

解説



CIMの現状

3. 製造システム

—製造工程の制御と管理†—

松家 英雄†

1. はじめに

昨今の製造業は、●市場ニーズの多様化からくる変種変量生産形態、●需要変動に対する迅速な対応と効率の良い生産性、●製品寿命の短命化による生産システムの早期安定化と維持更新、●労働力の高齢化、減少化に対する生産システムの自動化及び省人化、●国際化による工場のグローバルなオペレーション、といった数多くの問題点をかかえている。

これらの問題を解決する一つの手段として CIM が期待されている。CIM は変動する需要に対して迅速に対処できるよう、生産に直接及び間接に関連する生産活動を企業として集約できるプログラマブルな生産システムであると考えられる。プログラマブルな生産システムとはプログラムにより製造工程の作業内容を定義し、製造機械群によってその作業内容を実時間で実行制御できる生産形態をいう。これを可能にするためにセルという概念が導入された。セルという語は単独の NC 工作機械で素材のストック及び自動供給または着脱装置、工具変換などを備え、長時間無人に近い状態で複数の工程の加工処理が可能で、所定の目的の部品が製造できる FMC (Flexible Manufacturing Cell) から由来している。現在は加工に加えて、組立や検査にも同じ概念を適用して総称してセルと呼んでいる。すなわちセルは製品の流れと情報の流れを実時間で結びつけ整流化する『要』であり、CIM 成功の鍵をにぎっているといっても過言ではない。このためセルに対して厳しい要求と大きな期待がかけられている。たとえば

- (1) 多品種混流生産に対する迅速かつ柔軟なセル内及びセル間の操作に対して
- (2) 需要変動に対する WIP (Work In Process), 稼働率を考慮した効率の良い工程制御に対して
- (3) セル内、間の運転状況、生産実績の収集把握、蓄積などの統一監視に対して
- (4) 早期立上げ安定化に対するシステムの信頼性向上に対して
- (5) 日々更新されるシステム構築や他のシステムとの統合、融合について
- (6) 異常や機器故障に対する縮退運転について

筆者らもこの点に注目して数年前から、組立工程における製造セルについて研究開発及び現場での実証を繰り返して、高性能化、汎用化を試みてきた。これらの結果は Manufacturing Cell Control System (MaCCS)^{1),2)} あるいは Global Programming, Controlling and Monitoring Environmental for Distributed Manufacturing Cells (GPCME)³⁾, Advanced MaCCS⁴⁾ として発表してきた。これは工場フロアに構築されたネットワーク環境において分散的に稼働し、生産環境を認識して適切な工程制御を行えるソフトウェア環境である。また増田ら⁵⁾ は、自律分散システムとして高機能セルの試作を行っている。

一方、セルのソフトウェアについて、生産機械や機器間の通信のための統一的なプログラミング及び制御管理環境の立場で GPL⁶⁾ や MMS⁷⁾ などが提案され、国際的な標準化活動が活発になってきている。またこれらを受けて Workcell, Plant Control, EasyMAP などの商用ソフトウェアも普及し始めてきている。

本稿ではこのような背景から、原点に戻って製造システムについてシステムの機能やアーキテク

† Production System—ShopFloor Control—by Hideo MATSUKA
(IBM Japan Ltd, Plant Operation Technology Center).

† 日本 IBM(株) プラントオペレーション技術センター

チャを考え、システムの開発あるいは運用のためのソフトウェア環境について考察する。セルは製造システムの中核に位置づけられるので、セルのソフトウェア環境についての検討は製造システムへの展開を容易にするであろう。

2. 製造システム

製造業における生産システムは工場内の生産機械の自動化によるボトムアップ・アプローチと工場内の生産活動を情報の立場から管理・運営しようとするトップダウン・アプローチの両側面から行われてきた。ここでは生産システムについて生産管理を含めた製造企業のシステムとして考えてみる。

2.1 生産管理と製造システム

渡辺⁶⁾は製造企業のシステムを図-1に示す生産管理システムと技術情報システム、工程制御管理システム及び製造機械システムの4つに分けている。

生産管理システムとは、各種の製品の生産量と納期を考慮した生産計画を立て、これに基づいて必要な素材、部品を購入する資材計画を立案、外部への発注・納期を管理する購買・受入管理、現場での仕事を実行するための日程計画と作業計画を作り、日々生産される製品の出荷を行う出荷管理及び計画と実績との差を計画に反映する機能が含まれる。

この生産管理システムには従来から存在する(1)見込み生産、(2)部分見込み生産、(3)受注生産によって方式は異なる。見込み生産は顧客仕様の多様化に応じ、多種の製品見込在庫をもつという危険があり、また受注生産は製作期間が不確定で納期短縮に対処しにくい問題があり、市場環境の激化より、二つの短所を補う部分見込生産方式に期待がかけられている。

この生産管理情報に加えて、製品の設計情報に基づいて生産技術情報に変換するとともに製品の生産評価を行う技術情報管理システムがある。このシステムは、製品を構成する部品についての加

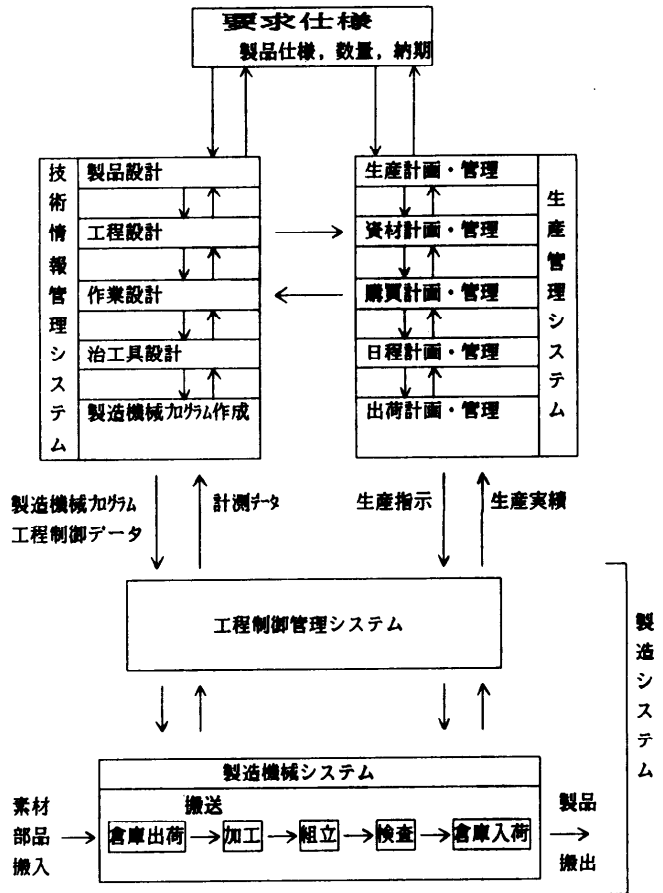


図-1 生産システムの構成

工・組立・検査などの工程を設計し、各工程における作業を各種生産機械に対してプログラムする役割をもっている。そして、製造システムから製品の品質についての製造情報を収集・分析し、設計にフィードバックする生産評価機能ももっている。

一方、製造システムは工程制御管理システムと製造機械システムから構成される工場フロアのシステムである。工程制御管理システムは、生産管理システムから生産管理情報を生産現場の作業員や生産機械に伝達し、また生産の進行状況を把握し、上位レベルの管理システムへ報告するとともに生産現場での素材・部品の搬送と工程管理情報の流れを同期化する。製造機械システムは、加工、組立、検査工程を経て直接素材を製品に変換する製造機械群と、素材や部品を運搬し貯蔵するなどといった補助的な生産活動にたずさわる補助製造機械群から成り立っている。

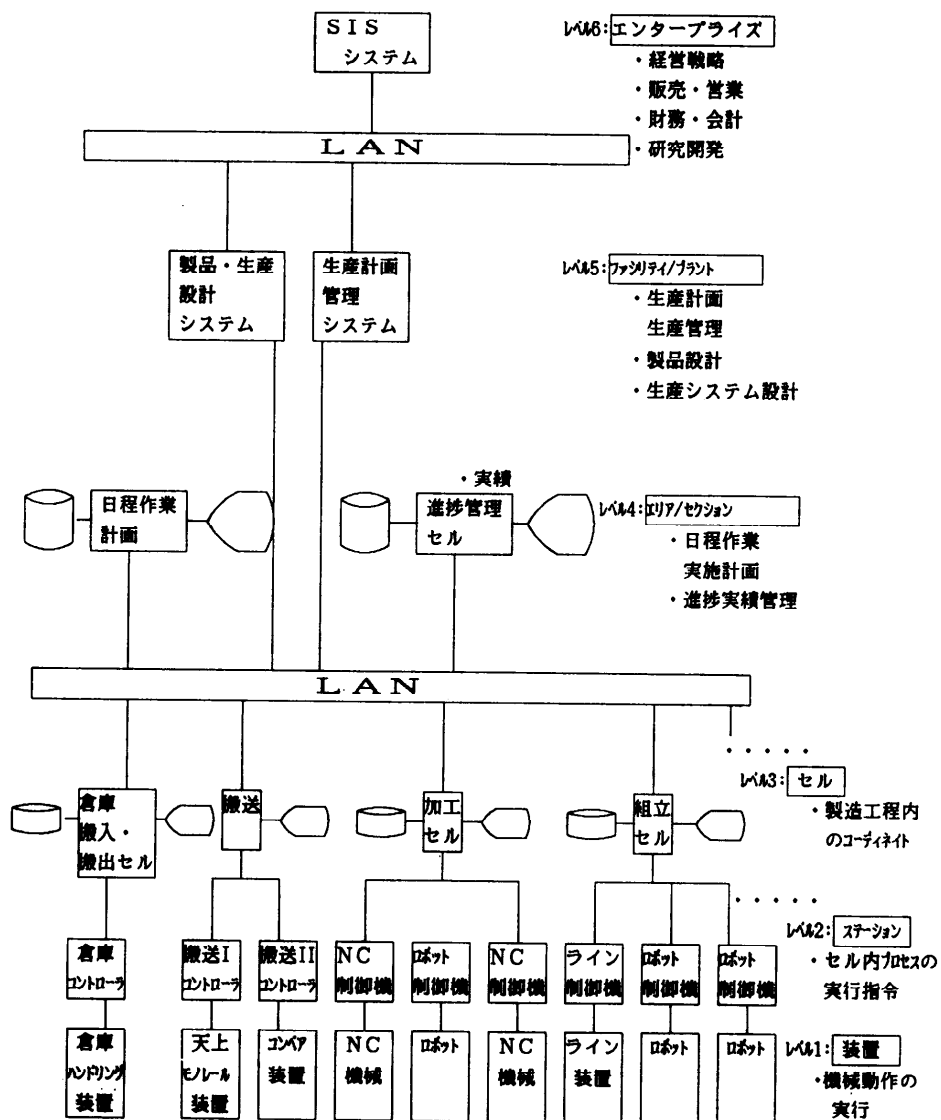


図-2 製造システム参照モデル

これら4つのサブシステムは、部分見込生産が浸透するに従って緊密な相互協調を要求されるようになってきている。情報を迅速に伝達し、現状の生産実績との差を生産計画に反映させ、より正確な情報を作り出すため、すなわちネットワーク上で相互通信を行い有効な生産情報を貯え、それを利用する高度なソフトウェア環境が必要となってきている。

2.2 製造システムの構成と構造

製造システムは工程制御管理システムと製造機械システムから成り立っている。ここでは ISO/TC 184 による FA 参照モデル⁹⁾に従って製造シ

ステムの構造について考えてみよう (図-2)。

レベル6のエンタープライズは需要/市場要求データを基に、販売営業、設計技術、生産管理などの部門が協力して企業としての生産戦略を策定し、実行にうつす活動を担当する。

レベル5のファシリティ/プラントは上位レベルの経営戦略に従って技術情報管理を含む生産管理システムから構成され、複数個のセクション/エリアを含んでいる。狭義の製造システムといえは一般に製造現場 (Shop Floor Production System) を指し、4以下のレベルをいう。

レベル4のエリア/セクションは製品を生産す

る投入工程から出荷までの工程を一括した製造活動総称名で管理と工程制御から成り立っている。管理とは補給管理、製造準備と開始、製造作業の開始、品質検査などの管理をいう。制御とは搬入・加工・組立・検査・搬出の工程などの実操作をいう。これらの工程を構築するのにセルという概念を導入し、論理的な組合せで各種の工程を定義し、制御をできるようにした。

レベル3のセルは産業用パソコンと称するコンピュータを中心にセル間の通信をするとともにインタフェースを介して各種のFA機器の制御装置と接続している。一般にセルは、

- (1) 一つのセルで複数の工程をこなし、プログラムの変更で工程変更も容易にできる。
- (2) 生産量が拡大するときはセルの台数を増やし、減少するときは1台のセルで複数の作業を実行させるため、柔軟に生産量の変化に対応できる。
- (3) セルを構成する製造機械群が、複数の工程に対して共通になるため装置の低コスト化がみられる。

このようなことから、セル型の工程制御システムは多品種少量生産の領域において優位性のあるシステムとしてこれから発展すると考えられている。

レベル2は一般にステーションと呼ばれる製造機械用制御装置から構成され、制御命令の順序に従ってプログラムを実行したり、セルからの割り込みを受け付けたりする。これらの制御装置ではバスラインやMini-MAPなどを介して、実時間で高速に通信が行われる。

レベル1はロボット、ベルトコンベアなどの生産機械である。

図-2をハードウェアの面から階層的に眺めてみよう。生産管理システムは汎用コンピュータで構築され、ここから生産計画指令がエリアに出される。エリアでは投入計画が実行され実工程の管理が行われる。セルは工程制御データに従って制御指令を専用制御装置に送りセル内の作業の進捗を司るとともにセル間での相互通信が行われる。専用制御装置はロボットやコンベアと直結している。このように製造システムは階層的にネットワークを介した分散システムを構成している。

2.3 生産工程形態

製造システムは生産形態によって工程形態が異なってくる。一般に、生産形態は個別、ロット、連続生産の三つに大別される。

ロット生産とは、同種類で、ほぼ同じ時期に納入する製品を集めてロットを形成し、各ロットに対して全体として共通のセル・人員を割り当てる生産形態である。連続生産は、あらかじめ製品の各種類に対し、専用の生産ラインを設計、各ラインで連続的に同一種類の製品を作る生産形態で、個別生産は共通の資源で個々の製品を加工・組み立てる形態である。個別生産は生産進行速度が遅く、生産コストの低減が難しい、また連続生産は顧客仕様の多様化や製品ライフサイクルの短命化を考えると、各専用ラインの稼働率や有効寿命の点で問題がある。いずれも部分見込生産と同じように、ロット生産形態に転換していく傾向がある。

これらの工程形態は物の流れとレイアウト方式によって図-3に示すようなジョブショップとフローショップと呼ばれる二つの方式に分けられる。

(1) フローショップ:

フローショップ型は製品の製造工程が連続的に流れる工程形態であり、混合製品に対しロット生産を切り換えることにより可能にしており、またフローショップを並列におくことにより並列生産もできる。大量生産に向けた稼働率の高い連続生産システムである。

(2) ジョブショップ:

ジョブショップ型は多種類の素材が異なるセルあるいは製造機械群に適当に配分され処理される方式で多品種少量生産に向けた柔軟なシステムである。納期確保、各機械群の稼働率の向上、仕掛品の低減という問題をかかえている。

フローショップ型は連続生産からロット生産形態(計画型前工程押出型)と、ジョブショップ型は個別生産型(現場型後工程引取型)に適しているといわれているが、工程の種類たとえば加工工程はジョブショップ型、組立工程にはフローショップ型の採用を使い分けしている場合が多い。

図-4に示すようにジョブショップであれフローショップであれ素材や仕掛品は各工程を左から右へ、そして各工程内のセルへと搬送され、加工、組立され、ある場合は中間でストックされ、

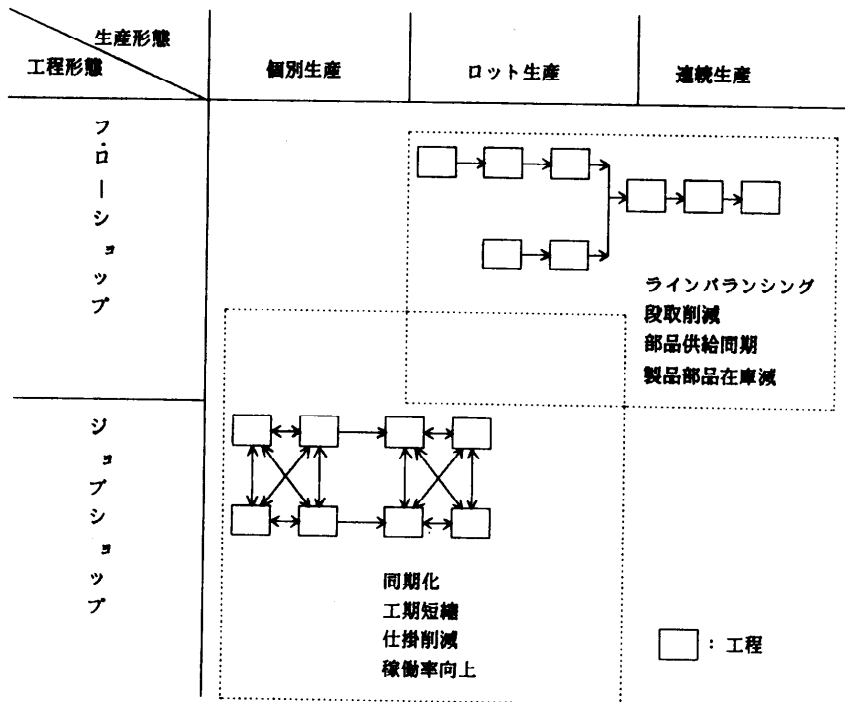


図-3 生産形態とジョブショップとの関係

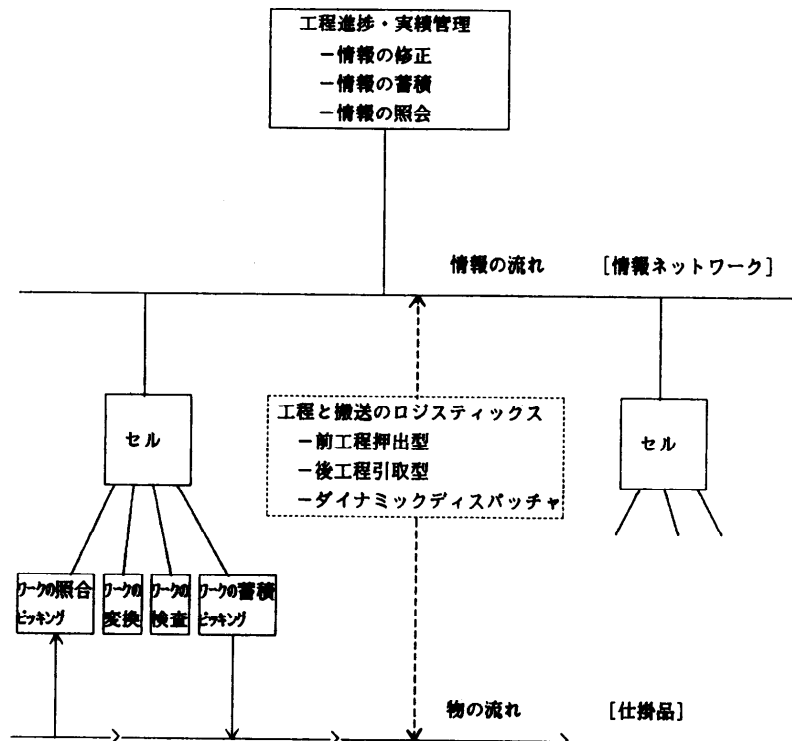


図-4 物と情報の流れ

検査されて出荷される。このような素材の流れに付随して製造情報も変換され、工程管理データと

照合されつつ転送されていく。従来の生産では物の流れと情報の流れの一体化は人間によって制御

されていた。しかし多品種少量を要求する部分見込型ロット生産が今後の傾向であるならば、この情報と物の流れの同期化が大きな問題となる。

図-4 に示した工程制御管理システムと製造機械システムとの連係、すなわち前工程押出型と後工程引取型による事象駆動の実時間制御、あるいはセル内での作業のプロセス制御と各種生産機械の動きの同期化が重要な課題となる。

3. 製造システムのソフトウェア環境

前章で述べたように製造システムを構築するためには各種異種機械の相互接続とともに、これらを製造工程のセルとして互いに通信で緊密に結び、物と情報の流れを整流化する工程制御管理システムが重要である。

ISO/TC 184/SC 5/WG 4 も製造システムソフトウェア環境のあるべき姿をグローバルプログラミング言語環境への立場から仕様を国際標準として提案しようとしてきている。1988年に MAPL (Manufacturing Automation Program Language)¹⁰⁾ 1990年に MAPLE (Manufacturing Automation Programming Language Environment)¹¹⁾ のシンポジウムを開催し、これらのシンポジウムで生産アプリケーションプログラミング言語のプログラムの設計と開発、生産ソフトウェアの操作及び管理について議論をしてきた。WG¹²⁾ として1992年までに、●生産データのハンドリングの観点、●アプリケーションソフトウェア開発の二つの観点から製造ソフトウェア環境をまとめる方向で進んでいる。データハンドリングではデータバッシング、データ伝送、可視化を含むデータ表現、データ収納と検索、データ編集と操作、データ入力などの標準課題があげられ、アプリケーションソフトウェア開発では仮想生産機械 (Virtual Manufacturing Device)、実生産機械 (Physical Manufacturing Device) へのマッピング、コミュニケーションデータフロ、及びアプリケーションプログラムの仕様記述などが対象となっている。

ここではジョブショップ型ロット生産を目的としたセルを取り上げ実働環境としての生産データハンドリング、開発環境としてのアプリケーションプログラムの仕様記述について事例をあげながら考えてみよう。

3.1 実行環境

図-1 に示したように生産計画で作られる日程計画とは、短期間中 (一週間単位) の製造すべき製品の品種、生産量の在庫状況、部品納入予定、機械稼働状況、生産実績データ、出荷計画、及び製品のマスター工程データなどを基にして、日単位またはシフト単位にスケジュール表を作ることを用いる。また工程設計とは製品の品種に対して製造されるべき工程及びその順序を定義しマスター工程データを作成することである。作業設計とは、工程を構成するセルの機械構成や機械が行うことのできる作業及び作業を組み合わせたプロセスを設計することをいい、結果がマスター作業データとなる。

これらのデータを基に日あるいはシフトごとにロット単位で処理される工程経路を決定し、この情報のもとに各工程ごとにセルを単位として各セルでの作業開始及び終了時刻の決定、パーツ投入順序の決定、また異常時のための代替え経路の決定を行い、工程制御管理データを作成する。これは一般にショップ・スケジューリング (詳細スケジューリング) と呼ばれている (図-5)。

実行時ではこのショップ・スケジューリングに従って工程の進捗を管理する。この管理方式は前工程押出型か後工程引取型のロジックから成り立っているが、特にジョブショップ型ロット生産では部品の不良や機械の故障などのために実行データとは異なってくる。そのため、ディスパッチング・ルールによるスケジューリングがセル単位で行われ、納期内仕上げ、マシンの稼働率、セル内

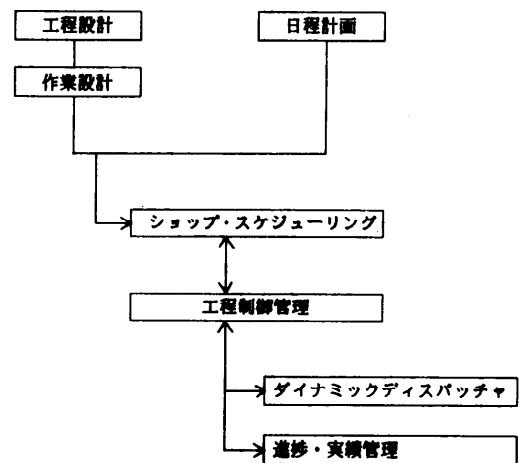


図-5 スケジューリングシステムの基本構造

でのブロッキングや手持ちの予防などを行う。たとえばある工程でのバッファから次にどの仕掛品を投入するかを決めるのに FIFO (First in First out), SPT (Shortest Processing Time First), MST (Minimum Slack Time First) などのルールや投入計画を生産状況に対応させて変更させるヒューリスティックルールなどが数多く開発されている¹³⁾。しかしこのようなディスパッチングルールは製造ラインの状況が変化したり、制約条件や評価基準が変わると性能が著しく低下することが多い。たとえば SPT ルールは mean flow time, makespan などの性能は良いが納期遅れを生じることがあるし、逆に MST ではこれと逆のことがいえる。これらの欠点を補うため、製造中に、生産状況や評価基準・制約条件の変化に応じてルール自体を切り換えていくことが望ましい。中須賀、吉田¹⁴⁾は実時間制御可能な IF-THEN 形式のダイナミックディスパッチャを作成するダイナミックスケジューリングを提案している。IF 部のオペレータはシステムの状態や制御条件の変数に関する等式、不等式の連言・選言により構成され、THEN 部はヒューリスティック・ルールで構成されている。

このように各工程内のダイナミックディスパッチャによる工程制御を行ったり、また計測データによりセル内の作業プロセスのパラメータを自動的にセットするインプロセス制御を行うために、製造工程管理データの変数の共有環境を実現化する必要性があり、ここにデータハンドリング機能が重要になってくる。

高田¹⁵⁾はデータプール、森¹⁶⁾はデータフィールドという概念を作り、作業プロセス間の同期や処理などを容易にしている。

データプールの基本機能は次のとおり、

- 名前によるデータのハンドリング
- 通信機能によるデータの共有
- データ変更の発見/通知
- データの依存関係に基づいた演算式の評価とデータの自動変更

である。図-6 に示すようにデータプールの構成はデータ管理とデータ監視からなる。データ管理は簡易データベースであり名前によるデータの指定ができる。データ監視すなわちデータ変化(トリガ)は次のように検出・処理する。

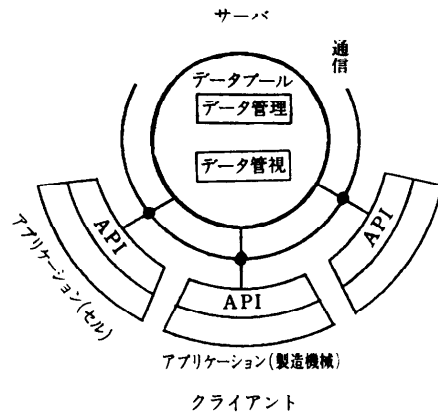


図-6 データプール概念

● 変数の値あるいは変数の属性が更新された場合には、その変数に対して指定された資源(生産機械群あるいはセル)に更新された変数の名前を通知する。

● あらかじめ名前を付けて登録された条件式を評価してその結果が真になったときに、その条件式に対して指定された資源へ条件式(条件式にはそのデータプール内の任意の変数・属性・他の条件式を含むことができる)の名前を送る。

データプールの実行環境はサーバとして働き、アプリケーション・プログラムはクライアントとしてデータを利用する。事例を図-7 に示す。その一つはデータ共有機能利用によるクライアント群からのデータの参照と変更である。2番目は、データ変更通知機能の利用である。個々のアプリケーションレベルでポーリングをする必要がなく、ある値が変化したときのプロセスを起動することができ、効率的にイベント駆動を行うことができる。たとえば監視のための実績データ変化による実時間生産管理図表示プログラムや、仕掛品の製品種類選別による製造機械の作業プログラムの稼働などが相当する。3番目は、依存データ自動変更機能による条件付イベント駆動である。名前同士の間には依存関係をもつ名前とその関係式を登録することにより、データプール上の依存する名前のデータ値の変更がおけるとその依存関係式を評価し、データの変更通信をクライアントに対して行う。そのため、データが一定値になったときか、複数データ間の条件が成立したときにプロセスを起動することができる。このようにデータプールはプロセス間通信を用いて複数のアプ

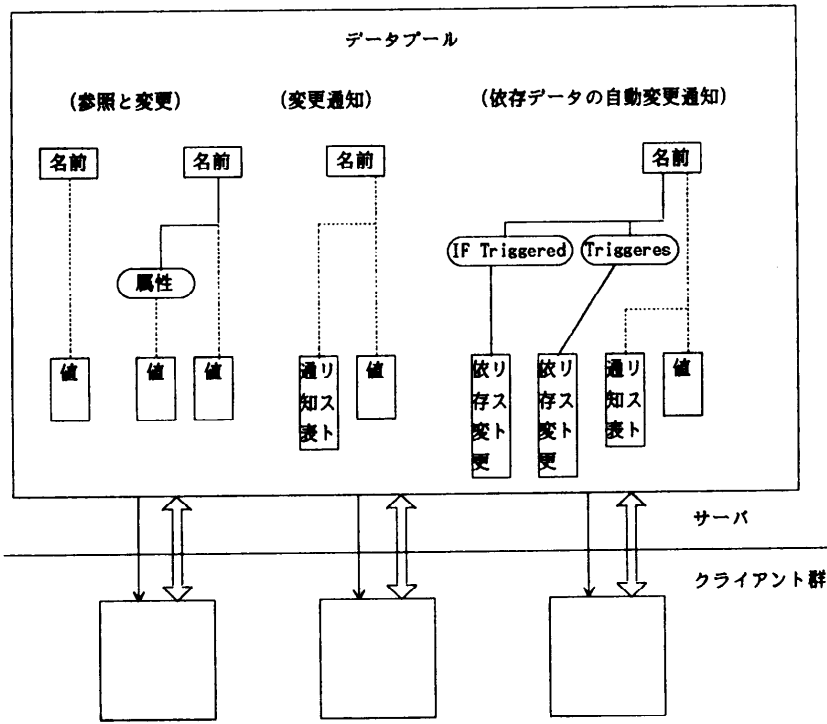


図-7 データベースの使用例

リケーションプログラムをコーディネートし、イベント駆動型のアプリケーションプログラムの開発/保守を容易にすることができ、またデータの共有により実時間制御のタイミングを容易調整するとともに分散環境における実時間製造工程管理を可能にする。

3.2 プログラミング環境

セルの作業をプログラミングするために上位層の工程、及び下位層の機器制御装置に独立に仕様を記述できるように、すなわち、『どのように』動作を行うかではなく『どのような目的のために』あるいは『どのような状態を得るために』動作を行うかをもって記述するように考えるべきである。そのため、レベルに応じてプログラミング環境を考える必要がある。

(1) 工程設計:

製品の品種ごとに処理すべき工程とその順序を記述することを目的とする。このうち工程そのものの記述は各セルに対して仕掛品に行うべき作業目的の指定として、順序についてはその工程に着手する前に完了していなければならない工程のリストとして考えられ

る。このためにこのプログラミングは施すべき工程をノードとして、順序関係は先だてて完了されていなければならない工程からその後に着手できる工程への有向アークとして表わす有向グラフで表現できる。工程記述のプログラミング環境はグラフィック・インタフェースを用いて有向グラフを作成修正するユーザフレンドリな能力をもち、製品の種類ごとのマスタ工程管理データを迅速に生成できるようになってきている。図-8 に工程記述環境上でのグラフ表示を示す。

(2) 作業設計:

セルがもつべき工程をどのような作業の組合せで行うかを記述する。作業はその作業の目的で、作業を開始するタイミングは先立つ作業を満たしているべき条件のリストとして、SFC (Segmental Function Chart)^{17),18)}を拡張したグラフで表現できる。作業開始条件には工程データ管理内の任意のデータの値や条件式の状態、ないしは状態遷移

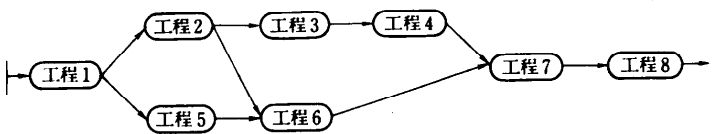


図-8 工程の有向グラフ表示

を用いることができる。作業記述環境はやはりグラフィックインタフェースを用いて、FCを拡張したグラフを生成・修正し、各セルに対して特定のワークの種類と工程に対する作業の制御を記述する。図-9に作業記述のプログラミング環境上でのセルの作業工程図

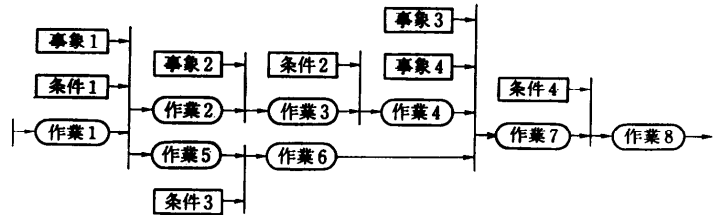


図-9 拡張 SFC 表示

を示す^{11), 3)}。この作業記述はセルのもつ機能に依存するものでセル内に蓄えられるが、セルに属する制御装置自体に対しては依然としてその作業の目的をもって記述し、制御装置固有の命令からの独立性を維持する。

(3) ステーションあるいは製造機械の動作記述:

下位レベルはテキストで記述されて言語処理形で解釈実行されるため、プログラミング環境の主要なツールはテキスト・エディタとなるが、最近では CAD システムの延長として、ロボットなどの動作をグラフィック上の動きから設計できるようになってきている。ロボットや NC 機械などの製造機械をオブジェクトとして捉まえ、それらの機械や制御方式をメソッドとして明確に扱えるようになっておくと、グラフィックデータは製造機械依存のテキストデータに変換されるので、その構文に対応した専用の構造化エディタも有効になってくる。このプログラムもセルあるいは個々の製造機械制御装置内に保管される。

プログラミング環境として忘れてならないのがシミュレーションとデバッグである。製造機械などの制御プログラムは実行時には物理的な機械要素を駆動するため、事前にその動作がある程度は意図したものに近いものになっていることの保証があることが望ましい。全ての機械をオブジェクトで扱うため、各機械の動作をプログラムでシミュレートして実機と同じ応答を返すような仮想機械を作れば、これを用いて機械制御プログラムに手を加えることなくシミュレーションを行い、デバッグを行うことができる。

これらのアプリケーション・プログラムにとって次に述べるようにエリア、セルあるいはステーション・装置レベルごとにライブラリをもつ必要がある。

- 各レベル内では製品の種類やクラスごとに整

理して保管し、実行時に高速に検索する。

- プログラム開発/保守時に必要な作業記述を分かりやすく提示し、バージョン管理を行う。

- 各レベルに必要なプログラムを転送する。

例にあげた3種類のプログラミング方式は『工程や作業の目的によって生産の過程を記述する方式』であり、基本的にはオブジェクト指向型のプログラミングパラダイムと同じである。なぜならば生産過程は内部状態をもつならんかのものに対して働きかけ、その状態を変更する操作であることとみなせる。これはオブジェクトに対してメッセージを送ることで全ての演算を実現するオブジェクト指向型の演算処理モデルと合致するからである。すなわち製造システムにおけるオブジェクト指向型のプログラミング環境は次の特徴をもつことになり、数多くの言語仕様で報告されている^{10), 11)}。

- セル/機器に依存しない生産手順の記述: オブジェクト指向型言語処理系ではオブジェクトはメッセージを受け取るとそれに対応して自分なりの方法で処理を行う。すなわちメッセージを送る側は受け取る側が適切に処理してくることを期待して、受取側に依存しない処理手順を記述できる。

- ブロック構造をもった手続き型の言語仕様: メッセージを受け取ったときの挙動の手順を容易に表現できるように、手続き型の記述を許し、さらにそれを構造化して処理の論理を分かりやすくすることができる。

- 言語レベルの排他制御と例外事象処理の記述: 同時に複数のアプリケーションが並行して稼働する環境において、プログラミング言語のレベルで同期や待ち合わせを実現できる。

- データプールによるデータ管理と他のアプリケーションとのデータ共用/通信: タイミング調整以外の目的のために一般的な情報のやり取りを

実現する。

以上オブジェクト指向型のプログラミング環境による工程順序及びセル製造機械の作業及び動作などの統一的な記述表現と、セル内とセル間のプロセスのダイナミック制御をするためのデータプールの考え方を考察してきた。最近ではセルを対象とした構成定義とその実行監視環境を設計するツールが商用ででてきており、急速にグローバルなプログラミング言語環境の考え方が産業に普及するであろう。特にモニタリング用の Visual 画面設計についても重要であるがこれらは Plantworks¹⁹⁾ を参照していただきたい。

4. おわりに

製造システムはますます自律分散、協調制御を指向して進展するであろう。このシステムの中核としてのセルの汎用性は大きな課題であり、ソフトウェア環境の新たな開発は必須である。この実現化のため、分散におけるデータプールの管理、このデータの駆動による工程の制御ならびに制御によって作動する機械の作業についてのプログラミングについて考察してみた。しかし、この新しい動きについて手際よくまとめることは、浅学な筆者にとってたいへん難しく、抽象的な話になり、分散ネットワーク上 VMD の概念にも言及することができなかった。しかし、ISO/TC 184/SC 5 の活発な標準化活動、また次世代の生産システム『知的生産システム；IMS』の国際共同開発プロジェクトにより、明確化された新しい概念の製造システムの普及が促進されるであろう。

最後に多くの話題を提供して下さった ISO/TC 184/SC 5/WG 4 の国内対策委員会のメンバーに感謝の意を表す。

参考文献

- 1) 長谷川雅樹：Manufacturing Cell Control System, 計測自動制御学会, 第 14 回シンポジウム (1988).
- 2) Temmyo, T. et al.: Distributed Control in Manufacturing Cells, Proc. of IEEE Workshop on Intelligent and Systems, pp. 25-530 (1988).
- 3) Matsuka, H. et al.: Global Programming, Controlling and Monitoring Environment for Manufacturing Cells, Proc. of Japan-U.S.A. Symposium on Flexible Automation, pp. 1283-1292 (1990).
- 4) 松家英雄他：知的製造セル制御, 精密工学会春季大会講演論文集, pp. 619-620 (1991).

- 5) 増田 茂：高機能化セルへの機能要求とシステムアーキテクチャ, 精密工学会春季大会シンポジウム資料, pp. 16-23 (1990).
- 6) Boudreaux, J.C.: Requirement for Global Programming Language, Proc. of the Symposium on Manufacturing Application Language, MAPLE 88, pp. 107-114 (1988).
- 7) ISO-MMS, Manufacturing Message Specification, Part 1, 2, ISO and DD 9506.
- 8) 渡辺 真：生産管理, 計測と制御, Vol. 26, No. 7, pp. 6-12 (1987).
- 9) ISO-SFPM: ISO. Reference Model for Shop Floor Production Standards, ISO TC 184 N 147 TR 10314.
- 10) National Research Council of Canada: Proc. Symposium on Manufacturing Application Programming Language (1988).
- 11) National Research Council of Canada: Proc. Symposium on Manufacturing Application Programming Language Environment (1990).
- 12) ISO/TC 184 国内対策委員会：ISO/TC 184 の活動状況等調査報告書, 国際ロボット FA 技術センター (1991).
- 13) Wu, S.Y.D.: Multi-Pass Expert Control System—A Control/Scheduling Structure for Flexible Manufacturing Cells, I. of Manufacturing System, pp. 107-120, Vol. 7, No. 2 (1986).
- 14) Yoshida, T. and Nakasuka, S.: A Dynamic Scheduling for Flexible Manufacturing Systems—Hierarchical Control and Dispatching by Heuristics, Proc. of 28th IEEE Conf. on Decision and Control (1989).
- 15) 高田昌之：生産制御用プログラミング/実行確認, FA におけるプログラミング言語の標準化に関する調査研究報告会, 国際ロボット FA 技術センター (1991).
- 16) 森 欣司：自律分散システムと制御分野での実用例, 計測と制御, Vol. 29, No. 10, pp. 47-52 (1990).
- 17) 笹島他：PC プログラミング言語と SFC, オートメーション, Vol. 37, No. 7, pp. 5-23 (1987).
- 18) 村田智洋他：ネット指向型シーケンスコントローラの動向, Vol. 27, No. 8, pp. 16-23 (1988).
- 19) PlantWorks: Application Automation Edition, SC 28-8257-00, IBM.

(平成 3 年 9 月 2 日受付)



松家 英雄 (正会員)

1942 年生。1967 年早稲田大学院計測制御工学修士課程専修。ロケット, 人工衛星の誘導・制御の研究開発を経て, 1970 年日本 IBM(株)サイエンティフィックセンタ (現東京基礎研究所) 入社。船舶, 建築等の CAD, 景観シミュレーション, 地域情報データベースの研究開発, 及びコンプライアント制御方式によるロボットの実用化をおこなう。現在, CIM プラントオペレーション技術センター長。CIM 構築に関する事業に従事。計測制御学会, 精密機械学会等各会員。