

하드웨어 소프트웨어 분할을 위한 ILP 구현

오 주 영^{*} 한갑수 박도순
 홍익대학교 컴퓨터 공학과
 {jyoh, kshan, dspark}@cs.hongik.ac.kr

ILP implementation for hardware software partitioning

Ju-Young Oh^{*} Kap-Su Han Do-Soon Park
^{*} Dept. of Computer Engineering, Hongik University

요 약

본 논문에서는 통합설계에서 시간제약하의 최소비용 시스템 구현을 위한 분할의 최적해 탐색을 위한 ILP형식을 정의하고 실험에 의해 결과를 평가한다. 분할 문제는 각 노드의 하드웨어 소프트웨어 파티션 선택과 소프트웨어 실행을 위한 대상 노드의 스케줄링 문제가 함께 고려되어야 한다. 본 논문에서는 분할 단계에서 스케줄링을 함께 고려하는 ILP형 정의를 위해 ASAP과 ALAP 스케줄에 의해 유도되는 모빌리티와 시간 제약 조건, 종속성 제약 조건을 형식화하고, 목적함수인 최소 시스템 설계 비용 탐색을 위한 형을 정의한다. 정의된 형에 의한 ILP 구현은 다양한 벤치마크 검증에 의해 최적의 해를 결과로 보인다.

1. 서론

통합설계 결과는 시스템 입력의 각 부분을 성능과 시간 대비에 따라 하드웨어 부분과 소프트웨어 부분으로 나누는 분할 알고리즘 결과에 의해 크게 좌우된다. 분할을 위한 기존의 다양한 알고리즘들[1]은 ILP 형식화에 의한 문제해결 방법의 시간 복잡도를 해결하기 위한 휴리스틱 방법에 의해 개발되어졌다. 이러한 방법들은 설계 대상 시스템의 상이함과 제약조건 및 대상 아키텍처의 차이에 의해 상대적 성능 검증이 불가능하였다. 본 논문에서는 시간 제약하에서 최소 시스템 설계 비용을 유도하는 최적해를 탐색하기 위한 ILP형식을 정의한다. 정의되는 ILP형식은 노드간 종속성 제약조건을 만족해야 하며 이를 위해 상위수준 합성에서의 스케줄링 기법[2][3]을 응용하여 각 노드의 실행 순서관계를 만족할 수 있도록 한다. 2절에서 ILP형 정의를 3절에서 실험 및 결과를 4절에서 결론을 각각 기술하였다.

2. ILP 형식 정의

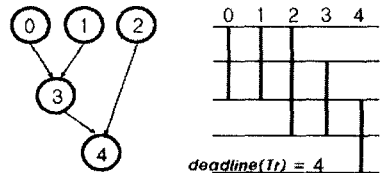
ILP형 정의를 위한 가정으로써, 분할 이후의 시스템 실행을 위한 타겟 아키텍처는 소프트웨어 실행을 위한 하나의 프로세서와 확장 가능한 ASIC 모듈로 하였다. 프로세서의 실행은 순차 실행으로 가정하였으며 통선에 소요되는 비용은 작은 것으로 가정하여 정형화 수식에서 제외하였다. 입력은 방향성 비순환 그래프(DAG)로 하였으며, 목적함수는 시간 제약(데드라인)을 만족하는 최소비용의 분할 결과를 탐색하는 것으로 한다.

2.1 ILP형식 정의

입력그래프 $G(V, E)$ 의 V 는 입력 그래프를 구성하는 각각의 노드로서 연산 혹은 테스크를 의미하며, E 는 노드간 종속관계를 의미한다. 설계대상 시스템에 부과되는 시간 제약은 Tr 로

표기한다. 식 ①은 입력 그래프내의 각 노드들을 Tr 제어단계 내에서 ASAP과 ALAP 스케줄에 의해 계산되는 각 노드의 모빌리티를 나타낸다.

$$mobility(V_i) = \{ S_j \mid E(V_i) \leq j \leq L(V_i) \} \text{---} \textcircled{1}$$



(a)입력 DAG (b)모빌리티 그래프
 그림[1] 입력 DAG 와 모빌리티 그래프

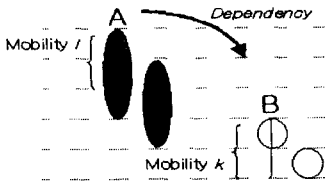
그림[1]은 입력의 한 예로서, 데드라인(Tr)시간이 4제어단계이고 각 노드의 실행시간이 1제어단계라고 가정하였을 때, 시스템에 대한 각 노드의 모빌리티 그래프를 나타내며, 노드 2의 스케줄 가능 제어 단계집합은 $S = \{0, 1, 2\}$ 로 계산된다. 각 노드는 자신의 모빌리티 구간인 처음 제어단계 $E(V_i)$ 와, 마지막 제어단계 $L(V_i)$ 사이의 특정 제어단계의 특정 파티션으로 분할되어 스케줄 될 수 있다. 각 노드 V_i 에 대한 분할집합 $PT(V_i) = \{ X_i^h, X_i^s \}$ 에서, 변수 X_i^h, X_i^s 는 0, 1의 정수 값을 가지며 각각의 노드 i 가 하드웨어, 소프트웨어로 분할

될 경우에는 해당 변수에 대한 정수 값을 1로 가진다. 변수 $TPT(V_i)$ 는 노드 V_i 의 해당 파티션으로의 할당시의 실행 시간을 의미하며, $CPT(V_i)$ 는 노드 V_i 의 해당 파티션으로의 구현시의 소요되는 비용을 의미한다. 파티션 단계에서 스케줄링을 함께 고려하기 위해 정의되는 변수 $X_{i,l}^h$ 와 $X_{i,l}^s$ 는 노드 V_i 가 하드웨어, 소프트웨어로 각각 분할되어 제어단계 l 에 스케줄 되는 경우에는 정수 값 1을 가지며, 그렇지 않은 경우는 0을 가진다. 변수 $X_{i,j}$ 는 노드 i 가 제어단계 j 에 스케줄되는 경우에 정수값 1을, 그렇지 않은 경우는 0을 가진다. $V_i \rightarrow V_j$ 는 노드 V_i 가 노드 V_j 의 선행 노드임을 나타낸다. 각 노드는 하드웨어 혹은 소프트웨어 파티션에 유일하게 분할되어야 하는데, 이는 식 ②로 표현된다.

$$\forall \text{node } V_i, \sum_{l=1}^{L(V_i)} (X_{i,l}^h + X_{i,l}^s) = 1 \quad \text{--②}$$

또한 각 노드에 대해, 타겟 아키텍처로 가정한 하나의 프로세서에 대해서 어느 시간에서 스케줄 가능한 각 제어단계별 소프트웨어 분할 노드의 개수는 1을 초과할 수 없다는 조건을 정형화한 것이 식 ③이다.

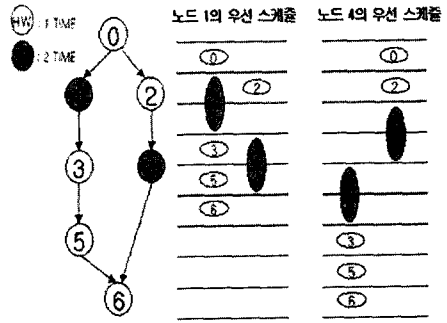
$$\forall i, 1 \leq i \leq n, \forall l, 1 \leq l \leq Tr, \sum_{k \in E(V_i)} X_{i,k}^s \leq 1 \quad \text{--③}$$



그림[2] 노드간 자료 종속성

그림[2]에서 노드 A가 스케줄 될 수 있는 모빌리티 구간과 노드 B의 모빌리티 구간에서 A, B의 스케줄 조합 결과가 종속성 제약 조건을 만족해야 하며, 이는 식 ④에 의해 정의된다. 식 ④는 선행 노드가 후행 노드의 스케줄 스텝이전에 스케줄 되어야 함을 명시하며, 노드의 하드웨어 소프트웨어 각각의 분할 결과에 대한 종속성 제약조건을 모두 만족하도록 유도한다.

$$\forall i, j, \sum_{l=1}^{L(V_i)} (TPT(V_i) + j \times X_{i,l}) - \sum_{k \in E(V_i)} (k \times X_{i,k}) \leq 0, \text{ for all } V_i \rightarrow V_j \quad \text{--④}$$



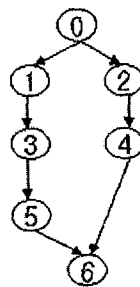
그림[3] 동일한 파티션에 대한 상이한 스케줄 결과

식 ③, ④는 그림[3]과 같이 동일한 분할 결과에 대해서 스케줄링을 어떻게 수행하는가에 따른 제약시간 만족상황을 모두 고려할 수 있도록 한다. ILP형식의 목적함수로서 제약 시간을 만족하며 최소의 구현 비용을 가지는 시스템을 합성하기 위한 형은 식 ⑤와 같이 정의될 수 있다.

$$\text{Minimize } \sum_{i=1}^n CPT(V_i) \times X_{i,l}^h \quad \text{--⑤}$$

3. 실험 및 결과

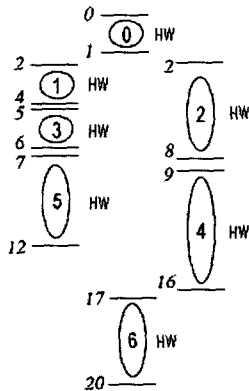
형식화된 ILP의 성능 평가를 위한 실험은 PeniumII 프로세서에서 그림[5]와 같은 벤치마크에 대해 수행하였다. 시스템 제약 시간은 입력 DAG에서 하드웨어로 구현시 최장시간이 소요되는 임계경로의 시간(hard)과 소프트웨어 실행시의 최장 경로 시간과 hard의 평균시간을 soft 시간으로 부여하여 각각 실험하였다. 그림[4]는 논문[4]의 벤치마크에 대한 ILP 실행 결과를 보여준다. (a)는 입력 DAG와 각 노드의 비용테이블이며, 이러한 입력 환경에 대해 (b)의 hard 시간 제약조건과 (c)의 soft 시간 제약조건하에서의 각각의 분할 결과는 최적 분할 결과를 유도한다.



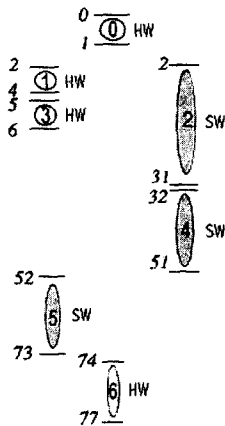
노드 번호	하드웨어 비용	소프트웨어 실행시간	하드웨어 실행시간
0	11	8	2
1	20	11	3
2	75	30	7
3	15	19	2
4	25	20	8
5	100	22	6
6	50	18	4

(a) 입력 DAG 와 각 노드의 비용

구되며, 프로그램 실행시간 경감을 위한 변수의 최적화에 대한 연구가 필요로 되어진다.



(b) $Tr=21$ 에서의 ILP 결과



(c) $Tr=78$ 에서의 ILP 결과

그림[4] ILP에 의한 노드 분할

각 벤치마크에 대한 실험결과는 그림[5]와 같이 hard와 soft 시간 제약에 대한 모든 경우에 대해 최적의 시스템 설계 비용을 유도하였으며 노드의 개수가 많아질수록 최소시스템 설계 비용 탐색을 위한 실행 시간은 지수배로 증가하였다

4. 결론

본 논문에서는 통합 설계의 성능에 가장 큰 영향을 미치는 분할 알고리즘의 성능 평가 기준으로서 ILP형을 정의하고 구현하였다. 구현된 ILP의 각 벤치마크에 대한 실험결과는 모든 경우에 대해 최적의 결과를 보였다. ILP실행은 분할 대상 노드가 많아질수록 노드의 지수배만큼 증가하는 스케줄링 경우 수에 의해 프로그램 실행시간에 대한 부담이 크게 작용한다. 이는 생산성 있는 시스템 설계에 대한 효율성이 떨어지게 된다. 따라서 분할 알고리즘을 위한 휴리스틱 알고리즘 개발에 대한 평가의 대상으로서 소규모 시스템 합성에 응용될 수 있으며, 향후 연구 방향은 분할시 부과되는 상이한 파티션간 나타나는 속성에 의해 부과되는 통신 비용을 고려한 ILP로의 확장이요

벤치마크	노드 개수	시간 제약조건	시스템 설계비용	
			최적 비용	ILP 비용
jam1.chs[4]	7	hard	296	296
		soft	147	147
GMDF-alpha[5]	9	hard	1058	1058
		soft	591	591
논문[6]	14	hard	522	522
		soft	397	397
3X3-det[7]	14	hard	603	603
		soft	523	523
modified-dhrc[7]	16	hard	331	331
		soft	233	233
FIR filter[8]	23	hard	665	665
		soft	465	465
AR-lattice filter[9]	28	hard	1169	1169
		soft	871	871

그림[5] 분할 결과 분석

참고문헌

[1] Giovanni De Micheli, Mariagiovanna Sami, " Hardware/Software Co-Design." Kluwer Academic Publishers, 1997.
 [2] L. Hafer and A.C. Parker, " A Formal Method for the Specification, Analysis, and Design of Register-Transfer Level Digital Logic", *Proc. of the 18th Design Automation Conference*, pp. 546-553, June 1981
 [3] J. Lee, Y. Hsu, and Y. Lin, "A New Integer Linear Programming for the Scheduling Problem in Data-Path Synthesis," *Proceedings of the International Conference on Computer-Aided Design*, pp. 20-23, 1989
 [4] Hidalgo, J.L.; Lanchares, J., "Functional partitioning of hardware-software codesign using generic algorithms," *Proceedings of the 23rd EUROMICRO conference*, Pages:631-638, 1997
 [5] F.Rousseau, J. Benzakki, J-M. Berge, M. Israel, "Adaptation of Force-Directed Scheduling Algorithm for Hardware-Software Partitioning," *proc. of Sixth Int'l workshop on Rapid System Prototyping*, pp:33-37, June 1995.
 [6] J.A. Maestro, "New methodologies for Hardware-Software Codesign Partitioning to Avoid High Communication Overhead", Departamento de informatica y Automatica Universidad complutense de Madrid
 [7] Chuck Monahan, Forrest Brewer, "Scheduling and binding bounds for RT-level symbolic execution," *Proc of the International Conference on Computer-Aided Design*, 1997
 [8] Ki soo Hwang, Albert E. Casavant, "Scheduling and Hardware Sharing in Pipelined data path," *IEEE Int'l conference on CAD*, Pages:24-27, 1989
 [9] Samit Chaudhuri Stepen A. Blythe, Robert A. Walker "A solution methodology for exact design space exploration in a three-dimensional design space," *IEEE transaction on VLSI systems*, Vol. 5.1, 1997