韓國컴퓨터情報學會 論文誌
第16卷 第7號, 2011. 7.

2011-16-7-1-9

정점 색칠 문제의 다항시간 알고리즘

이 상 운* 최 명 복**

A Polynomial Time Algorithm for Vertex Coloring Problem

Sang-Un, Lee * Myeong-Bok, Choi **

요 약

본 논문은 지금까지 NP-완전인 난제로 알려진 정점 색칠 문제를 선형시간 복잡도로 해결한 알고리즘을 제안하였다. 제안된 알고리즘은 그래프 $G=(V,E)$의 최소 채색수 $\chi(G)=k$ 를 결정하기 위해 사전에 k 값을 알지 못한다는 가정에 기반하고 있다. 단지 주어진 그래프를 독립집합 $\overline{C}$와 정점 피복 집합 C로 정확히 양분하여 $\overline{C}$에 색을 배정하는 방법을 적용하였다. 독립집합 $\overline{C}$의 원소는 $\delta(G)$인 정점 v가, C의 원소는 정점 v의 인접 정점들 u가 배정된다. 축소된 그래프 C는 다시 $\overline{C}$와 C로 양분되며, 이 과정을 C의 간선이 없을 때까지 수행한다. 26개의 다양한 그래프를 대상으로 제안된 알고리즘을 적용한 결과 정점 v를 선택하는 횟수는 정점의 수 n보다 작은 값을 나타내었으며, $\chi(G)=k$를 찾는데 성공하였다.

▸ 키워드 : 최소 정점 피복, 최대 독립 집합, 최소 차수, 정점 색칠 문제, 채색수

Abstract

The Vertex Coloring Problem hasn't been solved in polynomial time, so this problem has been known as NP-complete. This paper suggests linear time algorithm for Vertex Coloring Problem (VCP). The proposed algorithm is based on assumption that we can't know a priori the minimum chromatic number $\chi(G)=k$ for graph $G=(V,E)$. This algorithm divides Vertices V of graph into two parts as independent sets $\overline{C}$ and cover set C, then assigns the color to $\overline{C}$. The element of independent sets $\overline{C}$ is a vertex v that has minimum degree $\delta(G)$ and the elements of cover set C are the vertices u that is adjacent to v. The reduced graph is divided into independent sets $\overline{C}$ and cover set C again until no edge is in a cover set C. As a result of experiments, this algorithm finds the $\chi(G)=k$ perfectly for 26 Graphs that shows the number of selecting v is less than the number of vertices n.

▸ Keywords : Minimum Vertex Cover (MVC), Maximum Independent Set (MIS), Minimum Degree ($\delta(G)$), Vertex Coloring Problem, Chromatic Number ($\chi(G)$)

• 제1저자 : 이상운*, 교신저자 : 최명복**
• 투고일 : 2011.03.11, 심사일 : 2011.04.08, 게재확정일 : 2011.04.29.
* ** 강릉원주대학교 멀티미디어공학과 (Dept. of Multimedia Science, Gangneung-Wonju National University)

I. 서 론

정점 (Vertices, V) 수 n과 간선 (Edges, e) 수 m으로 구성된 무방향 그래프 (Undirected Graph) $G=(V,E)$, $u,v \in V$, $e=\{u,v\} \in E$에 대해, 정점 색칠 문제 (Vertex Coloring Problem, VCP)는 인접 (동일 간선 공유)한 정점들 간에는 서로 다른 색을 할당하여 그래프의 모든 정점들을 색칠하는데 소요되는 최소의 채색 수 (Chromatic Number, $\chi(G)=k$)를 찾는 문제이다[1]. 이는 기껏해야 k개의 색을 사용함을 의미하며, $k-$coloring이라 부르며, 정점 집합 V를 최대 독립집합 (Maximum Independent Set, MIS, $\overline{C}$) $\overline{C} \leq k$로 분할하는 문제와 같다. $V=\overline{C}+C$로 C는 최소 정점 피복 (Minimum Vertex Cover, MVC) 문제이다. 정점 색칠 문제는 일정 계획 (Scheduling), 컴파일러에서의 프로세서 레지스터 할당 문제, 무선 기지국 사이의 간섭을 제거하기 위한 주파수 할당, 지도의 색칠, 패턴인식 등 다양한 분야에 적용되고 있다.

$\chi(G)=k$를 찾는 방법은 NP-완전 (NP-complete, NPC)로 Karp의 21 NP-완전 문제들 중 하나이다.[2] NPC 문제는 빠른 해법이 알려져 있지 않다는 것이 가장 주목할 만한 특징으로 이들 문제를 빠르게 풀 수 있는지 여부를 결정하는 것이 오늘날 컴퓨터과학 분야에서 주요한 미해결 문제들 중 하나로 남아 있다[2].

사실, 지금까지 최적의 정점 색칠 문제를 찾기 위한 일반적인 해법은 정확한 알고리즘 (Exact Algorithm)인 전수탐색 (Exhaustive Search) 방법으로 채색수 $\chi(G)=k$를 알지 못한 상태에서 모든 정점들을 대상으로 한 정점에 특정한 색을 지정하고 인접한 정점들에 다른 색을 지정하는 경우의 수는 조합으로 n개의 정점에 k개의 색을 배정하는 방법으로 $\binom{n}{k}=\frac{n!}{(n-k)!k!}$의 가능한 경우수를 고려해야 한다. 이 방법은 일반적인 그래프에 대해서는 지수 (Exponential Time) $O(2^n)$시간이 소요되는 NP-완전 문제로 여겨져 왔다. 최적해에 근사한 해를 얻는 근사 알고리즘 (Approximation Algorithm)은 채색수 $\chi(G)=k$가 주어진 상태에서 $\chi(G)=k$를 검증하는 방법으로 4-칼라 평면 그래프 (Planar Graph) 문제는 $O(n^2)$, 보다 빠른 알고리즘이 제안되지 않고 있다 [3-5].

본 논문에서는 채색수 $\chi(G)=k$를 모르는 상태에서 k를 $O(n)$ 수행 복잡도로 근사해 (sub-optimal solution)를 찾는 알고리즘을 제안하여 정점 색칠 문제가 P-문제로 풀 수 있는 방법임을 보인다. 2장에서는 점점색칠에 관련된 기존의 알고리즘을 고찰하고 최소 채색수를 찾는 알고리즘을 제안한다. 3장에서는 제안된 알고리즘을 다양한 그래프에 적용하여 적용성을 검증한다.

II. 정점 색칠 알고리즘

그래프 G에서 각 정점 v에 부속한 (Incident) 간선 수를 차수 또는 결합가 (Degree or Valency, $\deg(v)$ or $d_G(v)$)라 한다. 최대 부속 간선 수를 갖는 정점 v의 차수를 최대 차수 (Maximum Degree) $\Delta(G)$로, 최소 부속 간선 수를 갖는 정점 v의 차수를 최소 차수 (Minimum Degree) $\delta(G)$로 표기한다[6].

$\chi(G)=k$를 찾기 위해 본 논문에서는 정점 집합 V를 k개의 MIS $\overline{C}$로 정확히 분할하고 각 MIS에 다른 색을 배정하는 방법을 적용한다. 참고로, MIS 문제도 NP-완전 문제에 속하고 있다. 대부분의 MVCP 알고리즘들은 k 값을 사전에 알고 있다고 가정하고 C를 찾는 근사 방법을 적용하였다. MVC 문제를 해결하기 위한 다양한 알고리즘들 중에 그림 1의 욕심쟁이 알고리즘[7,8]이 가장 간단한 장점을 갖고 있는 반면 항상 최적해를 찾지 못하는 단점을 갖고 있다[9].

```
C{ | }
while E  |
        1. 각 정점 v의 차수 deg(v) 계산.
        2. 최대 차수 ^(G)인 정점 v  V. 선택. (동일 차수
           정점들이 다수 존재시 임의로 하나 선택).
        C C*{v}.
        EEr{e E: v e}.
Return C
MVC : C.
MIS : C̄= V-C.
```

그림 1. MVC 욕심쟁이 알고리즘
Fig. 1. MVC Greedy Algorithm

반면에, 제안 알고리즘은 MVC인 C의 원소 개수 c와 MIS인 $\overline{C}$의 원소개수 k를 사전에 알지 못한다는 가정에 기반하고 $|\overline{C}|$를 최소로 하는 k를 찾아 $\overline{C_i},(i=2,\cdots,k)$에 서로 다른 색을 배정하는 방법을 적용하였다. k를 알고 찾을 경우 이 문제는 NP-완전 문제의 검증에 다항시간이 소요되는 경우로 NP-완전 문제에 대한 알고리즘이 될 수 없으며, k를 알지 못한 상태에서 k를 구하는 문제는 NP-완전 문제의 해를 찾는 방법이다. 따라서 채색수 문제는 P-문제로 근사해를 구

하는 방법을 제시한다. 또한, 임의의 정점을 선택하는 기존의 MVCP 알고리즘과 달리 $\delta(G)$인 정점 v를 선택하여 색을 배정한다. 왜냐하면 최대 차수 $\Delta(G)$ 정점에 색을 배정하는 방법을 적용한 결과 k의 최소값을 찾지 못하는 경우가 있어 이를 역으로 적용하였다. 제안 알고리즘은 그림 2와 같이 최소 차수 $\delta(G)$인 정점 v를 $\overline{C_k}$에, v 인접 정점들 u를 C_k에 배정하는 방법을 적용한다. 제안된 알고리즘은 두 정점 간에는 간선이 존재하지 않는 정점들의 집합인 독립집합에 대해 동일한 색을 배정할 수 있다는 개념을 적용하였으며, 이 때 최대 독립집합을 찾을 수 있는 방법이다.

그래프 $G=(V,E)$에 대해, 첫 번째 단계에서 주어진 정점들 $C_0=V$를 C_1와 $\overline{C_1}$로 정확히 양분 (Bipartite)하여 $\overline{C_1}$의 정점들을 동일한 색으로 설정한다.

k : 채색 수 (Chromatic Number, $\chi(G)$)
$\overline{C_k}$: k^{th} 색의 최대독립집합 (MIS), C_k : k^{th} 색의 최소정점피복 (MVC)
$C_0=V, E_1=E, C_1 \leftarrow \{\phi\}$
C_0 각 정점 v에 대한 $\deg(v)$ 계산.

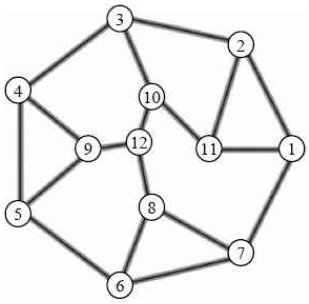
for $k=1$ to $E_k=\varnothing$
    while $E_k \neq \varnothing$
        E_k 각 정점 v의 $\deg(v')$ 계산.
        if $\delta(G)=0$ 정점 v 존재 then $\overline{C_k} \leftarrow \overline{C_k} \cup \{v\}$.
            /* (G) = 0정점은 독립집합으로 기존 색 배정 가능
        if
((G) ` 0)) ((G) ⌌ = 1) then $\overline{C_k} \leftarrow \overline{C_k} \cup \{v\}$
            /* 기존 색 배정 가능
        else if ((G) ` 0)) ((G) ⌌ e2) then
            /* (G) 의 deg(v')가 동일한 정점이 2개 이상 존재
            if deg(v)=모두 동일 then 임의 정점 v 선택
            else if deg(v)=모두 다름 then 최소 deg(v) 정점 v 선택
            end if
            $\overline{C}$/k⌌ $\overline{C}$/k⌌*{v} /* MIS 정점 집합.
        end if
        정점 v의 모든 인접 정점 u Ck-1 선택.
        Ck⌌ Ck⌌*{u}. /* MVC 정점 집합.
        Ek/Ekr⌌{u Ek}. /* u는 v와 다른 색을 배정해야하기 때문에 부속 간선 삭제.
    Return Ck
    $\overline{C}$k⌌= Ck-1r⌌Ck.
    Ek+1⌌=e Ck.
    Ck 각 정점 v의 차수 deg(v) 계산.
    if Ek+1⌌={|} then $\overline{C}$k+1⌌= Ck, 알고리즘 종료.
Next k
²(G)=k.

그림 2. 정점 색칠 알고리즘
Fig. 2. Vertex coloring algorithm.

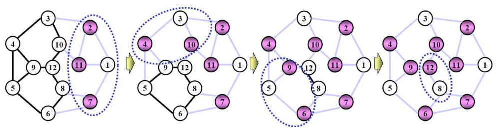
첫 번째 단계를 수행하면 원래 그래프의 정점들 $C_0=V$에서 이미 하나의 색이 배정된 $\overline{C_1}$을 빼면 나머지 정점들이 다른 색을 배정할 대상이 된다. 즉, $C_1=V \setminus \overline{C}$를 대상으로 두 번째 색을 배정한다. 이를 축소된 그래프라 한다. 축소된 그래프의 정점들 C_1의 간선들을 대상으로 다시 C_2와 $\overline{C_2}$로 정확히 양분하여 $\overline{C_2}$에 2번째 색을 지정한다. 이러한 과정을 축소된 그래프 C_{k-1}의 간선들이 없을 때까지 k회 수행된다. 결국, 제안 알고리즘은 $O(kc) \leq O(n)$회의 수행 복잡도로 $\chi(G)=k$를 얻을 수 있다. 여기서 c는 $\delta(G)$ 정점을 선택하는 횟수이다.

그림 3의 G_1 그래프를 대상으로 제안 알고리즘을 적용하여 보자. G_1 그래프는 Dharwadler[10]와 Mathworld[11]에서 인용되었다.

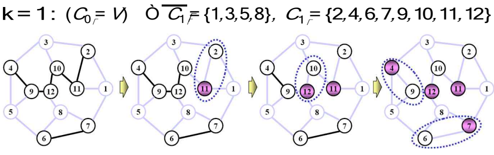
그림 3의 G_1 그래프에 정점 색칠 알고리즘을 적용한 결과는 그림 4에 제시되어 있다. 제안 알고리즘은 $n=12$인 G_1 그래프에 대해 n 보다 적은 7회 (4회+3회)를 수행하여 정확히 $\chi(G)=3$인 3-칼라로 색을 배정하였다.



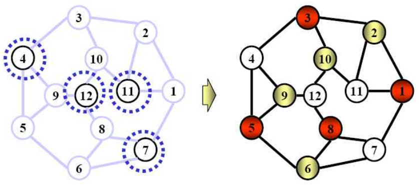
(n= 12,²(G)= 3)
그림 3 G_1 그래프
Fig. 3. G_1 graph



k=1: (C0⌌= V) Ò $\overline{C}$1⌌= {1,3,5,8}, C1⌌= {2,4,6,7,9,10,11,12}



k=2: C1⌌= {2,4,6,7,9,10,11,12} Ò $\overline{C}$2⌌= {2,6,9,10},
C2⌌= {4,7,11,12}



k = 3: C_2 = {4, 7, 11, 12} Ò $\overline{C_3}$ = {4, 7, 11, 12} Ó ²(G) = 3

k	(G)	deg(v)	deg(v')	MIS (v)	MVC (u)	$\overline{C_k}$	C_k
1	{1, 2, ï, 12}	3	3	{1}	{2, 7, 1	{1}	{2, 7, 11}
	{3, 6, 8, 10}	3	2	{3}	{4, 10	{1, 3}	{2, 4, 7, 10, 11}
	{5, 6, 8, 9, 12}	3	2	{5}	{6, 9}	{1, 3, 5}	{2, 4, 6, 7, 9, 10, 11}
	{8, 12}	3	1	{8}	{12}	{1, 3, 5, 8}	{2, 4, 6, 7, 9, 10, 11, 12}
2	{2, 4, 6, 7}	1	1	{2}	{11}	{2}	{11}
	{4,6,7} {10}	1 2	1 1	{10}	{12}	{2, 10}	{11, 12}
	{4,6,7} {9}	1 2	1 1	{9}	{4}	{2, 9, 10}	{4, 11, 12}
	{6, 7}	1	1	{6}	{7}	{2, 6, 9, 10}	{4, 7, 11, 12}
3	{4, 7, 11, 12}	0	0		{ㅣ}	{4, 7, 11, 12}	-

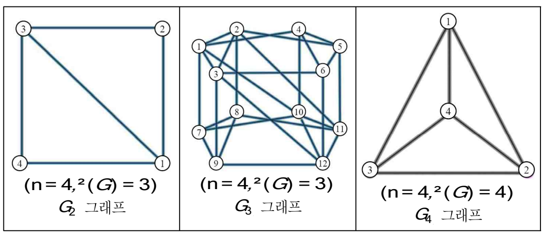
그림 4. G_1 그래프의 정점 색칠 알고리즘

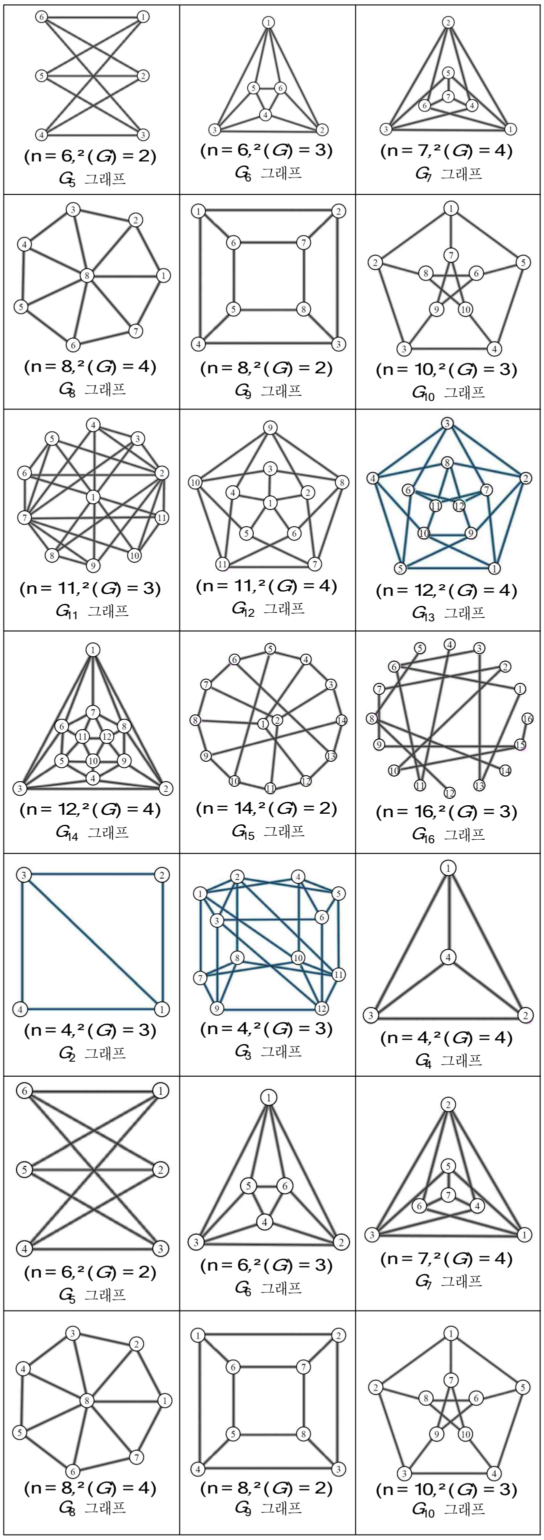
Fig. 4. Vertex coloring algorithm for G_1 graph

III. 실험 및 결과 분석

1. 실험 데이터

본 절에서는 가능한 다양한 형태의 그래프들을 대상으로 제안된 정점 색칠 알고리즘이 얼마나 빠르고 정확히 최적해를 구하는지 고찰해 본다. 이를 위해 실험 데이터는 그림 5와 같이 다양한 형태의 일반 그래프를 대상으로 선정되었다. $G_2 \sim G_4, G_6, G_7, G_{11} \sim G_{17}$은 Dharwadker[10]에서, G_5, G_8은 Dharwadler[10]와 Mathworld[11]에서, G_{10}, G_{18}은 Wikipedia[1]와 Mathworld[11]에서, G_{19}는 Hermann과 Hertz[12]에서, G_{20}은 Naserasr와 Tarchf[13]에서, G_{21}은 Wikipedia[14]에서, G_{22}, G_{23}은 Thomas[15]에서, G_{24}는 Wikipedia[16]에서 인용되었다. 실험 방법은 제안된 알고리즘을 구현한 프로그램을 적용하지 않고, 선정된 다양한 그래프에 대해 제안된 알고리즘을 적용하여 수기식으로 검증하였다.





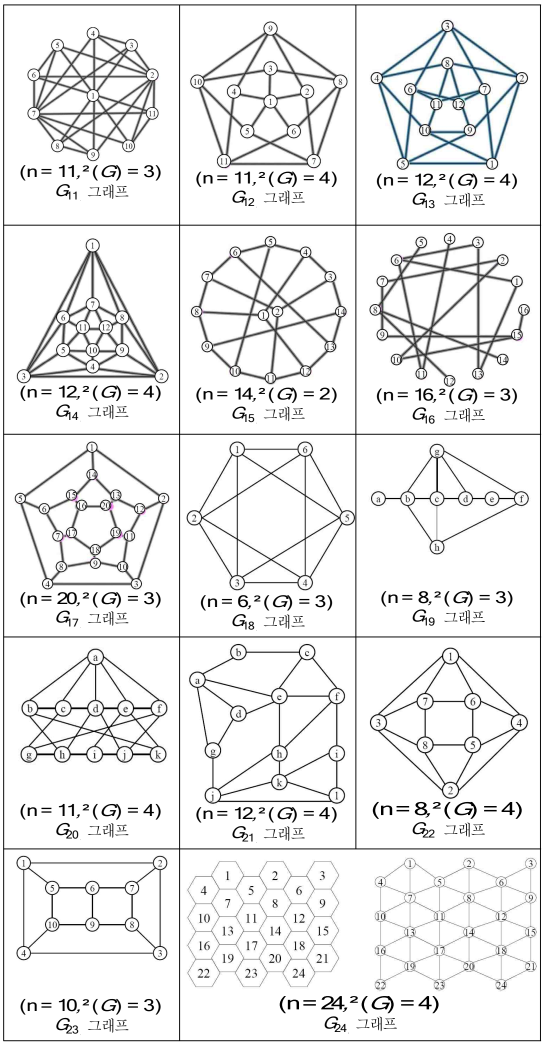


그림 5. 일반 그래프
Fig. 5. General graphs

2. 알고리즘 적용

그림 5의 23개 일반 그래프에 대해 정점 색칠 알고리즘을 적용한 결과는 표 1에 제시되어 있다.

표 1. 일반 그래프의 정점 색칠 알고리즘
Table 1. Vertex coloring algorithm for general graphs

(a) G_2 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{2,4}	{2}	{1,3}	{2,4}	{1,3}
2	{1,3}	{1}	{3}	{1}	{3}
3	{3}	{3}	–	{3}	–
$\chi(G)=3$		$\overline{C_1}=\{2,4\}, \overline{C_2}=\{1\}, \overline{C_3}=\{3\}$			

(b) G_3 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{4,5,6,7,8,9} {7} {2,3,11,12}	{4} {7} {2}	{1,5,6,10} {8,9} {3,11}	{4} {4,7} {2,4,7,12}	{1,5,6,10} {1,5,6,8,9,10} {1,3,5,6,8,9,10,11}
2	{1,5,6,8,9,10} {6,9} {9,11}	{1} {6} {11}	{3,10} {5} {8}	{1} {1,6} {1,6,9,11}	{3,10} {3,5,10} {3,5,8,10}
3	{3,5,8,10}	{3,5,8,10}	–	{3,5,8,10}	–
$\chi(G)=3$		$\overline{C_1}=\{2,4,7,12\}, \overline{C_2}=\{1,6,9,11\}, \overline{C_3}=\{3,5,8,10\}$			

(c) G_4 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,3,4}	{1}	{2,3,4}	{1}	{2,3,4}
2	{2,3,4}	{2}	{3,4}	{2}	{3,4}
3	{3,4}	{3}	{4}	{3}	{4}
4	{4}	{4}	–	{4}	–
$\chi(G)=4$		$\overline{C_1}=\{1\}, \overline{C_2}=\{2\}, \overline{C_3}=\{3\}, \overline{C_4}=\{4\}$			

(d) G_5 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,6}	{1}	{4,5,6}	{1,2,3}	{4,5,6}
2	{4,5,6}	{4,5,6}	–	{4,5,6}	–
$\chi(G)=2$		$\overline{C_1}=\{1,2,3\}, \overline{C_2}=\{4,5,6\}$			

(e) G_6 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,6}	{1}	{2,3,5,6}	{1,4}	{2,3,5,6}
2	{2,3,5,6}	{2}	{3,6}	{2,5}	{3,6}
3	{3,6}	{3,6}	–	{3,6}	–
$\chi(G)=3$		$\overline{C_1}=\{1,4\}, \overline{C_2}=\{2,5\}, \overline{C_3}=\{3,6\}$			

(f) G_7 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{4,5,6,7} {5,6}	{4} {5}	{2,3,7} {1}	{4} {4,5,6}	{2,3,7} {1,2,3,7}
2	{7} {1,2,3}	{7} {1}	– {2,3}	{7} {1,7}	– {2,3}
3	{2,3}	{2}	{3}	{2}	{3}
4	{3}	{3}	–	{3}	–
$\chi(G)=4$		$\overline{C_1}=\{4,5,6\}, \overline{C_2}=\{1,7\}, \overline{C_3}=\{2\}, \overline{C_4}=\{3\}$			

(g) G_8 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,7} {3,6} {5,6}	{1} {3} {5}	{2,7,8} {4} {6}	{1} {1,3} {1,3,5}	{2,7,8} {2,4,7,8} {2,4,6,7,8}
2	{2,4} {6,7}	{2} {6}	{8} {7}	{2,4} {2,4,6}	{8} {7,8}
3	{7,8}	{7}	{8}	{7}	{8}
4	{8}	{8}	–	{8}	–
$\chi(G)=4$		$\overline{C_1}=\{1,3,5\}, \overline{C_2}=\{2,4,6\}, \overline{C_3}=\{7\}, \overline{C_4}=\{8\}$			

(h) G_9 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,8} {3,5,7}	{1} {3}	{2,4,6} {8}	{1} {1,3,5,7}	{2,4,6} {2,4,6,8}
2	{2,4,6,8}	{2,4,6,8}	–	{2,4,6,8}	–
$\chi(G)=2$	$\overline{C_1}=\{1,3,5,7\}, \overline{C_2}=\{2,4,6,8\}$				

(i) G_{10} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,10} {3,4,6,8,9,10} {6,10}	{1} {3} {6}	{2,5,7} {4,9} {8}	{1} {1,3} {1,3,6,10}	{2,5,7} {2,4,5,7,9} {2,4,5,7,8,9}
2	{2,4,5,7,8,9} {4,5,7,9} {7,9}	{2} {4} {7}	{8} {5} {9}	{2} {2,4} {2,4,7}	{8} {5,8} {5,8,9}
3	{5,8,9}	{5,8,9}	–	{5,8,9}	–
$\chi(G)=3$	$\overline{C_1}=\{1,3,6,10\}, \overline{C_2}=\{2,4,7\}, \overline{C_3}=\{5,8,9\}$				

(j) G_{11} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{3,⋯,6,8,⋯,11} {5,6,8,9,10,11} {8,9,10,11} {10,11}	{3} {5} {8} {10}	{1,2,4,7} {6} {9} {11}	{3} {3,5} {3,5,8} {3,5,8,10}	{1,2,4,7} {1,2,4,6,7} {1,2,4,6,7,9} {1,2,4,6,7,9,11}
2	{4,6,9,11}	{4}	{1,2,7}	{4,6,9,11}	{1,2,7}
3	{1,2,7}	{1,2,7}	–	{1,2,7}	–
$\chi(G)=3$	$\overline{C_1}=\{3,5,8,10\}, \overline{C_2}=\{4,6,9,11\}, \overline{C_3}=\{1,2,7\}$				

(k) G_{12} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{2,3,4,5,6} {4,5} {5,6} {3,6}	{2} {4} {5} {3}	{1,7,9} {11} {10} {8}	{2} {2,4} {2,4,5} {2,3,4,5,6}	{1,7,9} {1,7,9,11} {1,7,9,10,11} {1,7,8,9,10,11}
2	{1} {7,8,9,10,11} {9,10}	{1} {7} {9}	– {8,11} {10}	{1} {1,7} {1,7,9}	– {8,11} {8,10,11}
3	{8} {10,11}	{8} {10}	– {11}	{8} {8,10}	– {11}
4	{11}	{11}	–	{11}	–
$\chi(G)=4$	$\overline{C_1}=\{2,3,4,5,6\}, \overline{C_2}=\{1,7,9\}, \overline{C_3}=\{8,10\}, \overline{C_4}=\{11\}$				

(l) G_{13} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,12} {9} {3,4,6,8,11} {8,11}	{1} {9} {3} {8}	{2,5,7,10} {12} {4,6} {11}	{1} {1,9} {1,3,9} {1,3,8,9}	{2,5,7,10} {2,5,7,10,12} {2,4,5,6,7,10,12} {2,4,5,6,7,10,11,12}
2	{2} {4,5,7,10} {6,7,11,12}	{2} {4} {6}	– {5,10} {11,12}	{2} {2,4} {2,4,6,7}	– {5,10} {5,10,11,12}
3	{5,12} {10,11}	{5,12} {10}	– {11}	{5,12} {5,10,12}	– {11}
4	{11}	{11}	–	{11}	–
$\chi(G)=4$	$\overline{C_1}=\{1,3,8,9\}, \overline{C_2}=\{2,4,6,7\}, \overline{C_3}=\{5,10,12\}, \overline{C_4}=\{11\}$				

(m) G_{14} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,12} {4,5,9,11,12} {11,12}	{1} {4} {11}	{2,3,6,7,8} {5,9,10} {12}	{1} {1,4} {1,4,11}	{2,3,6,7,8} {2,3,5,6,7,8,9,10} {2,3,5,6,7,8,9,10,12}
2	{2,3,5,6,7,8,10} {5,6,7,10,12} {5,6}	{2} {12} {5}	{3,8,9} {7,10} {6}	{2} {2,12} {2,5,12}	{3,8,9} {3,7,8,9,10} {3,6,7,8,9,10}
3	{3,10} {7,10} {9,10}	{3} {7} {9}	{6} {8} {10}	{3} {3,7} {3,7,9}	{6} {6,8} {6,8,10}
4	{6,8,10}	{6,8,10}	–	{6,8,10}	–
$\chi(G)=4$	$\overline{C_1}=\{1,4,11\}, \overline{C_2}=\{2,5,12\}, \overline{C_3}=\{3,7,9\}, \overline{C_4}=\{6,8,10\}$				

(n) G_{15} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,14} {3,5,7,9,11,13} {7,9,11,13} {5,9,10,11}	{1} {3} {7} {5}	{4,8,12 {2,14} {6} {10}	{1} {1,3} {1,3,7,13} {1,3,5,7,9,11,13}	{4,8,12} {2,4,8,12,14} {2,4,6,8,12,14} {2,4,6,8,10,12,14}
2	{2,4,6,8,10,12,14}	{2,4,6,8,10,12,14}	–	{2,4,6,8,10,12,14}	–
$\chi(G)=2$	$\overline{C_1}=\{1,3,5,7,13\}, \overline{C_2}=\{2,4,6,8,10,12,14\}$				

(o) G_{16} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{4,5,12,14,16} {5,12,14,16} {16} {9,10} {10} {1,3,6,13}	{4} {5} {16} {9} {10} {1}	{11} {8} {15} {7} {2} {6,13}	{4} {4,5,12,14} {4,5,12,14,16} {4,5,9,12,14,16} {4,5,9,10,12,14,16} {1,3,4,5,9,10,12,14,16}	{11} {8,11} {8,11,15} {7,8,11,15} {2,7,8,11,15} {2,6,7,8,11,13,15}
2	{8,13,15} {2,6,7,11} {6,11}	{8,13,15} {2} {6}	– {7} {11}	{8,13,15} {2,8,13,15} {2,6,8,13,15}	– {7} {7,11}
3	{7,11}	{7,11}	–	{7,11}	–
$\chi(G)=3$	$\overline{C_1}=\{1,3,4,5,9,10,12,14,16\}, \overline{C_2}=\{2,6,8,13,15\}, \overline{C_3}=\{7,11\}$				

(p) G_{17} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,⋯,20} {3,4,6,12,13,15} {6,8,9,11,12,13,15} {8} {11,12,13,16,17,18} {13,18} {16,18}	{1} {3} {6} {8} {11} {13} {16}	{2,5,14} {4,10} {7,15} {9} {12,19} {20} {17}	{1} {1,3} {1,3,6} {1,3,6,8} {1,3,6,8,11} {1,3,6,8,11,13} {1,3,6,8,11,13,16,18}	{2,5,14} {2,4,5,10,14} {2,4,5,7,10,14,15} {2,4,5,7,9,10,14,15} {2,4,5,7,9,10,12,14,15,19} {2,4,5,7,9,10,12,14,15,19,20} {2,4,5,7,9,10,12,14,15,17,19,20}
2	{2,4,5,7,9,10,12,14,15,17,19,20} {4,5,7,9,10,14,15,17,19,20} {7,9,10,14,15,17,19,20} {9,10,14,15,19,20} {14,15,19,20} {19,20}	{2} {4} {7} {9} {14} {19}	{12} {5} {17} {10} {15} {20}	{2} {2,4} {2,4,7} {2,4,7,9} {2,4,7,9,14} {2,4,7,9,14,19}	{12} {5,12} {5,12,17} {5,10,12,17} {5,10,12,15,17} {5,10,12,15,17,20}
3	{5,10,12,15,17,20}	{5,10,12,15,17,20}	–	{5,10,12,15,17,20}	–
$\chi(G)=3$	$\overline{C_1}=\{1,3,6,8,11,13,16,18\}, \overline{C_2}=\{2,4,7,9,14,19\}, \overline{C_3}=\{5,10,12,15,17,20\}$				

(q) G_{18} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	{1,2,3,4,5,6}	{1}	{2,3,5,6}	{1,4}	{2,3,5,6}
2	{2,3,5,6}	{2}	{3,6}	{2,5}	{3,6}
3	{3,6}	{3,6}	–	{3,6}	–
$\chi(G)=3$	$\overline{C_1}=\{1,4\}, \overline{C_2}=\{2,5\}, \overline{C_3}=\{3,6\}$				

(r) G_{19} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	$\{a\}$ $\{e\}$ $\{g,h\}$	$\{a\}$ $\{e\}$ $\{g\}$	$\{b\}$ $\{d,f\}$ $\{c\}$	$\{a\}$ $\{a,e\}$ $\{a,e,g,h\}$	$\{b\}$ $\{b,d,f\}$ $\{b,c,d,f\}$
2	$\{f\}$ $\{b,d\}$	$\{f\}$ $\{b\}$	– $\{c\}$	$\{f\}$ $\{f,b,d\}$	– $\{c\}$
3	$\{c\}$	$\{c\}$	–	$\{c\}$	–
$\chi(G)=3$		$\overline{C_1}=\{a,e,g,h\},\overline{C_2}=\{f,b,d\},\overline{C_3}=\{c\}$			

(s) G_{20} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	$\{g,k\}$ $\{b,i\}$ $\{d,e,i,j\}$	$\{g\}$ $\{b\}$ $\{d\}$	$\{c,f,h\}$ $\{a,k\}$ $\{e,j\}$	$\{g\}$ $\{b,g\}$ $\{b,d,g,i\}$	$\{c,f,h\}$ $\{a,c,f,h,k\}$ $\{a,c,e,f,h,j,k\}$
2	$\{h\}$ $\{c\}$ $\{e,f,j,k\}$	$\{h\}$ $\{c\}$ $\{e\}$	– $\{a\}$ $\{f,k\}$	$\{h\}$ $\{c,h\}$ $\{c,e,h,j\}$	– $\{a\}$ $\{a,f,k\}$
3	$\{k\}$ $\{a,f\}$	$\{k\}$ $\{a\}$	– $\{f\}$	$\{k\}$ $\{a,k\}$	– $\{f\}$
4	$\{f\}$	$\{f\}$	–	$\{f\}$	–
$\chi(G)=4$		$\overline{C_1}=\{b,d,g,i\},\overline{C_2}=\{c,e,h,j\},\overline{C_3}=\{a,k\},\overline{C_4}=\{f\}$			

(t) G_{21} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	$\{b\}$ $\{d,g\}$ $\{f\}$ $\{j,k,l\}$	$\{b\}$ $\{d\}$ $\{f\}$ $\{j\}$	$\{a,c\}$ $\{e,g\}$ $\{h,i\}$ $\{k,l\}$	$\{b\}$ $\{b,d\}$ $\{b,d,f\}$ $\{b,d,f,j\}$	$\{a,c\}$ $\{a,c,e,g\}$ $\{a,c,e,g,h,i\}$ $\{a,c,e,g,h,i,k,l\}$
2	$\{c,g\}$ $\{a,g,h\}$ $\{h\}$ $\{i,l\}$	$\{c\}$ $\{a\}$ $\{h\}$ $\{i\}$	$\{e\}$ $\{g\}$ $\{k\}$ $\{l\}$	$\{c\}$ $\{a,c\}$ $\{a,c,h\}$ $\{a,c,h,i\}$	$\{e\}$ $\{e,g\}$ $\{e,g,k\}$ $\{e,g,k,l\}$
3	$\{e,g\}$ $\{k,l\}$	$\{e,g\}$ $\{k\}$	– $\{l\}$	$\{e,g\}$ $\{e,g,k\}$	– $\{l\}$
4	$\{l\}$	$\{l\}$	–	$\{l\}$	–
$\chi(G)=4$		$\overline{C_1}=\{b,d,f,j\},\overline{C_2}=\{a,c,h,i\},\overline{C_3}=\{e,g,k\},\overline{C_4}=\{l\}$			

(u) G_{22} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	$\{1,2,3,4,5,6,7,8\}$ $\{2,5,8\}$	$\{1\}$ $\{2\}$	$\{3,4,6,7\}$ $\{5,8\}$	$\{1\}$ $\{1,2\}$	$\{3,4,6,7\}$ $\{3,4,5,6,7,8\}$
2	$\{3,4\}$ $\{4,5,6\}$	$\{3\}$ $\{5\}$	$\{7,8\}$ $\{4,6\}$	$\{3\}$ $\{3,5\}$	$\{7,8\}$ $\{4,6,7,8\}$
3	$\{4,8\}$ $\{7,8\}$	$\{4\}$ $\{7\}$	$\{6\}$ $\{8\}$	$\{4\}$ $\{4,7\}$	$\{6\}$ $\{6,8\}$
4	$\{6,8\}$	$\{6,8\}$	–	$\{6,8\}$	–
$\chi(G)=4$		$\overline{C_1}=\{1,2\},\overline{C_2}=\{3,5\},\overline{C_3}=\{4,7\},\overline{C_4}=\{6,8\}$			

(v) G_{23} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	$\{1,2,\cdots,10\}$ $\{3,10\}$ $\{7,10\}$ $\{9,10\}$	$\{1\}$ $\{3\}$ $\{7\}$ $\{9\}$	$\{2,4,5\}$ $\{8\}$ $\{6\}$ $\{10\}$	$\{1\}$ $\{1,3\}$ $\{1,3,7\}$ $\{1,3,7,9\}$	$\{2,4,5\}$ $\{2,4,5,8\}$ $\{2,4,5,6,8\}$ $\{2,4,5,6,8,10\}$
2	$\{2,8\}$ $\{4,6\}$ $\{5,6\}$	$\{2,8\}$ $\{4\}$ $\{5\}$	– $\{10\}$ $\{6\}$	$\{2,8\}$ $\{2,4,8\}$ $\{2,4,5,8\}$	– $\{10\}$ $\{6,10\}$
3	$\{6,10\}$	$\{6,10\}$	–	$\{6,10\}$	–
$\chi(G)=3$		$\overline{C_1}=\{1,3,7,9\},\overline{C_2}=\{2,4,5,8\},\overline{C_3}=\{4,7\},\overline{C_4}=\{6,10\}$			

(w) G_{24} 그래프

k	$\delta(G)$	v	u	$\overline{C_k}$	C_k
1	$\{3,22\}$ $\{2,22\}$ $\{1,22\}$ $\{10,22\}$ $\{22\}$ $\{11,23\}$ $\{23\}$ $\{12,24\}$ $\{21,24\}$	$\{3\}$ $\{2\}$ $\{1\}$ $\{10\}$ $\{22\}$ $\{11\}$ $\{23\}$ $\{12\}$ $\{21\}$	$\{6,9\}$ $\{5,8\}$ $\{4,7\}$ $\{13,16\}$ $\{19\}$ $\{14,17\}$ $\{20\}$ $\{15,18\}$ $\{24\}$	$\{3\}$ $\{2,3\}$ $\{1,2,3\}$ $\{1,2,3,10\}$ $\{1,2,3,10,22\}$ $\{1,2,3,10,11,22\}$ $\{1,2,3,10,11,22,23\}$ $\{1,2,3,10,11,12,22,23\}$ $\{1,2,3,10,11,12,21,22,23\}$	$\{6,9\}$ $\{5,6,8,9\}$ $\{4,5,6,7,8,9\}$ $\{4,5,6,7,8,9,13,16\}$ $\{4,5,6,7,8,9,13,16,19\}$ $\{4,5,6,7,8,9,13,14,16,17,19\}$ $\{4,5,6,7,8,9,13,14,16,17,19,20\}$ $\{4,5,6,7,8,9,13,14,15,16,17,18,19,20\}$ $\{4,5,6,7,8,9,13,14,15,16,17,18,19,20,24\}$
2	$\{4\}$ $\{5\}$ $\{6\}$ $\{15\}$ $\{24\}$ $\{14\}$ $\{13,16,19\}$	$\{4\}$ $\{5\}$ $\{6\}$ $\{15\}$ $\{24\}$ $\{14\}$ $\{13\}$	$\{7\}$ $\{8\}$ $\{9\}$ $\{18\}$ $\{20\}$ $\{17\}$ $\{16,19\}$	$\{4\}$ $\{4,5\}$ $\{4,5,6\}$ $\{4,5,6,15\}$ $\{4,5,6,15,24\}$ $\{4,5,6,14,15,24\}$ $\{4,5,6,13,14,15,24\}$	$\{7\}$ $\{7,8\}$ $\{7,8,9\}$ $\{7,8,9,18\}$ $\{7,8,9,18,20\}$ $\{7,8,9,17,18,20\}$ $\{7,8,9,16,17,18,19,20\}$
3	$\{7,8,9\}$ $\{16,18\}$ $\{17,18\}$	$\{7,8,9\}$ $\{16\}$ $\{17\}$	– $\{19\}$ $\{20\}$	$\{7,8,9\}$ $\{7,8,9,16\}$ $\{7,8,9,16,17,18\}$	– $\{19\}$ $\{19,20\}$
4	$\{19,20\}$	$\{19,20\}$	–	$\{19,20\}$	–
$\chi(G)=4$	$\overline{C_1}=\{1,2,3,10,11,12,21,22,23\},\overline{C_2}=\{4,5,6,13,14,15,24\},\overline{C_3}=\{7,8,9,16,17,18\},\overline{C_4}=\{19,20\}$				

3. 알고리즘 적용 결과 분석

본 논문에서 적용된 24개 그래프에 대해 정점 색칠 알고리즘의 성능을 분석하여 본다. 알고리즘 성능 분석 결과는 표 2에 제시되어 있다. 제안된 알고리즘은 24개 데이터 모두에서 최적해를 찾는데 성공하였다. 또한, v 정점을 선택하는 알고리즘 수행 횟수는 n보다 작거나 같아 $O(kc) \leq O(n)$의 복잡도임을 알 수 있다. 15개의 평면 그래프에서는 De Morgan의 ²(*G*) d 4의 법칙[1]이 성립함을 알 수 있다.

표 2. 알고리즘 성능 분석
Table 2. Analysis of algorithm performance

그래프	n	평면 그래프	$\chi(G)$	정점 색칠 알고리즘 수행횟수	정점 색칠 알고리즘 최적해
G_1	12	Ë	3	11	Success
G_2	4	Ë	3	3	Success
G_3	4	×	4	7	Success
G_4	4	Ë	2	4	Success
G_5	6	×	3	2	Success
G_6	6	Ë	4	3	Success
G_7	7	×	4	6	Success
G_8	8	Ë	2	7	Success
G_9	8	Ë	3	3	Success
G_{10}	10	×	3	7	Success
G_{11}	11	×	4	6	Success
G_{12}	11	×	4	10	Success
G_{13}	12	×	4	10	Success
G_{14}	12	Ë	2	10	Success
G_{15}	14	×	3	5	Success
G_{16}	16	×	3	10	Success
G_{17}	20	E	3	14	Success
G_{18}	6	×	3	3	Success
G_{19}	8	Ë	4	6	Success
G_{20}	11	×	4	9	Success
G_{21}	12	Ë	4	11	Success
G_{22}	8	E	3	7	Success
G_{23}	10	Ë	4	8	Success
G_{24}	24	Ë	4	20	Success

IV. 결론

본 논문은 정점 색칠 문제를 선형시간 복잡도로 해결하는 알고리즘을 제안하였다. 제안된 알고리즘은 그래프의 최소 정점 채색수 $\chi(G)=k$를 결정하기 위해 최대 독립집합 $\overline{C}$의 개수 k를 사전에 알지 못한다는 가정에 기반한다. 주어진 그래프의$\delta(G)$ 정점 v를 $\overline{C}$에, v에 인접한 정점들 u를 정점피복 집합 C에 포함시켜 그래프를 정확히 양분한다. 양분된 그래프의 $\overline{C}$에 색을 배정한다. 축소된 그래프 C에 대해 다시 독립집합과 정점피복 집합으로 양분하여 색을 배정한다. 이 과정을 축소된 그래프 C의 모든 정점들을 연결하는 간선이 없을 때까지 수행한다.

제안된 알고리즘을 24개의 일반 그래프와 인도와 미국 지도를 대상으로 적용 결과 수행 횟수는 정점 수 n보다 작은 결과를 얻었으며, 채색수 $\chi(G)=k$를 얻는데 성공하였다.

본 논문에서 제안된 정점 색칠 알고리즘은 지금까지 NP-완전인 난제로 여겨졌던 정점 색칠 문제를 선형 시간으로 근사해를 구하는 방법이 존재함을 보였다. 결국, 본 알고리즘은 정점 색칠 문제의 근사해를 도출하는 빠른 알고리즘으로 적용할 수 있을 것이다.

제안된 알고리즘은 n이 큰 경우에도 k를 구할 수 있는지는 증명되지 않았다. 이 경우는 실제 지도에 4가지 색을 칠할 수 있는지에 대한 4색 정리 (Four Color Theorem) 문제에 대해 추후 연구할 계획이다.

참고문헌

[1] Wikipedia, "Graph Coloring," http://en.wikipedia.org/wiki/ Graph_Coloring, Wikimedia Foundation Inc., 2008.

[2] Wikipedia, "NP-Complete," http://en.wikipedia.org /wiki/ NP-Complete, Wikimedia Foundation Inc., 2008.

[3] Wikipedia, "Four Color Theorem," http://en.wik ipedia.org/ wiki/Four_Color_Theorem, Wikimedia Foun dation Inc., 2008.

[4] J. M. Byskov, "Chromatic Number in Time $O(2.4023^n)$ Using Maximal Independent Sets," BRICS RS-02-45, 2002.

[5] Wikipedia, "Independent Set Problem," http://en.wikipedia.org/wiki/Independent_Set _Problem, Wiki media Foundation Inc., 2008.

[6] Wikipedia, "Degree (Graph Theory)," http://en.wikipedia. org/wiki/Degree_(graph-theory), Wikimedia Foundation Inc., 2008.

[7] R. V. Stee, "Approximations-und Online-Algorithme n: Vertex Cover und Scheduling Unrelated Machines,"http://algo2.iti.uni-karlsruhe.de /vanstee/courses/, 2007.

[8] M. A. A. Zito, " COMP309: Efficient Sequential Algorithms- Vertex Cover," University of Liverpool, http://www.csc.liv. ac.uk/~michele/TEACHING/ COMP309/2005/Lec10.4.4.pdf, 2005.

[9] Y. W. Chang, "Algorithms: Greedy Algorithms," http://cc.ee.ntu.edu.tw/~ywchang/Course s/Alg/unitf.pdf, 2007.

[10] A. Dharwadker, "The Vertex Coloring Algorithm," http://www.geocities.com/dharwad ker/vertex_ coloring/, 2006.

[11] E. W. Weisstein, "Maximal Independent Vertex Set," http://mathworld.wolfram.com/ MaximalIndependentVertexSet.html,Wolfram Research Inc., Mathworld, 2008.

[12] F. Herrmann and A. Hertz, "Finding The Chromatic Number By Means Of Critical Graphs," ACM Journal of Experimental Algorithms, pp. 1-9, 2002.

[13] R. Naserasr and C. Tardif, "The Chromatic Covering Number of a Graph," http://www.math .uwaterloo.ca/~naserasr/ pdfs/ccn4.pdf, 2005.

[14] Wikipedia, "Hadwiger Conjecture (Graph Theory)," http://en.wikipedia.org/wiki/Hadwiger_Conjectur e(graph_ theory), Wikimedia Foundation Inc., 2008.

[15] R. Thomas, "An Update on the Four-Color Theorem," Notices of the American Mathematical Society, Vol. 45, No. 7, 1998.

[16] Wikipedia,"Hadwiger-NelsonProblem," http://en.wikipeia.org/wiki/Hadwiger_Nelson _Problem,Wikimedia Foundation Inc. 2008.

저 자 소 개

이 상 운

1983년~1987년: 한국항공대학교 항공전자공학과 (학사)

1995년 ~ 1997년: 경상대학교 컴퓨터과학과 (석사)

1998년 ~ 2001년 : 경상대학교 컴퓨터과학과 (박사)

2003년 : 강원도립대학 컴퓨터응용과 전임강사

2004년 ~ 2007.2 : 국립 원주대학 여성교양과 조교수

2007.3 ~ 현재 : 강릉원주대학교 과학기술대학 멀티미디어공학과 부교수

관심분야 :소프트웨어 프로젝트 관리, 소프트웨어 개발 방법론, 소프트웨어 척도 (소프트웨어 규모, 개발노력, 개발기간, 팀 규모), 분석과 설계 방법론, 소프트웨어 시험 및 품질보증, 소프트웨어 신뢰성, 신경망, 뉴로-퍼지, 그래프 알고리즘

e-mail : sulee@gwnu.ac.kr

최 명 복

1992년 : 호서대학교 전자계산학과(학사)

1994년 : 아주대학교 컴퓨터공학과(석사)

2001년 : 아주대학교 컴퓨터공학과(박사)

1997~현재 : 강릉원주대학교 멀티미디어공학과(교수)

2004. 1~현재 : 한국인터넷방송통신학회 이사

관심분야 : 지능형 정보검색, 퍼지응용, 지식표현, 신경망, 지능형 교통제어, 소프트웨어 공학, 알고리즘

e-mail : cmb5859@gmail.com, cmb1@gwnu.ac.kr