

## エージェントに基づくサプライチェーンモデル

○谷口 憲 倉橋 節也 寺野 隆雄

筑波大学大学院 経営・政策科学研究科 企業科学専攻

東京都文京区大塚 3-29-1

Tel:03-3942-6816

Email:{taniguti,kurahasi,terano}@gssm.otsuka.tsukuba.ac.jp

あらまし 従来のサプライチェーンマネジメント (SCM) システムでは、それぞれのビジネス拠点の活動はグローバルな行動原理によって決定されている。しかし、なんらかの外乱がシステムに影響を与えるとき、その活動を適応的に変化させることは困難となる。その場合に、通常それぞれの拠点は、直前工程の情報だけを基に決定を変化させることになる。これに対し、本論文では、外乱に対してロバストなエージェントベースのSCMを提案する。知的エージェントとしてそれぞれの拠点をモデル化し、ブラックボードを介してそれぞれが情報交換を行うことができる。そして現在のSCMの状態を判断するために、情報エントロピーの概念を用いる。これは、各工程での情報の複雑さを表すものである。これらの考え方に基づき、進化的アルゴリズムを使用した学習機能を持つエージェントベースシミュレータを実装する。実験結果は、このシミュレータによって環境変化に対してロバストなSCMシステムをモデル化できることを示している。

キーワード

サプライチェーン、遺伝的プログラミング、マルチエージェント

## Agent-based Supply Chain Model

○Ken TANIGUCHI Setsuya KURAHASHI Takao TERANO

Graduate School of Systems Management, University of Tsukuba

3-29-1, Otsuka, Bunkyo, Tokyo

Tel:+81-3-3942-6816

Email:{taniguti,kurahasi,terano}@gssm.otsuka.tsukuba.ac.jp

Abstract In conventional supply chain management (SCM) systems, the activities of each firm are determined by a global optimization principle. However, it is very difficult to change the activities when some disturbance is input into the system. Such cases, each firm usually change their decisions based on the information of only other adjacent firms. Instead, this paper proposes a agent-based formulation of SCMs robust against the disturbance. We model each firm as an intelligent agent, which communicates each other via the blackboard architecture. To understand the states of the SCM system, we employ the concept of information entropy, which represents the complexity of the information they process. Based on the above ideas, we implement an agent-based simulator to learn 'good' decisions via evolutionary algorithms. Experimental results have shown that our simulator can model an SCM system robust against environmental changes.

key words

Supply Chain, Genetic Programming, Multiagent

## 1 はじめに

製造業の多くがサプライチェーンの再構築を急いでいる。今日の製造業の生産管理手法としては、「必要なものを、必要なときに、必要な量だけ、必要な場所に供給する」JIT(Just In Time)や、バケット型製造資源計画方式のMRP(Material Requirement Planning)、その発展型としてのMRP II(Manufacturing Resource Planning)、流通プロセスへ適用したDRP(Distributed Resource Planning)などが採用されている。ところが、これらの生産管理手法では対応できない事態に製造業は直面している。消費者の嗜好は日々変化し、市場での売れ行きに追隨できるサプライチェーンが求められている。パソコンメーカなどのダイレクト生産モデルに代表されるようなマーケットイン型事業モデルへの転換がそれである。

グローバルなサプライシステムにおいては、世界に広がるビジネス拠点の情報をリアルタイムに接続するネットワークがあつて初めて、グローバルな販売と生産、供給の同期化が実現される。在庫を最小にする限界のリードタイムで各生産拠点は対応し、納期どおりに出荷し、不測の事態には各工程の制約を踏まえた次善の最適供給計画が再構築される必要がある。また受注後のオーダー変更や需要予測の誤り、部品入荷遅れ、不良品発生など、さまざまな外乱・内乱が発生し、生産計画を乱すことになる。

このような事態に対し、サプライチェーンシステムは最適にマネージメントされなければならない。しかし、現状のSCM(Supply Chain Management)は世界に分散した拠点を高速なネットワークで接続することを前提とした中央集中型のアーキテクチャを取っている。拠点のリアルタイムな仕掛情報や販売情報を本社に設置したデータベースに集約し、最適な生産計画、需要予測を行う仕組みになっている。多くの企業が直面しているのは、労働集約型生産ラインの発展途上国への移転である。つまり、IT化の外に工場は次々と出て行っている。また、生産システムは拠点ごとに微妙に異なり、それぞれの中で独自にそして必死にBPR(Business Process Re-engineering)が行われている。これが、日本的な生産システムの特徴でもある。

このような自律分散的な生産システムの中で求められるSCMは、集中型ではない。それぞれの拠点で自律的に最適計画を目指しつつ全体最適をも考慮するような、自律分散型SCMが求められる。海外生産を含め生産・物流ラインが頻繁に変更される今日、その都度それぞれの拠点の特性や制約をすべて中央で管理し計画することは事実上不可能と云ってよい。ソフトウェアの維持費だけで膨大なものとなり、SCMの導入が一部の企業に集中する要因ともなっている。自律分散型SCMは、それぞれの特性はそれぞれで考え、かつ全体としての最適化を考慮する組織知能型のエージェント指向システムである。

## 2 エージェント技術によるSCM手法

SCMに知的エージェントを用いたモデルとして、資源分配をエージェント間のオークションに任せものがある[5]。市場動向に対応した生産環境の変化に素早く対応するために、サプライチェーン間の各部門が協調して資源分配を行う必要がある。このモデルではこれを解決するために知的エージェントを利用している。エージェントは、Management Agent, Account Agent, Resource Agent, Production Agentの4種類が存在する。それぞれ、全体管理のために線形計画法で資源分配を求めもの、製品と顧客を管理しオークションを行うもの、原料に関する需要予測と発注を行うもの、生産工程ごとに生産能力を管理するものである。Account Agentが将来にわたる内部利益を最大にするようにお互いにオークションによって資源を再調整を行っている。ここでは資源配分に重点がおかれているため、生産の同期化や仕掛在庫の削減などは対象外となっている。また、生産計画に対する実オーダーや生産ラインの乱れによる実績との差異などは考慮されない。このため、資源が十分あっても手待ちが発生したり、作りすぎによる仕掛の増加など、現実の生産現場で発生している問題に対応することが難しい。

アセンブリーラインのサプライチェーンを対象にマルチエージェントモデルを適応した研究[7],[4]では、進化シミュレーションツールのSwarmを用いて情報の価値を示している。従来の意思決定モデルは階層的な生産システムに効果を発揮したが、今日のサプライチェーンのような分散的な指示や制御を行う生産システムには、社会シミュレーションのようなテクニックが必要だとしている。生産形態を3タイプに分け、パソコンや家電のようなアセンブリーを主体とした生産ラインでシミュレーションを行った結果、オーダーに応じて最終組立を行う生産方式が他の在庫生産や受注生産などに比べて在庫コストとサイクルタイムの点で優れていることを見出している。そして、需要情報が在庫コストを減少させる(情報は在庫に置き換わる)としている。このモデルでは、サプライチェーンネットワークをオーダー、在庫情報、生産物流情報などで結合しているが、基本的な生産計画との結合がされていないため、需要情報や製造情報の扱いが現実的でない。実際の製造工程で広く用いられているMRPなどでは、これらの情報を前提に生産計画を立てている。

## 3 外乱制御のための情報量利用方式

本研究では、従来の中央集中型のSCMと異なり、各拠点に分散配置されたエージェントが自律的に生産計画を調整するモデルを提案する。また、実際の生産システムで用

いられている MRP, DRP を基本生産計画として用いることで、より現実的な SCM の実現を目指している。生産計画は図 1 に示したように、それぞれの工程で生産、需要、基準在庫をタイムバケット毎に計算して求めている。ここで需要とは後工程で生産に必要な中間品を表す。工程毎にあらかじめ決められた基準在庫を前提として生産数が決まる。ただし、仕掛製品は生産数の修正ができず、Time Fence 以降のタイムバケットに対してのみ生産計画の修正が可能となる。サプライチェーン全体は図 2 に示したような構成となる。このように MRP と DRP を結合したモデルでは、製造工程と同様に倉庫や販売拠点を扱うことができる。工程間の連結に示したように、後工程の生産数 P が前工程の需要数 S に相当する。また、図の中央に Web News が設置されているが、ここにはそれぞれの工程の PSI (生産、需要、在庫) 表が掲示されている。各工程はこの Web News を見ることで他の工程の生産状況を把握することができ、これらの情報をもとに自工程の生産計画を調整する。たとえば、プレス工程で生産遅れが発生すると、それは実績の変化に表れそれを他の工程が知ることができる。集中型の SCM ではこの Web News に相当する箇所ですべての生産調整を行っていたが、本モデルでは生産調整はそれぞれの工程に任せられている。このため工程の変更や工程間の連結の変更に対して必要な工程だけが自律的に生産調整を行うことができる。

		実績		仕掛	予定			
		N-2	N-1	N	N+1	N+2	N+3	N+4
生産	計画	100	+100	100	100	+40	100	150
	実績	90	120	100	100	70	100	150
需要	計画	-70	-70	-70	60	120	100	150
	実績	60	140	100	60	120	100	150
在庫	計画	-	-	+110	+150	70	70	70
	実績	100	80	80	120	-	-	-

基準在庫 (70個)

図 1: 生産計画と MRP

計画と実績の差の度合いはサプライチェーンの「複雑さ」と考えることができる [8]。計画通りに需要や生産実績が進めば、なにも SCM を行う必要はない。さまざまな外乱 (需要変動, 資材調達遅れなど) や内乱 (生産遅れ, 不良品発生など) が発生する結果、リアルタイムな生産計画の調整が必要となる。この計画値と実績値の差をできるだけ少なくすることが SCM の目的となる。本モデルでは、この複雑さを情報エントロピーとして捉え、これを情報量と

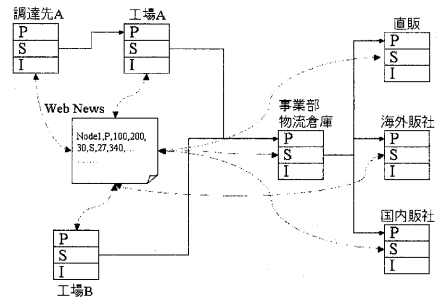


図 2: 自律分散サプライチェーンモデル

して次の式を与える。

$$H = - \sum_{i=1}^n p_i \log_2(p_i). \quad (1)$$

それぞれの工程は、MRP で与えられた生産計画だけでなく、Web News の情報をもとに自らの生産計画を調整し、実績と計画の差異を最小にしようとする。MRP や DRP だけでは、数期先の生産計画をもとに上流工程の計画を立てるため、上流になればなるほど不確定な需要予測に依存することになる [1]。そのため需要予測をできるだけ精度良く行うことに注目がいくが、現実の市場ではそのような予測は困難を極めていて、そこで、不確定な需要に対していかに柔軟に生産システムが対応するかが焦点となり、需要変動を見越した生産計画が求められる。いわゆる「工程の神様」と言われる微妙な生産調整を神業のように行う「工程マン」の知恵を本モデルでは再現しようとしている。

## 4 エージェントシミュレータ実験

各エージェントが需要の変動を受けた場合、どのように情報量が増えるかを示す。ここでは、サプライチェーンを工場/事業部/販社の 3 つのエージェントが構成し、工場で生産された製品が事業部の物流倉庫に入庫された後に販社向けに出荷される場合を考える。

### 4.1 Web News に掲載される基本情報

エージェントに関する基本情報は、以下のような Web News の内容として掲載される。

- 工場

- Agent name: a

- Safety stock: 0
- Logistics pattern: a --> b
- PSI information: 省略

● 事業部

- Agent name: b
- Safety stock: 100
- Logistics pattern: b --> c1
- Purchase lead time: 6 (day)
- PSI information:

```

Bucket : .., -2, -1, 0, 1, 2,..
P plan : ..,100,100,100,100,100,..
P results: .., 90,120, 0, 0, 0,..
S plan : .., 70, 70, 70, 60, 90,..
S results: .., 60,140, 0, 0, 0,..
I : ..,100, 80,110,150,160,..

```

● 販社

- Agent name: c1
- Safety stock: 0
- Purchase lead time: 1 (day)
- PSI information: 省略

PSI information の P,S,I とはそれぞれ Purchase(調達または生産), Sales(販売または出荷), Inventory(在庫)を表す(以下, 生販在を PSI と記述). Bucket(バケット)は PSI を管理する時間単位を表し, 通常, 月, 週, 日のいずれかの期間である. PSI information は PSI 実績/計画値をバケットごとに表した時系列データである. また, Safety stock は安全在庫数量, Purchase lead time は調達のための時間(この例では日数), Logistics pattern は物流の方向を表す.

## 4.2 情報量の計測

情報量は, 前述の PSI information における需要変動の度合いをサプライチェーンの複雑さとみなした場合の指標となる値である. 情報量の算出手順を以下に示す.

### 1. バケット別変動差分の算出

PSI information における P の情報を用いて実績値と計画値の差, および前回計画値と最新計画値の差をバケット別に求める.

### 2. グリッド別発生確率の算出

求められた変動差分をグリッドごとに割り振り, その発生頻度からグリッド別に対象範囲の変動差分が発生

する確率を求める. ここで, グリッドとは変動差分の範囲(例えば, グリッド幅が 10 であるとき, 0~9, 10~19, 20~29, ...)を表したものであり, あらかじめ定められているものとする.

### 3. バケット別発生確率の算出

バケット別変動差分をグリッド別発生確率に照らし合わせ, バケット別に変動差分が発生する確率を求める.

### 4. 情報量の算出

バケット別発生確率より, 前述の情報量算出式から情報量を求める.

## 4.3 需要変動時のエージェント間の情報伝播

例えば, 販社(エージェント c1)において, バケット 1(翌日)の 70 の販売計画数に対して, 100 のオーダーが発生した場合の各エージェント間の情報伝播は次のようになる.

1. 販社の PSI information のバケット 1 の S plan の値が 70 から 100 に変更となり, これに伴って同バケットの P plan の値も差分 30 だけ値が上乘せになる.
2. 販社から事業部への Purchase lead time は 1(出荷リードタイム 1 日)であるから, 事業部におけるバケット 0(当日)の S plan の販売計画数も 70 から 100 に変更になる. ここで, 事業部から工場への Purchase lead time は 6(生産リードタイム 6 日)であるから, バケット 5 までの事業部の P plan の変更は許されていない. したがって, この販売計画数の差分 30 はバケット 6(タイムフェンスに相当するバケット)に反映される. タイムフェンスバケットにおける事業部の P plan は, S plan と Safety stock の値によって決定される.
3. 事業部でのタイムフェンスバケット 6 の P plan の値は, 工場におけるバケット 0 の S plan の値に反映され, これに伴う工場の P plan の値も変更される.

このように, サプライチェーンの下流における需要情報の変動は上流に向けて伝播するため, このときの変動差分が各エージェントにおける情報量に影響を与える.

## 4.4 需要変動時の情報量の変化

各エージェントの PSI information(時系列データ)を設定し, 前節の情報伝播を発生させてシミュレーションを行った. 情報伝播の前後で, PSI 値およびバケット別発生確率がどのように変化したかを図 3,4,5 に示す. なお, 情報量算出の際に用いるグリッド幅は 10 とした.

情報伝播の前後での情報量の変化を表 1 に示す. 各エージェントはともに情報伝播後の情報量を増加させている. これは販社からのオーダー投入によってサプライチェーン

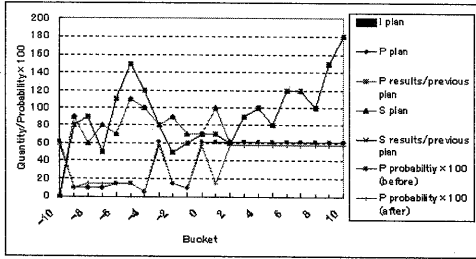


図 3: 販社 (エージェント c1) の需要変動

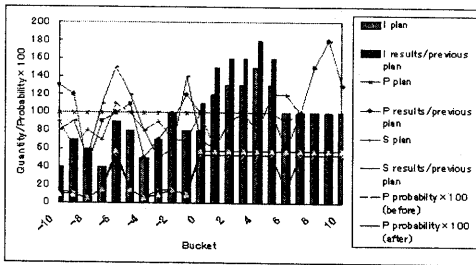


図 4: 事業部 (エージェント b) の需要変動

を構成するすべてのエージェントが変動の影響を受け、サプライチェーン自体の複雑さ、すなわち変動のあいまいさが増幅されたことを示している。今回のシミュレーションでは簡単のため販社の安全在庫数量をゼロとしたが、現実には事業部に対して安全在庫を加味した数量でオーダーする場合が多い。このような状況で複数の販社が存在する場合には、情報量の値はさらに増加することが予想される。

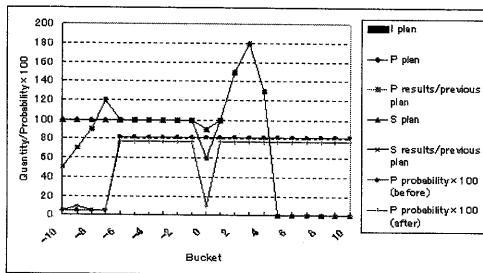


図 5: 工場 (エージェント a) の需要変動

エージェント名	情報伝播前	情報伝播後
c1(販社)	8.27	8.80
b(事業部)	8.44	8.85
a(工場)	5.03	6.06

表 1: 情報伝播前後での情報量

## 5 論理プログラミングによる GP 設計

本章では、SCMの複雑さを減少させるための手法として、ホーン節を用いた遺伝的プログラミングを提案する。探索空間が多峰性であるとき、木構造の多様性を維持するための機構を持たない GP での集団はすぐにある 1 つの局所解近傍へと収束する。木構造の多様性を維持するために、本研究では GP に Messy GA [2] 風の遺伝子表現、遺伝的操作を導入する。すなわち、遺伝子座のインデックスを利用して、木構造における多重定義、不足定義の処理を GP のメカニズムに採り入れる。多様性維持のための方法の 1 つとして sharing を導入した例を次節で述べる。木構造の表現方式としては前述のようにホーン節を用いるが、GP にて生成されるプログラムも同様にホーン節で表現されたものである。多重定義の遺伝子座に相当するノードを持つホーン節は、通常最初に現れたものを優先させるが、論理プログラミングによって実装することで、バックトラックによる優先度の低いノードを反映した別解も生成可能である。また、対象となる問題を複数のエージェントの協調によって解決させるといった場合に、すべてのエージェントが同一のプログラムによって動作する homogeneous モデルと、各エージェントが別々のプログラムを参照しながら動作する heterogeneous モデルの両者の設計を行った。

例えば、図 6 の木構造 (記号の右肩はノード番号) は、次のように記述される。

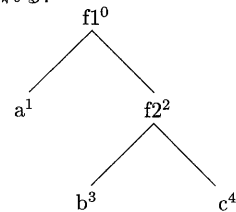


図 6: ホーン節に対応する GP の木構造

```

function(tree(1/agent1,0),f1) :-
terminal(tree(1/agent1,1),a),
function(tree(1/agent1,2),f2).
function(tree(1/agent1,2),f2) :-
terminal(tree(1/agent1,3),b),
terminal(tree(1/agent1,4),c).
terminal(tree(1/agent1,1),a).
terminal(tree(1/agent1,3),b).
terminal(tree(1/agent1,4),c).
  
```

ここで、述語 `function/2` は非終端 (関数) ノードを表現するのに用い、2 引数目にノード (`f1,f2`) を記述する。また、述語 `terminal/2` は終端ノードを表現するのに用い、同じく 2 引数目にノード (`a,b,c`) を記述する。集団に含まれる個体の番号 (木の番号) および Messy GA の遺伝子座のインデックスに対応するノード番号は、述語 `function/2,terminal/2` の 1 引数目において `tree`(個体番号/エージェント番号, ノード番号) の形式で表現する。ここで、このエージェント番号は `heterogeneous` モデルに対応できるようにするための識別子である。

このようなホーン節で表された木構造に対し、遺伝的操作をおおよそ次のように行う。

1. 可能な関数記号と終端記号を用い、ホーン節で表現されたランダム木の初期集団を生成する。ここで、`heterogeneous` モデルの場合、一個体はそれぞれのエージェントを表現する複数の木の集合を意味する。
2. 集団のそれぞれのランダム木の適合度を計算する。
3. 新しい集団を形成するために、選択、交叉、および突然変異を適用する。交叉では、2つの子を生成するために各親でランダムに交叉点 (ノード番号) を選び、その交叉点の下の部分木を交換する。ただし、`heterogeneous` モデルの場合には、対応する木 (エージェント) 同士で交叉を行う `Restricted Breeding` の方法をとる。突然変異では、ランダムに木のノードを選び、ランダムに生成された部分木とそのノード以下の部分木を交換する。
4. 終了判定を行い、条件を満たしていれば終了する。満たしていなければ、ステップ (2), (3) を繰り返す。

具体的な関数ノード、終端ノードは以下のようなものを用意する。

- `flatten(TF)`  
タイムフェンス TF 以降の調達計画を平滑化する。
- `get(Agent, Bucket)`.  
他エージェント Agent のオーダー情報を入手する。
- `do2(X, Y)`  
A, B を順に実行
- `do3(X, Y, Z)`  
X, Y, Z を順に実行
- `bucketChange`  
タイムバケットのインクリメント

我々は前述の情報量をサプライチェーン全体の目的関数として制御するモデルを考案している。そして情報量の値ができるだけ小さくなるような PSI 計画値立案を、この創発的計算手法によって実現することを目指している。

## 6 おわりに

本論文では、グローバルなサプライチェーンにおいて自律分散エージェントの手法が必要であることを議論し、販売・事業部・工場などの工程をエージェントとしたシミュレーション実験によってエントロピーを用いた複雑さの指標の有効性を実証した。そして、この複雑さを創発計算手法を用いて減少させることで、サプライチェーン全体の効率を向上させる可能性を示し、そのための遺伝的プログラミングの新たな手法を提案した。

## 参考文献

- [1] Hieber, R., Brutsch, D., Frigo-Mosca, F.: "How to Manage your Supply Network to get better results", *Strategic Management of the Manufacturing Value Chain*(ed; Bititci, U. S., Carrie, A. S.), pp.289-295, 1999.
- [2] Kargupta, H.: *SEARCH, Polynomial Complexity, And The Fast Messy Genetic Algorithm*, Illinois Genetic Algorithms Laboratory report no. 95008, 188pp., 1995.
- [3] Koza, J.: *Genetic Programming, On the Programming of Computers by means of Natural Selection*, MIT Press, 1992.
- [4] Lin, F., Strader, T. J., Shaw, M. J.: "Using Swarm for simulating the order fulfilling process in divergent assembly supply chains", *Economic Simulations in Swarm: Agent-Based Modelling and Object Oriented Programming* (ed; Luna, F., Stefansson, B.), Kluwer Academic Publishers, 2000.
- [5] 森 正勝, 松尾 博文, 小坂 満隆: "サプライチェーンマネジメントにおける知的エージェント技術を用いた資源配分に対する一方法", 計測自動制御学会, Vol.34, No.11, pp.1675-1683, 1998.
- [6] 岡田 和保: "DRP による在庫管理", *システム/制御/情報*, Vol.37, No.6, pp.350-355, 1993.
- [7] Schlueter-Langdon, C.: "Online Supply Chain Modeling and Simulation", *Economic Simulations in Swarm: Agent-Based Modelling and Object Oriented Programming* (ed; Luna, F., Stefansson, B.), Kluwer Academic Publishers, 2000.
- [8] Sivadasan, S., Efstathiou, J., Shirazi, R., Alves, J., Frizelle, G., Calinescu, A.: "Information Complexity as a Determining in The Evolution of Supply Chains", *International Workshop on Emergent Synthesis - IWES'99*, pp.237-242, 1999.
- [9] 谷口, 寺野: 論理プログラミングによる遺伝的プログラミングへの接近, 人工知能学会全国大会 (第 12 回) 論文集, 28-01, pp.481-482, 1998.
- [10] Wong, M. L. and Leung, K. S.: *Combining Genetic Programming and Inductive Logic Programming using Logic Grammars*, In the Proceedings of the 1995 IEEE International Conference on Evolutionary Computing, Vol.2, pp.733-736, 1995.
- [11] 吉原 賢治: "第 II 部 3 章 サプライチェーン", *生産管理の辞典* (編; 圓川 隆夫, 黒田 充, 福田好郎), 朝倉書店, 1999.