무게 있는 리프 이진 트리 균형 문제를 해결하는 알고리즘

이동규*†　백낙훈**　이종원***　류관우*
*경북대학교 컴퓨터공학과
**경북대학교 전자전기공학부
***조지 워싱턴 대학교 전산학과

Algorithms for Balancing Weighted-Leaf Binary Tree

Dongkyoo Lee*†　Nakhoon Baek**　J. Won Lee***　Kwan Woo Ryu*
*Dept. of Computer Science, Kyungpook National University
**School of Electronic and Electrical Engineering, Kyungpook National University
***Dept. Computer Science, The George Washington University

요 약

본 논문에서는 이진 트리 형태를 가지는 다관절체의 균형을 잡거나 이진 트리 모양으로 연결된 네트워크 상에서 단말 노드들의 부하를 균형 있게 하는데 이용할 수 있는 무게 있는 리프 이진 트리 균형 문제를 제안한다. 또한 무게 있는 리프 이진 트리 균형 문제를 리프들의 무게 변화량의 쌍의 l_1-norm, l_2-norm, l_∞-norm 각각을 최소로 하면서 해결하는 방법들을 제안한다. 이 방법들은 무게 있는 리프 이진 트리 균형 문제의 특성을 이용하여 n개 변수를 하나의 변수의 양의 상수배로 나타냄으로써 해결할 수 있음을 보인다.

1. 서론

본 논문의 문제를 제안하기 전에 먼저 몇 가지 용어를 정의한다. 무게 있는 리프 이진 트리(weighted-leaf binary tree)는 리프 노드(leaf node)들은 무게(weight)를 가지지만 모든 내부 노드들은 무게를 가지지 않는 이진 트리이다(그림 1 참조). 이때 내부 노드 I_i의 왼쪽 에지의 가중치를 e_{iL}, 오른쪽 에지의 가중치를 e_{iR}, 리프 노드 L_i $(1 \le i \le n)$의 무게를 w_i라 하고 이들은 모두 양수라 가정한다. 그리고 n개의 리프 노드를 가진 무게 있는 리프 이진 트리에서 임의의 내부 노드 $I_i(1 \le i \le n-1)$의 왼쪽 후손들에 속하는 리프 노드들의 무게의 총합을 왼-후손 무게(left-descendant weight) W_{iL}이라고 정의하고 오른쪽 후손들에 속하는 리프 노들의 무게의 총합을 오른-후손 무게(right-descendant weight) W_{iR}이라고 정의한다.

다음으로 본 논문의 문제를 정의한다. n개의 리프 노드를 가진 무게 있는 리프 이진 트리가 주어졌을 때, 임의의 내부 노드 $I_i(1 \le i \le n-1)$에서 왼-후손 무게 W_{iL}과 가중치 e_{iL}의 곱과 오른-후손 무게 W_{iR}과 가중치 e_{iR}의 곱이 같아지도록 각 리프 노드들의 무게를 구하는 문제를 무게 있는 리프 이진 트리균형 문제라 한다. 이 때, 새로이 구해진 리프 노드 $L_i(1 \le i \le n)$의 무게를 m_i라고 정의하면 무게 있는 리프 이진 트리 균형 문제는 리프 노드 $L_i(1 \le i \le n)$의 새로운 무게 m_i를 구하는 문제이다. 그런데 주어진 문제는 n개의 변수 m_1, m_2, ..., m_n에 대한 $n-1$개의 일차 조합(linear combination)이므로 해가 될 수 있는 m_1, m_2, ..., m_n의 값의 쌍은 무수히 많이 존재한다. 예를 들어 그림 1과 같은 무게 있는 리프 이진 트리가 주어졌을 때, 무게 있는 리프

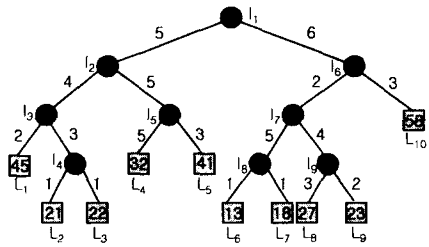

그림 1 무게 있는 리프 이진 트리

이진 트리 균형 문제를 만족시키는 m_1, m_2, ..., m_n의 값의 쌍은 (48, 16, 16, 24, 40, 16, 16, 24, 16, 48), (42, 14, 14, 21, 35, 14, 14, 21, 14, 42), ... 등 무수히 많이 존재한다. 따라서 본 논문에서는 각 리프 노드의 무게 변화량 $|m_i - w_i|$들의 l_1-norm, l_2-norm, l_∞-norm을 각각 최소로 하는 해를 찾고자 한다. 여기서 각 리프 노드의 l_1-norm는 각 리프 노드들의 무게 변화량의 합을 가장 작게 만들며, l_∞-norm은 한 리프 노드의 무게만 크게 변하지 않도록 하며, l_2-norm은 한 리프 노드의 무게만을 크게 변하지 않도록 하면서 각 리프 노드들의 무게 변화량의 합도 작게 만든다.

이 문제는 컴퓨터 그래픽스 분야에서 모빌, 사람 등과 같이 이진 트리 형태를 가지는 다관절체의 각 부분 물체들을 기존의 무게 계산 방법들[4, 5, 6, 7]을 이용하여 구한 후 이 물체가 균형을 잡도록 하는데 응용될 수 있다. 또한 이진 트리 모양으로 연결된 네트워크에서 단말 노드들의 부하(load)를 균형하게 하는데도 응용될 수 있다.

기존에 제안된 트리 균형(tree balancing) 방법들[2, 3, 8]은

이진 트리에 한 노드를 추가하거나 삭제할 때 이를 빠르게 하고 탐색을 빠르게 하기 위해 제안된 방법들이다. 이들 방법들은 한 노드를 추가하거나 삭제할 때 트리의 구조를 변화시킨다. 반면에 본 논문이 제안하는 문제는 트리의 구조를 바꾸는 것이 아니라 트리의 리프 노드들의 무게를 변화시켜서 균형을 맞추는 문제이므로 기존에 제안된 트리 균형 방법들을 사용할 수 없다.

본 논문에서 무게 있는 리프 이진 트리 균형 문제를 각 리프 노드의 무게 변화량들의 l_1-norm, l_2-norm, l_∞-norm 각각을 최소로 하면서 해결하려고 한다. 하지만 각 리프 노드의 무게 변화량들의 l_1-norm, l_2-norm, l_∞-norm은 각각 n개의 변수 m_1, m_2, ... m_n의 $n-1$개의 일차 조합이므로 이를 해결하는 것은 까다로운 문제이다. 우리는 $2 \le i \le n$일 때, m_i를 m_1의 양의 상수배로 표현할 수 있음을 증명함으로써 각 리프 노드의 무게 변화량들의 l_1-norm, l_2-norm, l_∞-norm을 모두 m_1의 함수로 만들 수 있음을 보이고 이를 이용하여 각 문제를 해결하는 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2절에서는 m_2, m_3, ..., m_n을 m_1의 양의 상수배로 표현할 수 있음을 보이고 이를 구하는 알고리즘을 제안한다. 3절, 4절, 5절에서는 2절의 성질을 이용하여 각 리프 노드의 무게 변화량들의 l_1-norm, l_2-norm, l_∞-norm을 최소로 하는 조건하에서 무게 있는 리프 이진 트리 균형 문제를 해결하는 알고리즘을 각각 제안한다. 그리고 6절에서는 결론과 앞으로 더 연구할 문제에 대해서 서술한다.

2. m_k를 m_1의 양의 상수배로 나타내기

본 절에서는 n개의 변수 m_1, m_2,..., m_n의 $n-1$개의 일차 조합(linear combination)으로 구성되는 무게 있는 리프 이진 트리 균형 문제에서 $n-1$개의 변수 m_2, m_3,..., m_n이 m_1의 양의 상수배가 됨을 보이고 m_i를 m_1의 양의 상수배로 나타내는 알고리즘을 제안한다.

n개의 리프를 가진 무게 있는 리프 이진 트리 균형 문제는 n개의 변수 m_1, m_2,..., m_n의 $n-1$개의 일차 조합이므로 m_1을 상수로 보고 m_2, m_3,..., m_n을 $n-1$개의 변수로 보면 m_2, m_3,..., m_n은 m_1의 배수로 표현할 수 있을 것이다. 다음의 소정리 1은 $m_i (2 \le i \le n)$가 m_1의 양의 상수배가 됨을 보인다.

소정리 1 $2 \le i \le n$일 때, m_i는 m_1의 양의 상수배가 될 수 있다.
증명 생략

다음으로 m_i를 m_1의 양의 상수배로 나타내는 방법을 제안한다. 주어진 무게 있는 리프 이진 트리를 깊이가 깊은 내부 노드부터 차례로 방문하면서 각 노드의 왼-후손 무게와 오른-후손 무게를 각각 변수 m_1, m_2, ..., m_n 중의 한 변수의 양의 상수배로 나타낸다. 이때 인덱스가 작은 변수의 양의 상수배가 되도록 한다. 이와 같이 하여 루트 노드까지 방문한 후 이제는 깊이가 얕은 내부 노드부터 차례대로 방문하면서 $W_{iL} \times e_{iL} = W_{iR} \times e_{iR}$임을 이용하여 m_k를 m_1의 양의 상수배로 나타낸다.

다음의 정리 2는 $O(n)$시간에 m_i를 m_1의 양의 상수배, 즉 $a_i m_1$로 나타낼 수 있음을 보인다.

정리 2 $2 \le i \le n$일 때, $O(n)$시간에 m_i를 m_1의 양의 상수배, 즉 $a_i m_1$로 나타낼 수 있다.
증명 생략

정리 2에 의하면 $n-1$개의 변수를 모두 m_1으로 나타낼 수 있으므로 각 리프 노드의 무게 변화량들의 l_1-norm, l_2-norm, l_∞-norm은 n개 변수의 함수가 아니라 한 변수의 함수가 된다. 다음 3, 4, 5절은 이러한 성질을 이용하여 문제들을 해결하는 알고리즘을 제안한다.

3. l_1-norm의 최소화

본 절에서는 정리 2를 이용하여 각 리프 노드의 무게 변화량들의 l_1-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 해결하는 알고리즘을 제안한다.

정리 2에 의하면 $m_i (2 \le i \le n)$는 $a_i m_1$이므로 각 리프 노드의 무게 변화량들의 l_1-norm은 m_1에 관한 함수 $f_1(m_1) = \sum_{i=1}^{n} |a_i m_1 - w_i|$이 된다. 이 때 정리 2에 의하여 a_i는 양수이며 w_i도 양수이다. 따라서 본 절의 문제는 $m_1 > 0$일 때, $f_1(m_1)$을 최소로 하는 m_1의 값을 구하는 문제가 된다. 함수 $f_1(m_1)$은 m_1의 구간에 따라서 일차 함수로 표현될 수 있으므로 $f_1(m_1)$을 최소로 하는 m_1의 값을 찾는 문제는 m_1의 각 구간의 경계점에서 $f_1(m_1)$이 가장 작을 때를 찾으면 되는데 이는 덧셈의 횟수를 고려하지 않는다면 $O(n)$시간에 해결될 수 있다. 하지만 덧셈의 횟수를 고려한다면 $O(n^2)$시간이 걸리므로 본 절에서는 m_1의 각 구간에서 일차함수를 바로 앞 구간의 일차함수를 이용하여 구함으로써 $O(n \log n)$시간에 해결하는 알고리즘을 제안한다.

제안하는 알고리즘은 먼저 $\frac{w_1}{a_1}$, $\frac{w_2}{a_2}$, ..., $\frac{w_n}{a_n}$을 오름차순으로 정렬하고 그 결과를 c_1, c_2, ..., c_n이라고 한다. 그런 다음 $m_1 < c_1$일 때의 일차 함수 $-(a_1+a_2+...+a_n)m_1 + (b_1+b_2+...+b_n) = A_0 m_1 + B_0$을 구한다. 그 다음으로 $c_1 = \frac{w_i}{a_i}$이면, $c_1 \le m_1 < c_2$일 때의 일차 함수 $(A_0+2a_i)m_1 + (B_0 - 2w_i) = A_1 m_1 + B_1$을 구하고 $c_2 = \frac{w_j}{a_j}$이면, $c_2 \le m_1 < c_3$일 때의 일차 함수 $(A_1+2a_j)m_1 + (B_1 - 2w_j) = A_2 m_1 + B_2$를 구한다. 이와 같이 하여 나머지 각 구간에서도 일차식을 구한다. 이때 A_0부터 A_{k-1}까지는 0보다 작다가 A_k가 처음으로 0보다 같거나 커지면, $f_1(m_1)$을 최소로 하는 값은 c_k이다.

다음의 정리 3은 각 리프 노드의 무게 변화량들의 l_1-norm

을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 $O(n \log n)$시간에 해결할 수 있음을 보여준다.

정리 3 각 리프 노드의 무게 변화량들의 l_1-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 해결하는데 $O(n \log n)$시간이 걸린다.

증명 생략

4. l_2-norm의 최소화

본 절에서는 정리 2를 이용하여 각 리프 노드의 무게 변화량들의 l_2-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 해결하는 알고리즘을 제안한다.

정리 2에 의하면 $m_i (2 \le i \le n)$는 $a_i m_1$이므로 각 리프 노드의 무게 변화량들의 l_2-norm은 m_1의 함수 $f_2(m_1) = \sqrt{\sum_{i=1}^{n} |a_i m_1 - w_i|^2}$이 된다. 이 때 정리 2에 의하여 a_i는 양수이며 w_i도 양수이다. 따라서 본 절의 문제는 $m_1 > 0$일 때, $f_2(m_1)$을 최소로 하는 m_1의 값을 구하는 문제로 바뀌게 된다. 함수 $f_2(m_1)$을 전개해보면 $\sqrt{(a_1^2 + a_2^2 + \ldots + a_n^2) m_1^2 - 2(a_1 w_1 + a_2 w_2 + \ldots + a_n w_n) m_1 + (w_1^2 + w_2^2 + \ldots + w_n^2)}$이 되므로 이를 최소가 되게 하는 m_1의 값은 $\frac{(a_1 w_1 + a_2 w_2 + \ldots + a_n w_n)}{(a_1^2 + a_2^2 + \ldots + a_n^2)}$이므로 주어진 문제를 쉽게 해결할 수 있다.

다음 정리 4는 각 리프 노드의 무게 변화량들의 l_2-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 $O(n)$ 시간에 해결할 수 있음을 보인다.

정리 4 각 리프 노드의 무게 변화량들의 l_2-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 해결하는데 $O(n)$시간이 걸린다.

증명 생략

5. l_∞-norm의 최소화

본 절에서는 정리 2를 이용하여 각 리프 노드의 무게 변화량들의 l_∞-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 해결하는 알고리즘을 제안한다.

정리 2에 의하면 $m_i (2 \le i \le n)$는 $a_i m_1$이므로 각 리프 노드의 무게 변화량들의 l_∞-norm은 m_1의 함수 $f_3(m_1) = \max_{i=1}^{i=n} \{|a_i m_1 - w_i|\}$이 된다. 이 때 정리 2에 의하여 a_i는 양수이며 w_i도 양수이다. 따라서 본 절의 문제는 $m_1 > 0$일 때, m_1의 함수 $f_3(m_1)$을 최소로 하는 m_1의 값을 구하는 문제가 된다. $1 \le i \le n$일 때, $|a_i m_1 - w_i|$을 모두 좌표 평면상에 나타내면, 함수 $f_3(m_1)$을 최소로 하는 m_1의 값을 구하는 문제는 m_1의 구간에 따라 가장 위에 있는 선분을 찾고 이 중에서 가장 작은 값을 가질 때의 m_1의 값을 구하는 문제가 된다. 따라서 이 문제는 선분들의 교차점 찾기(line segment intersection) 알고리즘[1]을 이용하여 쉽게 해결될 수 있다.

다음의 정리 5는 각 리프 노드의 무게 변화량들의 l_∞-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 $O(n \log n + k)$시간에 해결할 수 있음을 보인다. 이때, k는 직선들의 교점의 수이다.

정리 5 각 리프 노드의 무게 변화량들의 l_∞-norm을 최소로 하면서 무게 있는 리프 이진 트리 균형 문제를 해결하는데 $O(n \log n + k)$시간이 걸린다.

증명 생략

6. 결론

본 논문에서는 이진 트리 형태를 가지는 다관절체의 균형을 잡거나 이진 트리 모양으로 연결된 네트워크 상에서 단말 노드들의 부하를 균형 있게 하는데 유용한 무게 있는 리프 이진 트리 균형 문제를 제안하고, 이를 해결하는 알고리즘들을 제안하였다.

앞으로의 연구 과제는 리프들의 무게 변화량들의 l_1-norm, l_2-norm, l_∞-norm을 최소로 하는 문제 각각의 상한값(upper bound)을 구하는 문제와 리프들의 무게 변화량들의 l_1-norm, l_∞-norm을 최소로 하는 문제를 더 빠른 시간에 해결하는 방법들이 있다.

7. 참고문헌

[1] I. J. Balaban, "An optimal algorithm for finding segment intersections", In Proc. 11th Annu. ACM Sympos. Compt. Geom, pp. 211-219, 1995.

[2] R. Bayer and E. M. McCreight, "Organization and maintenance of large ordered indexes", Acta Informatica, 1(3), pp. 173-189, 1972.

[3] R. Bayer, "Symmetric binary B-trees: Data Structure and maintenance algorihms", Acta Informatica, 1, pp. 290-306, 1972.

[4] Carlos Gonazlez-Ochoa, Scott Mccammon and Jörg Peters, "Computing Moments of Objects Enclosed by Piecewise Polynomial Surfaces", ACM Transactions on Graphics, Vol. 17, No. 3, July, 1998, pp 143-157.

[5] Y. T. Lee and A. A. G. Requicha, "Algorithms for computing the volume and other integral properties of solids. II. A family of algorithms based on representation conversion and cellular approximation", Communcation of ACM, Vol 25, September, 1982.

[6] Sheue-ling Lien and James T. Kajiya, "A symbolic method for calculating the integral properties of arbitrary nonconvex polyhedra", IEEE Computer Graphics and Applications, 4(10):35-41, October, 1984.

[7] Brian Mirtich, "Fast and Accurate Computation of Polyhedral Mass Properties", J. of Graphics Tools, 1(2):31-50, 1996.

[8] Daniel D. Sleator and Robert E. Tarjan, "A data structure for dynamic tree", Journal of Computer and System Sciences, 26(30), pp. 362-391, 1983