

マクロレベルスケジューリングのための Lagrange 分解・調整法

成松 克己

1. はじめに

サプライチェーンマネジメント (SCM) で最も重要なのは、在庫を生み出している原因である「製品が流れるスピード」をコントロールすることである。このためには、製品に対して必要なさまざまな工程がうまく同期を取りながら生産が行われなければならない。特に、企業をまたがって生産活動が行われる場合には、製品毎に注目してそれぞれの足並みをそろえる作業が必要になる。大量生産の場合には、ラインキャパシティをうまく設計すれば個別企業内での管理でも足並みをそろえることができるが、受注生産 (BTO: Built To Order) [1] のように部品段階から紐付けられた細かいオーダーが入る場合には製品オーダーに注目した一段上位のレベルでの管理が必要であろう。近年ダイナミック MRP (Dynamic Resource Planning) やメタ製番管理 [2] などにより効果的に製品の流れをコントロールすることが主流になっているが、基本的には自社が管理する範囲のデータをすべて集めて制御することが念頭におかれている。本稿ではより階層的なデータの流れと意思決定のしくみを提案したい。

例えば図 1 のように 1 次加工業者、2 次加工業者、組立業者という 3 段階で生産が行われている場合、BTO ではオーダーが入ると 1 次加工業者から製造指示が入る場合もある。こういった場合には、それぞれの業者において、全体を通した視点から適切な時間にリソースが確保されないと足並みは崩れ、オーダーに対する納期回答とその保証が難しくなる。このため、1 次加工業者から組立業者までの全体を見てそれぞれの企業の時間配分を制御する仕組みによって全体の足並みを効果的にコントロールする必要がある。

一方、近年 Lagrange 分解・調整法によるスケジューリングが注目されている。この手法は制約条件を緩和し、緩和量に対してペナルティを課しながら収束計

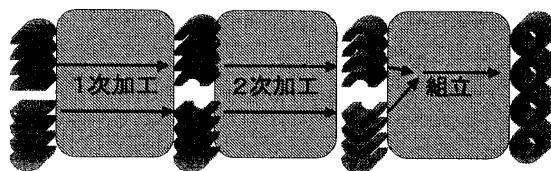


図 1 企業をまたがった生産

算することにより制約条件下での最適化を行うものである。実際に適用された例もいくつか見受けられるが、まだ一般に広まっているとは言い難い。

本稿では、上位レベルの意思決定をサポートするマクロレベルのスケジューラ [3] の概要と実用イメージ、Lagrange 分解・調整法との関係や今後の課題について紹介する。まだ実験段階ではあるが、マクロレベルスケジューリングと Lagrange 分解・調整法の組み合わせが今後の製造業にとって非常に重要な技術となることを示す。

2. マクロレベルスケジューリング

先に述べたような企業をまたがった生産活動において各企業の足並みをうまく調整するために、製品オーダーに注目した上位レベル (マクロレベル) と企業内の製造計画である詳細レベルの 2 つの階層を考える。階層は原理的には何階層あってもよいが、階層をあまり増やすと精度とスピードが落ちる。

詳細レベルのスケジューラは基本的にはマクロレベルから与えられたオーダー毎の投入日と納期にしたがって製造計画を作成し、状況を上位レベルに返すものであり、これまで数多く研究されているスケジューリングの枠組みに入るものである。しかし、マクロレベルのスケジューラには、大雑把な負荷状況を見ながら各オーダーの時期を調整する働きが必要となるため、従来の固定期間アクティビティのスケジューリングでは多少無理が生じる。

ここで、それぞれのアクティビティの期間を変動させながら最適化できれば、全体として効率的に生産を

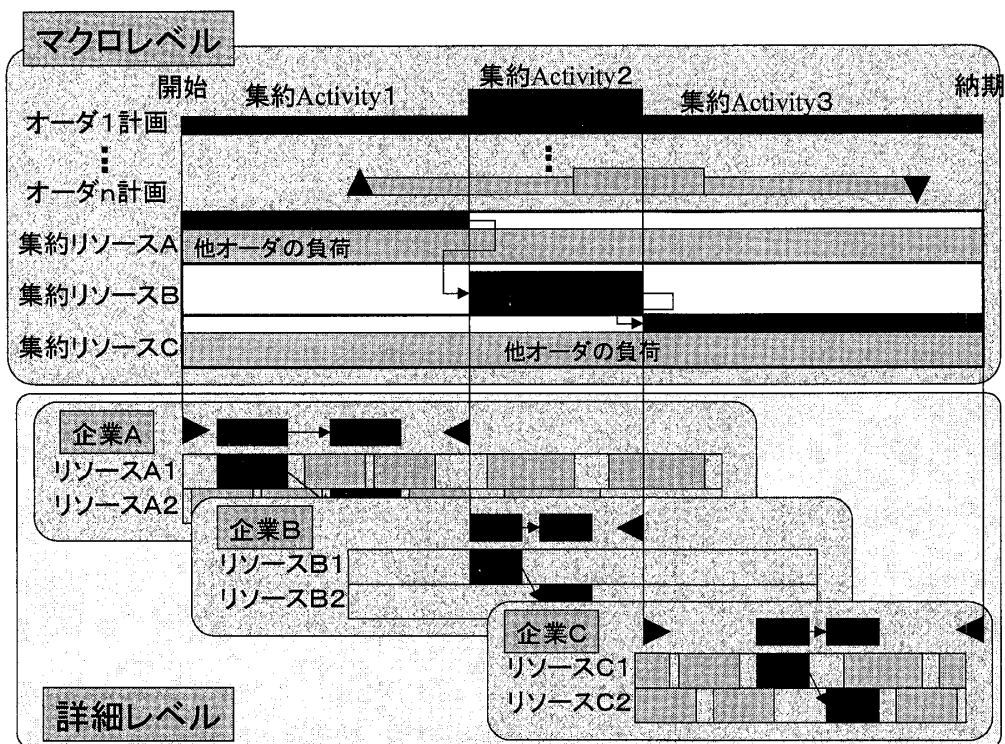


図2 マクロレベルと詳細レベル

コントロールすることができる。図2は、このような考え方に基づくマクロレベルと詳細レベルのスケジューラがどのように構成されるかを示したものである。異なる企業間を例にとっているが、このうちのいくつかの企業が社内の異なる部門であっても同じことができる。

図2では、製品の受注と納品に責任をもつマクロレベル、実際に製造を行う詳細レベルの企業A、B、Cが存在する場合の状況を、オーダー1のみに注目して負荷とスケジュールの調整を行う模様を表している。マクロレベルでは企業毎に集約された集約アクティビティおよび集約リソースに対し、前後の制約および他のオーダーとの兼ね合いを考慮して開始時刻と終了時刻を決定する。これに対し、詳細レベルに相当する各企業では、設定された開始、終了時刻を満たすように内部のスケジューリングを行う。この際、詳細レベルで調整不可能な場合にはマクロレベルにその旨を通知し、企業毎の開始終了時刻を再調整することになる。

このマクロレベルスケジューリングは、従来MRPによって行われていた中長期の生産計画（中日程と呼ばれることが多い）が持っていた役割の一部を果たす。大雑把な工程の期間をそれぞれ企業の負荷によって変動させながら他オーダーとの時期の調整を行うという点と、詳細レベルを扱う企業それぞれがある程度独立し

た意思決定を行うという点が特徴である。

2.1 集約リソース、集約アクティビティ

このモデルは、企業内で対象とする製品に関連するリソースを1つまたは数個の集約されたリソースとして表現することに重点が置かれている。ネック工程がはっきりしている場合には、そのネック工程で生産量がほぼ決まってしまうため、ネック工程を基準にして集約リソースを設定する。

再調整を少なくするには集約リソースを少なめに見積もって余裕分を確保すればよいが、稼働率は低くなる傾向になる。逆に稼働率を上げるために集約リソースを多めに見積もると余裕分が少なくなり、上下のレベル間での再調整が頻繁に発生するようになる。負荷をみながらこのあたりをうまくバランスさせることが非常に重要となる。

集約リソースが詳細レベルの企業のリソースを総括するのと同様に、集約アクティビティは対応する製品などに対して企業内で発生する作業を1つまたは数個に総括したものである。集約アクティビティの期間は変動させることができ、それに応じて消費する集約リソースの（単位時間あたりの）使用量も連動する。最も単純な例は、期間×使用量を作業量として固定で見積もり、期間と使用量を反比例させるもの（図3）で

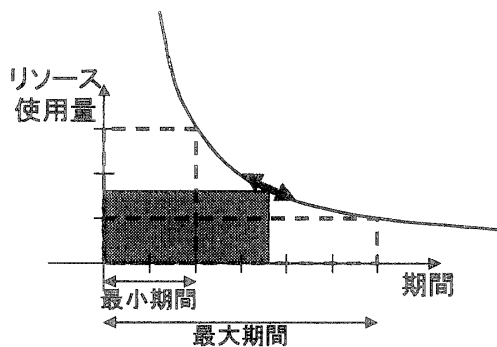


図3 期間と単位時間あたりの使用量

ある。

また、各集約アクティビティには最大期間と最小期間が与えられ、その範囲内で面積を一定にして期間を変動させることができる。期間が短くなったときに段取りなどのマージンがとれなくなり、クリティカルな状況が発生するような場合には、期間が短くなると作業量見積もり、すなわち長方形の面積も増えるようなモデルにすることも可能である。

2.2 決定変数と制約条件

マクロレベルスケジューラは、対応する製品オーダーに対してそれぞれの企業内を通過する時間的配分、すなわち集約アクティビティの開始時刻および終了時刻を決定する。また、アサイン可能な複数の企業から負荷やコストを考慮してアサインする企業を選択することもできる。

制約条件としては、基本的には各集約リソースのリソース容量制約と各アクティビティの開始/終了日に関する時刻制約を扱う。

容量制約は制約条件として扱われるが、もともと集約リソースの容量制約はあくまで目安としてのものである。従ってある程度の負荷バランスがとれていれば良い。詳細レベルでスケジュールしたときにあふれたオーダーがある場合には、そのオーダーに対する時刻変更をマクロレベルに申し入れ、再度調整を行うことにより解決される。この部分がこのスケジューラの最も大きな特徴である。

各アクティビティの開始/終了日制約については、すべてのアクティビティに制約がつけられるわけではない。制約条件の主な目的は、納期回答後のオーダーの確定である。遠い将来のものに関してはマクロレベルで自由に再スケジュールするが、ある程度時期が近づいてくるとあまり頻繁に変更するのは望ましくない。

このとき、詳細レベルでの実際の開始日と終了日をマクロレベルに戻し、この制約を守った上でマクロレベルのスケジューリングを行う。このやりとりについては後で詳しく述べる。

2.3 目的関数

マクロレベルのスケジューリングにおける代表的な目的関数として、本稿では次の2つの評価基準を紹介する。目的関数としては、各アクティビティに対するこれらの値に対して重み付き和をとるものとする。

1) (総)滞留時間

滞留時間は主に在庫コスト削減の目的で使用する。もちろん、製造工程であれば工程が進むにつれて中間生成物が大きくなり、製品全体を通した期間では在庫コストが均一でなくなるような場合もあるため、集約アクティビティ内もしくは集約アクティビティ間に対して在庫係数のようなものを用意することもできる。

2) 納期遅れペナルティ

オーダーの終了時刻から納期を引いた時間幅が正の値のときのみ2乗したものを納期遅れペナルティとして扱う。

3. マクロレベルスケジューラの使用法

これまで機能的な面を中心に説明してきたが、ここで実際にどのように使われるのかというイメージを紹介する。

重要なポイントは、オーダーの確定と計画を修正する各レベル間のやりとりと、上位レベルでの受注および納期回答の考え方である。

3.1 オーダーの受注および納期回答

受注時には納期を設定して実現可能性を検証したり、納期をかなり後にずらして前詰めした場合に納期がいつになるかをシミュレーションすることができる。このとき、すでに確定されたオーダーの負荷は全く変更しないで新規オーダーのみの負荷調整や、確定オーダーのスケジュール修正許容量を目的関数や制約に組み込むことによりある程度のやりくりをすることも可能である。納期回答は基本的に詳細レベルで実現性を検証してから行うが、正確性をそれほど必要としない暫定的な納期回答はマクロレベルのスケジューリングだけでも可能である。

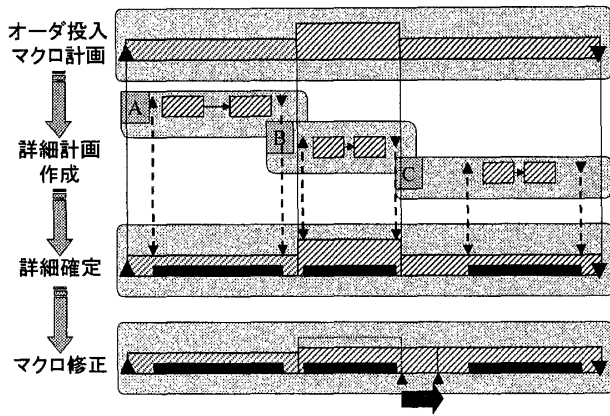


図4 オーダの確定とスケジュールの修正

3.2 オーダの確定と計画の修正

図4にオーダーの確定とスケジュールの修正を行う過程を示す。オーダーが投入され、まずマクロレベルで大雑把な負荷調整をもとに期間割当を作成し詳細レベルに提示する。次に、この期間割当をもとに詳細レベルで実際の製造計画を作成し、実現可能性をチェックする。必要に応じてマクロレベルの再調整を行う場合もある。ある程度時期が迫って来ると、オーダーやアクティビティを確定する。このとき、確定された集約アクティビティに関して実際に立てた製造計画に若干の余裕分を加えた期間を詳細レベルでの確定必要期間としてマクロレベルのスケジューラに返す。マクロレベルは必要に応じて計画を修正する際にこの確定必要時間を必ず含んだ期間割当を再度作成し、提示する。図4ではBの企業に新しい別のオーダーが入ったために、Bの企業の期間配分を増やして修正スケジュールを作成している。このように、マクロレベルでは詳細レベルの計画を包含するように期間配分し、詳細レベルではマクロレベルで計画された期間配分の範囲内で詳細計画を作成する。

4. マクロレベルスケジューリングと Lagrange 分解・調整法

4.1 Lagrange 分解・調整法の問題点

これまでのスケジューリング研究の主な領域はジョブショップ問題など作業機械レベルの製造計画が最も多く、Lagrange 分解・調整法もこの領域に対して多くの研究が続けられている。しかしながら、作業機械レベルでの製造計画において多くの場合は比較的厳密に定義されたリソース制約を守ることを要求される。例えば、品種変更による設備段取りを含む問題では、基本的に一つの機械は一度に一つのタスクしか処理す

ることができない。こういった問題に対しては、収束計算をベースとしたLagrange 分解・調整法では収束を保証するのが難しく、必ず計算した解をHeuristicsによって制約条件を満たすように解き直すが必要になる。Heuristicで解き直す際には、例えばリソース制約を緩和した場合には開始時刻順でソートしてその順序で並べるなどの方法が取られるが、この際にLagrange 分解・調整法による最適化の結果をできるだけ崩さず制約条件だけを満たす解を作成するのが非常に難しい。この手法がまだそれほど広まっていないのは、この部分が難しいためと考えている。

マクロレベルのスケジューラではアクティビティの期間を変動させ、変動した期間に応じて変動する負荷量を考慮した上でスケジュールの調整を行うものであった。この問題では、横軸に時間、縦軸にリソース使用量をとった図3のようなグラフにおける長方形の縦横の長さが変化する。

一般に、ディスパッチングルールによる解法では、縦横の長さが固定された長方形をリソース容量の中に埋めていくが、この形が変化する問題においては制御が非常に複雑になり、ルール自体も非常に複雑になる。厳密解法も決定変数が多く、計算時間の問題が大きい。しかし、Lagrange 分解・調整法を利用するとオーダー毎のサブ問題で動的計画法が使えるため[4]、これを拡張すれば変動する期間を考慮した最適化が可能となる。さらに集約リソースの容量制約が厳密なものではないという特徴があるため、収束しきれない部分をディスパッチで解き直すといった作業の重要性も低くなる。従って、Lagrange 分解・調整法は非常にマクロレベルスケジューラに向いている解法と言える。

5. 計算実験と課題

このLagrange 分解・調整法を用いたマクロレベルスケジューラは、まだ実験段階であるが、実装や実験結果の詳細については[4]に述べられている。ここでは、実験の内容と課題について簡単にまとめる。

5.1 計算時間

実験では、400 オーダ、4000 アクティビティ、500 時間程度の規模の問題を10分程度の時間で解き、良好と思われる解が得られている。この計算速度は、実験としては満足のいくものであるが、実用のためにはより大きな規模、特により広大な時間軸の計算が必要になる。これはアクティビティやオーダーをまとめるサ

イズの大きさには限界があるためである。計算時間の問題については、余分な計算を極力排除する細かな作業が必要であるが、さらに速度を上げるのは可能と考えている。また、サブ問題毎に独立に解けることを利用して並列計算機を利用して高速化を図ることも考えられる。

5.2 リソース集約方法

実験では、ごく簡単にリソースを集約しているが、実際にはかなり難しい作業となる。基本的にはボトルネックを中心にして見積もるが、プロダクトミックスの状況が変化するとボトルネックも変化するためである。このため、やはりある程度の需要予測をもとにプロダクトミックスの状況を予想したり、ボトルネックが発生する可能性のあるリソースを個別に扱うなどの対処も必要と考えられる。このあたりについては今後実例をもとにした実験を行って検証する必要がある。

6. おわりに

本稿では、企業や部門をまたがった生産活動におけ

る上位レベルの調整を行うマクロレベルスケジューリングについて紹介した。特に受注型非量産生産における生産計画の意思決定と管理に有効であり、Lagrange 分解・調整法との相性が良く、実用的な時間で仕掛かり在庫や納期充足について大雑把な最適化が可能になる。現在実験段階でありクリアしなければならない問題もあるが、今後実用化を進めていきたい。

参考文献

- [1] 野口, “超生産革命 BTO”, 日本能率協会マネジメントセンター刊, 1998 年
- [2] 山田, 奥村, “初めてのバリューチェーンマネジメント”, 工業調査会刊, 1999 年
- [3] Y. Zhang, P. B. Luh, K. Narimatsu, T. Moriya, T. Shimada, “A Macro-level Scheduling Method Using Lagrangian Relaxation”, Proc. of IEEE SMC, 1999
- [4] H. Chen, C. Chu, and J. M. Proth, “An improvement of the Lagrangean relaxation approach for job shop scheduling: a dynamic programming method”, *IEEE Transactions on Robotics and Automation*, Vol. 14, No. 5, pp. 786-95, 1998