

레터논문-11-16-1-16

H/W 복잡도 추정을 이용한 최적 FIR 필터 설계

김 인 철^{a)‡}

An Optimal FIR Filter Design Method Using H/W Complexity Estimation

Rinchul Kim^{a)‡}

요 약

본 논문에서는 CSD 계수로 표현되는 FIR 필터를 설계하는 방법에 관하여 고찰한다. 먼저, 게이트 수 측면에서 필터의 복잡도를 추정하는 방법을 제시한다. 이 방법을 통하여 최소의 복잡도를 가지면서 요구되는 성능을 만족시키는 필터를 설계할 수 있다. 다음으로 BonsaiG이라는 MILP 프로그램에 근거한 CSD 필터 설계 프로그램을 제시한다. 2가지 예제를 통해 설계된 필터의 게이트 수가 400-600 정도 차이를 나타내고, 이러한 비교를 통해 최적의 필터를 얻을 수 있음을 보인다.

Abstract

In this paper, we investigate a method for designing FIR filters with CSD coefficients. Firstly, the H/W complexity of a CSD FIR filter is estimated in terms of number of gates. Using the estimated complexity, an optimal filter that can meet the required performance with minimal H/W complexity can be designed. Next, based on the MILP problem solver called BonsaiG, we present a filter design program. From the two design examples, it is demonstrated that an optimal filter can be obtained by comparing the complexity of the candidate filters in terms of the gate counts, whose differences are estimated to be about 400-600 gates.

Keywords : FIR filter, CSD, MILP, gate count

1. 서 론

널리 사용되고 있는 FIR 필터^[1]를 효율적으로 구현하는 방법은 오래전부터 연구되고 있다. FIR 필터를 구현하려면 필터 길이만큼 곱셈 수가 요구되어 복잡도가 높아진다. 만약, 필터 계수를 2의 지수승으로 나타내되 각 자리(digit)를

-1, 0, 혹은 1으로 표현한다면, 필터에 사용되는 곱셈은 작은 개수의 덧셈기로 구현할 수 있다. 이러한 필터 계수 표현을 POT (power-of-two) 표현이라고 하고, 최소의 0이 아닌 값으로 표현된 것을 CSD (canonic signed digit) 표현^[2-4]이라고 한다. 필터 계수를 CSD로 표현하면, 최소의 덧셈기로 곱셈을 구현할 수 있게 된다.

실수 계수를 가지는 FIR 필터는 Parks McClellan 알고리즘 혹은 LP (linear programming) 기법으로 설계한다^[1]. 반면에 CSD FIR 필터는 MILP (mixed integer LP) 기법^[2], 유전 알고리즘 (genetic algorithm)^[3] 등을 통해 설계된다.

a) 서울시립대학교 전자전기컴퓨터공학부
Dept. of ECE, Univ. of Seoul

‡ 교신저자 : 김인철 (rin@uos.ac.kr)

※ 이 논문은 2009년도 서울시립대학교 교내학술연구비에 의하여 연구되었음.

· 접수일(2010년10월29일), 수정일(2011년1월18일), 게재확정일(2011년1월18일)

이 가운데 MILP는 가장 기본적인 CSD 필터 설계 기법이다. 그러나 CSD 필터를 설계하는 방법들은 대단히 복잡하여 필터 길이가 70 이상인 필터도 설계할 수 있다고 하나^[2], 실제적으로 길이가 긴 필터를 설계하기가 매우 어려웠다.

본 논문에서는 CSD 계수로 표현되는 FIR 필터를 구현하는 방법에 관하여 고찰한다. 먼저, 필터 길이가 60 이상인 CSD FIR 필터를 설계할 수 있도록 BonsaiG^[5]이라는 MILP 프로그램에 근거한 필터 설계 프로그램을 제시한다. BonsaiG은 MIPLIB 3.0에 제시된 대부분의 MILP 문제를 해결할 수 있는 효율적인 프로그램이다. 이 프로그램을 CSD 계수 처리가 가능하도록 수정하여 필터 설계에 응용하였다. 다음으로, 게이트 수 측면에서 필터의 복잡도를 추정하여 최소의 복잡도를 가지면서 요구되는 성능의 필터를 설계할 수 있도록 한다.

II. CSD 계수 필터 설계

LPF의 주파수 응답 $H(e^{j\omega})$ 는 $-\pi \leq \omega \leq \pi$ 주파수 범위에서 다음과 같이 규정된다.

$$|H(e^{j\omega})| = \begin{cases} 1, & |\omega| \leq \omega_p \\ 0, & \omega_s \leq |\omega| \leq \pi. \end{cases} \quad (1)$$

여기서 ω_p 와 ω_s 는 각각 통과대역과 정지대역 경계 주파수이다. 최적의 FIR 필터로 알려진 체비셰프 (Chebyshev) FIR 필터^[1]는 식 (1)의 조건을 근사화시켜 다음 조건을 만족시키는 필터 $A(e^{j\omega})$ 로 설계된다.

$$|H(e^{j\omega}) - A(e^{j\omega})| = \begin{cases} \delta_1, & |\omega| \leq \omega_p \\ \delta_2, & \omega_s \leq |\omega| \leq \pi. \end{cases} \quad (2)$$

여기서, δ_1 과 δ_2 는 통과 대역 및 정지 대역의 리플 크기이다. 만약 필터가 선형 위상이라고 하면 필터의 주파수 응답은 체비셰프 다항식으로 표현되어 필터 설계가 용이해진다. 한 예로 설계할 필터의 길이가 $N=2L+1$ 인 제1형 선형 위상 필터라고 하면, $A(e^{j\omega})$ 는 다음과 같이 표현된다.

$$A(e^{j\omega}) = h[0] + \sum_{n=1}^L 2h[n] \cos(\omega n) \quad (3)$$

식 (2)와 (3)으로 표현되는 필터 설계는 Parks-McClellan 알고리즘과 LP 기법을 통해 수행되고^[1], 이때 필터 계수는 실수 값으로 표현된다.

만약 필터 계수를 POT 값으로 표현한다면, 곱셈기 없이 필터를 구현할 수 있다고 알려져 있다^[2,4]. -1과 1사이에 B 비트 값 x 를 POT 값으로 표현하면 다음과 같다.

$$x = b_1 2^{-1} + b_2 2^{-2} + \dots + b_B 2^{-B} = \sum_{i=1}^B b_i 2^{-i} \quad (4)$$

여기서 b_i 는 -1, 0, 혹은 1 세 값 중의 하나이다. 같은 값에 대한 POT 표현은 여러 가지 있을 수 있는데, 0이 아닌 자리가 최소인 POT 표현을 CSD 표현이라고 한다.

최적화된 CSD 필터 계수를 구하는 가장 기본적인 방법은 MILP 기법^[2]이다. MILP에서는 먼저 LP 알고리즘을 통해 실수 계수 FIR 필터를 얻는다. 그런 다음, 분할 및 제한 (branch-and-bound)을 통해 각 계수 값의 범위를 순차적으로 제한시킨 상태에서 LP 최적화를 반복적으로 수행하여 최적의 CSD 필터 계수를 찾아낸다. CSD FIR 필터를 설계하기 위해서는 기존의 실수 계수 필터 설계에서 요구되는 ω_p , ω_s , δ_1 , δ_2 등 4 파라미터들이 설정되어야 한다. 여기에 덧붙여 CSD로 표현할 수 있는 값의 집합을 규정하는 두 파라미터, 즉 계수의 비트 해상도를 나타내는 최대 비트수 (maximum word length) B 와 0이 아닌 비트의 개수 (number of non-zero bits) Z 또한 필요하다. 결국, 필터 설계는 $\{N, \omega_p, \omega_s, B, Z\}$ 가 주어졌을 때 리플 크기를 최소화 할 수 있는 CSD 필터 계수 $\{h[n]\}$ 을 검색하는 문제로 표현된다.

III. 복잡도 분석

일반적으로 H/W로 구현하기 위한 CSD FIR 필터를 최적으로 설계하려면 요구되는 성능을 만족시키되 최소의 복잡

도를 가지도록 해야 한다. CSD FIR 필터의 복잡도는 N B Z 등 3 파라미터에 의해 결정되므로, 이 3 파라미터의 여러 조합에 대하여 MILP를 반복적으로 실행시켜 필터들을 설계한다. 설계된 필터들 중에서 요구되는 성능을 만족시키면서 최소의 H/W 복잡도를 보이는 필터가 최적의 필터로 선정된다. 그런데, 지금까지 N B , Z 3 파라미터를 개별적으로 비교하였을 뿐, 3 파라미터를 동시에 고려하여 복잡도를 비교하려는 시도가 없었다.

필터를 ASIC 등으로 H/W 구현할 때의 복잡도는 실리콘 면적 (die size)으로 표현되고, 이는 설계된 회로의 게이트 수와 밀접하게 관련된다. 이러한 점을 참조하여 본 논문에서는 N B Z 3 파라미터로부터 필터의 게이트 수를 추정한다. 그런 다음, 추정된 게이트 수로 H/W 복잡도를 비교함으로써 최적의 CSD FIR 필터를 선정한다.

배선(wiring), 제어 회로 등을 무시한다면, CSD 필터는 덧셈기와 레지스터(register)로 구현할 수 있다. 여기서 덧셈기와 레지스터는 각각 비트당 n_a , n_r 게이트가 요구된다고 가정하자. 이 값들은 반도체 회사의 합성 라이브러리(synthesis library)로부터 얻을 수 있다.

본 논문에서는 CSD 계수로 표현된 선형 위상 FIR 필터에 D 비트 입력이 인가되었다고 할 때, 게이트 수 측면에서 필터의 복잡도를 추정한다. 필터 길이 N 이 짝수일 때 선형 위상 필터의 구조^[1]는 그림 1에 제시한 바와 같다. 그림 1을 참조하면 선형 위상 필터는 레지스터 2개, 덧셈기 2개, 곱셈기 1개로 구성되는 모듈 $N/2$ 개를 종속 연결하여 구현된다.

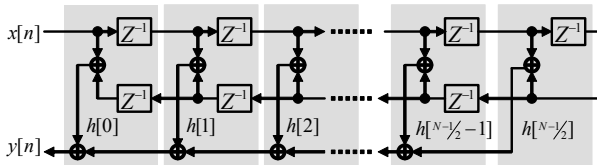


그림 1. 짝수 길이 선형위상 FIR 필터의 구조
Fig. 1. Structure of an even-length linear-phase FIR filter

한 모듈에서 요구되는 게이트 수를 추정하면 다음과 같다. 먼저, 2개의 레지스터에 $2D \times n_r$ Gates, 레지스터의 출력을 더하는 D 비트 덧셈기에 $D \times n_a$ Gates가 요구된다. 곱

셈기는 $Z-1$ 개의 덧셈기로 구현되는데, 여기에 사용되는 덧셈기는 CSD 계수가 군집되어 있는 정도에 따라 복잡도가 결정된다. 만약 CSD 계수에서 0이 아닌 비트들이 b_m 비트 내에 군집되어 있다면, $(D + b_m)(Z-1)n_a$ Gates가 곱셈기에 요구된다. 마지막으로 곱셈기의 출력을 다음 단계의 결과와 합하는 덧셈기는 유효자리가 $(D + b_a)$ 비트라고 할 때, $(D + b_a)n_a$ Gates가 요구된다. 각 연산에서 요구되는 게이트 수를 조합하면 $N/2$ 모듈로 구성되는 필터의 게이트 수는 다음과 같이 주어진다.

$$C = \frac{N}{2} N_a \{ (D + b_m)(Z-1) + 2D + b_a \} + \frac{N-3}{2} N_r (2D + b_a) \quad \text{(Gates)} \quad (5)$$

여기서, b_m 과 b_a 는 B 보다 작거나 같다. 식 (5)으로 주어지는 것은 개략적인 복잡도인데, 보다 근접된 복잡도는 각 모듈에 대한 Z , b_m , b_a 의 실제 값을 적용함으로써 얻을 수 있다.

IV. 필터 설계 예

본 절에서는 2가지 필터 설계 예제에 대해 MILP를 이용하여 CSD FIR 필터를 설계하고, 제3절에서 얻은 게이트 수 추정 값으로 H/W 복잡도를 비교한다. 필터 설계에서 핵심인 MILP 알고리즘은 1999년에 소개된 BonsaiG 프로그램^[5]을 이용하였다. BonsaiG은 MIPLIB3에 제시된 대부분의 MILP 문제를 해결할 수 있는 프로그램으로 알려져 있다. 본 논문에서는 BonsaiG을 CSD 값으로 표현된 변수들을 처리할 수 있도록 확장하여 CSD FIR 필터 설계에 적용하였다.

첫 번째 설계 예제는 표 1에 제시된 사양의 필터를 설계하는 것이다. 표 1의 사양은 $N=17$ 이상의 실계수 필터를 통해 구현될 수 있다. 표 1의 사양에 대해 다양한 N B Z 값에 대해 MILP를 적용시켜 얻은 필터의 성능은 표 2에 제시한 바와 같다. 표 2를 참조하면, $(N$ B $Z)$ 가 각각

(18, 11, 3), (20, 11, 3), (24, 12, 2)일 때 필터의 성능 요구 조건을 만족시킴을 알 수 있다. 이 3가지 조합에 대한 H/W 복잡도는 표 3에 제시한 바와 같다. 표 3에서는 $n_a = n_r = 7$ Gates/bit 이고 b_m 과 b_a 는 $B/2$ 에 비례한다고 가정하고, 식 (5)를 통해 게이트 수를 추정할 것이다. 표 3을 참조하면 게이트 수 측면에서 복잡도를 나타냄으로써 3가지 $N B Z$ 조합에 대해 복잡도를 비교할 수 있고, 그 결과 $(N B, Z) = (18, 11, 3)$ 일 때가 최적의 필터가 됨을 알 수 있었다.

표 1. 첫 번째 설계 예제의 사양
Table 1. Specification of the first design example

	band edge	ripple size	# of taps
passband	0.152 cycles/sample	0.1dB	17
stopband	0.324 cycles/sample	65dB	

표 2. 첫 번째 설계 예제에서의 필터 성능 (단위 dB)
Table 2. Performance of filters in the first design example. (unit: dB)

B/Z \ N	18	20	24
11/5	67.84	66.23	74.48
11/4	67.83	67.82	74.33
11/3	65.30	65.31	69.16
11/2	56.94	58.64	64.35
12/2			65.98

표 3 첫 번째 설계 예제에서의 게이트수 추정치
Table 3. Gate count estimates in the first design example

N	18	20	24
B/Z	11/3	11/3	12/2
# of gates	5481	6090	5964

표 4. 두 번째 설계 예제에서의 게이트수 추정치
Table 4. Gate count estimates in the second design example

N	60	62	64
B/Z	14/4	12/3	12/3
# of gates	23100	19205	19824

두 번째 예로, 통과대역 및 정지 대역 에지가 각각 0.224, 0.272 cycles/sample이고, 통과 대역 및 정지대역 리플 크기는 각각 0.1과 65dB로 규정된 필터에 대해 살펴본다. 이 사양에 대해 MILP를 이용하여 필터를 설계하였고, 그 결과 표 4에 보인 바와 같이 필터 길이 N 이 60, 62, 64일 때 최소의 B, Z 값, 그리고 각 조합에 대해 게이트 수 추정치를 얻었다. 표 4를 참조하면 $(N B, Z) = (62, 12, 3)$ 일 때가 요구되는 성능을 만족하면서 게이트 수 측면에서 최소의 H/W 복잡도를 가지는 것을 알 수 있다.

V. 결론

본 논문에서는 먼저 BonsaiG을 이용하여 MILP 기법으로 CSD FIR 필터를 설계하는 프로그램을 제시한다. 그런 다음, 게이트 수 추정치를 이용하여 H/W 복잡도를 비교함으로써, 요구되는 성능을 만족시키면서 최소의 H/W 복잡도를 가지는 필터를 얻을 수 있게 하였다. 그리고 2가지 예제를 제시하여 게이트 수 추정을 통해 H/W 복잡도 비교가 가능함을 보였다. 본 논문에서 제시한 방법을 이용한다면, 필터 설계시 H/W 복잡도를 고려하여 최적의 필터를 얻을 수 있을 것이다.

참고 문헌

- [1] A.V. Oppenheim and R.W. Schaffer, Discrete-Time Signal Processing, 3rd Ed., Pearson, 2010.
- [2] Y.C. Lim, "Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude," IEEE Trans. on Circuits and Systems, vol. 37, no. 12, pp. 1480-1486, Dec. 1990.
- [3] G. Wade, A. Roberts, and G. Williams, "Multiplierless FIR filter design using a genetic algorithm," IEE Proceedings- Vision, Image and Signal Processing, vol. 141, no. 3, pp. 175-180, June 1994.
- [4] D.M. Kodek, "Performance limit of finite wordlength FIR digital filters," IEEE Trans. on Signal Processing, vol. 53, no. 7, pp. 2462-2469, July 2005.
- [5] L. Hafer, "BonsaiG; Algorithms and Design," SFU- CMPT TR 1999-06, Simon Fraser Univ., Dec. 1999.