

共同利用(産業利用トライアルユース:先端研究基盤共用・プラットフォーム形成事業  
『みんなのスパコン』TSUBAMEによる日本再生)  
成果報告書 平成27年度 新規利用拡大「商用アプリバンドル型トライアルユース」

産業用ゴムベルトの有限要素法による構造解析  
Structural analysis of industrial rubber belts by finite element method

徳田 明彦  
Akihiko Tokuda

三ツ星ベルト株式会社  
Mitsuboshi Belting LTD.  
<http://www.mitsuboshi.co.jp>

当社は自社製品である産業用ゴムベルトの開発に有限要素法による構造解析を利用しているが、近年、大規模化・高速化のニーズが高まっている。そこで平成 25 年度に同課題名にて商用アプリバンドル型トライアルユースを利用し、Marc の領域分割法と GPU 利用の組み合わせによる高速化の試行を行ったところ、単体コアのみ利用時に比べ約 27 倍の高速化を図る事が出来た。本課題では領域内の計算にスレッド並列を追加することで、前回よりさらなる高速化を実現できた。また、前回とは異なるベルトシステムを対象とし、複数の異なる境界条件を適用したところ、いずれも安定して解が得られた。これらの結果を報告する。

The structural analysis by finite element method has been used in the development of industrial power transmission rubber belt products in our company. Recently, the need for large-scale and high-speed analysis has been increased. Therefore, we tried to speed up the analysis by using DDM (domain decomposition method) of Marc and GPU in 2014, and then speed of the analysis was 27 times faster than that in the single core used. In this study, we have achieved higher analysis speeds than the previous by adding thread-level parallelism to the DDM. In addition, we got results stably at plural conditions in the analysis of different belt system from the previous study.

*Keywords: Belt, Rubber, FEM, Marc, DDM, GPU*

## 1. 背景と目的

当社は自社製品である産業用ゴムベルトの開発において有限要素法による構造解析を実施し、性能および耐久性を評価している。しかし社内の計算機には製品全体を高精度な要素分割で実用的な時間内に計算する能力が無く、精度を妥協するか、製品の一部分モデルでの計算にとどまっている。そこで平成 25 年度の TSUBAME トライアル利用(以下「前回」とする)において、無段階変速機用ベルト(変速ベルト)の検証例題を商用アプリの Marc を用いて実施し、製品全体を高精度で高速に解析できる可能性を検討した。その結果、領域分割法と GPU 利用を組み合わせる事で、単体コア実行時に対して最大約 27 倍の計算速度が得られた。しかし、Marc が有する並列機能の全てを試行できず、更なる高速化の余地が残されていた。そのため本課題においては、引き続き Marc を用い、領域内においてスレッド並列を併用することで前回よりさらなる高速化を

検討した。また、実際のものづくりにシミュレーションを活用する場合、様々な種類の製品の解析について、計算速度(納期)、計算精度(品質)を安定して得られることが重要となる。そこで前回とは別のベルトシステムである V リブベルトの検証例題を用い、複数の異なる境界条件下において計算し、いずれも安定して高速に計算が出来るか、検証した。

## 2. 変速ベルト解析の高速化(概要)

図 1 に変速ベルトの断面画像および有限要素モデルを示す。これは前回使用したモデルと同一であり、要素数は 1,177,600、節点数 1,185,602 である。ベルトは幅方向に対称形状であるので、対称面から半分のみをモデル化し、帆布は省略した。また、心線中心部には張力を受け持つトラス要素を配置している。図 2 は解析手順である。これも前回と同一である。

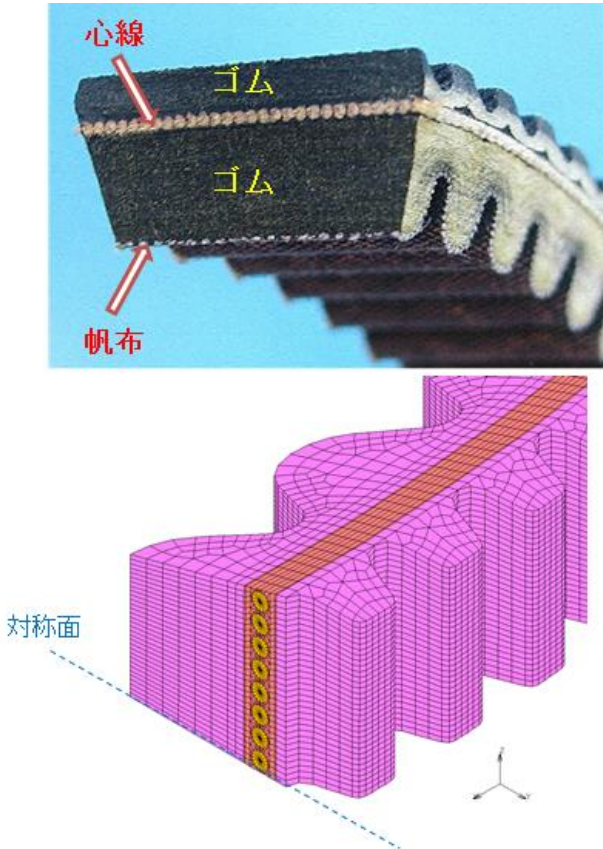


図1 変速ベルト(断面写真と有限要素モデル)

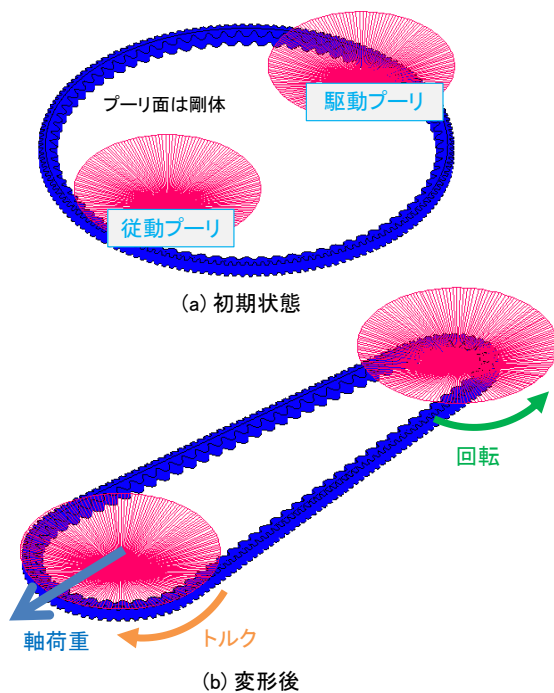


図2 解析手順

図 2(a)は初期状態である。駆動プーリおよび従動プーリは剛体面とし、ベルト近傍に配置する。図 2(b)は変形

後である。従動プーリに軸荷重を与えてベルトを張り、更に負荷トルクを追加する。その状態で駆動プーリを回転させ、ベルトの運転状態を模擬する。本来、摩擦力が定常になるまでプーリを回転させる必要があるが、これに多大な計算時間を要する。今回は計算速度検証目的の試行であるため、駆動プーリを回転開始から 78° 回転させた時点で打ち切る事にした。

図 3 は解析結果出力の一例で、駆動プーリ内でのベルトの Mises 応力分布である。この応力値をなるべく下げることがベルトの長寿命化に繋がる。

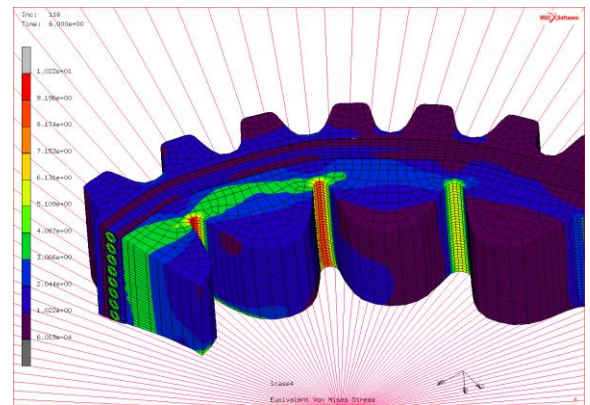


図3 結果出力例(Mises 応力)

上述の解析について、前回 Marc の領域分割法 (DDM)とソルバーの GPU 支援を組み合わせる試行したところ、図 4 に示すとおり、48 並列・GPU 有りで、単体コア・GPU 無しの約 27 倍の計算速度が得られた。

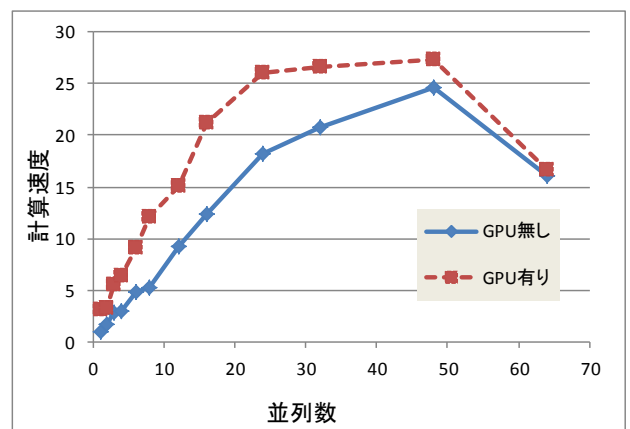


図4 前回の試行結果(DDM & GPU)

今回、さらなる高速化を目指し、前回の設定に加え、各領域内でのスレッド並列化を組み合わせる試行した。

前回・今回ともにHキューを用いたため、1ノード当たりのGPUは3基である。Marcは1領域あたり1基のGPUを割り当てるため、1ノードは最大3領域までしか受け持つ事が出来ない。前は領域内の計算について並列化を行わなかったため、物理CPUコアは1ノードあたり3コアの稼働にとどまり、残りの9コアは使われない状態であった。そこで今回、この残りコアを活用するため、領域内を並列化するオプションを追加して試行した。

1ノード(12CPUコア・3GPU)が3領域の受け持ちなので、CPUコアは1領域あたり4まで使用できることになる。MarcのDDMと領域内のスレッド並列を併用する場合、「剛性マトリクス組立および応力計算(剛性・応力)」部分と、「ソルバー」部分を分けてそれぞれにスレッド並列数を指定できる。そのため、DDM分割数、GPU使用の有無と合わせると膨大な数の組み合わせになり、その全てを網羅して試行する事は難しい。そこで、DDM分割数を24と48に絞り、領域内スレッド並列は①並列無し(前回の設定)に加え、②「剛性・応力」のみ4並列、③「剛性・応力」4並列+「ソルバー」2並列、④「剛性・応力」4並列+「ソルバー」4並列、の4通りを試行し、計算時間を相対的に比較した。

### 3. 変速ベルト解析の高速化(結果および考察)

結果を図5に示す(ただし、DDM24並列の①②は未実施となった)。

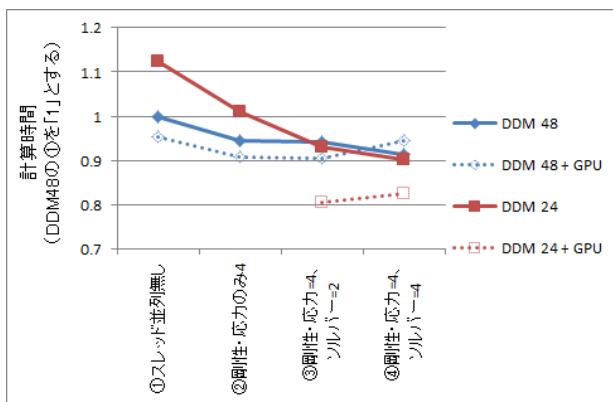


図5 領域内スレッド並列の効果

最も高速となったのは、DDM24並列+GPUで、③「剛性・応力」4並列+「ソルバー」2並列とした場合であ

った。前は①のみの実施で、GPUの有無にかかわらずDDM24並列よりもDDM48並列の方が高速であったが、領域内スレッド並列の併用によってこれが逆転した。つまり、DDM並列数の効果が頭打ちになっていない24並列においてスレッド並列の効果が大きい。ただしソルバーのスレッド並列は多すぎると逆効果である事が分かる。今回の結果を前回の結果である図4に追加すると図6となる。

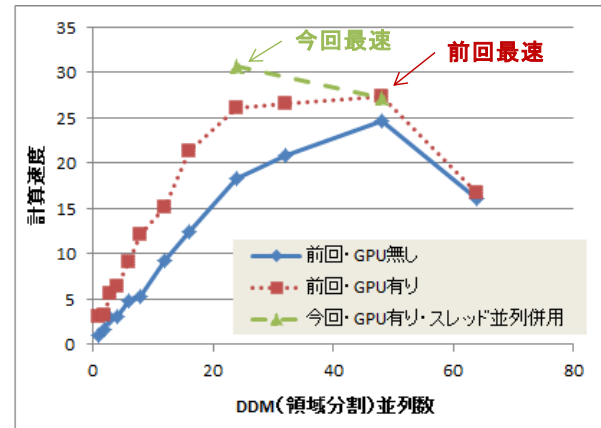


図6 前回からの高速化

前回の最速ケースから12%の高速化となった。このようにスレッド並列との組み合わせでは、DDM並列数を減らしたほうが高速になる場合もあることが分かった。この場合、DDM並列だけの場合と比べて計算機資源(CPUコア)を無駄なく活用できるメリットがある。前回最速ケース(DDM48並列)では1JOBの実行に16ノード必要だが、今回最速のケースでは半分の8ノードで済むので、計算機利用コストの観点からは100%の高速化(2倍速)に相当する。更に1JOBが12%高速化するので、実質124%の高速化に相当する成果が得られた。

### 4. Vリブベルトの4軸解析(概要)

図7に示すように、Vリブベルトはプーリとの接触面にV字の突起(リブ)を数個横に並べた形状であり、それが同じくV字状の溝を並べたプーリに巻きかかる事で、平ベルトに比べて狭い幅で大きな動力を伝達できるベルトで、主に自動車エンジンの補機類の駆動に用いられている。今回、このベルトを図8に示すように4つの

プーリ(4軸)間に張り、回転させる解析を行った。



図7 Vリブドベルトとプーリ

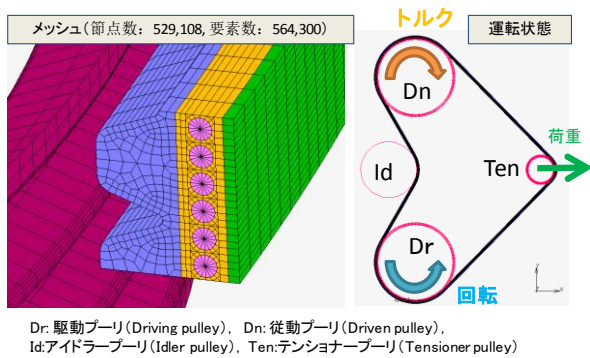


図8 Vリブドベルトのメッシュと運転状態

ベルトは対称性により幅方向に半分のみで、プーリ面(剛体)とベルト間の摩擦係数は0.3である。Ten プーリに荷重、Dn プーリにトルクを与えてベルトを張り、Dr プーリを摩擦の状態がほぼ定常になるまで、約 210° 回転させた。各条件は表 1 に示す組み合わせで実施した。計算には H キューを用い、2 ノードにて DDM24 並列、GPU なし、領域内スレッド並列無しで実施した。

表 1 Vリブドベルトの 4 軸解析条件

No.	条件名	Dn 負荷 トルク [N・m]	Ten 荷重 [N]	Ten プーリ径 [mm]	Ten V溝 角度[°]
1	標準	17.1	559	45	40
2	Dn 負荷トルク大	<b>34.2</b>	<b>730</b>	45	40
3	Ten 荷重大	17.1	<b>870</b>	45	40
4	Ten 径大	17.1	559	<b>55</b>	40
5	Ten V 角度狭	17.1	559	45	<b>38</b>

## 5. Vリブドベルトの 4 軸解析(結果)

結果、表 1 全ての条件について、それぞれ 80 時間～120 時間程度で正常に計算が終了した。結果出力の一

例として、図 9 に Ten プーリ上における Mises 応力分布図を示す。荷重条件やプーリ形状による応力状態の変化は妥当なものである。

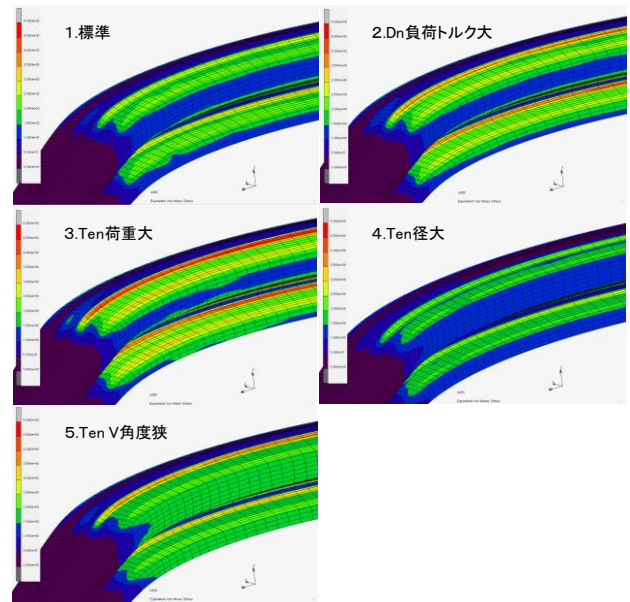


図9 Vリブドベルト Ten プーリ上 Mises 応力分布

## 6. まとめ

(1) TSUBAME において、FEM 解析ソフトウェア Marc の領域分割法(DDM)による並列化と領域内スレッド並列を組み合わせる事により、前回比で 12% の高速化を半分のノード数で実現する事が出来た。これは、計算機資源の利用効率を考慮すると、124% の速度向上に相当する成果となる。

(2) Vリブドベルトの 4 軸解析を複数の条件で実施し、いずれも想定内の時間で安定して適切な解を得る事が出来た。このことから、TSUBAME と Marc によるベルトの大規模解析は、納期厳守が必須である企業内での設計支援業務に安心して適用出来る事が確認出来た。

## 謝辞

本課題へ多大なご助力を頂きました、東京工業大学学術国際情報センター共同利用推進室、エムエスシーソフトウェア株式会社の皆様に感謝申し上げます。