의거 $N_C(u)$ 들 중 최 외곽 노드들은 제외시킨다.

본 장에서는 [규칙 1]과 [규칙 2]를 적용하지 않고, [규칙 3]만을 적용하여 신규개설 점포의 위치를 결정하는 방법을 제안하며, 이 알고리즘은 독점업체 F_A 의 근방인 $N_C(u)$ 노드들만을 대상으로 F_B 업체의 개설 점포 위치를 결정하므로, 이를 근방 탐색(neighborhood search, NS)^[7] 알고리즘이라 하자.

위치	주민수	F_A 점포 위치				F_B 점포 위치				F_A 확보 고객 수				F_B 확보 고객 수			
		1	16	29	41	2	33	31	6	1	16	29	41	2	33	31	6
1	120	1.00	21.40	17.45	14.85	3.16	16.63	13.71	10.61	120	0	0	0	0	0	0	0
2	114	3.16	20.09	14.29	12.17	1.00	13.47	10.68	8.61	0	0	0	0	114	0	0	0
3	110	7.63	15.71	9.69	17.35	4.47	9.00	6.08	12.40	0	0	0	0	110	0	0	0
4	108	5.00	23.09	13.70	9.85	3.00	16.47	10.09	5.61	0	0	0	0	108	0	0	0
5	105	2.00	25.34	16.70	12.85	5.16	18.63	13.09	8.61	105	0	0	0	0	0	0	0
6	103	10.61	28.11	16.44	4.24	8.61	21.40	12.83	1.00	0	0	0	0	0	0	0	103
7	100	10.23	19.95	7.22	12.15	7.07	13.24	3.61	9.34	0	0	0	0	0	0	100	0
8	94	4.47	16.93	12.85	15.33	3.16	12.16	9.24	11.09	0	0	0	0	94	0	0	0
9	91	7.00	25.18	15.07	7.85	5.00	18.47	11.46	3.61	0	0	0	0	0	0	0	91
10	90	13.00	31.18	21.07	6.08	11.00	24.47	17.46	5.00	0	0	0	0	0	0	0	90
11	88	3.41	26.75	18.11	11.97	6.57	20.04	14.50	7.73	88	0	0	0	0	0	0	0
12	87	30.88	15.31	16.18	33.11	26.31	12.84	19.79	32.62	0	0	0	0	87	0	0	0
13	87	2.24	23.64	19.88	14.21	5.40	18.87	16.08	9.97	87	0	0	0	0	0	0	0
14	85	31.70	10.30	23.90	38.71	29.75	16.28	25.39	38.22	0	85	0	0	0	0	0	0
15	83	12.31	25.78	10.61	6.32	9.15	19.07	7.00	5.83	0	0	0	0	0	0	0	83
16	82	21.40	1.00	23.96	32.10	20.09	6.71	20.91	28.11	0	82	0	0	0	0	0	0
17	80	24.81	11.81	9.86	26.79	21.65	9.34	17.08	26.30	0	0	0	0	80	0	0	0
18	79	18.18	23.96	5.39	12.72	15.15	18.10	4.47	12.23	0	0	0	0	0	0	79	0
19	79	14.03	14.96	3.61	18.42	10.87	9.10	5.10	17.93	0	0	79	0	0	0	0	0
20	77	21.06	39.24	25.18	8.25	19.06	32.53	21.57	12.49	0	0	0	77	0	0	0	0
21	76	22.00	40.18	30.07	12.72	20.00	34.47	26.46	14.00	0	0	0	76	0	0	0	0
22	74	20.87	6.71	11.86	26.67	17.71	4.24	13.35	26.18	0	0	0	0	74	0	0	0
23	72	19.42	17.20	4.47	21.72	16.26	14.49	8.08	20.91	0	0	72	0	0	0	0	0
24	70	28.08	36.62	19.67	18.84	24.92	32.38	18.75	21.60	0	0	0	0	0	0	70	0
25	69	19.59	33.06	18.97	4.12	16.29	26.35	14.28	8.36	0	0	0	69	0	0	0	0
26	69	25.25	29.41	12.46	18.38	22.22	25.17	11.54	19.30	0	0	0	0	0	0	69	0
27	64	27.80	6.40	24.97	39.78	26.20	13.11	26.46	34.13	0	64	0	0	0	0	0	0
28	63	26.49	18.81	11.54	28.88	23.33	16.34	15.15	27.98	0	0	63	0	0	0	0	0
29	62	17.45	23.96	1.00	16.44	14.29	12.71	3.61	16.44	0	0	62	0	0	0	0	0
30	61	12.63	10.71	8.71	21.39	9.47	4.00	10.20	17.40	0	0	0	0	61	0	0	0
31	60	13.71	20.91	3.61	13.32	10.68	14.20	1.00	12.83	0	0	0	0	0	0	60	0
32	58	15.08	6.32	17.71	25.94	13.77	5.00	18.47	21.70	0	0	0	0	0	58	0	0
33	57	16.63	6.71	12.71	25.64	13.47	1.00	14.20	21.40	0	0	0	0	57	0	0	0
34	55	8.08	13.32	15.08	18.94	6.77	12.00	11.47	14.70	0	0	0	0	55	0	0	0
35	54	26.62	41.09	21.24	16.21	26.62	33.95	20.32	20.14	0	0	0	54	0	0	0	0
36	53	19.02	32.21	13.64	9.78	19.02	26.35	12.72	12.54	0	0	0	53	0	0	0	0
37	51	16.76	38.47	28.88	13.89	20.23	32.28	25.27	12.81	0	0	0	0	0	0	51	0
38	49	18.25	7.21	20.92	29.15	16.98	9.47	21.68	24.91	0	49	0	0	0	0	0	0
39	47	24.85	19.69	33.40	39.70	29.46	21.95	32.40	34.93	0	47	0	0	0	0	0	0
40	44	20.27	23.27	33.28	33.19	23.43	25.53	30.80	28.95	44	0	0	0	0	0	0	0
41	43	14.85	32.10	16.44	1.00	12.17	25.64	13.32	4.24	0	0	0	43	0	0	0	0
42	42	5.99	22.92	11.46	9.34	2.83	16.30	7.85	5.10	0	0	0	0	42	0	0	0
43	41	10.00	15.46	20.03	24.32	12.85	15.32	17.55	20.08	41	0	0	0	0	0	0	0
44	40	4.12	22.74	21.44	17.04	7.28	20.75	17.83	12.80	40	0	0	0	0	0	0	0
45	39	11.08	10.32	13.71	21.94	9.77	9.00	14.47	17.70	0	0	0	0	39	0	0	0
46	37	11.42	20.46	25.03	24.25	14.49	20.32	22.55	20.01	37	0	0	0	0	0	0	0
47	35	6.71	28.42	21.72	14.68	10.08	23.34	18.11	11.34	35	0	0	0	0	0	0	0
48	34	24.06	37.53	19.64	8.59	20.76	30.82	18.72	12.83	0	0	0	34	0	0	0	0
49	33	26.59	40.06	24.89	11.12	23.29	33.35	21.28	15.36	0	0	0	33	0	0	0	0
50	33	18.20	39.91	33.21	24.33	21.57	34.83	29.60	22.83	33	0	0	0	0	0	0	0
51	32	29.28	48.46	33.91	17.47	27.28	40.75	30.79	21.28	0	0	0	32	0	0	0	0
52	26	22.83	36.08	37.84	30.80	26.20	39.46	34.23	27.46	26	0	0	0	0	0	0	0
53	25	12.37	34.08	27.38	20.34	15.74	29.00	23.77	17.00	25	0	0	0	0	0	0	0
54	24	24.65	13.61	27.32	35.55	23.38	15.87	29.34	31.31	0	24	0	0	0	0	0	0
55	21	15.00	15.02	25.03	29.32	17.85	17.28	22.55	25.08	21	0	0	0	0	0	0	0
3575										702	351	276	471	523	456	378	418
										1800				1775			

그림 2. NS의 MS-엑셀 검증
 Fig. 2. MS-Excel verification for NS

NS 알고리즘은 컴퓨터 프로그램의 도움 없이 단지 그림 2의 MS-Excel을 활용하여 n 과 주민 수가 변경되는 어떠한 경우라도 쉽게 적용하여 빠르게 해를 찾을 수 있는 장점이 있다.

NS 알고리즘은 다음과 같이 수행된다.

Step 1. 주어진 망에 대해 그림 4와 같이 ‘ F_A 점포 위치’의 k 개 점포(u)에 대해 노드들 간 거리 데이터를 배치시킨다.

Step 2. [규칙 3]에 의거 $v \in N_G(u)$ 에 대해 ‘ F_B 점포 위치’에 거리 데이터를 넣어 v 의 확보 고객 수를 결정한다. 여기서는 “=IF(MIN(\$C3:\$J3)=C3, \$B3, 0)” 수식을 적용하면 쉽게 구할 수 있다.

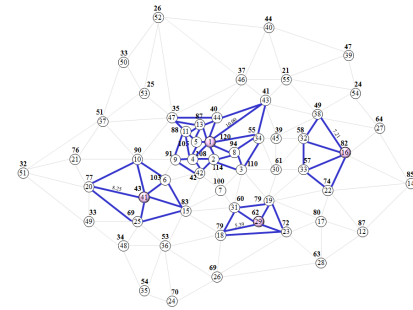
Step 3. ‘ F_A 점포 위치’ 각 $u_i, i=1,2,\dots,k$ 의 $N_G(u_i)$ 의 확보 고객 수들 중 최대로 고객을 확보한 노드들을 F_B 의 후보 노드들 k 개로 결정한다.

Step 4. Step 3에서 구한 F_B 의 후보 노드들 k 개를 그림 4와 같이 ‘ F_B 점포 위치’에 모두 배치하여 F_A 들 중 T 이하가 되는 노드들을 삭제한다.

Step 5. 남은 ‘ F_A 점포 위치’들을 대상으로 최소 고객 확보 노드 u_i 에 대응(인접)하는 ‘ F_B 점포 위치’ $v_i \in N_G(u_i)$ 를 ‘ F_B 점포 위치’에서 삭제(u_i 를 폐쇄시킬 수 있는 후보 위치가 존재하지 않기 때문에)하고, F_A 중 최대 고객을 확보한 노드 u_j 를 폐쇄시키기 위해 $N_G(u_j)$ 를 대상으로 u_j 를 폐쇄시킬 수 있는 노드들 v_j 를 선정하고, ‘ F_B 점포 위치’에 v_j 가 추가될 경우 최대 고객을 확보하는 위치 v_j 를 결정하여 v_i 를 v_j 로 대체한다.

IV. 알고리즘 적용성 평가

본 장에서는 그림 1의 Swain-55 망에서 표 1의 $F_A = \{1,16,29,41\}$ 에 대해 가능한 많은 점포를 폐쇄시키고 F_B 신규 입점 점포들이 시장을 최대한으로 확보할 수 있는 위치를 제한된 NS 알고리즘으로 구하여 본다. 먼저, ‘ F_A 점포 위치’의 k 개 점포(u)에 대해 $v \in N_G(u)$ 인 ‘ F_B 점포 위치’ 후보들은 그림 3과 같이 결정되었다. F_B 각 후보 노드들을 단독으로 개설(‘ F_B 점포 위치’에 배치)할 경우 확보 고객 수를 구하여 ‘ F_A 점포 위치’의 별로 최대 확보 고객 수를 나타낸 그림 2와 같이 $F_B = \{2,6,31,33\}$ 을 얻는다.

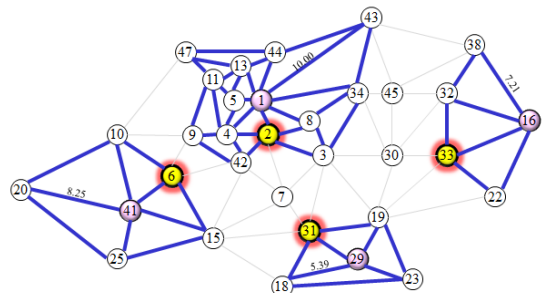


F_A	F_B 후보 위치	개수
1	{2,3,4,5,8,9,11,13,34,42,43,44,47}	13
41	{6,10,15,20,25}	5
29	{18,19,23,31}	4
16	{22,32,33,38}	4

그림 3. F_B 점포 후보 위치

Fig. 3. Candidate stores location of F_B

$F_A = \{1,16,29,41\}$ 와 $F_B = \{2,6,31,33\}$ 을 배치시켜 그림 4와 같이 확보 고객 수를 계산하면 $F_A = \{1,16,29,41\}$ 들 중 1(702), 16(351), 29(276), 41(471)을 얻어 16과 29는 $T=357.5$ 보다 작아 폐쇄된다. 따라서 ‘ F_A 점포 위치’에서 {16, 29}를 삭제하면, $F_A = \{1,41\}$ 와 $F_B = \{2,6,31,33\}$ 가 남게 된다. 이 경우 1(702): 2(523), -:33(807), -:31(654), 41(471):6(418)이 된다.

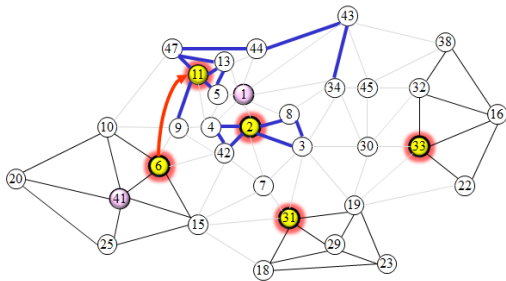


F_A	F_B 후보 위치	선정
1	max{2(753),3(456),4(565),5(640),8(577),9(485),11(649),13(506),34(502),42(648),43(310),44(302),47(170)} = 2(753)	2
41	max{6(460),10(417),15(360),20(269),25(313)} = 6(460)	6
29	max{18(325),19(289),23(371),31(488)} = 31(488)	31
16	max{22(359),32(350),33(456),38(323)} = 33(456)	33

그림 4. F_B 단독 입점시 점포 위치

Fig. 4. Isolated standing stores location of F_B

여기서, 41(47)은 {16,29}가 존재할 때와 삭제된 이후에도 동일한 값으로 $F_B = \{2,6,31,33\}$ 의 영향을 전혀 받지 않아 $F_A = \{41\}$ 점포를 폐쇄시킬 노드는 없음을 알 수 있다. 따라서 이에 대응하는 $F_B = \{6\}$ 을 제외시키고, $F_A = \{1(702)\}$ 를 폐쇄시킬 수 있는 또 다른 노드를 찾고자 Step 5에서 $N_G(1) \setminus N_G(2)$ 인 $\{5,9,11,13,34,43,44,47\}$ 의 8개 노드에 대해 $F_A = \{1,41\}$, $F_B = \{2,6,31,33\}$ 에 대해 F_B 에 개별적으로 추가된 경우 각 노드의 확보 고객 수를 계산하였으며, 그 결과 $F_A = \{1\}$ 을 폐쇄시킬 수 있는 후보로 $\{5,11,13\}$ 을 얻었다. $\{5,11,13\}$ 들 중에서 $F_A = \{41\}$ 의 고객을 최대한 빼앗는 위치를 결정하기 위해, $F_A = \{41\}$, $F_B = \{2,6,31,33\}$ 에 각 노드를 추가한 경우 최종적으로 고객을 추가로 최대한 확보하는 위치로 그림 5와 같이 $\{11\}$ 로 결정되었다.



F_A	F_B 후보 위치 (F_A 확보 고객 수)	선정
1	$F_A = \{1\}$ 폐쇄 후보 위치 $\{5(346.5), 9(702), 11(346.5), 13(287), 34(559), 43(533), 44(400), 47(583)\}$ $= \{5(346.5), 11(346.5), 13(287)\}$ $F_A = \{41\}$ 고객을 최대한 빼앗는 위치 $\max\{5(0), 11(51), 13(0)\} = 11(51)$	11

그림 5. F_B 최종 점포 위치

Fig. 5. Final stores location of F_B

$F_A = \{1\}$ 을 폐쇄시킬 수 있는 후보 $\{5,11,13\}$ 들 중에서 $\{5(346.5), 11(346.5), 13(287)\}$ 로 $F_B = \{13\}$ 노드가 최적의 위치인 것으로 판단되나, $F_B = \{11\}$ 로 결정된 이유는 다음과 같다. 여기서, 13(287)은 $F_A = \{13\}$ 노드를 선택시 $F_A = \{1\}$ 점포가 빼앗기는 고객 수를 의미한다. 이는 $F_A = \{41\}$ 을 고려하지 않고, 또한 $F_A = \{1\}$ 이 폐쇄되면 $F_B = \{2\}$ 가 확보하는 고객 수와 중첩되는 부분이 된다. 따라서 $F_A = \{41\}$, $F_B = \{2,6,31,33\}$ 에 대해 $\{5,11,13\}$ 들 중 $F_A = \{41\}$ 의 고객을 가장 많이 빼앗는 위치로 결정되어야만 하기 때문이다.

Han et al.^[2]의 MICL 알고리즘은 Step 1에서 51개 노드들 중 $F_B = \{2,33,31,49\}$ 를 얻었다(51회). Step 2에서 $F_B = \{33\}$ 에 따라 $F_A = \{16\}$ 을 폐쇄하였으며(1회), Step 3에서 $F_B = \{2,33,31,49\}$, $F_A = \{1,29,41\}$ 에서 $F_A = \{29\}$ 가 폐쇄되었다(1회). 이 결과 $F_B = \{2,33,31,49\}$, $F_A = \{1,41\}$ 로 결정되었으며, 마지막으로 추가된 $F_B = \{49\}$ 를 제외시키고, 가장 많은 고객을 갖고 있는 $F_A = \{1\}$ 을 폐쇄시키기 위해 $\{5,11,13\}$ 중 $F_A = \{1\}$ 고객을 최대한 빼앗는 위치인 $F_B = \{11\}$ 로 결정되었다(48회). 따라서 51+1+48=101회가 수행되었다. 반면에, 본 논문에서 제안된 NS는 그림 4의 Step 2에서 26개 노드를 수행하였으며, 그림 5의 Step 5에서 11개 노드에 대해 알고리즘이 수행되어 37회 수행되었다.

또한, 제안된 NS는 Han et al.^[2]의 MICL에 비해, 최적해를 찾아가는 명확한 규칙을 발견하였다는데 의미가 있다.

V. 결론

본 논문에서는 경쟁업체 F_A 가 k 개 점포로 시장을 독점하고 있는 상황에서, 동종업계 후발업체 F_B 가 동일한 점포 수 k 개를 신규로 개설하여 경쟁업체 점포 고객을 한계수준 T 이하로 폐쇄시키면서 역으로 시장을 독점할 수 있는 점포 위치를 결정하는 문제를 다루었다.

제안된 알고리즘을 독점업체 점포를 폐쇄시키기 위해서는 인접한 위치에 점포를 개설해야 한다는 이론에 기반하여 신규 개설 점포 위치를 독점업체 F_A 의 이웃으로 한정시켜 k 개의 신규 점포 개설 위치 초기 해를 구하였다. 이들 중 F_A 의 점포를 폐쇄시키지 못하면서, F_A 의 확보 고객 수가 가장 적은 위치를 커버하는 신규 개설 점포 위치를 포기하고, 최대 고객 수를 가진 F_A 점포를 폐쇄시키기 위해 최대 고객 수 F_A 점포를 폐쇄시키면서, 포기를 F_A 점포로 부터는 가장 가까워 이 점포의 고객을 가장 많이 빼앗는 위치로 이동시키는 방법을 적용하였다.

이와 같이 명확한 이론에 근거하여, 제안된 알고리즘은 최적 해를 찾아가는 명확한 규칙을 제시하였으며, 기존의 알고리즘들에 비해 해가 존재하는 영역을 최대한 축소시켜 그것도 컴퓨터 프로그램 도움 없이 단지 MS-Excel을 활용해 최적 해를 빠르고 정확하게 구할 수 있어 실제 상황에 누구나 활용할 수 있도록 하였다.

References

- [1] D. Serra, C. ReVelle, and K. Rosing, "Surviving in a Competitive Spatial Market: The Threshold Capture Model", *Journal of Regional Science*, Vol. 39, No. 4, pp. 637-650, Dec. 2002, doi:10.1111/0022-4146.00153
- [2] T. Y. Han, S. S. Jeong, and S. U. Lee, "Sports Facility Location for Surviving in a Competitive Markets," *Journal of the Korean Society of Sports Science*, Vol. 20, No. 4, pp. 437-450, Aug. 2011, uci:G704-001369.2011.20.4.042
- [3] D. Serra and C. ReVelle, "Market Capture by Two Competitors: The Pre-Emptive Location Problem", *Economics Working Paper 39*, Dept. of Economics and Business, Universitat Pompeu Fabra, 1993.
- [4] S. B. Choi, B. G. Kim, and T. Y. Han, "Sports Shop Location Strategy to Maximize the Market Share Using Divide-and-Conquer Strategy," *Journal of the Korean Society of Sports Science*, Vol. 23, No. 2, pp. 309-320, Apr. 2014, uci:G704-001369.2014.23.2.018
- [5] S. U. Lee, Y. S. Lee, S. B. Choi, and T. Y. Han, "Location Strategy of Sports Outlets to Maximize the Market Share," *Journal of the Institute of Internet, Broadcasting and Communication*, Vol. 13, No. 3, pp. 93-101, Jun. 2013, doi:0.7236/JIIBC.2013.13.3.93
- [6] S. U. Lee, "New Store Positioning Algorithm to ensure Competitive Advantage in Monopoly Market," *Journal of the Institute of Internet, Broadcasting and Communication*, Vol. 18, No. 5, pp. 251-257, Oct. 2018, doi:0.7236/JIIBC.2018.18.5.251
- [7] R. Battiti, M. Brunato, and F. Mascia, "Reactive Search and Intelligent Optimization," Springer Verlag, 2008, ISBN: 978-0-387-09623-0.

저자 소개

이 상 윤(정회원)



- 1987년 : 한국항공대학교 항공전자공학과 (학사)
 - 1997년 : 경상대학교 컴퓨터과학과 (석사)
 - 2001년 : 경상대학교 컴퓨터과학과 (박사)
 - 2003년 : 강원도립대학 컴퓨터응용과 전임강사
 - 2004년 ~ 2007년 2월 : 국립 원주대학 여성교양과 조교수
 - 2007년3 ~ 2015년 3월 : 강릉원주대학교 멀티미디어공학과 부교수
 - 2015년 4월 ~ 현재 : 강릉원주대학교 멀티미디어공학과 정교수
- <주관심분야 : 소프트웨어 프로젝트 관리, 개발 방법론, 분석과 설계 방법론, 시험 및 품질보증, 소프트웨어 신뢰성, 최적화 알고리즘>
- E-Mail : sulee@gwnu.ac.kr