모서리값 확장 그래프를 사용한 함수구성에 관한연구

박춘명*

A Study on the Constructing the Function using Extension Edge Valued Graph

Chun-Myoung Park*

요 약

본 논문에서는 최근의 디지털논리시스템의 함수구성시에 도입되고 있는 그래프이론에 바탕을 둔 새로운 형태의 데이터구조 형태인 모서리값 확장 그래프를 추출하는 알고리즘을 제안하였다. 이를 위해 수학적 배경으로는 리터럴 함수와 리드 물러 확장에 대해 논의하였으며, 본 논문의 근간인 모서리 확장 그래프의 도출에 대해 논의하였다. 또한, 모서리 확장 그래프로부터 임의의 m 치 n 변수의 축약된 함수구성을 도출하는 알고리즘을 제안하였으며 이를 예에 적용하여 그 타당성을 보였다. 제안된 알고리즘의 규칙성을 고려하여 동일부분을 모듈화함으로써 일반성을 가짐을 보였다.

ABSTRACT

In recently years, many digital logic systems based on graph theory are analyzed and synthesized. This paper presented a method of constructing the function using edge valued extension graph which is based on graph theory. The graph is applied to a new data structure. from binary graph which is recently used in constructing the digital logic systems based on the graph theory. We discuss the mathematical background of literal and reed-muller expansion, and we discuss the edge valued extension graph which is the key of this paper. Also, we propose the algorithms which is the function derivation based on the proposed edge valued extension graph. That is the function minimization method of the n -variables m -valued functions and showed that the algorithm had the regularity with module by which the same blocks were made concerning about the schematic property of the proposed algorithm.

키워드

디지털논리시스템, 그래프, 모서리값 확장 그래프, 모듈

Key word

Digital logic systems, graph, edge valued graph, module

* 정회원 : 한국교통대학교(교신저자, cmpark@ut.ac.kr)

접수일자 : 2012. 08. 03

심사완료일자 : 2012. 09. 18

I. 서 론

현재 사용되고 있는 디지털논리시스템 및 이를 근간으로 하는 컴퓨터 하드웨어 분야는 2진논리에 기초를 두고 집적회로기술의 비약적인 발전으로 회로의 형태가 VLSI, ULSI화 되어 단일 칩상에 방대한 양의 회로를 집적할 수 있게 되었다. 이렇게 회로가 대형화 될수록 상호결선은 더욱 복잡해지고, 연산속도의 제한성, 정보전송량의 방대함에 따른 정보전송시간 지연 및 외부 단자수의 증가로 인한 칩면적의 효율성 저하 및 단자수의 제한 등의 문제점들이 대두되기 시작하였다.[1-3] 특히 최근에는 그래프이론[4-6]에 바탕을 두고 디지털논리시스템을 해석하려는 연구가 시도되고 있다. S.B.Aker[7]는 최초로 방향성 비순환 그래프형태의 데이터구조인 결정도의 개념을 정립하였으며 R.E.Bryant[8]는 S.B.Aker가 제안한 개념으로부터 최초로 2진 그래프를 사용하여 부울함수구성의 추출 및 축약에 적용하였다. 또한, D.M.Miller[9]는 S.B.Aker가 제안한 개념을 확장하여 좀 더 효율적으로 함수구성의 추출 및 축약에 적용하였다. 앞에서 언급한 연구들은 각각 그래프에 바탕을 둔 2진 논리시스템 및 확장 논리시스템의 연구의 효시가 되고 있으며 그 후 다수의 연구들이 발표되고 있다.

한편, S.B.K.Vrudhula 등[10]은 기존의 2진 그래프가 부울함수식만을 표현하는 단점을 보완하여 새로운 확장체인 모서리값 2진 결정도를 제안하였는데, 이는 부울 표현식의 벡터를 필요로 하는 대신 더 높은 레벨의 특성을 갖는 상태에서 도출될 수 있으며 계층적으로 검증할 수 있다는 장점이 있으므로 임의의 정수함수식에 대한 함축적이고 전형적인 표현방식이다. 또한, C.M.Park[11]은 그래프에 기반을 둔 스위칭함수 구현과 FFDD를 사용하여 함수구성을 하였다. 본 연구에서는 모서리값 2진 그래프의 장점을 유지하면서, 확장 논리함수를 구현하는데 적합한 리터럴 함수를 통해서 모서리값 2진 그래프를 모서리값 확장 그래프로 전환하는 알고리즘을 제안하고, 이를 바탕으로 임의의 m 치 n 변수의 다치논리함수 구성 및 함수축약에 적용하는 한가지 방법을 제안하였다.

본 논문에서는 노드수의 감소에 의해서 처리속도의 감소, 회로구성의 간략화 및 비용감소 등의 장점을 도출하였다. 노드별로 동일하게 적용되는 에지화 과정을 하나의 모듈로 구성하여 각각의 노드에 반복 적용하여 제

안된 알고리즘이 정규성이 있음을 보였다. 본 논문의 서술과정은 다음과 같다, II장에서는 모서리값 그래프에 대해 논의하였으며, III장에서는 본 논문의 수식전개의 바탕이 되는 수학적 배경을 논의하였다. 그리고 IV장에서는 모서리값 확장 그래프를 정의하고 모서리값 2진 그래프로부터 모서리값 확장 그래프를 생성하는 알고리즘을 제안하였다. V장에서는 모서리값 확장 그래프로부터 함수를 도출에 대한 내용을 논의하였으며 이를 확장 다변수인 경우로 확장하여 일반식을 도출하였다. VI장의 결론에서는 본 논문에서 제안한 모서리값 확장 그래프에 기반을 둔 함수구성 방법의 특징과 장점에 대해서 서술하였다.

II. 모서리값 그래프

모서리값 그래프는 2진 그래프의 확장으로 산술연산 함수의 효과적인 처리를 수행한다. 변수의 브랜치를 모서리화한다는 것은 2진 결정도에 비해서 부울연산 표현식의 벡터들을 필요로 하는 반면에 모서리 변수화 그래프는 좀 더 높은 레벨의 특성을 갖는 상태에서 유도될 수 있고 계층적 검증이 가능하다는 장점을 갖는다. 모서리값 2진 결정도는 부울 연산식을 정수함수식으로 전환하여 결정도를 생성하는 과정에 중점을 두고 있기 때문에 2진 논리상에서 나타나는 한계를 벗어나지 못하였다. 본 논문에서는 모서리값 2진 결정도의 장점을 유지하면서 2진에 있어서의 노드 에지화를 확장하여 적용하는 알고리즘에 대하여 논하였다.

2.1. 부울 표현식을 산술연산함수로의 전환

정수 상에서 0과 1이 부울 대수의 참과 거짓을 각각 나타냄을 이용하여 부울 연산을 산술연산으로 바꾸는 기본 수식은 다음 식들과 같다.

$$x \wedge y = xy, x \oplus y = x + y - 2xy$$

$$x \vee y = x + y - xy, x' = 1 - x$$

따라서 부울 함수는 정수 함수의 특별한 경우로 볼 수 있고 2진 결정도는 모서리값 2진 그래프의 특별한 경우로 볼 수 있다.

2.2. 모서리값 2진 결정도의 변수확장

모서리값 2진 결정도의 노드에는 각각 변수가 할당되어 변수값에 따라서 브랜치가 결정된다. 변수는 모두 $\{0, 1\}^n$ 의 값을 가지며 만일 함수 중의 변수가 2진 값을 갖지 않고 정수의 값으로 입력된다면 해당 변수를 2진 벡터로 표현해야한다.

III. 수학적 배경

3.1. 리터럴 함수

2진 결정도상의 2진을 확장하기 위한 기본개념이 되는 리터럴 함수를 다음과 같이 정의한다.

[정의 1] X_i 집합 $P_i = \{0, 1, 2, \dots, (p-1)\}$ 중의 임의의 값을 취하는 다치변수라 한다. 임의의 부분집합 $S_i \subseteq P_i$ 에 대하여 $X_i^{S_i}$ 를 다음 식(1)의 내용을 표현하는 literal이라 한다.

$$X_i^{S_i} = \begin{cases} 1 & \text{if } i \in S_i \\ 0 & \text{if otherwise} \end{cases} \quad (1)$$

3.2. 리드 물러 확장

보다 일반적인 경우인 m 치 n 변수로 확장하는 방법은 부울연산식을 모듈연산에 의해서 확장시킨 리드 물러 확장으로 도출할 수 있으며 이로부터 다치 다변수 예지 값 결정도의 일반식을 유도한다.

부울연산 내의 논리합과 논리곱은 모듈연산에 있어서의 모듈2합과 모듈2곱으로 대치될 수 있음을 이용하여 논리곱과 논리적으로 구성되어 있는 부울연산식의 일반식을 모듈합과 모듈적으로 재배열한 것을 리드 물러 확장이라 하고 그 일반식은 다음 식(2)와 같다.

$$F(x_n, \dots, x_1) = \sum_{i=0}^{2^n-1} C_i x_n^{e_{i,n}} x_{n-1}^{e_{i,n-1}} \dots x_1^{e_{i,1}} \quad (2)$$

over GF

여기서, e_{ij} 는 0 또는 1이고 C_0 는 상수항이다.

또한, 계수 선택은

$$T_1 = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}, T_n = \begin{pmatrix} T_{n-1} & 0 \\ T_{n-1} & T_{n-1} \end{pmatrix}$$

로써 선택할 수 있다.

IV. 모서리값 확장 그래프의 정의

4.1. 모서리값 확장 그래프의 정의

본 절에서는 2진 상에서 정의된 모서리값 2진 그래프를 m 치로 확장하기 위해서 각각의 노드에 m 개의 브랜치를 할당하고 리터럴함수를 적용하여 변수를 모서리화 하기위한 기본 개념을 정의한다.

[정의 2] 모서리값 확장 그래프 $\langle c, f \rangle$ 의 쌍으로 구성되며 c 는 상수 값이며 f 는 다음 2가지 경우의 노드를 갖는 방향성 비순환 그래프이다.

- (1) 0에 의해서 표현되는 하나의 터미널 노드 0가 존재한다.
- (2) 터미널이 아닌 노드 v 는 $\langle \text{variable}(v), \text{child}_1(v), \text{child}_{M_1}(v), \dots, \text{child}_{M_m-2}(v), \text{child}_r(v), \text{value} \rangle$ 의 $m+2$ 개의 쌍으로 구성되며 $\text{variable}(v)$ 는 변수이다.

[정의 3] 모서리값 m 치 그래프 $\langle c, f \rangle$ 는 산술 연산 함수 $c+f$ 를 나타내며 f 는 f 에 의해서 나타내지는 함수다. 0는 상수함수 0를 나타내며 $\langle x, l, M_1, \dots, M_{m-2}, r, v \rangle$ 는 산술연산함수 $x(1+v) + (x-1)M_1 + \dots + (x-(m-2))M_{m-2} + (x-(m-1))r$ 이다. 각각의 노드에는 식(1)의 리터럴 함수를 적용하여 변수입력이 M 인 경우만을 모서리화 시킨다.

4.2. 모서리값 2진 그래프에서 모서리값 확장 그래프를 생성하는 알고리즘

정의된 모서리값 확장 그래프를 모서리값 2진 그래프로부터 생성하는 과정을 모서리화 되는 노드에 중점을 두고 브랜치 삽입, 리터럴 함수 적용, 터미널 노드에 이를 때까지 반복적용 등의 단계별로 기술하면 다음과 같다.

[STEP 1] 터미널 노드가 아닌 노드에서는 모서리값 확장 그래프의 산술연산함수를 나타내는 m 개의 쌍 $\langle x, l, M_1, \dots, M_{m-2}, r, v \rangle$ 를 만족하는 노드로 만들기 위해서 중간 브랜치를 삽입한다.

[STEP 2] 중간 브랜치를 삽입한 후 각각의 노드의 산술연산함수는 $x(1+v) + (x-1)M_1 + \dots + (x-(m-2))M_{m-2} +$

$[x-(m-1)]r$ 로 전환된다.

[STEP 3] 각각의 노드에 스레쉬홀드 검출기인 리터럴 함수를 적용하면 다음 식(3)이 된다. 예를 들어 3치인 경우 $m=2$ 이므로 변수 x 의 입력이 2이면 모서리화되어 브랜치가 가중치를 갖고 그 외에 0, 1이면 비모서리화되어 브랜치가 가중치를 갖지 않는다.

$$X = \begin{cases} 1 & \text{if } x = m \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

위 STEP2의 논리식에 의해서 입력 변수의 값이 m 치일 때만 모서리화되어 논리 1의 값을 갖게되고 나머지 브랜치는 논리 0인 비에지화 브랜치로 된다.

V. 모서리값 확장 그래프에 대한 함수식

5.1. 모서리값 확장 그래프에 대한 함수식

III장 2절에서의 n 변수 함수식인 전개식에서 각항의 계수의 개수는 다음과 같다.

$F(x_1, \dots, x_n) \rightarrow$

${}_nC_0$: 상수항 계수의 개수

${}_nC_1$: 1차항 계수의 개수

${}_nC_2$: 2차항 계수의 개수

$\vdots$

${}_nC_{n-1}$: $n-1$ 차항 계수의 개수

${}_nC_n$: n 차항 계수의 개수

여기서, ${}_nC_0 + \dots + {}_nC_n = \sum_{i=0}^n {}_nC_i = 2^n - 1$

따라서, 일반식은 다음 식(4)와 같이 표현 할 수 있다.

$$F(x_1, \dots, x_n) = C_0 + \sum_{i=1}^{2^n-1} C_i x_1^{e_{i,1}} x_2^{e_{i,2}} \dots x_n^{e_{i,n}} \quad (4)$$

이제 위 식(4)에 리터럴 함수(1)을 적용하여 주어진 변수의 순서에 따라 모서리화하는 과정은 다음과 같다.

$f_{x1} : x_1$ 의 변수 에지화

$f_{x2} : x_2$ 의 변수 에지화

$\vdots$

$f_{xn1} : x_n$ 의 변수 에지화

위 과정으로부터 각각의 모서리화 노드를 추출하기 위해 먼저 변수 x_1 에 대해 모서리화하는 과정은 다음과 같다. 우선 2진인 경우의 모서리화는 식(5)와 같고 m 치로 확장하면 다음 식(6)과 같다.

$$f_{x1} = C_0 + x_1 \quad (5)$$

$$\begin{aligned} & \cdot (C_1 + \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}} \\ & + \sum_{i=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) + (1 - x_1) \\ & \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \end{aligned}$$

$$\begin{aligned} f_{x1} &= C_0 + x_1 \cdot (C_1 + \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}} \\ & + \sum_{i=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) + (x_1 - 1) \\ & \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \\ & \quad \vdots \\ & \quad \vdots \\ & + (x_1 - (m-1)) \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \\ & = C_0 + x_1 \cdot (C_1 + \sum_{i=2}^{2^n-1} C_i x_2^{e_{i,2}} x_3^{e_{i,3}} \dots x_n^{e_{i,n}} \\ & + \sum_{i=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \\ & \quad \vdots \\ & \quad \vdots \\ & + \sum_{k=1}^{m-1} (x_1 - k) \cdot (\sum_{j=2}^{2^n-1} C_j x_2^{e_{j,2}} x_3^{e_{j,3}} \dots x_n^{e_{j,n}}) \end{aligned} \quad (6)$$

여기서, 상수항을 고려하기 위해서 $f_{x0}=f_0=C_0$ 라고 하면 주어진 n 변수 일반식은 다음 식(7)과 같다.

$$F = (x_n, \dots, x_1) = \sum_{i=0}^n f x_i \quad (7)$$

여기서,

$$\begin{aligned}
 fx_0 &= f_0 = C_0 \\
 fx_1 &= \sum_{i=1}^{2^n-1} C_i x_1^{ei,1} x_2^{ei,2} \dots x_n^{ei,n} \\
 fx_2 &= \sum_{i=2}^{2^n-1} C_i x_2^{ei,2} x_3^{ei,3} \dots x_n^{ei,n} \\
 &\vdots \\
 fx_{n-1} &= \sum_{i=n-1}^{2^n-1} C_i x_{n-1}^{ei,n-1} x_n^{ei,n} \\
 fx_n &= \sum_{i=n}^{2^n-1} C_i x_n^{ei,n}
 \end{aligned}$$

이다.

위 내용에서 상수항인 f_{x_0} 를 제외한 각각의 변수에 대하여 m 치로 예지화 시키면 다음 항이 앞의 항에서 반복하여 나오는 형태가 되므로 이를 정리하면 다음 식(8)과 같다.

$$\begin{aligned}
 fx_1 &= x_1 (C_1 + fx_2 + fx_2') + \sum_{k=1}^{m-1} (x_1 - k) fx_2' \\
 fx_2 &= x_2 (C_2 + fx_3 + fx_3') + \sum_{k=1}^{m-1} (x_2 - k) fx_3' \\
 &\vdots \\
 fx_{n-1} &= x_{n-1} (C_{n-1} + fx_n + fx_n') + \sum_{k=1}^{m-1} (x_{n-1} - k) fx_n' \quad (8)
 \end{aligned}$$

따라서 m 치 n 변수 모서리화 변수화 일반식은 다음과 같이 표현할 수 있다.

$$\begin{aligned}
 F(x_1, \dots, x_n) &\equiv fx \\
 &= x_i (C_i + fx_{i+1} + fx_{i+1}') + \sum_{k=1}^{m-1} (x_i - k) fx_{i+1}' \\
 (i &= 1, 2, \dots, n)
 \end{aligned}$$

여기서 $f_{x_0}=f_0=C_0$ 이고 fx_i' 는 비모서리화 브랜치가 갖는 함수식이다.

또한, 위의 일반식을 모서리값 확장 그래프로 표시하면 그림1과 같다.

그림 1. m 치 n 변수의 일반식에 대한 모서리값 확장 그래프

Fig. 1 Edge valued extension graph of General expression for m valued n variable

5.2. 모서리화 모듈 블록도

각각의 변수에 대하여 리터럴 함수를 적용하여 모서리화하고 다음단의 함수식을 찾아내는 과정을 하나의 모듈로 생각하여, 만일 n 개의 변수입력 이 있다면 한 개의 모듈과 상수항으로 모듈화 해서 블록도를 그릴 수 있다. 예를 들어 n 개의 변수입력을 갖는 함수식에서 fx_i 를 변수를 포함하는 함수식 중에서 x_i 변수를 제외한 함수식이라 하고 fx_i' 을 x_i 변수를 포함하지않는 함수식이라 할 때 변수 x_i 에 대한 모듈구성은 다음 그림2와 같고 여기서 모듈 I는 다음단 변수의 모듈 I+1의 입력 모듈이 된다. 변수의 개수가 n 개인 경우는 상수항을 제외하고 각각의 변수에 동일한 모듈을 적용하여 반복함으로써 모든 변수를 모서리화할 수 있다.

그림 2. 변수 x_i 에 대한 모서리화 블록다이아그램

Fig. 2 Edged block diagram for variable x_i

VI. 결 론

본 논문에서는 최근에 디지털논리시스템의 함수구성시에 도입되고 있는 그래프이론에 바탕을 둔 새로운 데이터구조 형태인 모서리화 2진 그래프로부터 임의의 m 치 n 변수의 함수를 다룰 수 있는 모서리화 확장 그래프를 도출하는 한 가지 방법에 대하여 논의하였으며, 예제를 통하여 타당성을 살펴보았다. 제안된 모서리화 확장 그래프를 사용하여 노드수를 줄일 수 있었으며, 이는 함수의 축약과 회로의 간략화를 수행 할 수 있음을 의미하며 처리속도의 감소를 기대할 수 있다. 또한 알고리즘의 규칙성을 고려하여 모듈구성을 통한 정규성에 대한 가능성에 관하여 논하였다. 향후 연구과제로는 제안한 모서리화 확장 그래프를 시뮬레이션, 조합논리시스템 및 순차논리시스템 등의 분야에 효과적으로 적용하는 부분의 연구가 요구된다.

참고문헌

- [1] D.Green, Modern Logic Design, Addison-Wesley Publishing Company, 2008.
- [2] David C.Rine, Computer Science and MVL Theory and Applications, North-Holland Publishing Company, 1977.
- [3] M.K, "Toward the age of the beyond binary electronics and system," ISMVL 90, pp.162-166, North Carolina U.S.A. May. 1990.
- [4] R.Gould, Graph Theory, The Benjamin/Cummings Publishing Company, Inc. 1988.
- [5] R. J.Wilson and J.J.Watkins, GRAPH an Introductory Approach, John Wiley & Sons, Inc. 2006.
- [6] D. B.West, Introduction to Graph Theory, Prentice-Hall, 2003.
- [7] S.B.Aker, "Binary Decision Diagrams," IEEE Trans. Comput., vol.C-27, no.6, pp.509-516, Jun. 1978.
- [8] R.E.Bryant, "Graph-Based Algorithms for Boolean Function Manipulations," IEEE Trans., Comput., vol.C-35, no.8, pp.677-691, Aug. 1996
- [9] D. M. Miller, "MVL Design Tools," IEEE Proc. of ISMVL, Sacramento, California, pp.2-11, may. 1993.

- [10] Yung-Te Lai, M. Pedram and S. B. K. Vrudhula, "Formal Verification using Edge-Valued Binary Decision Diagrams," IEEE Trans. On Computers, vol.45, no.2, Feb. 1996.
- [11] C.M.Park, "Switching function Implementation based on Graph," JKIMICS vol.15, no.9, pp1965-1970, Sep., 2011.

저자소개

박춘명 (Chun-Myoung Park)

한국정보통신학회논문지
제15권 제9호 참조