

# 판금형 해석을 위한 정적/외연적 유한요소 프로그램의 병렬화에 관한 연구

## On The Parallel Implementation of a Static/Explicit FEM Program for Sheet Metal Forming

전석기 (연세대 자동차기술연구소), 정동원 (제주대 기계공학과), G. P. Nikishikov, 牧野内 昭武  
Sukky Jun (Yonsei Institute of Automobile Technology), Dong Won Jung (Cheju National University)  
Gennardy P. Nikishikov (GKSS Research Center, Germany), Akitake Makinouchi (RIKEN, Japan)

### ABSTRACT

A static/implicit finite element code for sheet metal forming (ITAS3D) is parallelized on IBM SP 6000 multi-processor computer. Computing-load-balanced domain decomposition method and the direct solution method at each subdomain (and interface) equations are developed. The systems of equations for each subdomain are constructed by condensation and calculated on each processor. Approximated operation counts are calculated to set up the nonlinear equation system for balancing the compute load on each subdomain. The square cup tests with several numbers of elements are used in demonstrating the performance of this parallel implementation. This procedure are proved to be efficient for moderate number of processors, especially for large number of elements.

**Key Words :** sheet metal forming (판 금형), static/implicit FEM (정적/외연적 유한요소법), parallel finite element analysis (병렬 유한요소 해석), domain decomposition method (영역 분할법)

### 1. 서 론

일반적으로 판금 성형을 위한 유한요소 해석은 재료의 대변형, 마찰, 접촉, 공구의 형상 등 여러 가지 요인을 모두 고려하는 복잡한 과정이다. 그러므로 보다 정확한 성형 결과를 얻기 위해서는 이러한 각각의 요인들을 위한 효율적이고 개선된 알고리즘이 필수적으로 개발되어야 한다. 그러나 현재 주어진 프로그램을 이용하여 실 시간 내에 최대의 성과를 얻으려고 한다면 (i) 변형이 심한 국소 부위의 요소망을 세분화하는 적응기법이나 (ii) 프로그램의 병렬화를 이룩하여 많은 수의 요소를 고려하는 대규모 문제의 정확한 해석이 가능하도록 하여야 한다.

본 논문에서는 판금 성형 해석 전문 유한요소 프로그램인 ITAS3D<sup>(1)</sup>를 MIND 형 병렬 컴퓨터에서 실행이 가능하도록 병렬화를 수행하였다. ITAS3D는 개정 라그랑지안을 정식화(updated Lagrangian rate formulation)를 이용하는 정적/외연적 프로그램이며 대부분의 경우, 축퇴된 (degenerated) 셸 요소를 사용한다. 이 프로그램을 사용하면 스탬핑 과정과 스프링백 과정 모두를 정확히 실행할 수 있지만 계산 시간의 장기화 때문에 만족할만한 결과를 실 시간 내에 얻기가 어려운 단점 또한 갖고있다. 알고리즘 자체의 개선을 통해서도 실행 속도를 높일 수 있지만 보다 근본적인 계산 시간의 향상을 위해서 여러

개의 연산장치를 갖는 병렬 컴퓨터 상에서의 실행 능력을 극대화 하도록 프로그램의 병렬화를 수행하였다.

유한요소법의 행렬 방정식을 계산하는 방법으로는 축차(iteration) 해법과 직접(direct) 해법이 있는데 병렬화 방법을 위해서는 반복 해법이 선형 확장성 (Scalability)을 갖는 장점이 있다. 즉, 연산장치 수의 증가에 따라 실행 능력이 선형적으로 증가한다. 그러나 이 방법은 종종 수렴성의 문제를 초래하는 결정적인 단점을 갖고있으므로 ITAS3D는 수렴의 안정성을 보다 높이기 위해서 직접 해법을 사용한다. 또한, 직접 해법을 이용할 경우, 계산 시간의 예측이 가능하고 프로그래밍이 간단해지는 장점이 있다. 따라서 그에 따른 병렬화 과정에서 많은 숫자의 연산장치는 필요치 않고 보통 8~12 개 정도까지의 연산장치를 이용하여 병렬화 작업을 수행하였다.

ITAS3D의 병렬화 작업 중에서 가장 중요한 과정은 문제의 영역을 여러 개의 부수 영역 (subdomain)으로 나누는 요소 분할(mesh partitioning) 과정이다.<sup>(2)</sup> 보통, 문제의 영역을 나눌 때는 병렬 컴퓨터의 연산자 수와 부수 영역의 수가 일치하도록 하는데, 이 경우, 각 연산자의 실행 시간이 균형을 이루도록 하고 각 연산자간의 정보 전달을 최소화 하는 것이 중요하다. ITAS3D는 전체 행렬 방정식을 각 부수 영역의 행렬 방정식으로 나누고 이 각각의

행렬 방정식을 각 연산자에서 균형적으로 동시에 실행하도록 병렬화를 진행시켰다.

본 논문에서는 먼저, ITAS3D의 정식화 과정을 간단히 소개하고 병렬화 전략을 제시하였으며 이어서 균형된 계산 하중을 고려하는 영역 분할법을 설명하였다. 또한 여러 가지 요소망을 이용하여 병렬화 된 ITAS3D의 수행성의 향상을 검증하였다.

## 2. ITAS3D의 정식화와 병렬화

ITAS3D는 대변형을 일으키는 금속판의 기술을 위하여 개정 라그랑지안을 정식화를 사용한다. 가상 속도 원리(virtual velocity principle)에 의한 비율(rate) 형태의 평형 방정식은 아래와 같이 표현될 수 있다.<sup>(3)</sup>

$$\int_V \{ (\sigma_{ij}^j - 2\sigma_{ik} D_{kj}) \delta D_{ij} + \sigma_{jk} L_{ik} \delta L_{ij} \} dV = \int_{S_i} \frac{df_i}{dt} \delta v_i dS \quad (1)$$

여기에서  $V$ 와  $S$ 는 임의의 시간  $t$ 에서의 내부 영역과 경계를 의미하며  $S_i$ 는 트랙션의 비율  $df_i/dt$ 가 서술된 자연 경계를 나타낸다.  $\delta v$ 는 필수 경계에서  $\delta v = 0$ 을 만족하는 가상 속도장(virtual velocity field)이고  $L = \partial v / \partial x$ 로 주어지는 velocity gradient tensor이다. 또한,  $D$ 와  $W$ 는 각각  $L$ 의 대칭과 비대칭 성분이다. Jaumann 응력 비율(rate of stress)  $\sigma_{ij}^j$ 의 구성 방정식은

$$\sigma_{ij}^j = C_{ijkl}^{ep} D_{kl} = C_{ijkl}^{ep} L_{kl} \quad (2)$$

이다.

관급 성형 해석을 위해 내연적 유한요소법을 사용할 경우에는 수렴성의 문제가 발생할 수 있으므로 ITAS3D 프로그램은 시간 적분을 위해 다음과 같은 정적/외연적 방법을 이용한다. 먼저, 충분히 작은 시간의 증분  $\Delta t$  동안에, 한 적분 점에서 탄성으로부터 소성으로의 변환이나 한 경계 점에서 비접촉으로부터 접촉으로의 변환 등과 같은 상태의 중요한 변화가 없다고 가정한다. 이런 가정이 타당하도록 한 스텝의 크기를 결정하기 위하여 ITAS3D에서는 r-최소화 방법<sup>(4)</sup>을 이용한다. 그러므로 이러한 시간 증분  $\Delta t$  동안 식 (1)의 모든 비율(rate)량들은 아래와 같은 증분 값들로 대체될 수 있다.

$$\Delta u = v \Delta t, \quad \Delta \sigma^j = \sigma^j \Delta t, \quad \Delta L = L \Delta t \quad (3)$$

이러한 가정으로부터 유한요소법의 행렬 방정식을 유도하면 다음과 같은 결과를 얻는다.

$$K \Delta u = \Delta F + \Delta C \quad (4)$$

여기에서  $K$ 는 강성 행렬이고  $\Delta u$ 는 절점에서의 변위 증분 계수이다. 또한,  $\Delta F$ 와  $\Delta C$ 는 식 (1)에서의 우변 항에 관계된 항들로서,  $\Delta F$ 는 외력의 증분에 의한 항이고  $\Delta C$ 는 외력 벡터의 회전에 의한 항이다. ITAS3D에서는 매 시간 증분마다 축차법으로 식 (4)를 풀지만 이 과정 동안 강성 행렬  $K$ 는 변화가 없이 일정하다고 가정하므로 준 정적(semi-implicit) 혹은 정적/외연적 방법이라 불리운다.

병렬화를 위하여 영역 분할법 (domain decomposition method, DDM)을 이용하면 위의 행렬 방정식은 다음과 같은 각 부수 영역마다의 행렬 방정식으로 나누어진다.

$$\begin{bmatrix} k_{ii} & k_{ib} \\ k_{bi} & k_{bb} \end{bmatrix} \begin{Bmatrix} u_i \\ u_b \end{Bmatrix} = \begin{Bmatrix} f_i \\ f_b \end{Bmatrix} \quad (5)$$

아래 첨자  $i$ 는 각 부수 영역 내부의 노드를 의미하며  $b$ 는 각 부수 영역의 계면(interface)을 의미한다.  $[k]$ ,  $\{f\}$ ,  $\{u\}$ 는 각 부수 영역의 강성 행렬, 하중 벡터, 변위 벡터이다. 식 (5)는 내부 영역에 해당하는 자유도를 소거함으로써 다음과 같이 압축(condensation)시킬 수 있다.

$$\begin{aligned} [\bar{k}_{bb}] \{u_b\} &= \{\bar{f}_b\} \\ [\bar{k}_{bb}] &= [k_{bb}] - [k_{bi}] [k_{ii}]^{-1} [k_{ib}] \\ \{\bar{f}_b\} &= \{f_b\} - [k_{bi}] [k_{ii}]^{-1} \{f_i\} \end{aligned} \quad (6)$$

그리고 식 (6)을 모든 부수 영역에 대하여 합성(assembly)하면 하나의 전체 상호 경계 행렬 방정식을 얻을 수 있고 이를  $\{u_b\}$ 에 대하여 풀면 그 값들로 각각의 부수 영역의 내부 영역 노드의 변위를 다음 식으로 구할 수 있다.

$$[k_{ii}] \{u_i\} = \{f_i\} - [k_{ib}] \{u_b\} \quad (7)$$

상호 경계 방정식을 풀 때 이용하는 forward/backsolve 과정은 계산 시간이 매우 짧은 반면 데이터의 이동이 많으므로 병렬화에는 적합하지 않다. 따라서, 이 과정은 하나의 연산자에서 수행하도록 하였다.

ITAS3D에서 전체 강성 행렬은 항상 대칭성을 갖지만 때때로 positive definite 하지 않기 때문에 식 (7)을 위해 다음의 LDU 인수분해(factorization)를 이용한다.

$$\begin{aligned} [k] \{u\} &= \{f\} \\ [k] &= [L][D][U] \end{aligned}$$

$$\{u\} = [U]^{-1} [D]^{-1} [L]^{-1} \{f\} \quad (8)$$

위에서 대칭 강성 행렬의 경우에는  $[L]=[U]^T$  이므로 응축 과정을 거치면 아래와 같이 유도할 수 있다.

$$\begin{aligned} [k_{ii}] &= [U]^T [D] [U] \\ [\bar{k}_{ib}] &= [U]^{-T} [k_{ib}] \\ [\bar{k}_{bb}] &= [k_{bb}] - [\bar{k}_{ib}]^T [D]^{-1} [\bar{k}_{ib}] \\ \{\bar{f}_i\} &= [U]^{-T} \{f_i\} \\ \{\bar{f}_b\} &= \{f_b\} - [\bar{k}_{ib}]^T [D]^{-1} \{\bar{f}_i\} \end{aligned} \quad (9)$$

따라서 각 부수 영역마다의 내부 노드의 변위 벡터  $\{u_b\}$ 는 아래와 같이 구할 수 있다.

$$\begin{aligned} \{\bar{f}_i\} &= \{\bar{f}_i\} - [\bar{k}_{ib}]^T \{u_b\} \\ \{u_i\} &= [U]^{-1} [D]^{-1} \{\bar{f}_i\} \end{aligned} \quad (10)$$

### 3. 계산 하중 균형을 고려한 영역 분할법

전체 계산 시간을 줄이기 위해서는 각 연산 장치에서의 계산 시간을 최소화함은 물론이고, 특히, 직접 해법을 이용한 경우, 연산장치끼리의 계산 하중(compute load)이 균형을 이루어야 한다. 이를 위하여 먼저, 각 부수 영역마다 절점들의 번호를 새로 부여하여 부수 영역 방정식의 프로필을 최소화하고, 연산 횟수(operation count)에 근거한 계산 하중의 균등 배분을 고려하여 영역 분할법을 실행하고 또한 상호 경계 방정식의 크기를 최소화한다. Fig.1은 부수 영역 강성 행렬의 전형적인 구조를 보여준다.

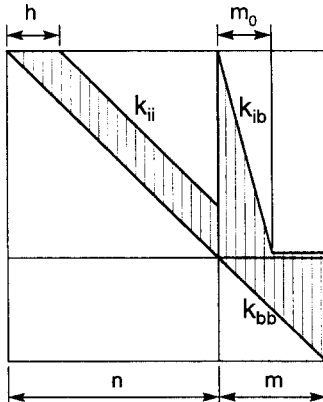


Fig. (1) 부수 영역 강성 행렬의 일반적 형태

이와 같은 강성 행렬을 압축하고 합성하는 과정이 전체 계산 시간의 상당한 부분을 차지하므로 계산 시간의 정확한 예측을 위해 이 과정의 연산 횟수를 대략적으로 아래와 같이 추정할 필요가 있다.<sup>(5)</sup>

$$C = c^{el} e + h^2 (n - \frac{2}{3} h) + 2(n-h)mhf + \frac{4}{3} nm^2 f^2 \quad (11)$$

여기에서  $e$ 는 부수 영역 내부 요소의 수이며  $c^{el}$ 은 강성 행렬의 계산에 대한 연산 횟수이다. 또한,  $f$  ( $0 < f < 1$ )는  $[k_{ib}]$ 의 fill factor로서

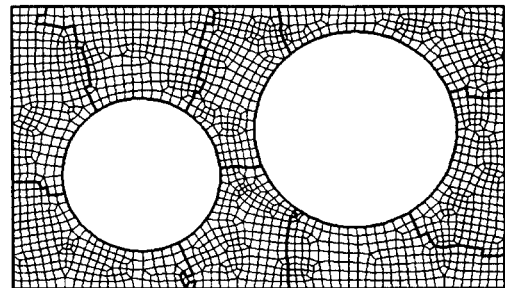
$$f = \frac{\sum_{i=1}^m h_i}{mn} \quad (12)$$

와 같이 주어지는데  $h_i$ 는  $i$ 번째 열의 높이를 의미한다.  $n$ 은 내부 절점들의 수이고  $m$ 은 상호 경계 상의 절점 수이며  $h$ 는  $[k_{ii}]$ 의 반(半) 밴드 폭이다. 전체 영역을 모두 같은 수의 요소를 갖는 부수 영역으로 나누는 것이 계산 하중의 균등한 분배를 의미하는 것은 아니므로 위와 같은 연산 횟수를 고려하여 영역을 분할하면 일반적으로 각 부수 영역마다의 요소들의 갯수는 약간씩 다르게 된다. 균등한 계산 하중을 부여하는 조건은 다음과 같이 얻어질 수 있다.

$$\begin{aligned} C_1 - C_2 &= 0 \\ C_2 - C_3 &= 0 \\ &\dots \\ C_{s-1} - C_s &= 0 \end{aligned} \quad \text{그리고, } \sum e_i = E \quad (13)$$

여기에서  $s$ 는 부수 영역의 수이며  $e_i$ 는  $i$ 번째 영역의 요소의 갯수, 그리고  $E$ 는 전체 영역의 요소 숫자이다. 이 비선형 방정식을 영역 분할 과정의 마지막 단계에 도입하여 풀면 계산 하중이 균등하게 분배된 분할 영역들을 구할 수 있다.

아래의 Fig. 2에서는 불규칙적인 요소망을 8개의 부수 영역으로 분할한 경우, 요소 숫자가 모두 같은 단순 분할과 계산 하중을 고려한 균형 분할의 예를 보였다.



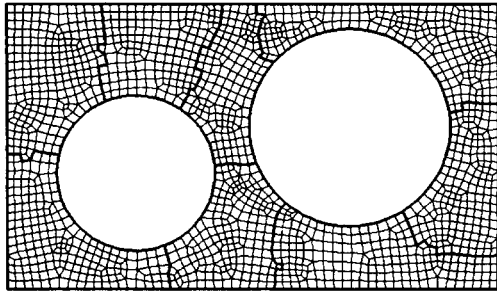


Fig.(2) 불규칙한 요소망의 8 개 부수 영역으로의 (위) 단순 분할과 (아래) 균형 분할

#### 4. 예제

본 논문의 예제를 위하여 최대 12 개의 연산자를 갖는 IBM SP2 (RS6000)를 dedicated 모드에서 사용하였고 코드의 병렬 포팅을 위해 IBM MPL (Message Passing Library)을 이용하였다.<sup>(6)</sup> 병렬화 된 프로그램의 수행성을 비교하기 위하여 사각컵 문제를 풀었고 병행 결과는 Fig. 3 에 보였다. 펀치의 이동 거리는 5mm 이며 금속 판의 크기는 150mm 이다.<sup>(7)</sup> 사용된 전체 요소의 수는 256, 1024, 2304, 4096 이며 각 경우 모두 4 개의 축소된 헬요소를 사용하였다.

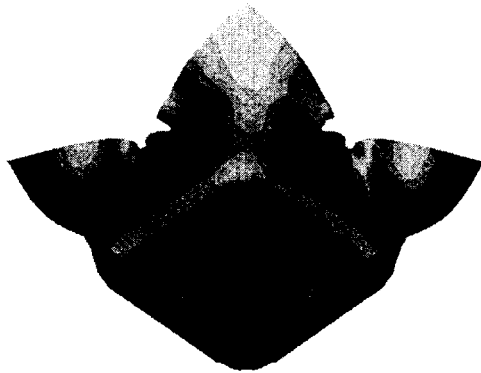


Fig.(3) 사각컵 문제의 변형 결과

계산 시간의 비교를 위해서 Speedup 이란 지수를 도입하였는데 순차(sequential) 계산 시간을 병렬 계산 시간으로 나눈 값이다. 각 요소 수와 연산자 수별로 Speedup 의 결과를 Fig.4 에 보였다. 8 개와 12 개 연산자를 사용할 때 요소 수가 증가할수록 수행성이 꾸준히 증가함을 알 수 있다. 그러나 연산자의 수가 증가할수록 확장성이 저하되는데 이는 앞에서 지적하였듯이 직접 해법을 사용하였기 때문이다.

#### 5. 결론

판 금형 해석 전문 유한요소 프로그램인

ITAS3D 의 병렬화를 수행하였다. 계산 하중을 고려한 효율적인 영역 분할법을 사용하였으며 그 결과를 몇 가지 요소별 그리고 연산 장치 수별로 비교하였다. ITAS3D 에서는 직접 해법을 이용하므로 많은 수의 연산자를 필요로 하지는 않는다. 사각컵 문제를 통하여 병렬화에 따른 확장성이 연산 장치의 수가 8 개일 때 가장 우수하고 12 개부터는 수행 효율성이 다소 떨어짐을 확인하였다.

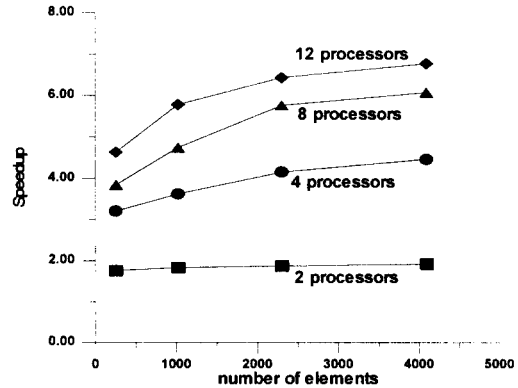


Fig.(4) 병렬 계산의 Speedup 비교

#### 참고문헌

1. A. Makinouchi and M. Kawka, "Process Simulation in Sheet Metal Forming", Journal of Materials Processing Technology, Vol. 46, pp. 291-307, 1994.
2. I. Babuska and H. C. Elman, "Some Aspects of Parallel Implementation of Finite Element Method on Message Passing Architectures", Journal of Computational and Applied Mathematics, Vol. 27, pp. 157-187 1989.
3. R. M. McMeeking and J. R. Rice, "Finite Element Formulation for Problems of Large Elastic-Plastic Deformation", International Journal of Solids and Structures, Vol. 11, pp. 601-616, 1975.
4. Y. Yamada, N. Yoshimura and T. Sakurai, Int. J. Mech. Sci., Vol. 10, pp. 343-354. 1968.
5. G. P. Nikishkov, A. Makinouchi, G. Yagawa and S. Yoshimura, "Performance Study of the Domain Decomposition Method with Direct Equation Solver for Parallel Finite Element Analysis" Computational Mechanics, to appear, 1997.
6. IBM AIX Parallel Environment : Programming Subroutine Reference (SH26-7228), IBM, 1994
7. A. Makinouchi, E. Nakamachi, E. Onate and R. H. Wagoner (Eds.), Numisheet '93, 2<sup>nd</sup> International Conference, Isehara, Japan, 1993.