

インターフェイス・マトリクスによる構工法計画の理論と手法

安 藤 正 雄

インターフェイス・マトリクスによる構工法計画の理論と手法

第1章 序論

- 1. 1 構法・工法から構工法へ
 - 1. 1. 1 構法と工法のちがい
 - 1. 1. 2 構工法という表現をもたらした状況
 - 1. 1. 3 設計・計画手法としての構工法の選択・組み合わせ手法の必要性
- 1. 2 本論文の内容と構成
 - 1. 2. 1 「もの」と「仕事」を統合する概念・手法としてのインターフェイス・マトリクス
 - 1. 2. 2 本論文の内容と構成

第2章 インターフェイス・マトリクスの定義

- 2. 1 はじめに
- 2. 2 ビルディング・システムと工程要素
- 2. 3 要素間のインターフェイスと工程関係
- 2. 4 インターフェイス・マトリクス
- 2. 5 工程要素の合成と分解
- 2. 6 本章の結論

第3章 インターフェイス・マトリクスによるネットワーキング

- 3. 1 はじめに
 - 1. ネットワーキングの意味
 - 2. 手法化の手順
- 3. 2 先行・後続関係の演算
 - 1. 要素間の直接先行・後続関係
 - 2. 先行要素、後続要素と誘引要素
- 3. 3 トポロジカル・オーダリング
- 3. 4 ループ探索
- 3. 5 ループ解消による単工区モデルの作成
- 3. 6 本章の結論

第4章 インターフェイス・マトリクスによるサブシステムの抽出

- 4. 1 はじめに
- 4. 2 サブシステムの抽出
- 4. 3 サブシステム間のインターフェイスとシステムズ・ビルディング
- 4. 4 本章の結論

第5章 インターフェイス・マトリクスによる工程計算

- 5. 1 はじめに
- 5. 2 最早開始可能行列と最遅開始可能行列
- 5. 3 TF、FF、クリティカルパスの計算
- 5. 4 本章の結論

第6章 工区分割問題とインターフェイス・マトリクス

- 6. 1 はじめに
- 6. 2 多工区分割と工程同期化
- 6. 3 ジョブ工区への工程の割付け
 - 1. サイト工区とジョブ工区
 - 2. 垂直・水平の先行・後続関係とモード
- 6. 4 多工区同期化工程の作成法
- 6. 5 本章の結論

第7章 時間の関数としてのインターフェイス・マトリクス

- 7. 1 はじめに
- 7. 2 サイクル工程の進行と I F M の書換え
- 7. 3 サイクル数を変数とした I F M (t)
- 7. 4 リアルタイムの工程管理手法への I F M (t) の応用
- 7. 5 本章の結論

第8章 インデックス・テーブルを媒介とした工区工程の計算

- 8. 1 はじめに
- 8. 2 I F M (V)、I F M (H) とインデックス・テーブル
- 8. 3 複合多工区同期化工法の工程計画
- 8. 4 本章の結論

第9章 プロダクト・モデリングからプロセス・モデリングへ

- 9. 1 はじめに
- 9. 2 プロダクト・モデルからプロセス・モデルへ
 - 1. 建築基本モデル
 - 2. 足し算の建築と引き算の建築
 - 3. 部分構工法の定義
 - 4. 部分構工法群からプロセス・モデルへ
 - 5. 4次元構工法モデリング
- 9. 3 工区分割を変えた 4次元構工法モデリング
- 9. 4 部分構工法の選択組合せを変えた 4次元構工法モデリング

第10章 結論

英文要旨

既報論文・報告一覧

記号・表記一覧

謝辞

第1章 序論

1. 1 構法・工法から構工法へ

1. 1. 1 構法と工法のちがい

工法に比べて構法という言葉は新しい。「構法計画」という概念が打ち出されたころ、すなわち1975年前後にこの言葉の定義を巡る議論が多くあったように思う。この二語の区別を、簡単に言うならば、次のようになる。すなわち、構法が建築の「ありよう」(what to build)をあらわす言葉であるのに対し、工法はその「やりよう」(how to build)をあらわす言葉である。したがって、ふつうには、構法はより設計に、また工法はより施工に近い概念として理解されていると考えられるが以下のような混乱もある。

(ケース1)「工業化構法の代表的な例であるPCa工法」という記述はありふれた言い方である。しかし、その場合、いったいどちらにどの字をあてればいいのか。特定の設計をより強く意識していれば構法が適当なようでもあるが、しよせんは同じRC造の建築物を現場打ちにするかPCにするかという程度のニュアンスのちがいならば、工法がよいような気もする。しかし、前者に工法の字をあてたとしても大きな誤解を招くわけではないし、後者の場合のふたつのやりかたは詳細図レベルではまったく異なる。

(ケース2)VE(Value Engineering)といえ、日本では施工段階におけるより合理的な解答の提案と理解されることが多いが、アメリカでは設計段階での検討が不可欠とされているとの指摘を聞いたことがある。事実、提案の内容によっては、施工段階ですらいわゆる構法の変更を伴うことが多くあるだろう。このような場合、ものの「ありよう」、「やりよう」に関する一連の意思決定のプロセスを一線で画し設計と施工とに分けることは不可能であるし、いわんや前者を構法、後者を工法と呼ぶと決めつけることなどもはや非現実的といつてよかろう。

(ケース3)昨今多く用いられるカイザー版、オムニア版などの半PC版は構法か、それとも工法と呼ぶべきか。これらは本設の構法の一部であつて、タイルを打ち込んだ見え懸かりの部材などは意匠設計上重要な対象である。いっぽうこれらは、同時に仮設工法をかねる。仮設工法は従来純粹に「やりよう」に属する問題と考えられていたはずである。

1. 1. 2 構工法という表現をもたらした状況

●在来構法の多様化

構工法という曖昧な言葉をもちいることで問題がすっきりと解決するわけではないが、以上の例から、構法、工法のいずれかでは納まりがつかない対象があることは、確かであろう。では、なぜこのようなことが近年になって問題にされるようになってきたのであろうか。

第一に言えるのは、在来構法がすでに安定的、限定的な存在ではなくなったことである。構法計画、建築構法、あるいは各部構法などと呼ばれている教科目は、かつては一般構造と呼ばれていた。少し前の一般構造の教科書、参考書に示されているような構法の中には今日もはや一般的に用いられなくなったものも多い。技術は日々新しく、それらを体系的に再編成することは不可能のようにさえ思える。

ボザール全盛の時代の図面には見えががりの意匠は精密に描かれてはいても、その背後に隠された構造や下地などの詳細が示されることはない。この空の部分をポシェというが、それは「在来」構法が安定的に存在して初めて成立する設計技法である。実に、古典の時代にはディテールという言葉は細部意匠を意味していた。それが近代に入ると技術的な詳細、すなわち現在のディテールの意味に変質してくる。工学的な世界観が世を覆い始めると同時に在来構法はその輪郭を失い始める。

在来構法がその実体を失うということは、また、その対照物としての工業化構法が実体を失うということでもある。私見では、このことがもっとも鮮やかに認められるのは、住宅公団が標準設計の採用を全面的に廃し、民間で開発された工業化構法を買い取る方式、いわゆる性能発注制度が導入されたとき（1976年）である。集合住宅の躯体工法にそくしていえば、大型PC版工法から複合化工法に大方の関心が移行した時期と言い替えてもよい。SPH（中層プレハブ住宅規格統一要項、1970年制定）をはじめとする大型PC版工法がなぜこの時期妥当性を欠くことになったかについて、ここで詳述するつもりはない。ただ言いたいのは、このとき、ほぼ100%工業化されたPC工法一辺倒になるよりも、プロジェクト条件や環境条件に応じ、工業化手法と在来的手法を適宜織り混ぜて用いる複合化工法が望ましいものとして選択されようとしていたということである。その組合せ方には種々の考え方がある。しかし部分的にみれば、状況によるとはいえ、工業化手法も在来的手法も等しく選択可能な対象となった。不特定の工事に用いられる「オープン」な技術としての工業化手法はすでに在来工法と呼ぶにふさわしい。

かくして在来構法は、きわめて多様な手法、生産システムの様態を含んだ混成系と化した。それとともにかつてはいずれかを指示することによって矛盾なく他方をも了解させていた「ありよう」と「やりよう」の一対一の対応関係も失われた。

●技術革新と産業構造の変化

新しい構工法の普及は、しかしながら、機能的要請や工学的技術の発展のみに由来するのではない。技術革新は一面では企業活動の論理であり、経済的現象である。そしてこのことは、構工法の新機軸を打ち出す主体として、またその生産供給を常時担う主体としての新しい部門が絶えず創出されることを意味する。これはまた、既存の部門に影響を及ぼさずにはおかない。すなわち、産業構造の絶えざる変化である。構法、工法を分離してとらえることをもはや不可能としている第二の理由はここにある。

技術革新、あるいは産業化の進展は、構工法に関連して次のような二つの大きな変化をもたらした。ひとつには、かつては存在しなかった部位が出現し、それを供給する産業が出現したことである。設備はそのようなものの典型であり、このところ、情報技術がもたらす新たな市場とリエンジニアリングの可能性がずっと注目されてきている。ふたつ目の変化は、供給されるものとその供給方式の変化である。伝統的な建築材料にとってかわった新しい建材は、販売店、販工店などの流通ネットワークを出現させ、末端の工事部門の機能をも少なからず変えた。

産業構造の変化という点で注目すべきは、サブコントラクターの変化である。わが国のサブコンは、それがいかに巨大な企業であっても、あるいはその保有技術がいかに高度であっても、下請けであるサブコンに変わりはない。しかし、イギリスなどでは従来の職種と新しい職種を分けてあらわすのが普通のようなものである。例えば、大きな区分として、スペシャリスト・サブコントラクターとトレード・サブコントラクターという分けかたがある。後者が伝統的、在来的技能区分にもとづいた職種であるのに対し、前者は技術革新とともに出現した比較的新しいサブコンである。スペシャリスト・サブコントラクターはまた材工一式の請負で特徴づけることができるであろう。

スペシャリト・サブコントラクターがスペシャルと呼ばれる所以は、その設計・生産に関わる技術がそのサブコンに独占的に保有されているからである。もちろん、スペシャリスト・コントラクターのなかにも設計・生産はせず、組立・取付のみを行とするものもある。しかし、おおまかにくれば、これらは専門的な情報、技術を保有するサブコンと理解してよからう。

情報・技術がこのように特定のサブコン（のちに述べるようにサブシステムと言い替えたほうがよいかも知れない）に偏在してくると、設計・施工のプロセスはどのような影響をうけるであろうか。当然のことながら、その部分の仕様に関する詳細な意志決定はサブコンからの情報提供がない限り不可能であるし、さらにはサブコンが決定されられぬ限り仕様の検討など始められないということにもなる。すなわち、技術革新にともなう産業構造の変化は必然的に生産プロセスの変化をもたらし、その結果として、早期の段階での構工法の決定不能性がもたらされるのである。

●設計・施工のプロセスと組織の一体化

ゼネコンによる設計施工一貫方式は日本建設業のもつ特質の一つとされてきたが、近年、欧米を中心に、広い意味での設計施工（design and build）方式が増加しつつある。プロジェクトの性質と景気動向にもよるが、発注者側がその利益とリスクを天秤にかけた場合、デザイン・ビルドが選択されるケースも多いのである。従来の設計施工分離方式は必ずしも理想的な形態とはみなされなくなりつつあり、そのことは新しい多様なプロジェクト・マネジメント方式の出現に明確にあらわれている。

古典的な設計施工分離方式（a）は、入札の前後で設計、施工が完全に別個のプロセスとして分離されている方式である。イギリスはこの方式をいち早く成立させた国であるが、それは19世紀のことであり、その時期はまた、建築家、工事請負者といった諸職能が確立された時期でもあった。それ以前は、建築家が施工者を兼ねたこともあり、また施工業者が投機的建設業者として活動することも多かった。投機的（スペキュラティブ）ビルダーとは、要するに、デヴェロッパーをかねた建設業者、建売り業者のことを指す。重要なことは、専門的な職能が確立されたこの時期、それぞれの職能団体が設立されるのであるが、建設業者の団体はこれらスペキュラティブ・ビルダーを排除するかたちでつくられたという事実である。専門的職能は、その専門技能によってのみ自らを守ることができる。と同時に、依頼者、各関与者の利益は、権利の行使と義務の遂行によって厳密に保証されなければならない。職業的倫理が近代的制度として規定されたのがこのときだったのである。設計施工分離方式は、それゆえ、侵すべからざる規範として今日まで保持されてきた。

ところが、近年、この方式にかわる発注契約方式が多く用いられるようになってきた。CM方式、マネージメント・コントラクティング方式、デザイン・ビルド方式などがその例である。

なぜ古典的な設計施工分離方式にかえてこれらの方式がとられるようになったのであろうか。それは、ひとえに工期とコストの不確定性を回避することが多く要請されるようになったことによる。ではなぜそのような要請が強くなってきたのか。その理由のひとつに、大量の民間資本が投入される大規模な開発事業が多くなってきたということがあげられる。投下資本の早期回収、これこそ公共から民間へと需要がシフトした時代にふさわしい行動原理である。大規模開発は例外なく投機的事業である。投機的建設業者を排除するかたちで成立した設計施工分離方式が、ふたたび投機的事業の要請によって変化しようとしているのである。

●サブコン／サプライヤーの機能変化

こうした新しい状況は構工法の位置づけにどのような影響をおよぼすであろうか。新しい発注契

約方式はいずれも設計・施工の組織とプロセスとを一体的に運営することを目指したものである。これは構工法の実質的な担い手であるサブコントラクターにも大きな影響を及ぼさざるを得ない。サブコンは早期から設計のプロセスに巻き込まれる。かくして従来建設プロセスの上流にのみ位置づけられていた設計という行為の一部は、下流に移転される。同時に従来のプロセス機能をより上流のプロセスを包含するように変質させようとするモメントが存在する。設計はより上流の企画を取り込もうとし、従来の施工部門ですら設計をとおり越し、さらに上流の機能をもとうとする。

サブコンの機能は変化しつつある。サブコンはその職域を垂直方向に拡大しつつあり（サブコンが変質して一種のゼネコン化し重層構造を一層強化するようにはならないという希望的観測のうえでの記述であるが）、専門性と自律性を高めつつある。

●生産設計の重要性

サブコンはこのように変貌しつつあるのだが、技術開発の最大のポテンシャルを有しているセクターは現時点ではやはりゼネコンであり、一部の設計事務所を含むエンジニアリング・コンサルタントであるといつてよい。これらより上流側に位置するセクターも現状に手をこまねいているばかりではいられない。上流の設計情報をいかに有効に、しかも矛盾なく下流に受け渡していけるか、あるいは逆に、下流から提供される生産情報をいかに的確に設計に取り込んでいけるかが、焦眉の課題となっている。

アメリカでコンストラクタビリティ（あるいはコンストラクティビリティ）、イギリスでビルダビリティと呼ばれている考え方は、それぞれ少しずつ異なるとはいえ、大きくはこのような状況に呼応した動きとみなしてよかろう。これらの考え方は、組織としては従来の設計・施工の独立した機能イメージを否定しないまでも、もはやいっぽうが他方から完全に分離独立したプロセスとしては存在し得ないという前提に立っている。設計された建築物をはたしてそのとおりにつくることができかどうかは、生産情報を現実に把握していない限り知りようもないというわけである。したがって、ここでも、構法と工法を不可分のものとして考えざるを得ない立場があることが理解されるであろう。

目下建設業界で先端的な研究開発の最重点テーマのひとつとされているC I C、情報化施工といったことがらもまた、まさに同じ関心のうえに立脚したものである。その手法、またそのための要素技術にはさまざまなものがある。しかし、要は、従来の設計に生産設計をいかに接続できるか、あるいは組み込めるかということがその核心にある。設計はもはや物的要素を机上で組み立てるだけではすまされない。要素とは物であり、その物をつくる技術であり、その技術を保有している主体である。つまり今のところは輪郭のぼやけたサブシステムとしか言いようのない、しかし、間違いなくより多義的な対象なのである。構工法とはそのようなものにほかならない。生産設計とは、数ある構工法のなかから、目的と条件に応じて適当なものを選択し、組み合わせる技術である。それが「協調的」な手法でなければならないことは、言うまでもないことであろう。

1. 1. 3 設計・計画手法としての構工法の選択・組み合わせ手法の必要性

構工法という言葉が最初に使われた例は、菅見する限り、このようなコンテキストにおいてであった（「構工法計画のモデル化と利用方法に関する研究」、山崎雄介ほか、第6回建築生産と管理技術シンポジウム、日本建築学会、1990）。その発表の席であったか、あるいは別の機会であったか、やはりこの「構工法」という言葉が物議をかもしたらしい（いずれにせよ私はその場にいなかった）。しかし、この研究が「全生産過程で設計と施工が生産の合理化に関して協調して検討するこ

と」を目的の一つに掲げている以上、やむを得なかったことであろう。繰り返すが、構工法という言葉が認知されるべきだといっているのではない。しかし、何か、このような概念に適切な言葉があることは事実なのである。

さて、構工法の選択・組合せに関する手法については、そのいずれもが開発途上であるため、その行方を予想し、確たる評価を下すことはできない。しかし、現時点で予想される問題点を指摘しておくことはそれほど大それた企てとはいえない。

予想される問題点はおおまかには二つに分けられよう。ひとつは開発の前提として想定されている生産システムの将来イメージに関することである。誰が、生産プロセスのどの段階で意志決定に参加するか。イニシアティブが要求されるとすれば、それは各プロセス段階で誰によって提示されるべきなのか。情報の提供、およびその他の知的生産活動に対する代償は、片務的にではなく遂行されるのか。日々更新される情報は、誰によって、どのようにアップデートされ、維持されるのか。DBの構築、維持には絶えずそのような問題が付きまとう。「協調的」な手法は確かに求められるべきであるが、それはしかるべき生産システムの想定がない限り無意味である。

もうひとつの問題群は、計画・設計手法としてのシステム開発にからむすべての問題であり、膨大な事項が含まれる。これらは大ざっぱには構工法計画・設計のプロセス・モデル化と構工法そのもののモデル化に関する問題とに分けることができよう。構工法選択システムの開発において盛んに試みられている知識工学の応用は、それ自体の手法化に関する検討をなお多く要する段階だとはいえ、それ以前に、建築生産固有の対象をモデルとして記述するための有効な手法がぜひとも必要である。そして、ここにおいて初めて、構工法というこれまではあいまいに語ってきた対象を、今後いかに厳密に記述できるかという大きな問題に直面するのである。

複合化工法のように、各「部分」について、いくつかの構工法が選択されうるケースを想定してみると、あるいは分かりやすいかも知れない。そもそも、構工法はどのような単位に対応しているのだろうか。部位による分割はもはや明らかに不適當である。では、何をよりどころに単位をイメージすべきか。作業や請負区分の単位を加味するためには、ワークパッケージといった概念が有効かも知れない。しかし、これらもプロセス・モデルをつうじた全体的評価の結果確定する筋合いのものであって、「部分」をア・プリオリに規定するものではなさそうである。さらには、これらの分割・組合せに付随する問題がある。構工法の分割点は、はたして整合的に整理されうるか。フライング・ショアや床用の半PC版などのように、従来の分割点の基本であった仮設と本設、異なった部位、異なった職種を縦横に横断する構工法の例はいくらでも見いだせる。もはや、一時代前の工業化、部品化の場合のように、機能的、空間的、生産的分割点がきれいにそろった状態で、構工法を可視的に理解することは不可能である。構工法とはこのように難しい対象である。しかし、この問題を解かない限り、新しい時代は見えてこないと思われるのである。

1. 2 本論文の内容と構成

1. 2. 1 「もの」と「仕事」を統合する概念・手法としてのインターフェイス・マトリクス

本研究は構工・工法を一体的に扱うことを可能にするために著者が定義したインターフェイス・マトリクスの行列演算をいろいろな変数とリンクさせながら数理的な計画手法に展開する理論を構築し、それをさまざまに応用可能な構工法計画手法に展開した結果をまとめたものである。

1. 2. 2 本論文の内容と構成

本論文の内容と構成を表 1-1 に示す。

本論文は著者が独自に定義したインターフェイス・マトリクス（IFM）を用いて、さまざま構工法計画手法を開発、実用化した成果をまとめたものである。

表 1-1 本論文の内容と構成

章	変数	IFM自 体の構 造	職種	施工量	施工能 率	資源水準	工区分 割	工期 (TM)	時刻 日付	複数の 工区分 割法	複数の 構成材 分割法
				作業時間							
第1章	序論										
第2章	IFMの定義										
第3章	IFMによるネットワーク	●									
第4章	IFMによる構工法のシステム化	●	●								
第5章	IFMによる工程計算	●			●						
第6章	工区分割問題とIFM	●	●	●	●	●	●	●			
第7章	時間の関数としてのIFM	●	●	●	●	●	●	●	●		
第8章	垂直・水平の工区複合モードを組み込んだ多工区同期化工程の作成	●	●	●	●	●	●	●		●	
第9章	プロダクト・モデリングとプロセス・モデリングの統合ー4次元構工法モデル	●	●	●	●	●	●	●			●
第10章	結論										

序論と結論を除き、本論文は大きく三つのパートに分かれる。

第一篇は、IFM の定義とその工程計画への応用に関する研究をまとめた第 2 章から第 5 章までの部分である。

第二篇は第 6 章から第 8 章までで、いずれの章も多工区同期化構工法のサイクル工程に関するの各種計画手法のための理論化とツール開発を扱ったものである。

最後の第三篇は第 9 章の 1 章のみからなる。これも多工区同期化構工法の構工法計画の開発であるが、次の点でそれまでの章とは異なっている。すなわち、第 1 章から第 8 章までは「構法」を所与としたうえで工程計画を中心とするさまざまな構工法計画への展開がはかられているが、第 9 章では同一の建物の設計（建物基本モデル）から「構法」をさまざまに変化させ、「工法」と連動した計画手法を完成させたという点である。本論文の基底にある認識は、構法と工法は不可分であるということにあり、各章も何らかの意味でその認識に基づいた研究となっているが、第 9 章はその両方を双方向に操作可能な属性と値としてまとめあげたところに大きな特徴がある。その意味で、この章は本論文の結論部といってよい。

表 1-1 に示すように、本論文は IFM と名づけた行列の定義に始まり、その要素にさまざまな属性を持たせることによってより高度な理論と手法への展開をはかってきた経過を跡付けるものとなっている。

以下、第 2 章から第 9 章にいたる各章の内容について概要を記す。

第 2 章は IFM の定義を行った部分である。IFM は n 個の構成材（あるいは工程要素と解釈してもよい）の関係を表す行列である。前述のように、本論文は「構法」と「工法」を結ぶことを目的に構成材と工程要素を 1 対 1 に対応付けることから出発しているから、行列の軸を構成材と見れば行列の要素として表現された「関係」は構成材間の物理的なインターフェイスであり、またそれを工程要素と見れば同じ「関係」は工程順序として読める。この点が IFM の大きな独創性である。また、IFM は対角要素を 0 とせず 1 とすることによって、後章の手法の開発に有機的に役立っていると同時に、アロー型ダイアグラムとフロー型ダイアグラムの長所を兼ね備えた行列の表現となっている。この点も行列を応用した他の工程計画手法と大きく異なる点である。

第3章は、ランダムな順番に並んでいる IFM の軸を工程順に並びかえるトポロジカル・オーダリングを扱った章である。工程をあらわす行列から到達可能行列を求めそれを三角行列に変換するという手法はすでに存在しているが、本論文では IFM の特性に基づいた定式化を行った。その特徴は、後章でさまざまな工程分割、構成分割を行うために、構成材間の IF で最初に定義された要素間の関係が持つ冗長性をいっさい排除せずにトポロジカル・オーダリングを可能にした点である。また、後章での展開のために、IFM から求められるさまざまな行列を写像として用意し、工程要素（またはその群）をあらわすベクトルとの演算が可能ないように準備した。

第4章では IFM の工程要素に職種という変数を付加し、任意のまとまりの工程要素群（＝サブシステム）を抽出し、あるいは最適化する方法についてまとめた。これは第3章で IFM から定義した誘引要素の概念に主としてもとづいたものである。

第5章は、IFM の工程要素に作業時間という変数を連動させることにより、簡易な方法で最早開始日程や最遅終了日程、クリティカル・パスなどが導かれるように手法化した結果をまとめたものである。

第6章以降は、IFM にもとづき、繰返し型の工程を持つ建築に応用できる多工区同期化構工法の理論と手法を構築した結果をまとめたものである。この段階では、施工量や資源、工区分割、工期など構工法計画に関連する多くの変数が新たに IFM と関連付けられている。手法の基礎は工区分割を IFM の部分行列化として扱うというユニークな方法にある。また、工程分割と作業空間分割を区別して扱うためにジョブ工区とサイト工区という概念を導入したが、このことによって工区間関係を明解に説明できるようになり、手法化が可能となった。

本章で理論化・手法化した多工区分割同期化法についてもう一つ特筆すべき点は、これが各作業チームの完全同期化（工程的な矛盾なく各チームが 100%稼働できること）を条件とした計画法であるために、構工法計画の目標を満足する解の存在が常に保障されており、しかも多様な解の導出が可能とされていることである。多工区同期化手法に類似するラインバランシング手法や多工区同期化よりはるかに単純なタクト工程について平準化された最適解としての工程を求める試みがなされているが、本手法はこの問題が最適化問題ではないことを明らかにし、多様な解を用意できるようにしたことに意味がある。

第7章は、IFM に時間という変数を加え、IFM の時間的変化を表現する方法についてまとめた章である。また、行列の要素にサイト工区番号を記すことにより、基準階工程をあらわす行列の次元を増やすことなく、工程の進捗状況をあらわす IFM を作ることができた。この方法はリアルタイムの工程管理手法として実用化されている。

一方、第8章は、ジョブ工区またはそれに含まれる工程要素についてサイト工区間のインターフェイスを考慮することにより、各ジョブ工程（工程要素）に適当な多様なサイト工区分割法を矛盾なく混在させる方法を導いたものである。これにより、多工区同期化手法のバリエーションが各段に拡張されている。

第9章はすでに触れたように、建築の「ありよう」、すなわち構法も適宜変更することによって構工法計画の手法としての完成を提示した部分である。この場合の出発点は「建物基本モデル」と呼ぶ基本設計であってその全体的な形状、品質は不変であるから、ここでいう構工法計画は生産設計に対応することになる。その手法は、3次元のCADオブジェクトとして定義された建物モデルをブール演算によって適宜分割し（あるいは分割されたオブジェクトを合成し）、そこにつくりだされた構成材＝工程要素をIFMの演算と連動させることによって、構工法計画の評価を行うというものである。3次元CADと時間に関する工程計画を結んだという意味で、この手法を4次元の構工法モデリングと呼ぶ。この手法は次の点で今までにないものである。

その第一は、「関係」(IF)と量を媒介させることにより、本手法は構法、工法のいずれからも双方向的にアプローチできるようにされていることである。この点が、単にCADオブジェクトに工程順序を与えるだけの方法とは根本的に異なっている。

第二は、この手法が慣習的な、固定的な建築の部分概念を打破して、計画対象の自由な意味分節を計画者に委ねることを許している点である。この点も、オブジェクト指向等の概念を利用した類似の試みの限界を超えるものと考ええる。

建物基本モデルが計画者の介在によってさまざまに変化しつつ最終出来型に収斂してゆくプロセスに対応したこの手法は、設計の意味を再考するうえでも有効であることを最後に触れた。

終章である第10章では、本論文の成果をまとめるとともに、本研究が持つ意味について総括した。

第2章 インターフェイス・マトリクスの定義

2. 1 はじめに

- * 「もの」(プロダクト=構成材~構法)と,「こと」(プロセス=工程要素~工法)を同等に扱う
- * すなわち,「もの」も「こと」も同じ部分構工法の異なるファセットとみなす
- * 「構成材」の取合い(接合)関係と,「工程要素」の工程順序の関係を同じ関係(インターフェイス)の異なるファセットであるとみなす
- * その際,「インターフェイス」は「もの」(=構法)の側がア・プリオリに持つ情報であるとみなす
- * 「構成材」ないしは「工程要素」間の関係を行列(インターフェイス・マトリクス: IFM)で表現し,これをもとに構工法のパフォーマンス(資源量・コスト・工期等)を算出できるようにする

2. 2 インターフェイス

2. 2. 1 構成材の集合としての構法と工程要素の集合としての工法

建築はそれを構成する「もの」と,それをつくり,あるいは組み立てる「仕事」とから成り立っている。前者の物的構成という見方をとれば,建築の全体は構成材(component)から成り立っているという言い方ができる。構成材によって表現された建築,ないしは建築の構成をここではビルディング・システムと呼んでおこう。ビルディング・システムは,建築の実体であり,設計された建築であり,あるいは構想された建築全体またはその部分に含まれる構成材とそれらの間の関係でもある。本論文の目的からして,ここでは,構法を建築全体に対応するものとし¹,それは互いに関係付けられた構成材の集合によって構成されているものとする。このとき,建築=構法は構成材とその関係の集合として,また,構成材はその属性によって表現される。

「(「ありよう」としての)建築=構法」= {{(ひとつの)構成材}, {それが他の構成材との間に持つ関係}}

{{(ひとつの)構成材} = {(ひとつの)構成材 | 名辞(番号), 種類(名辞), 機能, 他の種類との包含関係, 量, 寸法, 位置, 材料, 性能, 物性, 供給元 etc.}}

一方,建築をつくる(組み立てる)「仕事」として建築を見る場合,それには一般に工法という言葉が当てられる。構法が建築の「ありよう」(what to build)をあらわすとすれば,工法は建築の「やりよう」(how to build)をあらわす。建築=工法もまた,その要素と要素間の関係,すなわち工程要素と工程によって表現される。

「(「やりよう」としての)建築=工法」= {{工程要素}, {工程要素間の関係}}

{工程要素} = {工程要素 | 番号(名辞), 種類(名辞), 資源, 技能・技術, 職種, 必要な資機材, 量, 時間, 他の種類との包含関係, etc.}

本論文の基本的なアイデアは,構法における構成材と工法における工程要素を1対1に対応付け

¹ 以後,建築の部分を指して構法という概念を当てはめようとするときは,「部分構法」という。

ることが可能であるとする点にある。「もの」と「仕事」を一組にして概念化するこの考え方は、システムズ・ビルディング (systems building) の考え方と共通したものである。システムズ・ビルディングにおいては、それを構成する要素をサブシステムと呼び、さらにその下位にある要素のまとまりの概念が必要であればサブ・サブシステムというような言い方をする。システムズ・ビルディングで扱うサブシステムのまとまりは一般的により大きい²が、本論文でいう構工法計画はその考え方のコアにおいてシステムズ・ビルディングのそれと変わるところはない²。

本設と仮設

2. 2. 2 要素間のインターフェイス (IF) と順序関係

さて、これら構成材と工程要素はどのように対応付けられるべきであろうか。

いま、簡単のために、構成材も工程要素もそれぞれ n 種類 n 個 (すなわち 1 種類 1 個) ずつあり、それぞれが 1 対 1 の対応関係にあるとしてみよう。すると、この対応の組には一つの共通した名辞を与えることが可能であり、それらの組の間関係は構成材間関係と工程要素間関係を合成したものと扱うことができる。

● インターフェイス

構成材または工程要素間の関係が存在するときにインターフェイス (IF) があるといい、その関係を単にインターフェイスと呼ぶことにする。IF は作業の順序関係を定める。この点において、構成材間の IF を工程要素間の IF と等価に扱うことができるのである。この順序関係については、後続の構成材・工程要素が先行の構成材・工程要素に IF をもつと定義する。以後、この関係を次のように \rightarrow で表わす。

$$A \leftarrow B \quad (\text{「}A\text{が先行, }B\text{が後続」} = \text{「}B\text{が}A\text{に IF をもつ」})$$

IF は次のように分類される。

a) 構成材間の直接的な取合いによって決められるインターフェイス

a - 1) 支持関係による IF

建築は重力に抗して下から積みあげられる、という宿命を持つ³。したがって、部材 B を支える部材 A があれば、部材 A が先に施工されなければならない。このとき、支持される (後続の) 構成材 A が支持する (先行する) 構成材 B に IF をもつ。

(例 1) 下階の柱は上階の柱に対して IF をもつ。

(例 2) 梁は柱に対して IF をもつ⁴。

a - 2) 仕上げ・下地の区分による IF

² ただし、システムズ・ビルディングでは、必ずしも「もの」と「仕事」を分けて取り扱うことが必要とされるわけではない。

³ 重力に縛られない組立系では指示関係による IF を考えなくてよい。例えば、無重力の宇宙空間に建設する構造物は上から (地上での概念では) つくってもよい。また、定盤上で組み立てられる部品は二次元の組立であるから、これも指示関係による IF を持たない。例えば、サッシの組立に際して、上枠、立枠の組立順はどちらでもよい。

⁴ 梁を先に取り付ける場合には仮設支保工がいる。

一般的に仕上げ材は下地材が用意された後に施工される。したがって、仕上げ (材) は下地 (材) にインターフェイスをもつ。下地材は仕上げ材を支持することが一般的であるから、これを支持関係による IF の特別な場合と解することもできる。

(例) 壁紙は下地 PB に対して IF をもつ。

a - 3) 勝ち負けによる IF

建築を端的に表現すれば、線材、面材を用いて直方体をつくることとすることができる。このように、建築には部材が直角に取合う場面が多くある。直角に取合う部材は一般に対照の関係ではなく、一方が伸びて他方を支持し、あるいは隙間なく接合されるようになっている。このとき、伸びている方を「勝っている」といい、他方を「負けている」という。勝ち負けの関係は支持関係や次に述べる作業上の都合による工程関係と密接に関わっているから、IF を定める。この場合、一般的に、負けているほうが (後続で) 勝っているほうに IF をもつ。

(例) 可動間仕切壁は床に対して IF をもつ (負けている)

a - 4) 「重い・硬い・強い構成材ー軽い・軟らかい・弱い構成材」間の関係による IF

軽い構成材、軟らかい構成材、弱い構成材は先に施工されると傷んだり、精度の保持が困難であつたりといった問題が生じやすい。そこで、一般的には、「重く、硬く、強い」構成材が先に施工され、「軽く、硬く、弱い」構成材がそれに IF をもつ。

(例) 左官仕上げと石仕上げが取合っているとき、左官仕上げが石仕上げに IF を持つ。

b) 作業上の取合い (工定要素の取合い) によって決められるインターフェイス

b - 1) 同一構成材に付随する一連の作業の性質によって定義される IF

一つの構成材の施工に、時間的に分離された一連の作業が含まれることがある。先に、構成材と工程要素は 1 対 1 に対応しているものとしたが、このような場合も一般的に生ずる。このとき、各作業を独立した工程要素とみなせば、一連の作業の中で先行する工程要素に後続する工程要素が IF をもつ。

(例 1) 鉄骨歪み直しは鉄骨建方に IF をもつ。

(例 2) 鉄骨 HT ボルト本締めはボルト仮締めに IF をもつ。

(例 3) 型枠解体は型枠組立に IF をもつ⁵。

b - 2) 作業箇所にアクセスできることによって関係付けられた IF

構成材間には直接的な取合いはなく、したがって IF もないが、ある構成材 B が先に施工されてしまうと構成材 A を取り付けるのに手が届かないといった場合である。ある構成材の施工がその手前や表に構成材があると妨げられてしまう場合、および作業足場として用いられる構成材がある場合などがこれに相当する。

(例 1) 天井仕上げはダクト取り付けに IF をもつ。

(例 2) 梁主筋圧接は梁主筋圧接用の吊り足場取り付けに IF をもつ。

(例 3) バルコニー PC a 板が鉄骨建方の足場として利用される場合、鉄骨建方はバルコニー PC a 板にたいして IF をもつ。

このうち、構成材間の直接的な取合いによって定義される IF は、建築の「ありよう」、すなわち

⁵ 後述するように、仮設材の場合はひとつの構成材に対して必ず「組立」、「解体」というふたつ一組の工程要素が生ずる。

構法から一意的に定められる IF である。すなわち、構法由来の IF は工法の如何によらず、ア・プリオリに定義されている。一方、建築のやりよう、すなわち工法から決められる IF のうち、(b-1) のすべて、(b-2) の (例1) などは、「ありよう」(設計=構法) によってあらかじめ明らかにされているという条件と言える。

したがって、次のふたつの重要な事実が明らかとなる。

①工程を決める順序関係は、そのほとんどが構成材間の取合い（～「ありよう」～構法～設計）によって半ばア・プリオリに与えられている。

②組立の順序関係に関するかぎり、したがって、構成材間の関係、および工程要素間の関係はほぼ共通であると考えてよく、また、構成材と工程要素を一組の同一概念として扱ってよい。

以上の事実が、本論文の主題である構工法計画、すなわち構法と工法を一組の概念として扱うことの理論的な基盤をなしている。

2. 3 インターフェイス・マトリクス (IFM)

2. 3. 1 インターフェイス・マトリクスの定義

構成材、ないしは工程要素間の関係を行列として表現したものがインターフェイス・マトリクス (IFM) である。工程ネットワークを表現するのに行列を用いることは一般的であるが、IFM には次に述べるような特徴がある。

すなわち、(n×n) 型の行列である IFM は、その n 行 (列) が n 個の構成材 (または工程要素) に対応し、その i 番目の対角要素で i 番の構成材を施工する作業を表わし、行列の (i, j) 要素 (i ≠ j) で i 番目と j 番目の構成材との関係 (取合いの有無; 取合いがあるときに 1, その他は 0) を表わした行列である。

行列の対角要素とは非対角要素はこのように性格が違うが、後章でさまざまに応用がはかられるように、対角要素も 1 とした行列をインターフェイス・マトリクスと呼ぶことにする。この場合、対角要素の意味は、「すべての構成材 (工程要素) はそれ自体と関係をもつ」と解釈される。IFM は対角成分を 1 とし、j 番目の構成材が i 番目の構成材に取合いをもつとき行列の (i, j) 成分を 1 とし、その他を 0 とした行列である (図 2-1)。

R =

c _i	1	2	3	4	5	6	7	8	9
1	1	0	0	0	0	0	0	0	0
2	1	1	0	0	0	0	0	0	0
3	0	1	1	0	0	0	0	0	0
4	1	0	0	1	0	0	0	0	0
5	1	0	1	1	1	0	0	0	0
6	0	0	1	1	0	1	0	0	0
7	0	0	0	0	1	1	1	0	0
8	0	0	0	0	1	0	0	1	0
9	0	0	0	0	0	0	1	1	1

図 2-1 IF 行列

2. 3. 2 n次元の構成材空間と構成材空間内の写像としての IFM の意味

IFM は n 個の構成材（工程要素）が張る n 次元空間で定義された写像である。IFM の対角要素以外の要素を 0 とした単位行列は、1 から n までの構成材のおののに対応する行（列）ベクトル、（i 番目の要素を 1 としその他を 0 としたベクトル）を並べたものである。

IFM の i 番目の行は i 番目の構成材に IF をもつ構成材（それ自身を含む）を示す行ベクトルであり、また j 番目の列 j 番目の構成材が IF をもつ構成材を示す列ベクトルである（図 2-2）。構成材（群）をあらわすベクトルと IFM の演算により、工程要素間の先行後続関係に関するさまざまな演算が可能となるが、詳細は第 3 章で述べる。

他の構成材に IF をもたない（先行する要素を持たない）要素を自由先行要素と呼び、また他の構成材からの IF をもたない（後続する要素を持たない）要素を自由後続要素と呼ぶ。

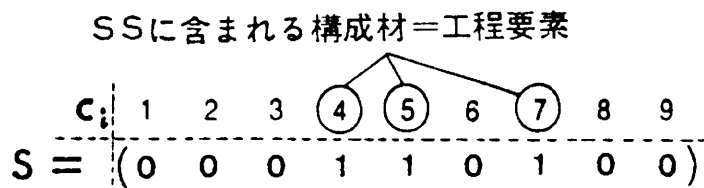


図 2-2 SS ベクトル

2. 3. 3 本設資材と仮設資材の扱い

IFM の縦軸，横軸，対角要素は構成材をあらわす。

構成材には本設資材と仮設資材とがある。本設資材が「ありよう」としてすでに記述されている構成材であるのに対し、仮設資材は「やりよう」によって定義され新たに構成材に組み込まれる資材である。仮設資材は建築プロセスの終了後、建築として残らない資材であるから、「組立」、「解体」の 2 つの工程要素としてあらわれる。

n 個の（本設）構成材の「やりよう」から m 個の仮設資材が定義されたとしたら、IFM は $((n + 2m) \times (n + 2m))$ の行列となる。すなわち、IFM の各軸には（対角要素にも）、仮設資材は 2 度ずつ現れる。

2. 4 ネットワークとしての IFM

2. 4. 1 IFM のネットワーク表現

図 2-1 の IFM をネットワーク（工程グラフ）として表現したものが図 2-3 である。このネットワークは、自分自身への関係（あるいは作業；IFM の対角成分）を自己ループとして持つネットワークである。個のネットワークを通常のアロー型ダイアグラムとして表現すれば、例えば図 2-4 のようになる。両者の主な違いは次の点にある。

①アロー型ダイアグラムでは、工程要素はノード間のアローで表わされる。IFM にしたがつた工程グラフでは、工程要素は必ずノード（あるいはノードからノードへの自己ループ）に対応する。

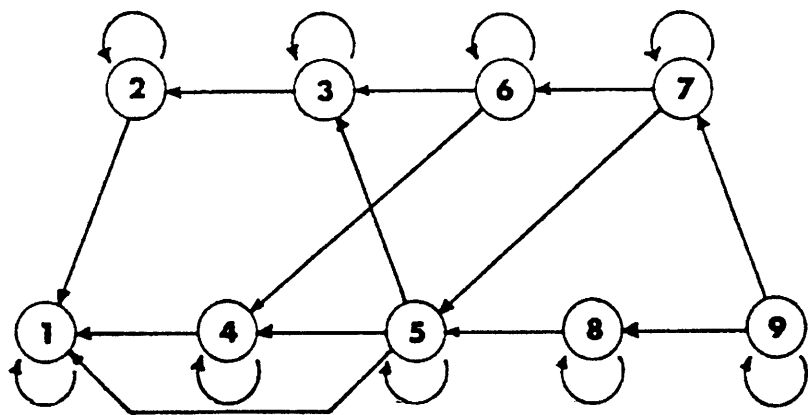


図 2-3 インターフェイスによる工程グラフ

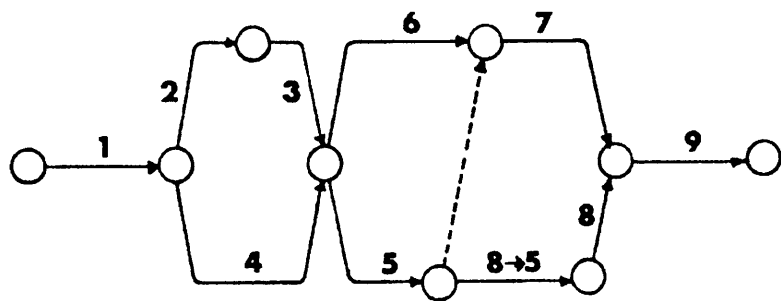


図 2-4 ネットワークによる図 2-3 の表現例

IFM 工程グラフをアロー型ダイアグラムの一変種として理解することもできる。この場合、これまで単に関係として説明されてきた IF を対角要素と同じように「仕事」として解釈する。インターフェイス、すなわち取合いを完成させるのに、一定の作業を要すると考えるのである。ただし、その作業量は 0 であってもかまわない。すると、これらの仕事は「中味の仕事」と「インターフェイス」の仕事に分けられることになる（図 2-5）⁶。

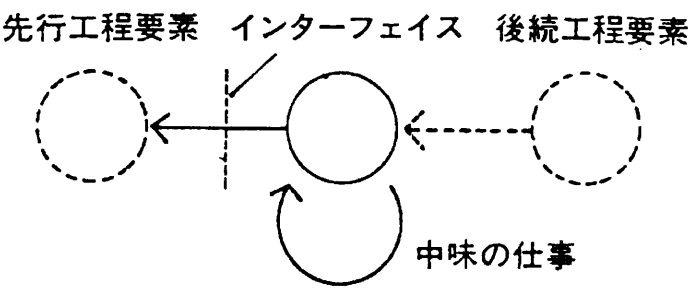


図 2-5 インターフェイスの概念

②IFM 工程グラフのアローの向きは、アロー型ダイアグラムとは逆向きにつけられている。これは、工程の流れというよりは、ある構成材が他の構成材にインターフェイスをもつという事実を明示的に表現するためである。

③IFM 工程グラフでは、IF の仕事は必ずノード間のアローで示されるが、従来のアロー型ダイアグラムではこれが消滅してしまうおそれがある⁷。

2. 4. 2 インシデンス型、プレシーデンス型ネットワークとしての IFM

従来のネットワーク手法にも、主流であるアロー型ダイアグラム（プレシーデンス型ネットワークともいう）のほかに、インシデンス型ダイアグラム（インシデンス型ネットワークともいう）と呼ばれる表現法がある。ノードに作業を対応させているという意味で、インシデンス型ネットワークは IFM 工程グラフに近い。

要するに、IFM 工程グラフは、インシデンス型のネットワークとプレシーデンス型のネットワークの長所を兼ね備えたグラフ表現であり、IFM はその性質をそのまま演算可能な行列に置き換えた工程の表現法であるといえる。

2. 4 工程要素の合成と分解

2. 5 本章の結論

⁶ IF の仕事を定義することは IFM の演算上得策ではない。そこで、本論文では作業を対角要素のみで表現する方法をとっている。ただし、次項であつかうように、工程要素の合成と分解を扱うときには IF の仕事の定義が有効である。
⁷ 図 2-3 のアロー（①←⑤）は図 2-4 では表現されていない。

第3章 インターフェイス・マトリクスによるネットワーキング

3. 1 はじめに

＊目的：ランダムにならぶ構成材（工程要素： C_i ）によって定義されたIFMを，行列演算により工程順序にしたがって並びかえる方法を求める

＊方法

・IFM から到達可能行列 R_s を求め，これにより先行・後続の順序構造を持つ $\{C_i\}$ の部分行列の列 $\{\{A_1\}, \{A_2\} \cdots \{A_{k-1}\}\}$ を求める

・言い換えると，任意のIFMを三角行列に並びかえる手続きである

＊手法の特徴

- ・ダミーを挿入せずに済み，また冗長性も許される
- ・冗長性が許容されることは，IFM の持つ関係構造（＝インターフェイス）が変更されずに保存されることを意味する
- ・このことは，工区分割法など後の手法の展開にとって決定的に重要である
- ・ループの探索と解消も，IFM の演算によって可能である

3. 1. 1 ネットワーキングの意味

ここに言うネットワーキングとは，「①インターフェイス（IF：前章 2. 3）によって工程要素間の工程関係を定義し，②個々の工程要素を工程順に並び替えると同時に要素間の工程関係をわかりやすく表現すること」を指す。工程要素を関係の有無，あるいは親近度によって順序構造化することをトポロジカル・オーダリングというが，本章ではマトリクスの行（列）をあらわす要素を工程順に並び替えることがそれに相当する。インターフェイス・マトリクス（IFM）は，本来，要素とその間の工程関係を表現したものに他ならず（2章 2. 4），要素の並び替えによって順序関係は変化しない。したがって，IFM によってネットワーキングを行うとは，マトリクスの成分として工程関係が示されたIFMの行および列の要素を工程順に並び替えることを意味する。

工程関係に矛盾のない（i.e.ループを生ずることのない）一群の工程要素の集合とそれらの間のIFによってIFMをつくり，そのIFMの行（同時に列）の要素を入れ替えることにより三角行列を得ることができる。こうして得られた三角行列がネットワークとしてわかりやすく表現されたIFMである。

3. 1. 2 手法化の手順

IFMによるネットワーキング手法の手順は概略次のとおりである。

- ①工程要素およびその間のIFは所与のものとしてIFMrをつくる。この段階でのIFMrの行（列）の要素はランダムに並んでおり，順序構造化されていない。
- ②IFMr，その転置行列^tIFMrの積の演算を工程順序の検索手法として意味付ける。すなわち，前者の積によって後続の要素が，また後者の積によって先行の要素が算出できる。
- ③IFMrの積より到達可能行列を求める。到達可能行列とは，ある要素から別の要素の間に（順行の）経路が存在するか否かを表す行列である。

3. 1. 3 手法の独自性

3. 2 先行・後続関係の演算

3. 2. 1 要素間の直接先行・後続関係

最初に、IFM（以下、行列としてあらわすときは \mathbf{R} と表記）によってある要素 c_i に直接後続する要素および直接先行する要素を導く方法を示す。ここでは前者を直接後続要素、後者を直接先行要素と呼ぶことにし、それぞれを $s(c_i)$ 、 $p(c_i)$ と表す¹。

行列 \mathbf{R} においては、先行要素 c_i と後続要素 c_j とが直接的な工程順序関係を持つとき、すなわち c_j が c_i に対してインターフェイスを持つときに限ってその成分 r_{ij} は 1 となり、その他の場合は 0 としてあるから（2 章 2. 4）， \mathbf{R} の第 i 行の非零要素群 $\{r_{ij}\}$ に対応する要素の集合 $\{c_j\}$ が、要素 c_i の直接後続要素 $s(c_i)$ に対応することになる。あるいは、IFM の第 i 行をベクトルとみなし、それ自体を直接後続要素を表すベクトルとみることもできる。この場合、 s を太字で $\mathbf{s}(c_i)$ と記すことにする。 c_i の直接後続要素 $\mathbf{s}(c_i)$ には c_i 自身も含まれる。

\mathbf{R} の (h, i) 成分が 1 であれば要素 c_h は c_i に先行することになるから、同様に \mathbf{R} の第 i 列は c_i の直接先行要素全体を表すベクトル $\mathbf{p}(c_i)$ とみなすことができる。

また、要素 c_i を要素全体が張る n 次元空間のベクトル \mathbf{c}_i として表すことにすれば²， $\mathbf{s}(c_i)$ ， $\mathbf{p}(c_i)$ をそれぞれ $\mathbf{s}(c_i)$ ， $\mathbf{p}(c_i)$ のように書き表すこともできる。 n 項列ベクトル $\mathbf{p}(c_i)$ および n 項行ベクトル ${}^t\mathbf{s}(c_i)$ を用いて \mathbf{R} を表すと次のように書ける。

$$\mathbf{R} = (\mathbf{p}(c_1) \cdots \mathbf{p}(c_i) \cdots \mathbf{p}(c_n)) = \begin{pmatrix} {}^t\mathbf{s}(c_1) \\ \vdots \\ {}^t\mathbf{s}(c_i) \\ \vdots \\ {}^t\mathbf{s}(c_n) \end{pmatrix} \quad (3.2.1)$$

直接先行要素および直接後続要素は、要素を表すベクトルの IFM による写像として求めることができる。

$$\mathbf{p}(c_i) = (\mathbf{p}_k) = \mathbf{R} \mathbf{c}_i \quad (3.2.2)$$

$${}^t\mathbf{s}(c_i) = (\mathbf{s}_k) = {}^t\mathbf{c}_i \mathbf{R} \quad (3.2.3)$$

$$\mathbf{p}_k = \sum_{h=1}^n r_{kh} \mathbf{c}_h = r_{ki}$$

$$\mathbf{s}_k = \sum_{h=1}^n \mathbf{c}_h r_{hk} = r_{ik}$$

¹ それぞれ、predecessor, successor の頭文字をとったもの。
² i 番目の要素が 1，それ以外の要素が 0 である行列。

上式の線形性からあきらかのように，任意の要素の組を同様にベクトル \mathbf{c} として表し，それらの直接先行要素の集合および直接後続要素の集合をベクトルの形で導くことができる。すなわち，

$$\mathbf{p}(\mathbf{c}) = \mathbf{R} \mathbf{c} \quad (3.2.4) \quad {}^t \mathbf{s}(\mathbf{c})$$

$$= {}^t \mathbf{c} \mathbf{R} \quad (3.2.5)$$

\mathbf{c} は任意の要素の組に相当する成分を 1 とし，他を 0 としたベクトル。

ただし， ${}^t \mathbf{s}(\mathbf{c})$ ， $\mathbf{p}(\mathbf{c})$ の成分には 2 以上の非零要素が出てくることがある。この場合には，非零要素全体を先行ないし後続要素とみなす。

\mathbf{R} の転置行列， ${}^t \mathbf{R}$ を用いて同様の演算を定義することもできる。 ${}^t \mathbf{R}$ の (i, j) 成分を ${}^t r_{ij}$ とすると，定義により次式のように書ける。

$${}^t r_{ij} = r_{ji} \quad (3.2.6)$$

要素 c_i の直接先行要素は， $r_{ji} = 1$ である j に対応するすべての c_j であるからその集合 $\mathbf{p}(c_i)$ は \mathbf{R} の j 行目の n 項行ベクトルで与えられ，また同様に要素 c_i の直接後続要素 $\mathbf{s}(c_i)$ は \mathbf{R} の i 列目の n 項列ベクトルとして与えられる。IFM の要素のベクトルで表現すると，以上は次式の演算としてあらわされる。

$$\mathbf{s}(c_i) = {}^t \mathbf{R} \mathbf{c}_i \quad (3.2.7)$$

$${}^t \mathbf{p}(c_i) = {}^t \mathbf{c}_i {}^t \mathbf{R} \quad (3.2.8)$$

同じく，任意の要素の組を表すベクトル \mathbf{c} の直接後続要素，直接先行要素は次式で与えられる。

$$\mathbf{s}(\mathbf{c}) = {}^t \mathbf{R} \mathbf{c} \quad (3.2.9)$$

$${}^t \mathbf{p}(\mathbf{c}) = {}^t \mathbf{c} {}^t \mathbf{R} \quad (3.2.10)$$

式(3.2.2)，(3.2.3)と式(3.2.7)，(3.2.8)の関係，あるいは式(3.2.4)，(3.2.5)と式(3.2.9)，(3.2.10)の関係はつぎのことからも明らかである。

$$\mathbf{s}(c_i) = {}^t ({}^t \mathbf{s}(c_i)) = {}^t ({}^t \mathbf{c}_i \mathbf{R}) = {}^t \mathbf{R} {}^t ({}^t \mathbf{c}_i) = {}^t \mathbf{R} \mathbf{c}_i$$

$${}^t \mathbf{p}(c_i) = {}^t (\mathbf{R} \mathbf{c}_i) = {}^t \mathbf{c}_i {}^t \mathbf{R}$$

$$\mathbf{s}(\mathbf{c}) = {}^t ({}^t \mathbf{s}(\mathbf{c})) = {}^t ({}^t \mathbf{c} \mathbf{R}) = {}^t \mathbf{R} {}^t ({}^t \mathbf{c}) = {}^t \mathbf{R} \mathbf{c}$$

$${}^t \mathbf{p}(\mathbf{c}) = {}^t (\mathbf{R} \mathbf{c}) = {}^t \mathbf{c} {}^t \mathbf{R}$$

また，(3.2.1) 式にならって ${}^t \mathbf{R}$ を $\mathbf{s}(c_i)$ ， $\mathbf{p}(c_i)$ であらわすと次のようになる。

$${}^tR = (s(c_1) \cdots s(c_i) \cdots s(c_n)) = \begin{pmatrix} {}^t p(c_1) \\ \vdots \\ {}^t p(c_i) \\ \vdots \\ {}^t p(c_n) \end{pmatrix} \quad (3.2.11)$$

3. 2. 2 先行要素, 後続要素と誘引要素

前項ではある要素または要素の組に直接先行あるいは後続する要素を求める方法について述べたが, ここではそれを拡張し, 間接的に先行, 後続する要素の全体を求める方法について述べる。

ある要素 c_i に先行する要素の集合は, IFM をあらかず行列 R と c_i をあらかずベクトル c_i により, $R c_i$ と求められる。さらにこの要素の集合に先行する要素の集合は, 同じく R を用いて, $R(R c_i)$ と求めることができる。よく知られているように, ベクトルの j 成分は 1 または 0 以外の値をとることがあり, グラフ理論においてはその値はある節点から別の節点に 2 つの辺を経由して至る経路の本数として意味付けられている。グラフ理論ではインターフェイス・マトリクスに相当する行列を接続行列と呼ぶが, 一般に接続行列を k 乗して得られる行列の (i, j) 成分の値は, 節点 i から辺の向きにしたがって k 個の辺をたどって節点 j に到達するときの経路の数を表す。

ここでは, 主たる目的が要素間の順序関係を明らかにすることにあるため, 節点間の辺や経路の数を知ることとは必要ではなく, それらの有無がわかりさえすればよい。したがって, IFM の積, ベキ乗の演算においては, ブール代数によりその成分の値を 1, 0 に限定しておく。すなわち, R の成分 r_{ij} (1 または 0) に関し, その積, 和の演算を次のように定義する。

$$\begin{aligned} 0 \times 0 &= 0, & 1 \times 0 &= 0, & 1 \times 1 &= 1 \\ 0 + 0 &= 0, & 1 + 0 &= 1, & 1 + 1 &= 1 \end{aligned}$$

上記の演算における R のブール積を R^k と表記することにする。 R^k の成分 r_{ij} の値は 1 または 0 である。さらに, 1 または 0 を成分とする $(1, m)$ 型行列 A と (m, n) 型行列 (ベクトル) B のブール積については, $A \cdot B$ のように表記することにする。このとき, 任意の要素 c_i をあらかずベクトル c_i の直接先行要素 $p(c_i)$ および直接後続要素 $s(c_i)$ は次のようにあらわされる。

$$p(c_i) = R \cdot c_i = {}^t({}^t c_i \cdot {}^t R) \quad (= R c_i) \quad (3.2.12)$$

$$s(c_i) = {}^t R \cdot c_i = {}^t({}^t c_i \cdot R) \quad (= {}^t R c_i) \quad (3.2.13)$$

同様に, 任意の要素の組 c の直接先行要素 $p(c)$ および直接後続要素 $s(c)$ は次のようにあらわされる。

$$p(c) = R \cdot c = {}^t({}^t c \cdot {}^t R) \quad (\neq R c) \quad (3.2.14)$$

$$s(c) = {}^t R \cdot c = {}^t({}^t c \cdot R) \quad (\neq {}^t R c) \quad (3.2.15)$$

$p(c)$ の直接先行要素の集合 $p(p(c))$, $s(c)$ の直接後続要素の集合 $s(s(c))$ は, それぞれ次のようになる。

$$p(p(c)) = R \cdot (R \cdot c) = R^2 \cdot c \quad (3.2.16)$$

$$s(s(c)) = {}^t R \cdot ({}^t R \cdot c) = ({}^t R)^2 \cdot c \quad (3.2.17)$$

$s(s(c))$ を ${}^t R$ によらず R と c のみであらわすと次のようになる。

$$\begin{aligned} s(s(c)) &= {}^t ({}^t ({}^t R \cdot ({}^t R \cdot c))) \\ &= {}^t ({}^t ({}^t R \cdot c) \cdot {}^t ({}^t R)) \\ &= {}^t ({}^t c \cdot R \cdot R) \\ &= {}^t ({}^t c \cdot R^2) \end{aligned} \quad (3.2.18)$$

c から k 個以下の辺を経て先行する要素の集合を c の k 次の先行要素と呼ぶことにし、 $p_k(c)$ とあらわす。また、 c の k 次の後続要素を $s_k(c)$ とあらわす。すると、直接先行要素、直接後続要素はそれぞれ1次の先行要素、1次の後続要素と言い換えることができる。このとき、一般に次の式が成立する。

$$p_k(c) = R^k \cdot c \quad (3.2.19)$$

$$s_k(c) = {}^t R^k \cdot c = {}^t ({}^t c \cdot R^k) \quad (3.2.20)$$

c からちょうど k 個の辺をへて先行または後続する要素の集合は、それぞれ次のベクトルに対応する (E は単位行列)。

$$\begin{aligned} p_k(c) - p_{k-1}(c) &= (R - E)^k \cdot c \\ s_k(c) - s_{k-1}(c) &= ({}^t (R - E))^k \cdot c \end{aligned}$$

n 個の要素からなる工程グラフが連結であるとする、ブール演算を施した場合の R^k は $k = n - 1$ までの段階で収束し、それ以上変化しない。これを R の(逆行の)到達可能行列と呼び、 R_p とあらわす。同様に、 R の順行の到達可能行列 R_s は、 ${}^t R^k$ の収束した行列として定義される。 c の先行要素全体の集合、後続要素全体の集合をそれぞれ大文字で $P(c)$ 、 $S(c)$ と書くと、次のようになる。

$$P(c) = R_p \cdot c = {}^t ({}^t c \cdot R_s) \quad (3.2.21)$$

$$S(c) = R_s \cdot c = {}^t ({}^t c \cdot R_p) \quad (3.2.22)$$

$$R_s = {}^t R_p \quad (3.2.23)$$

$$R_p = {}^t R_s \quad (3.2.24)$$

特に $c = c_i$ のとき、上式はブール積ではなく、通常の行列とベクトルの演算としても成り立つ。 $P(c)$ 、 $S(c)$ より、次のように c に関する誘引要素 $I(c)$ 、平行要素 $L(c)$ が定義される。

$$I(c) = P(c) \cap S(c) \quad (3.2.25)$$

$$L(c) = \{c_i \mid c_i \notin P(c) \cup S(c)\} \quad (3.2.26)$$

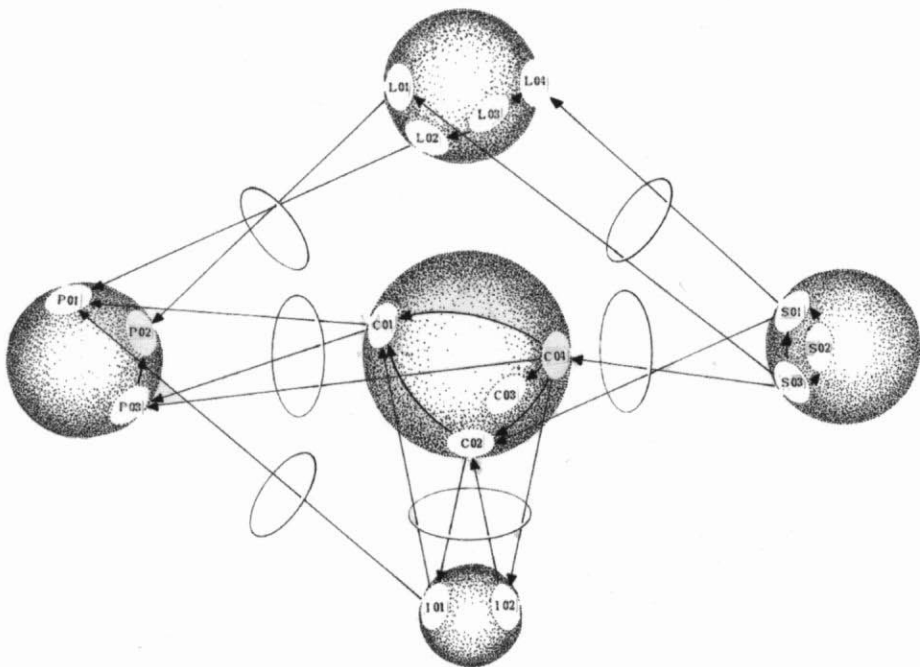
誘引要素 $I(c)$ は c に先行すると同時に後続する要素の集合である。誘引要素 $I(c)$ は c を含む。

誘引要素の先行要素全体、後続要素全体の集合は、 c の先行要素全体、後続要素全体と同じである。

$$P(c) = P(I(c))$$

$$S(c) = S(I(c))$$

一方、平行要素 $L(c)$ は、 c に先行も後続もしない要素の集合、すなわち $P(c) \cup S(c)$ の余集合であって、 c との順序関係は決められないものである(図)。ネットワーク、および c の選び方によっては、 $I(c)$ または $L(c)$ が存在しないこともある。



3. 3 トポロジカル・オーダリング

3. 3. 1 トポロジカル・オーダリングに関する既往の手法

工程要素をその工程順序にしたがって並び替えることをトポロジカル・オーダリングという。並び替えはある種の工程順序に関する情報にしたがって遂行されるわけであるが、その情報は並び替えの前後で変化しないと前提するのが妥当であろう。また、並び替えの演算には、これらの情報の表現と直接結びついた簡便なものであることが望ましい。

トポロジカル・オーダリングについてはこれまでにさまざまな手法が提案され、用いられてきているが、その代表的なものに尻取り法 (threading method) と位取り法 (ranking method) がある。IFM によるトポロジカル・オーダリングを示す前にこれらの手法の概要を示し、上記の前提にしたがって評価を行ってみる。

(a) 尻取り法

この方法ではランダムに節点番号をふったアロー・ダイアグラムにもとづき、その節点番号を工程

順に付け替えることを行う。したがって、ダミー作業をまず準備段階で定義しなければならないという点が厄介である。

次に、ダミー作業を含めた作業リストをつくり、これを小さい順に並べておく。すなわち、作業 (ij) の i について大分類し、ついで j についての小分類を行う。この作業リストをまずニューメリック・エリアと呼ばれる作業場所に置く。このほかに、ロジカル・エリア、ターミナル・エリアと呼ばれる作業場所を用意する。並び替えの手順は、概略次による。

①ニューメリック・エリアの中の最初の作業の集合 (i が同じ番号の集合) をロジカル・エリアに移す。

②ロジカル・エリアの中の最初の作業の j により、その直接後続の節点番号もつ作業の集合をニューメリック・エリアの中から検索する。もしあれば、その集合をロジカル・エリアの最後に続ける。

③ロジカル・エリアの中の第 2, 第 3 の作業を順次索引要素としながら、それらに直接後続する節点番号を持つ作業の集合をニューメリック・エリアとロジカル・エリアの中で探す。もしニューメリック・エリアの中で見つければ、それらをロジカル・エリアの最後に移す。もしロジカル・エリアの中で見つかり、かつそれがいま索引要素となっている作業より後にリストされていれば、そのまま次の作業に進む。もし前にリストされていれば、その集合は後続すべきものであるからロジカル・エリアの最後に移す。いずれにも見つからなければ、次の索引要素に移る。

④以上の手順でロジカル・エリア内の作業による検索をすべて終えたなら、これ以上これらに後続するものがないということであるから、ロジカル・エリア内の全内容をそのままターミナル・エリアの先頭に移す。すでにターミナル・エリアの中に作業が存在する場合には、それらの前に移す。

⑤ニューメリック・エリアの中にまだ作業が残っていれば、①～④を繰り返し、ニューメリック・エリアとロジカル・エリアが空になるまで続ける。

最終的にターミナル・エリアの中で得られた配列が工程順序に従った配列となっているから、この順に番号を付け替えるとトポロジカル・オーダリングが完成する。要するに、ターミナル・エリアにランダムに配列された作業の集合から一部分を抜き出してロジカル・エリアに移し、そこで誘引要素も含めた順序化の手続きを施し、さらにそのブロックを単位とした順序化をターミナル・エリアで行っているわけである。

この方法では、ダミーも含めた作業とその番号付けをあらかじめ確定しておくなど準備作業が面倒であるうえに、計算も複雑である。

(b) 位取り法

この方法は、より計算機に適した手法として考えられたものである。その手順は概ね次のとおりである。

①後続要素を持たない要素の集合として、ネットワークの始点を求める。ここでも、アロー型のネットワークを前提としているから、手続きとしては先行結合点を持たない結合点を探すことになる。そのような集合が存在しなければ全体がループになっている。もし存在すれば次のステップに進む。

②見つかった結合点に通し番号を付ける。

③それらの結合点の集合をネットワークから削除する。

④残りのネットワークから同様に先行結合点を持たない結合点を探す。この手続きを繰り返し、ネットワークが空集合となったら、オーダリングは終了する。

⑤途中でそれ以上番号付けができなくなれば、それはループが存在する場合である。

位取り法の原理は、次項に述べる方法、すなわち到達可能行列によってトポロジカル・オーダリン

グを行う方法とよく似ている。しかし、位取り法では作業を結合点の組としてのアローで定義するために準備作業が面倒であること、および特にカーンの手法では作業表 (activity list : A表)、結合点表 (event list : N表) を用いた計算を行うため、手続きもやや煩雑になるという欠点がある。

3. 3. 2 到達可能行列によるトポロジカル・オーダリング

前項で扱った順序化の手法は、いずれもアロー型のネットワークをつくる手法である。アロー型のネットワークでは、一つの作業に対応する節点の組をそれぞれ工程的に矛盾のないように配列することによって、オーダリングは基本的に完了する。ただし、アロー型ネットワークには常にダミー作業をつくっておかなければならないという厄介が付きまとうことは既に述べたとおりであるし、また冗長性を排除することが要請されることもある。しかし、フロー型のネットワークでは少なくともダミーを導入しなくても作業の定義はできる。IFM はアロー型とフロー型のどちらもともいえる定義となっているので、それをフロー型とみなして直接オーダリングを行うことができる。以下に、前節で導いた到達可能行列を用いてトポロジカル・オーダリングを行う方法を示す。

トポロジカルという言葉には、本来、近い関係にあるものを近いところに移すという意味が含まれているはずであるが、そうした演算法については次節に譲る。残る問題は、いかにして工程順序に従った (少なくとも矛盾のない) 配列を得ることができるかということになる。これを行列の属性に即して表現すれば、ある要素の先行要素がすべてその前に配置されていればよいということになり、すなわち、いかにして IFM を上三角行列に変形するかという問題と同値となる。

IFM をあらわす行列 R から導かれる順行、逆行の到達可能行列 R_s 、 R_p と特定の要素の組 c の関係は、それぞれ次の式で定義されている。

$$P(c) = R_p \cdot c = {}^t(c \cdot R_s) \quad (3.2.21)$$

$$S(c) = R_s \cdot c = {}^t(c \cdot R_p) \quad (3.2.22)$$

c を特定の要素 c_i と置き換えてみれば明らかなように、 R_s の第 i 行は要素 c_i に後続する要素全体をあらわすベクトルとなっており、また同じく第 i 列は先行要素全体をあらわすベクトルとなっている。

もし R_s の第 i 行の第 i 成分 (すなわち自分自身) 以外がすべて 0 であれば、要素 c_i に後続する要素がないことを意味する。また、 R_s の第 i 列の第 i 成分以外が 0 であれば、要素 c_i に先行する要素がないことを意味する。

したがって、 R_s の第 i 列の第 i 成分以外が 0 である要素の集合を取り出して、これらを先頭の要素群 A_1 とすることができる。すなわち、

$$\begin{aligned} A_1 &= \{c_i \mid P(c_i) = c_i\} \\ &= \{c_i \mid {}^t(c_i \cdot R_s) = c_i\} \end{aligned} \quad (3.3.1)$$

先頭が確定したら、これを R_s から取り除く。具体的には、 A_1 の後続要素とのインターフェイス、すなわち R_s の第 i 行 ($i \in \{i \mid A_1 = \{c_i\}\}$) の非零成分 (ただし対角成分を除く) をすべて 0 とし、この行列を R_{s1} と書き直す。 R_{s1} によって、改めて、先行要素を持たない要素の集合 A_2 が求められる。 A_{20} に含まれているが A_1 に含まれない要素の集合 A_2 が、この段階で新たに先頭に位置付けられたものである。

$$\mathbf{A}_2 = \mathbf{A}_{20} - \mathbf{A}_1 \quad (3.3.2)$$

$$\mathbf{A}_{20} = \{ \mathbf{c}_i \mid {}^t \mathbf{c}_i \cdot \mathbf{R}_{s_1} = \mathbf{c}_i \} \quad (3.3.3)$$

以後同様に、残りの要素の集合が ϕ になるまでこの手続きを繰り返す。最終的に \mathbf{R}_{s_k} は単位行列のかたちになる。

こうして得られた $\{i\}$ の列、 $\mathbf{A}_1 \mathbf{A}_2 \cdots \mathbf{A}_{k-1}$ が、工程順に並べられた要素の番号である。また、この順番に \mathbf{R} の行、列の要素を入れ替えた行列 \mathbf{R}_{i_0} がトポロジカル・オーダリングのなされたネットワークそのものの表現となっている。

前節で用いた例に基づいて、図表 3.3.1 に到達可能行列によるネットワーキングを例示する。

3. 3. 3 IFM とアロー型、フロー型ネットワークの関係

前項で述べたように、トポロジカル・オーダリングの施された IFM は、それ自体でネットワークをわかりやすく表現したものとなっている。つまり、それは対角成分に作業を工程順に配し、作業間の取り合いの関係をインターフェイスとして行列の上三角に与えたかたちになっている。後の章にも示されるように、クリティカル・パスの計算などネットワークの応用に関してはこのような定義でいっこうに問題はない。ここでは専らネットワークの表現上の問題、すなわち IFM によるネットワーク表現と慣用的なアロー型のネットワークとの関連について考察してみる。

まず、両者を対照しながら、アロー型のネットワークの特徴を以下に列記してみる。

①アロー型ネットワークでは作業が前後の結合点の組として定義される。一方 IFM によるネットワークでは、作業が対角成分 (i, i) として定義されるので、前後に同じ結合点を持つ作業ができてしまう。

②上記からもわかるように、IFM によるネットワークではループ（特に自己ループ）を許しているが、アロー型ではこれは許されない。

③アロー型ネットワークでは、平行作業（前後ともに同じ結合点を持つ複数の作業）が生ずると、同一結合点で複数の要素の先行後続関係が区別されなくなるのを防ぐためにダミー作業を必要とするが、IFM によるネットワークでは少なくとも後者に対する配慮は要求されない。

④アロー型ネットワークは必ず始点と終点を持つが、IFM によるネットワークでは特にそれを要求しない。むしろ、始点、終点を設けずに自由に先行し、あるいは後続する要素間の順序構造をそのまま保存することによって、後の章で示されるように、複数工区、複数プロジェクト間の工程問題を解く手法の開発を可能にしている。

前述のように、IFM によるネットワークはアロー型とフロー型のネットワークの中間的な性格を持つが、事実それは両者の長所を兼ね備えたものとなっている。フロー型ネットワークの長所としては、それがダミー作業を必要としないために表現の一意性（グラフとしてではなく結合の構造としての）を持っていることがあげられるが、IFM もその点では同様である。このことはすなわちアロー型の欠点となっていて、アロー型からは一意的にフロー・ダイアグラムへの変換ができるが、逆にフロー型からアロー型への一意的な変換はできないという、両者の半双対の関係をもたらしている。一方、アロー型の長所としては、結合点に番号（記号）をつけるだけで順序構造と作業の両方をわかりやすく表現でき、また工程計算の対象として扱いやすいということがあげられるが、この点もまた IFM に共有された長所である。

要するに、IFM によるネットワークは行列の対角成分 (i, i) によって作業をフロー型に近い

たちで表現し、その他の成分 (i, j) がアロー型に近いたちで順序関係をあてているということができる。あるいは、IFM をフロー型ネットワークとして解釈すれば (i, i) が作業をあらわし (i, j) が関係をあらわすとみなすことができるし、他方それをアロー型ネットワークとして解釈しようとするのであれば、 (i, i) が実際の作業を、また (i, j) がダミー作業をあらわしているとみなすことができる。

IFM をアロー型に変換するときには、したがって、ダミー作業をふんだんに含んだネットワークとすればよい。ダミーの冗長性については別途配慮する必要があるが、本来アロー型が一意的に決まらないものであることを考えればそれほど重視すべきことでもない。

以上に述べた IFM とその他のネットワークの表現の関係を、図 3.3.2 に示す。

3. 3. 4 IFM とネットワークの冗長性

トポロジカル・オーダリングにおいては、専ら工程要素の順序構造化の問題を扱っているわけであるが、単に順序付けを行うことと、要素間の順序関係の構造を保ちながら順序付けを行うことでは、その意味合いが違う。単に要素を順番どおりに並べるだけでよいのであれば、それに必要十分な情報が含まれていればよく、また計算も早い。しかし、並び替えを行う前にある要素の集合が有していた順序関係の構造がその後の演算のためにも必要な場合、その構造をオーダリングの前後で不変に保っておくべきである。一般に、要素間の関係（グラフのパス）が先行後続といった順序をあらわす以上のものとして期待されていなければ、構造そのものは失われても構わないといえる。一方、グラフのパスに、例えばその長さが時間をあらわすといったように、情報が付加されており、それがあつた種の演算の対象となる場合には、いかなるパスも省略することはできない。このように、オーダリングの手法の適否はその目的によって違ひう。IFM は要素間の順序構造に基づき、他の変数を組み合わせてさまざまな手法に発展させることを目的としたものなので、その構造が絶えず保存されていることを前提としている。

トポロジカル・オーダリングを行うのに不必要なパスを持つとき、そのネットワークは冗長であるという（図 3.3.3）。IFM には要素間の取り合い、すなわちインターフェイスがすべて表現されているため、一般に IFM は冗長性を持ったネットワークとなっている。IFM を完全に非冗長なネットワークに置換する必要はないが、以下に IFM と冗長性との関連を明らかにしておく。

順序付けに不必要なパスを最大限に持つネットワークを、完全冗長なネットワークと呼ぶことにする。したがって、完全冗長なネットワークとは、間接的な先行・後続関係がある場合に、それらをすべて直接的な関係に置き換えたものということになる。それゆえ、到達可能行列は完全冗長性を備えた行列であるし、IFM から到達可能行列を求めることは、任意のネットワークを完全冗長なネットワークに変換することを意味する。

完全非冗長性とは、逆に、間接的な先行・後続関係がいつさい排除されたネットワークの属性を表した言葉である。したがって、完全非冗長なネットワークの任意の要素 c_i を取り出すとき、 c_i はその直接先行要素 $p(c_i)$ を除く先行要素全体、すなわち、 $a) R_p \cdot c_i \cap p(c_i)$ に対してインターフェイスをもたず、また、 $b) p(c_i)$ の要素の中で他のいずれかに先行する要素に対してもインターフェイスを持たない。これが、あるネットワークが完全非冗長であるための必要十分条件であり、この場合、任意の要素 c_i はその直接後続要素 $s(c_i)$ を除く後続要素全体に直接先行せず、 $s(c_i)$ の要素間にインターフェイスを持つこともない。

a), b) の条件は結局 c_i の直接先行要素の中で c_i にただ一つのパスをもつ要素ということになるから、ブール積によらない行列の積で、 R_p に相当する次数だけ R を掛けたときのその成分の値に

よってそれを判定できる。すなわち R_p の次数を d として、 R^d の対角成分($=1$)のほか値が d の成分だけを残し、その値を1に替えた行列が、完全非冗長なネットワークをあらわす行列である。

(図 IFM とその完全冗長、および完全非冗長なネットワークの行列表現)

3. 4 ループ探索

3. 4. 1 ループの意味とその取扱い

通常ネットワーク計算においては、ループが存在しないことを前提として計算手法が構築されている。つまり、もしループが存在する場合は、それを排除しなければならないようになっている。ところが、建築のように人為的なプロセス、あるいは物的要素の複雑な組立といった問題を扱うとき、ループは不可避免に出現しがちである。したがって、これらの存在を念頭に置いたネットワーク計算の手法が必要となるはずであるし、また、それらの構造化を取り扱うことができるような手法が望まれるわけである。

ループが出現する可能性は次の理由による。

①入力ミスまたは定義ミス

実際にはループが存在しないインターフェイスの関係であるのに、誤ったデータが入力されてしまう場合がある。特に、繰り返し型のプロセスでは、単一工区での要素間のIFの有無を判定、定義する際に、異なった工区間のIFと同一工区間のIFが混同されてしまうことが多い。こうしたループをネットワーク計算の過程でチェックしたり、矛盾のないネットワークに変換することもIFMの機能の一つとして必要である。

②手戻り

工程自体に手戻りがある場合、すなわちある工程要素が終了した後いくつかの要素を経て再びその要素にプロセスが再帰する場合、その間のプロセス全体がループをなす。このようなケースは、ふつうプロジェクトの実施段階の異常作業として現れ、計画段階では積極的に意味付けられることはない。しかし、手戻りが本質的に発生しがちなプロセスであれば、プロセス制御の観点から、手戻りが生じた場合にも備えた計算処理ができるようになっていることが望ましい。

③同時作業あるいは組作業

いくつかの要素がほぼ同時に開始され終了することになっていながら、なおそれらの間の先行・後続関係を明らかに定義できないとき、これらは同時作業とみなされ、ループとして表現される。特に異なった職種の間での協力的な同時作業が要請される場合は、これを組作業という。K 未作業は、建築の組立にはごく一般的に見られるループである。

④自己ループ

IFM ではその行列 R の成分(i, i)を自己ループとして扱い、これに作業を対応させている。その理由については前節に述べたとおりである。

以上のように、構工法計画あるいは建築生産の分野においては、ループは不可避免に出現するものであり、したがってこのような分野を対象としたネットワークにはループを積極的に組み込んだ演算手法が要求されるわけである。このようなループに関する演算を取り入れたことは、本論文であつかう IFM によるネットワーク手法の大きな特徴の一つである。

3. 5 ループ解消による単工区モデルの作成

3. 6 本章の結論

【成果】

- *IFM のべき乗から得られる到達可能行列を用いて工程順序にしたがった構成材（工程要素）の並び替え（トポロジカル・オーダーリング）を行う方法を求めた
- *その際、ダミー作業を加えたり、冗長性を除くために IFM の順序構造を変更しなくて済むことを示した
- *到達可能行列は、完全冗長性を持つ IFM の表現であることをあきらかにした
- *ループ探索とその解消を行う方法を示した

【第3章に関する主要既発表論文・講演】

(略)

第4章 インターフェイス・マトリクスによる構工法のシステム化

4. 1 はじめに

本章で扱うのは「ジョブ・コーディネーション」といわれる問題である。建築は多くの専門職種が錯綜した工程にしたがって、順次作業を行う。この場合、工程を整理し、計画管理しやすくするには大別して二つの方法がある。ひとつは、本論文の後半で扱うことになる多工区同期化構工法のように、作業空間を分けることによって同時作業を可能にしながら同時作業をはかる方法である。もうひとつは、工程や構法、あるいは担当職種を変えて、同種の作業のまとまりをなるべく大きくすることである。これにより、全体を手切れのよいいくつかの部分に分けて進行させることができる。

後者が、システムズ・ビルディングの基本的な考え方である。システムズ・ビルディングにおいては、全体（トータル・システム）はいくつかのサブシステムに分割される。各サブシステムは「もの」とそれに対応する「仕事」とから成る。サブシステム間のインターフェイスを最小限にとどめることによって、各サブシステムの開発・設計・生産・施工を他から独立して進められるようにすることが、システムズ・ビルディングの目的であり、手法である。

本章では、IFMにより、さまざまな構成材（＝工程要素）から成るトータル・システムからいかにしてサブシステムを抽出するかという問題と、抽出されたサブシステムをいかにまとまりよく、手切れのよいサブシステムにするかという問題を扱う。

システム化に際して IFM 以外に用いられる外部変数（属性）は、工程要素の担当職種である。

4. 2 IF によるサブシステムの抽出

4. 2. 1 サブシステム化のための IMF の応用

IFM の演算によって全体工程の中からサブシステムを抽出し、その範囲を確定する方法について述べる。

いま、工程全体に含まれる工程要素の数を次元数とするベクトル空間を考える。ある空間的まとまり、機能的まとまりを持ち、かつ同一の職種によって施工される工程要素（構成材）群をとりあげ、それをあるサブシステムの構成要素の候補とする。これらの工程要素のそれぞれには、担当職種があらかじめ与えられているものとする。

サブシステムの構成要素の候補とされた要素群を SS_1 とすると、前章で示したように、 SS_1 に対応するベクトル SS_1 が定義できる。その先行要素 $P(SS_1)$ 、後続要素 $S(SS_1)$ により、 SS_1 の誘引要素 $I(SS_1)$ が求められる（図 4-1、4-2）。 $I(SS_1)$ は SS_1 を含む。

サブシステム化にあたって検討の対象となるのは、 SS_1 および $I(SS_1)$ である。

誘引要素のうち、 $SS_1 \cap I(SS_1)$ に相当する部分（狭義の誘引要素）は、の担当職種ではない職種によって施工される工程要素群である。必要であれば、これらの担当を変え、 SS_1 に統合できるか否かを検討する。この手続きを終えると、工程要素の全体は、先行要素、抽出されたサブシステム (SS_1)、狭義の誘引要素、後続要素の 4 つに分けられる。

サブシステム化の評価は、次による。

- ①狭義の誘引要素が十分に少ないか、またはそれらによるサブシステム工程の複雑化の度合いが適正にとどまっているか。
- ②サブシステムと先行要素、後続要素との間のインターフェイスが量的、質的に適切であるか。

4. 2. 2 検討事例－1 複合化構工法の部分構工法の生産性比較¹

以下は、SRC・RC 系の複合化構法集合住宅躯体工事の部分構工法の実産性を比較するために、IFM によるサブシステムの抽出を応用した事例である。労務量による比較分析は、細かく分節された工程要素について行われた実際の躯体工事における調査データに基づいている。

●比較のための条件

部分工法を抽出して比較するためにはまず以下の条件の違いを検討する必要がある。

- [条件 1] インターフェイスの物理的条件が同一
- [条件 2] 構成材を扱う職種、技能、機器が同一
- [条件 3] 性能、精度が同一
- [条件 4] 作業条件が同一

また、比較の方法としては以下の 2 つがある。

- ① 同じ BE 量の部位を抽出し比較する。
- ② BE 量と労務量の相関を用いて比較する。

[条件 1 ～ 4] は構成材の種類と関連が深い。在来工法は構成材の種類がほぼ同じで [条件 1 ～ 4] にも標準が設定できた。そのため方法②によって標準労務量（＝歩掛り）が把握された。しかし、複合化による省力化工法が導入され異種の構成材が用いられると [条件 1 ～ 4] 中でも 1 と 2 が変化し比較が難しい。この場合は、方法①により [条件] の影響を考慮しながら比較することが必要である。

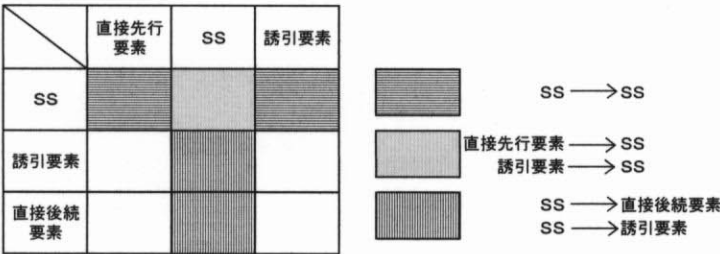
●ケーススタディ

2 工事について観察による現場労務量調査を行ない部分構工法の比較を行った。部位、職種、構成材の種類等に着目し B - 84、C - 88 の工程要素に分割し各工程要素の関係をインターフェイスによって定義した。部分工法の抽出対象はいろいろあるが、ここでは住戸床及び小梁について検討する。なお、調査は B 9 階、C 4 階で行ない数十個分のデータを平均した値を用いてある。

■住戸床

両工事の躯体形状は同じで、住戸床の BE 量も等しい。工法は B がオムニア版床工法、C が在来床工法である。方法①によって 2 つの部分構工法を比較する場合、構成材を同一の範囲で切り取ることが大切であるが、実際は完璧な分割を行いがたい、この場合コンクリート打設は全部位に及び各部位に分割できないため双方から除いて比較した。その結果、B - 7、C - 4 の工程要素を抽出し、さらに先行、後続する工程要素を図 4-1 のように並び換えて表わした。

図 4-1 工程要素の表わし方



¹ 昭和 56 年大会梗概

それぞれ床工法の工程要素は図 4-2、図 4-3 に示すとおりである。さらに労務量を表 4-1、図 4-3 のように集計した。それによると B の法がトータルで 2 割程度、構成材自身の作業だけでは 1 割強省力化されている。しかし、それ以上に担当する職種の変動が大きく一概に比較できない。さらにこの内同種の構成材を扱う鉄筋工の労務量生産性をみる。各々の資材量は B-0.29 t、C-0.94 t でありこれらの数値で労務量を割ると、B-875.9 人分/t、C-177.7 人分/t である。B の方が著しく生産性が悪い。[条件 1] に違いがあること理由の 1 つとして考えられる。

■小梁

労務量の集計結果のみを表 4-3、図 4-5 に示す。B は P C 工法，C は在来工法である。ここでも双方からコンクリート打設作業を除き，各々らの工程要素を抽出した。B は C に比べ 1/2 以上の省力化になっている。

以上から，次のことが言える。標準労務量は構工法（特に構成材の種類とジョブコーディネーション）が安定した状態においては把握しやすい。それは先に示した[条件 1～4]が標準化されているからである。工法の多様な分化はさまざまな質的变化をもたらし，それによって標準を設定しづらくしている。

図 4-2 B 工事床工法工程要素図

	12	14	15	22	23	24	25	26	27	28	57	59	62	63	69	70	71	94	84
	柱鉄筋着床 9	P C 小梁設置 1-6	P C 小梁上部主筋配筋 3	通路側梁主筋圧着 5	バルコニー側梁主筋圧着 5	戸境側梁主筋圧着 5	戸境側梁主筋圧着 3	バルコニー側梁配筋 3	戸境側梁配筋 3	戸境側梁配筋 3	P C 小梁支保工解体 6	住戸オムニア版支保工組立	住戸オムニア版取付調整	住戸オムニア版着床	住戸床電気配管	住戸床電気配管	住戸床電気配管	住戸オムニア版支保工解体	コンクリート打設
59											0	27							
62							0	0	0	0		0	176						
63				5	5	5	5						14	1					
69			0	0									70		5				
70			0	1	1	1		2					4			15			
71			0	0	0	0	0	0	0				192		0	0	62		
94												0						45	0
84	コンクリート打設 7・9・10・12・13													0	0	0	0		
74	通路電気配管 10														2				
75	住戸内電線取付 10												8		6		0		
81	給排水、雑排水用配管 12												2				1		

図 4-3 C 工事床工法工程要素図

	19	20	21	34	37	43	51	52	55	58	59	61	69	71	73	84	99	70	83
	戸境側梁主筋圧着 5	通路側梁主筋圧着 5	バルコニー側梁電気配管 0	戸境側梁電気配管 10	住戸電気配管 10	通路側梁電気配管 10	コンクリート打設 (V) 7・13	足場解体 7	戸境 A P S 脱型 1	通路側 A P S 脱型 1	小梁型枠組立 2	小梁型枠組立 1	住戸床型枠組立	住戸床配筋	住戸床電気配管	住戸床電気配管	小梁電気配管	小梁配筋	コンクリート打設 (H)
69							53	0	0	0	0	10	202						
71	0	0	0										18	145				0	4
73				6	2	10							20	10	150		10	0	
84													0			135			0
99	小梁電気配管 10												0						
70	小梁配筋 3												0						
83	コンクリート打設 (H) 7・9・10・12・13												0	0	0				
77	衛生配管 12												2						

表 4-1 職種コード

コード	職 種
1	大 工
2	解 体 工
3	鉄 筋 工
4	溶 接 工
5	圧 接 工
6	マ 工
7	土 工
8	断 工
9	左 官 工
10	電 工
12	南 生 工
13	ポン プ 工
16	鉄 骨 工

表 4-2 床工法の労務量比較

区分	職種							計
	大 工	解体工	鉄筋工	溶接工	マ 工	電 工	南生工	
B	63	—	4	—	—	28	—	95
	202	135	163	—	—	180	—	680
	—	—	—	—	—	—	2	2
	計	265	135	167	—	208	—	777
C	—	マ	—	20	—	5	—	25
	—	—	254	15	248	94	—	11
	—	—	—	—	—	16	3	619
	計	—	254	35	248	115	3	655

表 4-3 小梁工法の労務量比較

区分	職種							計
	大 工	解体工	鉄筋工	溶接工	マ 工	電 工	南生工	
B	8	—	2	—	—	—	—	10
	138	35	114	—	—	—	27	314
	10	—	4	—	—	—	—	14
	計	156	35	120	—	—	27	338
C	20	—	—	—	—	—	—	20
	16	—	5	—	99	—	—	120
	—	—	12	5	—	—	—	17
	計	36	—	17	5	99	—	157

図 4-4 床工法の労務量比較

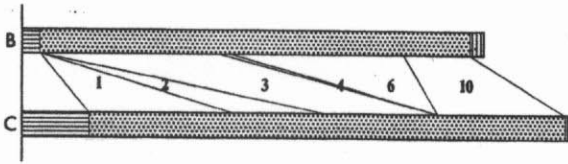
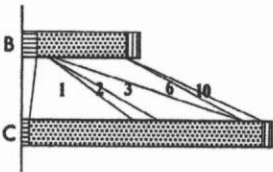


図 4-5 小梁工法の労務量比較



4. 3 サブシステム間のインターフェイスとシステムズ・ビルディング

4. 4 本章の結論

- 【成果】
- *IFM に示された工程要素とその工程順序関係に、「職種」の情報を加えることにより、その職種の担当する工程要素群（＝「もの」と「こと」のまとまり＝サブシステム）に対応するベクトルcを抽出することができる
 - *また、IFM のべき乗から得られる到達可能行列を用いて、サブシステムcに対応する誘引要素I(c)、平行要素L(c)を算出できる
 - *誘引要素の多寡や、誘引要素・先行要素・後続要素と当該サブシステムのインターフェイスの質・量によりサブシステム化の評価が可能であることを示した

【応用事例】

- ・住宅用設備ユニットのジョブ・コーディネーションに関する研究
- ・集合住宅躯体工事の部分構工法の生産性に関する分析

【第4章に関する主要既発表論文・講演】

(略)

第5章 インターフェイス・マトリクスによる工程計算

5. 1 はじめに

＊目的：IFM に示された工程要素にその作業時間の情報を加えることにより，各工程の最早（遅）開始時刻，最早（遅）終了時刻，クリティカル・パス，トータル・フロートなどを IFM に基づいた行列演算によって算出する方法を見出す

＊方法

- ・「こと（＝工程要素）」に対応する IFM の対角要素にその工程要素の作業時間を組み込む
- ・上の形に変形された IFM より，最早日程（終了時刻）行列 EF と最遅日程（終了時刻）行列 LF を求める
- ・最早日程（終了時刻）行列 EF と最遅日程（終了時刻）行列 LF より，クリティカル・パスおよびトータル・フロートを求める

5. 2 最早終了時刻行列 EF

R と各工程要素の作業時間 T_i から最早終了時刻行列（または最早日程行列：以後最早時刻行列を採用）， $\mathbf{EF} = \{E_{ij}\}$ を求める¹。**R** は n 個の工程要素と工程要素間のインターフェイスの有無によって定義された $n \times n$ 型の行列，すなわち，対角要素を 1，非対角要素のうち工程要素 c_i が工程要素 c_j に直接先行する場合を 1（ $r_{ij} = 1$ ），その他の要素を 0 とした行列である。

ある工程要素 c_i の最早開始時刻を ES_i とあらわす²。 c_i をもっとも早く開始できる時刻は，それに直接先行する工程要素群 $\{c_h\}$ の最早終了時刻 EF_h のうちのもっとも遅い時刻である。各工程要素 c_h の最早終了時刻はそれぞれの最早開始時刻に作業時間を加えた値である。また， \mathbf{EF} と \mathbf{ES} の間には次の関係がある。

$$EF_h = ES_h + T_h \tag{5.2.1}$$

c_i の最早開始時刻 ES_i は， EF_h （ $=ES_h + t_h$ ）のうち最大のものとなる。

いま，最早終了時刻行列 $\mathbf{EF} = \{E_{ij}\}$ の要素を次のように定義する。

$$E_{xi} = \left\{ \begin{array}{ll} EF_i = ES_i + t_i = \text{Max} \{EF_h\} + t_i & \cdots (x = i \text{ かつ } \{h\} \neq \phi) \\ EF_h = ES_h + t_h & \cdots (r_{xi} = 1) \\ 0 & \cdots (r_{xi} = 0) \end{array} \right\} \quad (x \neq i) \tag{5.2.2}$$

ただし， c_h は c_i の直接先行要素（すなわち $r_{hi} = 1$ ）

¹ \mathbf{EF} は Earliest Finish Time の略。
² \mathbf{ES} は Earliest Start Time の略。

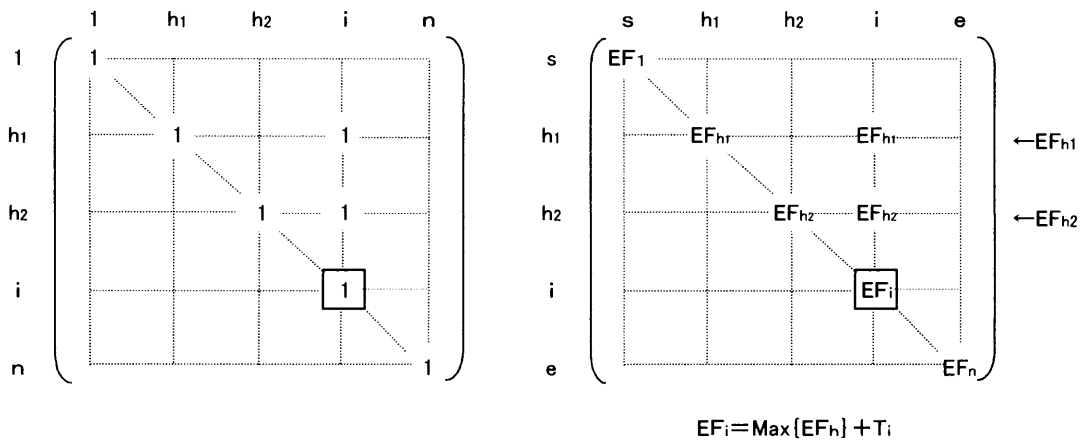


図 5-1

この行列の意味するところは次のとおりである。すなわち、 $EF = \{E_{ij}\}$ はその i 番目の対角要素に c_i の最早終了時刻をあらわした行列である。 c_i の最早終了時刻は、行列の i 列 ($i \neq j$) に含まれる値の最大値に T_i を加えた値である。また、 E_{ij} の非零要素は i について (i 行で) j によらず一定の値、すなわち i の最早終了時刻の値をとる (図 5-1)。この性質により、 EF を最早終了時刻行列と呼ぶ。 c_i の直接先行要素が ϕ の場合、 E_{ii} は T_i となる。

同様に最早開始行列 ES を定義することも可能であり、これは EF の i 行の各成分の値から T_i を引いた値を持つ。さらには、 ES の対角成分を EF_i に変えた行列も考えうる。

IFM, R から EF を求めるのは次の手順による。

- ① トポロジカル・オーダリングを施した R について、直接先行要素を持たない要素の対角要素 ($r_{ss} = 1$) を $T_s (=EF_s)$ に置き換える³。また、各々の直接後続要素 c_i について r_{si} を i によらず (s 列すべて) EF_s に置き換える。
- ② 次に、トポロジカル・オーダーにしたがい、直接先行要素を持つ要素 c_i について、順次つぎの操作を行う。すなわち、 c_i の直接先行要素 (r_{hi} が 1 である h 番目の要素) のすべての $r_{hi} (=EF_h)$ の最大値に T_i を加えた値を EF_i とする。また、 c_i の後続要素 c_j ($r_{ij} = 1, i \neq j$) すべてについて、 r_{ij} を j によらず (i 列すべて) EF_i に置き換える。
- ③ 最後まで同じ操作を繰り返す。 EF_i の最大値がクリティカルパス上の工期 (T) をあらわす⁴。

図 5-2 に EF の例を示す。

図 5-2 EF の例

5. 3 最遅終了時刻行列 LF

同様に、最遅終了時刻行列 LF を定義することができる。いまある工程要素 c_i の最遅開始時

³ ここでも、IFM によるオーダリング、工程計算は始点、終点が複数あってもよいとしていることに注意。

⁴ 終点がひとつであれば EF_n が T となる。ここでは始点、終点が複数存在してもよい。

$$LS_i = LF_i - T_i \quad (5.3.1)$$
$$LF_i = \text{Min} \{LS_k\} \quad (5.3.2)$$

$$LS_k = LF_k - T_k \quad (5.3.3)$$

$$L_{i_y} = \begin{cases} LF_i = \text{Min} \{LS_k\} = \text{Min} \{LF_k - T_k\} & \cdots (y = i \text{ かつ } \{k\} \neq \phi) \\ LF_k = LS_k + T_k & \cdots (r_{i_y} = 1) \\ 0 & \cdots (r_{i_y} = 0) \end{cases} \quad (y \neq i) \quad (5.3.4)$$

Figure 5-2 consists of two grid diagrams illustrating the calculation of LF values. The left grid shows the initial state with '1' values on the diagonal and the first row. The right grid shows the updated state with LF values calculated for the first row and column. Arrows indicate the direction of calculation from the first row and column into the rest of the grid.

Left Grid (Initial State):

	1	i	k1	k2	n
1	1				
i		1			
k1			1		
k2				1	
n					1

Right Grid (Updated State):

	1	i	k1	k2	n
1	LF ₁				
i		LF _i	LF _{k1}	LF _{k2}	
k1			LF _{k1}		
k2				LF _{k2}	
n					LF _n

Below the right grid, arrows point to the columns k1 and k2, labeled LF_k. The formula LF_i = Min {LF_k - T_k} is shown.

⁵ LF は Latest Finish Time の略。
⁶ LS は Latest Start Time の略。

c_i の直接後続要素が ϕ の場合、 L_{ij} は T となる。

R から LF を求める手順は次のとおりである。

- ① トポロジカル・オーダリングを施した R において直接後続要素を持たない要素の集合 ($\{c_e\}$) の r_{ee} を $LF_e (=T)$ と置き換える。また $\{c_e\}$ の直接先行要素との $IF (= \{r_{ie} \mid r_{ie} = 1\})$ をすべて T と置き換える。
- ② 次に、直接後続要素を持つ要素 c_i についてトポロジカル・オーダーの逆行順に LF_i を求める。 LF_i は c_i の直接後続要素 c_k の LF_k から T_k を引いた値の最小値である。また、 $r_{hi} (=1 : c_h \text{ は } c_i \text{ の直接先行要素})$ をすべて LF_i に置換する。
- ③ トポロジカル・オーダーの先頭にいたるまで、同じ操作を繰り返す。

最遅開始時刻行列 LS は $\{L_{ij}\}$ の第 j 列成分 ($=LF_j$) から T_j を引いた値であらわされた成分を持つ行列である。

R の転置行列 tR は c_i の直接後続要素を求める関数であるから⁷、これにより最遅終了時刻行列 LF' を導くこともできる。この場合、 ${}^tR ({}^t\{r_{ij}\})$ に対応する最遅終了時刻行列 $LF' = \{L'_{ij}\}$ の要素は次のように定義される。

$$L'_{yi} = \left\{ \begin{array}{ll} LF_i = \text{Min} \{LS_k\} = \text{Min} \{LF_k - T_k\} & \cdots (y = i \text{ かつ } \{k\} \neq \phi) \\ LF_k = LS_k + T_k & \cdots (r_{yi} = 1) \\ 0 & \cdots (r_{yi} = 0) \end{array} \right\} (y \neq i) \quad (5.3.5)$$

ただし、 c_k は c_i の直接後続要素 (すなわち $r_{ik} = 1$)
 $y = i$ かつ $\{k\} = \phi$ の場合、 $L'_{ii} = T$

すなわち、 $L'_{ij} = L_{ji}$ が成り立つから、

$$LF' = {}^tLF \quad (5.3.6)$$

となる (図 5-3)。 R から LF' を導く手続きは上記①～③と同様である。

⁷ 式(3.2.7)参照。

クリティカルパスは、 TF_i が 0 である要素 c_i を IF によって結んだパスである。このとき (5.4.1), (5.4.3) より次式が成り立つ。

$$LS_i = ES_i \quad (5.4.6)$$

$$LF_i = EF_i \quad (5.4.7)$$

5. 6 本章の結論

【成果】

*IFM に各工程要素の「作業時間」の情報を加えることによって、最早日程行列、最遅日程行列を簡単に計算できることを示した

*また、再送日程行列、最遅日程行列から、トータル・フロート、フリー・フロート、クリティカル・パス等を産出する方法を示した

【第 5 章に関する主要既発表論文・講演】

(略)

第6章 工区分割問題とインターフェイス・マトリクス

6. 1 はじめに

繰り返し型の工程を持つ工事を多工区分割し、かつ同期化を図る方法については、ラインバランシングやマルチ・アクティビティ・チャートなど、これまでもいくつかの考え方が提案されてきている。しかしこれらを「手法」としてみると、特定の解にいたる過程を必要十分に操作し再現できるようになっているとはいいがたく、したがってそれだけ応用の範囲を狭めているといつてよいと思われる。

本章ではこの点に注目し、単工区モデルの基準工程をあらわす IFM という一定不変の「関係構造」を出発点として、工区分割問題を IFM の部分行列化の問題として扱えるようにする手法を新たに提案するものである。工程を分割し特定の工区に割り付けることを厳密に数理的な演算に置き換えることにより、一連の過程から偶然になされる選択決定を排除しておこうということである。その意味ではこの手法は最適解を自動的にもたらすものではなく、むしろそのような解を得るための支援手法であるといえる。

実際、後に述べるように、この手法は「完全同期化」、すなわち各職種（作業チーム）が手待ちや作業時間の重複なくフル稼働しているという条件を前提としているため、「最適解」はひとつではなくいくつも得られる。

本章では最初にこの「完全同期化」という概念を説明し、また工区分割が IFM の部分行列化とどのように対応しているのかを示す。次いで、「サイト工区」と「ジョブ工区」という概念を導入することにより、作業空間を分割することと作業プロセスを分割することの違いを明らかにするとともに、両者を関係付ける。そして最後に、任意の工区分割数を持つ工程計画の作成法を示す。

6. 2 多工区分割と工程同期化

6. 2. 1 工程の完全同期化

工区分割の目的は、本来順序関係が存在するために同時に作業することができない工程を、作業空間を分けることにより同時作業が可能にようにすることにある。

異なる職種の担当作業間にインターフェイスが存在するとき（作業の順序関係が存在するとき）、単工区モデル（基準階工程）では同一作業空間内の作業となるため、ある作業が進行している間に他の作業が休止状態となるため、全体的な稼働率のレベルは低い。しかし、これをいくつかの工区に分割することができれば、各職種は平行して作業を行うことができるようになって稼働率は上り、作業者の数は同じでも工期は短縮される。

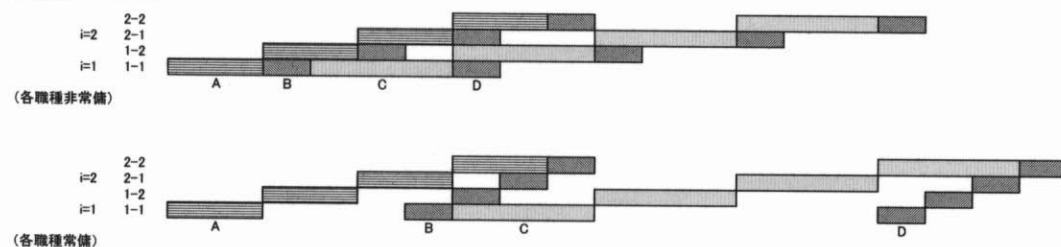
●「各職種常備」と「各職種非常備」のストラテジーの違い

いま、基準階工程1サイクルを2回繰り返す（2階建て建築物の）を例に取り上げる。基準階工程（のクリティカルな作業）は、図 6-1（a）の $n=1$ に示されるように直列に連結しているものとする。基準階全体を1工区とした単工区モデルの場合（a）、各職種の作業時間の不揃いにより、サイクルを繰り返すうちに手待ちが生ずる。2階（2サイクル）分の工程では、先行工程の職種（A、C）より作業時間が相対的に短いことにより、職種B、Dに手待ちが生じ、C以外はクリティカルな工程を外れている。

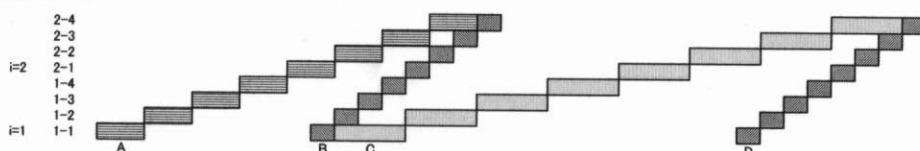
(a)水平1工区分割



(b)水平2工区分割



(c)水平4工区分割



(d)水平8工区分割

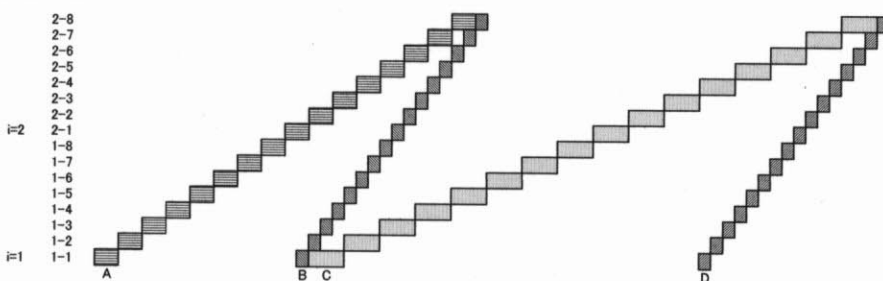


図 6-1

このようにクリティカルでない作業を含む工程を重ね合わせ、異職種間の同時作業を許すには、2つのストラテジーがありうる。このことを水平2工区分割で例示する。図 6-1 (b) の上段は各職種の作業を最初のサイクルの最早開始時刻で始めるという戦略であるが、この場合全体工期は短くなるが、直前工程よりも作業時間の短い職種には常に手待ちが生ずる。その意味でこの戦略を「各職種非常備」の戦略と呼ぶことができる。一方 (b) の下段は、各職種の作業を最終サイクルを基準とした最遅開始時刻から始めるとした場合の工程である。こちらは、全体工期は延びるものの、各職種が連続的に作業を行うことができるため、「各職種常備」の戦略をとった場合の工程といえることができる。いずれの場合も、各階 (工区) のサイクル工期は等しくなることに注意する必要がある。

作業量、作業人数の多い職種に関しては、常備的に取り扱うのがふつうであるから、図 6-1 の (c)、(d) には各職種常備とした場合の水平4工区分割、同8工区分割の例を示した。(水平) 工区分割数が多いほど全体工期は短くなる。

●投入資源水準一定下での「同期化」

各職種の手待ちをなくし、全工程をクリティカルな工程とする場合、一般にタクト工程と呼ばれている工程，すなわち各職種の作業時間を等しくする方法が有効である。このようなケースを，「同期化」された状態ということにする。

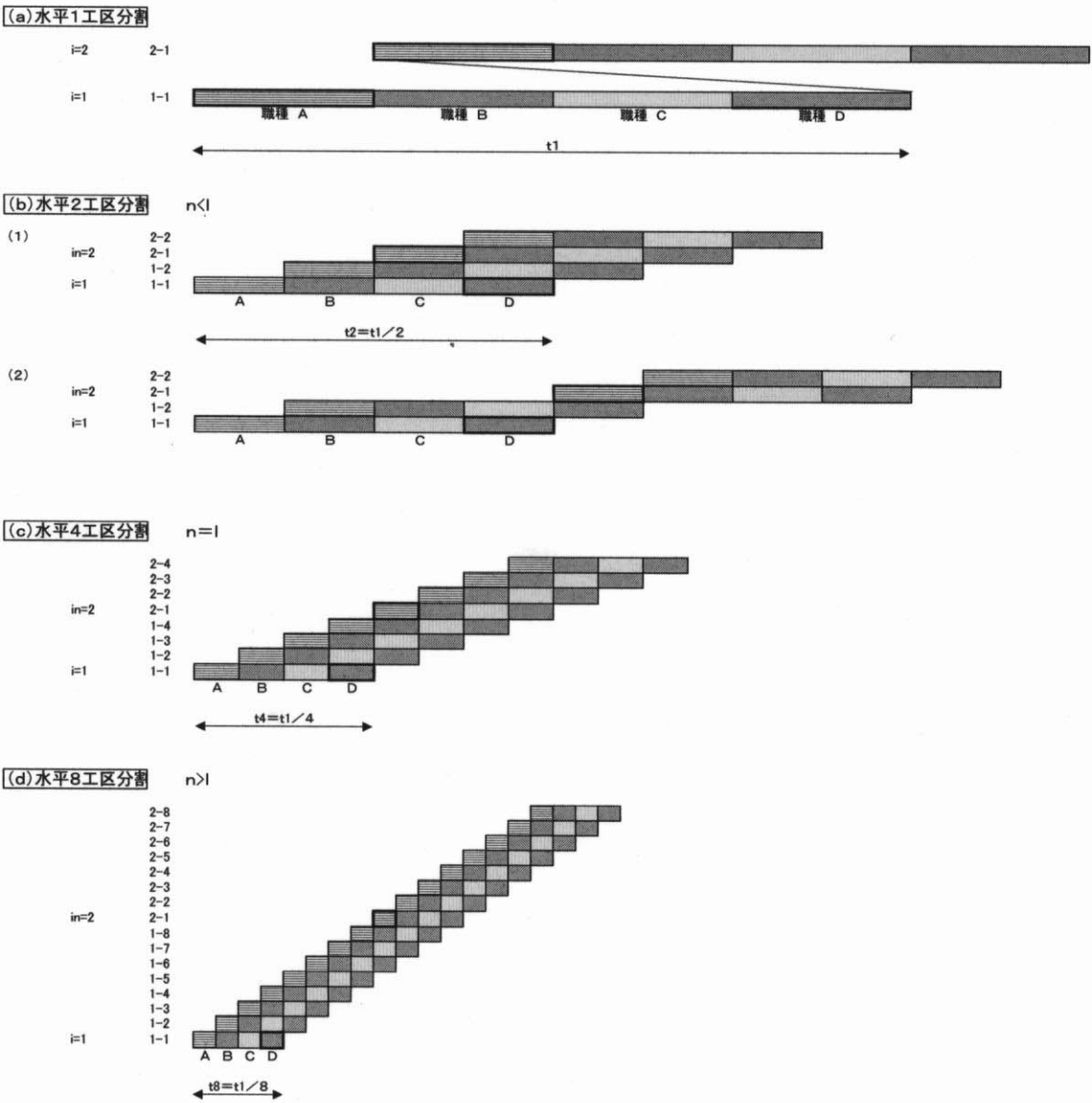


図 6-2

基準階単工区工程を図 6-2 (a) に示す。職種数 (工程分割数) を ℓ , 基準階を n 工区分割した場合のサイクル工期を t_n とし, このときのタイムモジュール, すなわち繰返し作業の 1 単位分の時間を M_n とすると, 両者の間には次の関係がある。

$$t_n = M_n \times \ell \quad (6-1)$$

$$M_n = M_1 / n \quad (6-2)$$

$$t_n = t_1 / n \quad (6-3)$$

上記の関係は投入資源水準 (各職種の作業者の数など) が一定であることを前提としている。各投入資源を単純に倍にすればサイクル工期 t_1 は半分になるというように, t_1 が (したがって $M_1 = t_1 / \ell$ も) 一定しないからである。しかし, 投入資源一定としても, 実際には次のような制限がある。すなわち, 工区分割数が増えるとそれに反比例して 1 工区あたりの作業空間は小さくなる。一定の大きさの作業空間で同時に作業ができる人数には限りがあるから, (その状態を資源密度の上限と呼ぶことにすれば) 実は工区分割数にも資源密度に応じた上限が存在する。以下は, 「資源密度が上限にあり」かつ「その条件下で投入資源水準が一定であり」またかつ「工程分割数 (ℓ) が一定である」ことを前提とした工区分割数の検討である¹。

基準階工程は上階でも繰り返されるが, 一般に, 下階での自職種の作業の終了後ただちに職種 A が上階での作業にかかれるわけではなく, 最も遅い場合には下階での最終工程 D が終了した後に作業を開始しなければならないので, 単工区の基準階工程 2 階分の全体工期は, $t_1 + M_1$ と $t_1 \times 2$ の間の値をとる。水平 2 工区分割の場合, 上階第 1 工区 (2-1) の工程 A が直下階の工程 C 以降の工程要素にインターフェイスを持たなければ, A は下階第 1 工区 (1-1) の作業が終了し次第開始できるので, 各職種は手待ちなく, 順次, 次工区の作業にかかることができる² (図 6-2 (b) (1))。しかし下階の最終工程が終了しない限り上階の最初の工程が始められないとすると, (2-1) の A は (1-1) の D の終了後に開始されることになり, 各職種には手待ちが生ずる (図 6-2 (b) (2))。

●投入資源水準一定下での「完全同期化」

しかしながら, 水平工区分割数 n が工程分割数 (ℓ) と等しいかそれ以上の場合には (図 6-2 (c), (d)), 上階の最初の工程の開始以前に下階の最終工程が終了していることが保証されているため, 各職種の作業に手待ちが生ずることはない。そこで, このような同期化状態を「完

¹ 資源密度最大限, 資源投入水準一定, 工程分割数一定がワンセットの条件であることは次のように証明される。

いま, 基準階単工区での職種 k の作業の施工量を L_{k1} とする。このとき資源水準 (～資源密度) R_{k1} が上限 (R_{kmax}) に達していれば ($R_{k1} = R_{kmax}$), 施工歩掛りを E_k (施工量/単位資源量・時間) とし, $t_{k1} = L_{k1} / (E_k \times R_{kmax}) = M_1 = t_1 / \ell$ 。すなわち, $R_{kmax} = (L_{k1} \times \ell) / (E_k \times t_1)$ 。

基準階を水平 n 工区分割した場合の職種 k の施工量は L_{k1} / n であるから, $t_{kn} = L_{kn} / (E_k \times R_{kn}) = (L_{k1} / n) / (E_k \times R_{kn}) = M_n = M_1 / n$ 。これより, $R_{kn} = (L_{k1} / n) / (E_k \times (M_1 / n)) = L_{k1} / (E_k \times M_1) = (L_{k1} \times \ell) / (E_k \times t_1) = R_{kmax}$ 。すなわち, 基準階単工区で資源密度が上限にあれば, 投入資源水準を一定とする限りそれをいくつの工区に分割しようとも資源密度は上限にあり, かつそのときの工程分割数も一意的に決まる ($R_{kn} = R_{kmax} = (L_{k1} \times \ell) / (E_k \times t_1)$)。

² 厳密に言えば, 上階の作業が開始できるのは, 上階での先頭の工程要素 (群) がインターフェイスを持つ直下階 (またはさらに下階の同一サイト工区) の工程要素群のうち, もっとも遅く終了する工程が終わった後である。

全同期化」と呼ぶことにする。完全同期化状態下（ $n \geq \ell$ ）では、水平 n 工区分割時の全体工期を T_n 、階数を m として、次の関係が成り立つ。

$$T_n = M_n \times n \times m + M_n (\ell - 1)$$

式(6-1)、(6-2)、(6-3)より、

$$T_n = M_1 \times m + M_1 (\ell - 1) / n \quad (6-4)$$

$$= t_1 \times m / \ell + t_1 \times (\ell - 1) / (n \times \ell) \quad (6-5)$$

式(6-4)、(6-5)にみるごとく、完全同期化状態下（ $n \geq \ell$ ）では、水平工区分割数 n が大きくなるほど全体工期 T_n は短くなる。最初の工程の最終工区での終了日程（時刻）に相当する両式の第1項は工区分割数によらず一定であり、第2項が最終工区での残余の職種の作業に要する工期をあらわす。工区分割数 n に応じて短縮されるのはこの第2項である³。

●投入資源水準を増やす場合（＝工程要素数（職種数）を増やす場合）

ここで、「 $n > \ell$ （図6-2（d））である」という条件が持つ現実的な意味を考えてみたい。 $n > \ell$ という条件だけに着目すると、水平工区分割数 n に対して同時作業にかかる職種数（工程要素数）はそれより少ない ℓ であるから、 ℓ を大きくとればさらに同時作業を増やすことができ、その結果工期短縮をはかる可能性がありそうである。あるいは、各職種の投入資源水準を一律に上げて、サイクル工期を短縮することも考えられる。しかし、この一連の検討では、 ℓ 分割された工程を担当する職種の資源密度が上限に達しているということを前提としているため、後者の投入資源水準増加という方策はとりようがない。これ以上1工区あたりの投入資源を増加できないという意味で、 n はすでに上限にはりついているのである。よって、これ以上の工期短縮は工程分割数（ ℓ ）を（ ℓ' ）に増やし、同時作業を促進するほかないということになる。

分割された各職種（作業チーム）の工程はいくつかの工程要素群からなる。完全同期化条件下での工程分割の増加は、工程分割数変更後の各作業チームの作業時間をそろえながら、担当工程要素群の割付けを変えることを意味する。投入資源水準が不変で各工程要素の作業能率もチームによらず一定であるならば、当然、各作業チームの作業時間は工程分割数の増加に反比例して減少する。しかし、ここでの検討は、投入資源水準が一様に上限に張り付いていることを条件としているから、この作業時間の減少を埋めるだけ（つまり工程分割数の増加に比例して）各作業チームの投入資源水準を増加させることが必要となる⁴。すなわち、工程分割の増加は、同時に投入資源水準の増加を意味する。工程分割数を（ ℓ ）から（ ℓ' ）に変更することにより、(6-1)、(6-2)は次のように書き換えられる。

$$t'_n = M'_n \times \ell' \quad (6-1')$$

³ これには、あくまで「投入資源水準（作業者の数など）が同一である限り」という条件がつく。

⁴ 実際には投入資源量は離散的な自然数の値をとるため、必ずしも連続量として取り扱えるわけではない。

$$M'_n = M'_1 / n \quad (6-2')$$

このとき(6-3)式, $t_n = t_1 / n$ は, ℓ , ℓ' によらず成り立つから,

$$t'_n = t_n = t_1 / n \quad (6-3')$$

したがって⁵,

$$M'_n \times \ell' = M_n \times \ell \quad (6-6)$$

また, 式 (6-2'), (6-6)より,

$$M'_1 = M_1 \times \ell / \ell' = (t_1 / \ell) \times \ell / \ell' = t_1 / \ell' \quad (6-7)$$

が成り立つ。このとき, 新しい工程分割 (ℓ') に対応する全体工期 T'_n は,

$$T'_n = M'_1 \times m + M'_1 (\ell' - 1) / n \quad (6-4')$$

$$= t_1 \times m / \ell' + t_1 \times (\ell' - 1) / (n \times \ell') \quad (6-5')$$

であるから, (6-4')式に(6-6)式を代入するとともに, (6-5')式を変形して次を得る。

$$T'_n = M_1 \times m \times \ell / \ell' + M_1 \times \ell \times (\ell' - 1) / (n \times \ell') \quad (6-4'')$$

$$= (t_1 / \ell') (m + (\ell' - 1) / n) \quad (6-5'')$$

式(6-5'')より, 工程分割数を増やした場合の工期短縮効果は次のとおりである。

$$T'_n - T_n = m t_1 (1 / \ell - 1 / \ell') + (t_1 / n) ((\ell - 1) / \ell - (\ell' - 1) / \ell') \quad (6-8)$$

$$= m t_1 (\ell' - \ell) / \ell \ell' - (t_1 / n) (\ell' - \ell) / \ell \ell'$$

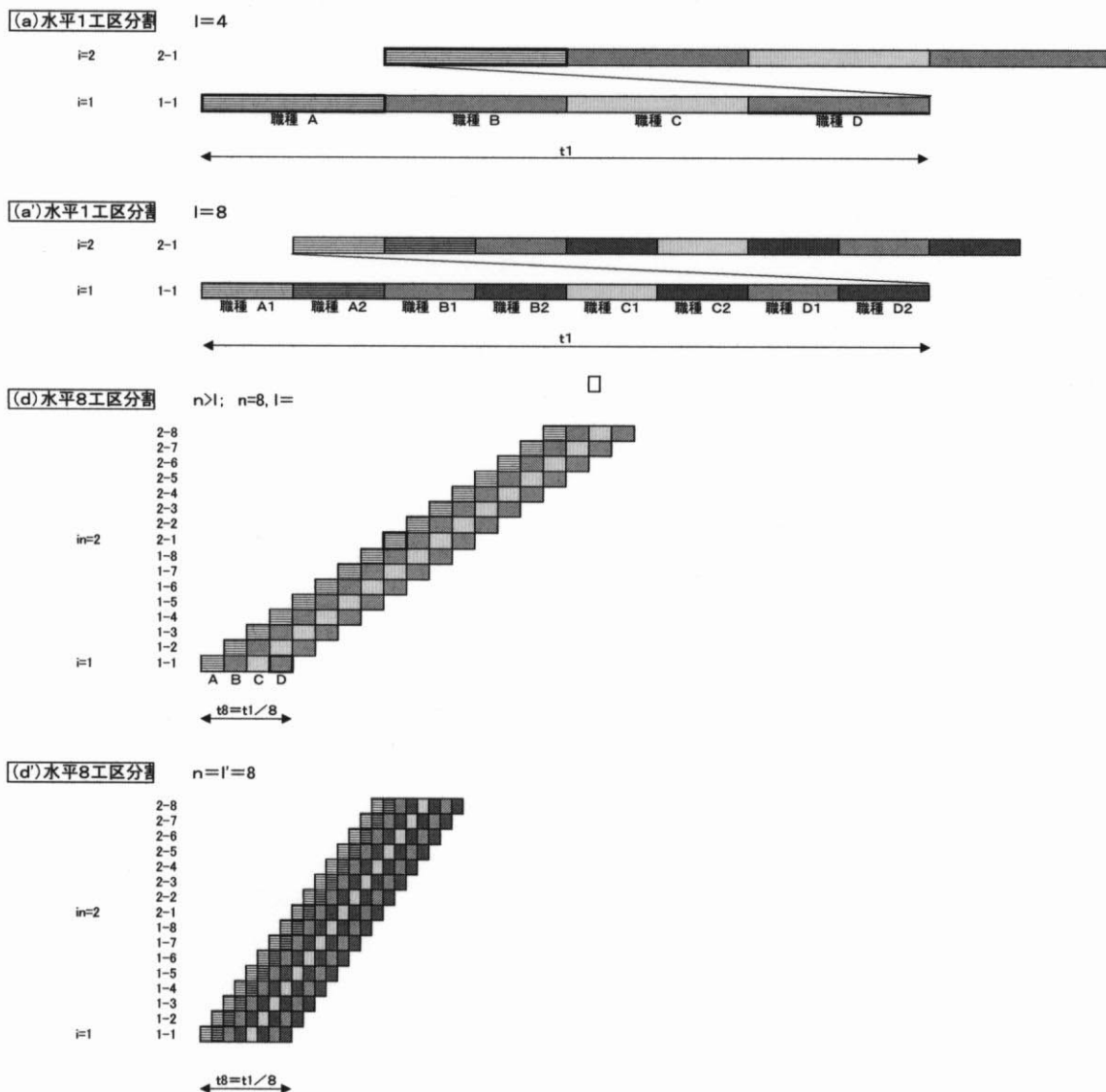
$$= t_1 (\ell' - \ell) / \ell \ell' (m - 1 / n) \quad (6-8)$$

n , m , t_1 は一定であるから, 工程分割数 ℓ^6 を増やし, その分投入資源を増やせば工期短縮効果があるということになる。

図 6-3 に, $n = 8$, $\ell = 4$, $\ell' = 8$ の例を示した。この場合, 4 工程分割のときの各工程がそれぞれ 2 分割されているが, 各工区の投入資源水準は不変であり, 全体の投入資源水準は最大で 2 倍となっている (8 サイクル~16 サイクル)。また, $\ell = 4$, $\ell' = 8$ のいずれの場合も, サイクル工期 $t_8 (= t_1 / 8)$ は一定であるが, 全体工期は, $t_1 = 32$ としたとき, 19 か

⁵ 脚注 4 と同じ理由で, 常に(6-4)式が成り立つわけではない。

⁶ 6.3 節に述べるように, 工程分割数はジョブ工区分割数とほぼ同義である。



ら 11.5 に減少している。

図 6-3

後に示すように、上下階にまたがる2以上のジョブ工区を一つのジョブ工区とみなすことが可能であるから、水平工区分割数 n_a を ($n_a < n$) 増やすことなく、実質的に l ($= n$) を増やし、より多くの資源を投入することが可能である。

●多工区分割同期化構工法計画と工程分割問題

多工区同期化構工法計画においては、MAC (Multi-Activity Chart)、もしくはDOC (One-Day One-Cycle) 工程表などと呼ばれる工程を作成する。これは、次の点においてこれまで扱ってきたタクト工程と異なっている。

①多工区同期化構工法計画においては、ジョブ工区という呼称で1タイムモジュールに相当す

る工程分割を行うが、1 ジョブ工区に含まれる工程要素群は同一職種（作業チーム）に属するものとは限らず、複数作業チームの分担する工程要素が混在している。

②逆に、ひとつの作業チームが担当する工程要素は一つ以上のジョブ工区（＝工程）にまたがって存在しているため、各チームはジョブ工区（したがってサイト工区）間を移動しながら、1 タイムモジュールの作業を完了する。

③しかし、各作業チームの1 タイムモジュールの作業時間はフル稼働を前提として計画されるため、完全同期化条件は基本的に満足されている。この点は、タクト工程の場合と同一である。

④完全同期化条件は、ほかならぬ工程分割＝ジョブ工区分割によって保証されている。タクト工程では同一作業チームの担当工程要素群を一連の大きな単一の工程にまとめ、工区と工程、職種間に1 対1 の対応をつけるのに対し、多工区同期化ではジョブ工区の異なる工程要素間の順序関係を取り除くことにより、工程要素群を一連の工程としてまとめる必要性を無くし、完全同期化条件の導入をはかるのである。

次項において、諸概念の定義と関係を整理する。

6. 2. 2 工程分割とジョブ工区分割

工区分割の基本となるのは、繰り返し施工されるすべての工程要素の（各一つの）組とそれらの間のインターフェイスで定義された単工区の基準階 IFM である。これを異なる数の工区に分割することは、IFM を部分行列に分割することであるが、このとき異なる工区間の IF は基本的に消滅し、代わりに分割および分割後の条件によって工区間に新たに IF が付加される。分割された工区は単に施工プロセスにおける工程要素の分節を示すものであり、空間的な分割単位との対応付けはこの段階ではなされていない。この両者の違いを考慮して、本論文では前者を「ジョブ工区」、後者を「サイト工区」と呼ぶ。すると、多工区分割問題とは、基準階工程をジョブ工区工程に分節しそれらをサイト工区に割り付けること、と言い換えられる。ジョブ工区分割数とサイト工区分割数は見かけ上同じとして理解するのが簡単であるが、後に精密に議論するように、現実には一致しない。

●ジョブ工区分割と工程分割のための準備と方針

ジョブ工区分割とは、基準階工程をそれぞれ連続するいくつかの部分工程に分割することである。ジョブ工区分割は比較的長期間継続する工程でかつクリティカルな工程要素（例えばクレーンを比較的長期間用いる作業）を、異なるジョブ工区の主要作業として割り付けることによって行われる。

完全同期化条件を満たすジョブ工区分割を多様におこなうことができるようにするためには、準備段階での整理、ないしは方針の確認が必要である。以下にその概要を記す。

①前項に示したように、多工区分割同期化による工期短縮、生産性向上を目指すのであれば、サイト工区分割数（ n ）をある程度大きくし、またそれにもなつて工程分割数（ l ）も多くできるように準備しておく必要がある。

②そのためには、工程分割の境界となりうる工程要素をできる限り細かい単位として用意しておく必要がある。これは、工区分割・工程分割のための境界を随所に用意するためばかりではなく、完全同期化を保つ（すなわち各作業チームの作業量を手待ちなしに等水準にそろえる）ためにも重要である。

③また、各工程要素について、構工法計画に関わる諸元を明らかにしておくことが必要である。特に重要なのは、技能域による担当職種、基準階あたりの施工量、施工能率（歩掛り）、必要な仮設資機材、異職種間での組作業の必要の有無である。

④一般に同一職種の作業とみなされる工程要素であっても、工程要素の独立性、動き方の特性によっては別の作業チームの作業として用意したほうがよいこともある。逆に、特に特殊な技能、資機材を要しない工程要素に関しては、ある程度、担当職種の割り当てを自由に変更することによって労務資源量・作業量の平準化をはかれるようにしておくことが望ましい場合がある。

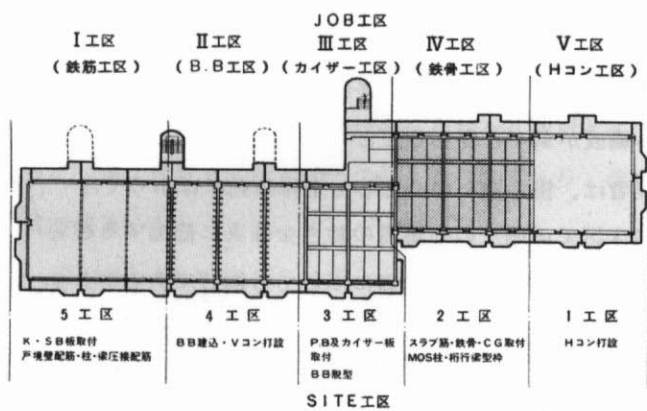
⑤高層オフィスビルや複合化構工法による RC・SRC 造集合住宅等、繰り返し型の構工法には典型的なシステムがある。これらについては工程要素、および工程要素間のインターフェイスをその都度洗い出すのではなく、データベースとして蓄積されうる。過去のデータから構工法を組み立てる時には、躯体数量等施工量のみをパラメトリックに扱えばよい場合も多い。新たに部分構工法を案出する必要がある場合には、その都度、構工法を定義し、データベースに加えることになる。

●多工区同期化工程におけるジョブ工区分割と工程分割の関係

前項で、タクト工程と多工区分割同期化工程の違いに触れた。しかし、完全同期化という前提に関しては両者は同じであり、サイクル工期やタイムモジュールなどの概念はほぼそのまま準用してよい。ここではそのことについて、やや詳細に述べる。

サイト工区とは作業空間・建築空間の幾何学的分節をあらわす概念であり、施工順に番号を付けるとすれば一般には水平工区分割番号と階数を複合した方法により、建物下層より順次空間座標系内に絶対的な工区番号を付けることができる。これに対してジョブ工区とは、特定サイト工区で特定タイムモジュール内が特定工程を施工中であるという「状態」をあらわす概念であり、「サイト工区の状態」という側面と「分割された基準階工程」という側面の 2 面をもつ。ジョブ工区にも工程の進行順に番号を付けるとすれば、ある時点でのサイト工区の状態をあらわすジョブ工区は、先行サイト工区（番号の大きい工区）から後続サイト工区（番号の小さい工区）に向かって並んでいる（図 6-4）。引き続き述べるように、互いに隣接する一連のサイト工区に隙間をあげずにジョブ工区を割り付ける必要は必ずしもない。

いま簡単のために、水平 n サイト工区分割、n ジョブ工区分割の多工区分割同期化工程を考える。このとき、タイムモジュールを M として、（基準階および 1 サイト工区の）サイクル工期は nM となる。



基準階工程には工程要素 { $c_{kj}(i)$ } が含まれるものとする。それぞれは特定の作業チーム (k) により、所定のジョブ工区 (j) において作業時間 $t(c_{kj}(i))$ で施工される。このとき、各作業チームの作業時間の合計は M に等しい⁷。これが多工区同期化工程における完全同期化条件である。すなわち、すべての k について、次が成り立つ。

$$\sum_j \sum_i t(c_{kj}(i))=M$$

ある作業チームの各ジョブ工区における作業時間は等しくなく、まったく作業のないジョブ工区もある。

基準階工程からタクト工程、多工区同期化工程を導く方法を示したのが図 6-5 である。この基準階工程は、それぞれ 3 つの工程要素からなる 3 つの職種 (作業チーム) の作業から構成されている。これをそれぞれ 3 つのタクト工程、多工区同期化工程に分割する方法を示したのが、図 6-3 (a) である。タクト工程の場合は同一職種の工程要素を大きくまとめるようにする。一方、多工区同期化工程の場合には、それにこだわらず、単に工程を 3 分割すればよい。タクト工程、多工区同期化工程のいずれの場合も、各チームの工程要素に要する作業時間の合計がそろっている ($=M$) が必要であるが (完全同期化の条件により) 、多工区同期化の場合に各ジョブ工区の工期がそろっている必要はない。タクト工程の工区 ($=$ 工程、多工区同期化工程のジョブ工区に相当) を作業チームの名にあわせて A, B, C 工区と呼ぶ。また、多工区同期化工程のジョブ工区を 1, 2, 3 工区とする。

タクト工程をつくる場合、同一作業チームに属する工程要素の誘引要素が無ければ工程分割は直ちに完成するが、A 工区に誘引された工程要素 (B1) があるように、一般的には基準階工程を作業チームの作業のまとまりとして直列に切り分けることはできない。そこで (c) に示すように、B 職種の技能域に属する別の作業チーム (B' 1) を A 工区に加え、タクト A を編成することになる⁸。

多工区同期化工程をジョブ工区ごとに指名したのが (b) の上段である。このとき、一つのジョブ工区には複数チームの工程要素が含まれる。各ジョブ工区はそれぞれ独立した作業空間である別個のサイト工区に対応する。したがって、ジョブ工区をまたがる工程関係 (インターフェイス) は消滅し、同一ジョブ工区内の工程関係のみが残る。一方、同一作業チームの作業は各ジョブ工区 (サイト工区) で重なってはいならないから、その順序関係を調整する必要がある。その結果を指名したのが (b) の下段である。これをみると、各ジョブ工区の工期は M と一致しないが、各作業チームの全工区での作業時間の合計は常に M であること、また同一作業チームの 1 タイムモジュール内での工程は複数ジョブ工区 ($=$ サイト工区) にまたがり、しかも後続サイト工区の工程要素が先行サイト工区の工程より先行して施工されることがあることがわかる。

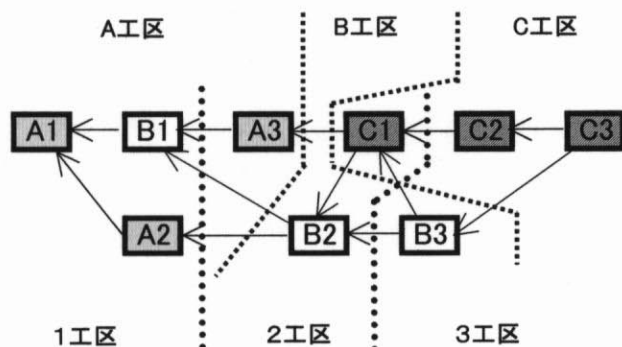
⁷ タクト工程の場合には、ひとつのサイト工区で完了するひとつの作業チームの作業時間が M となる。
⁸ この場合、B 1 が除かれた B 工区、B 1 が加えられた A 工区の工期を調整することが必要である。

(a) 基準階工程のタクト工程, 多工区同期化工程への分割

タクト工程

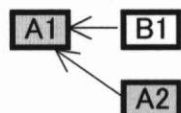
基準階工程

多工区同期化工程

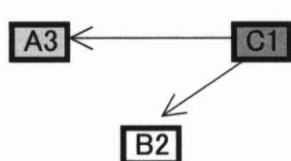


(b) 多工区同期化工程のジョブ工区

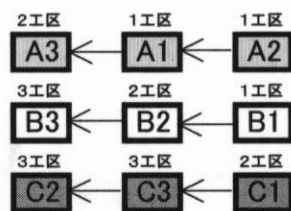
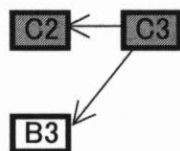
1工区



2工区



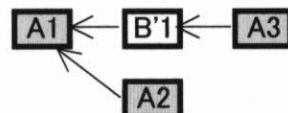
3工区



各作業チームの工程

(c) タクト工程の工区

A工区



B工区



C工区



図 6-5

以上をまとめると次のようになる。

①タクト工程，多工区同期化工程のいずれも各作業チームの作業時間をそろえて完全同期化条件を満たそうとすると同じである。

②タクト工程の同一作業チームの作業は同一ジョブ工区(=同一サイト工区)の作業である。しかし，多工区同期化工程ではそれが複数のジョブ工区(=サイト工区)に分割されており，この別個のサイト工区での作業という事実により，同一作業チームに属する工程要素の工程順序関係が抹消され，完全同期化が保証されることになる⁹。要するに，タクト工程と多工区同期化工程の違いは，工程の分割を同一サイクル内で完結させるか否かという考え方の違いにあるといえる。

③建築工事のように数多くの職種の作業が複雑に錯綜する場合にはタクト工程をつくることのほうが困難であり，作業チームのサイト工区間移動が生じて多工区同期化工程を用意することのほうが現実的である。

すると，問題は，いかなる基準階工程においても完全同期化の必要条件（すなわち各作業チームの作業時間が等しいこと）を満たしていれば，いつも多工区同期化工程を作成することが可能かどうかということになる。

●ジョブ工区分割数の決定

ジョブ工区とは作業空間(サイト工区)を分けることを前提に分割された基準階工程である。いま，すでに完全同期化の必要条件，すなわち，各作業チームの作業時間がすでにあるタイムモジュールにそろえられていることが満たされていることを前提としよう。すると，ジョブ工区のサイト工区への割り付けには次の二つの問題が存在する。

(1) ジョブ工区分割をどのように行うか。

(2) 分割されたジョブ工区をサイト工区にどのように割り付けるか。

第一の問題は単純ではない。その理由は，基本的に解答がいくつも存在するというところにある。例えば，基準階工程が m 個の工程要素から構成されているとしよう。ここで m 個のジョブ工区分割を行うことにすると，各ジョブ工区はそれぞれ一つの工程要素からなることになり，しかもそれゆえそれらは自由先行であり，自由後続である。したがって， m 以上のサイト工区を用意すれば，これらを各サイト工区に最大1つまで割り付けることがいつも可能であり，解が複数存在することがわかる。実際には，サイト工区数を必要以上に多くすることは不合理であるため，工程要素をまとめてジョブ工区の数減らすことが望ましい。一方，多くの作業を同時に行えるようにするには，ジョブ工区数，サイト工区数をそれぞれある程度多くすることが必要であるから，結局，ジョブ工区分割数とサイト工区数をほぼ同じになるように決めることが現実的となる。

ジョブ工区(サイト工区)分割の数は，次のように決めるのがよい。すなわち，多工区分割同期化構工法は与えられた資源水準や工期に応じて最大限の生産効率を求めるために行うのであるから，ここでは完全同期化という必要条件が満たされていることを前提として，

1) 所与の資源水準に対して最短工期を設定するか，あるいは

⁹ 図6-3のタクト工程の場合でも，A工区の(A1, A2)と(B1, A3)を分けて4つのジョブ工区に分割すれば，同一チームの作業に異なるサイクルの作業が混ざるが，見かけ上のタクト工程が完成することが理解されよう。

2) 所与の工期に対して最小資源水準を決めればよいということになる。

サイト工区数とジョブ工区数が等しい場合、サイクル工期 (t) とタイムモジュール (M)、およびジョブ工区数 (ℓ = サイト工区数 = n) との間には、

$$t = M \times n \quad (6-1)$$

という関係が成り立っている。

1) 所与の資源水準から最短工期を求める場合には、もっともタイトな資機材・作業チームの作業時間をタイムモジュールとすればよい。この場合、あらかじめ設定した構工法に応じて基準階施工数量を与え、施工能率と各職種（作業チーム）の調達可能資源水準から作業時間を求める。この値が M であり、そのときのサイクル工期は $M \times n$ である。サイクル工期の望ましい値より工区分割数 n を決め、このジョブ工区分割数に合うように多工区同期化工程の分割を検討する手順となる。

2) 逆に、工期が所与の条件として与えられているときは、サイクル工期 $M \times n$ を先に決め、これを満たす範囲で M と n ($=\ell$) を選ぶことになる。

●ジョブ工区のサイト工区への割り付け

以上、ジョブ工区分割数とサイト工区分割数が等しいものとし、一つのジョブ工区が一つのサイト工区に対応するものとして考察を進めてきたが、実際にはこのように単純ではない。

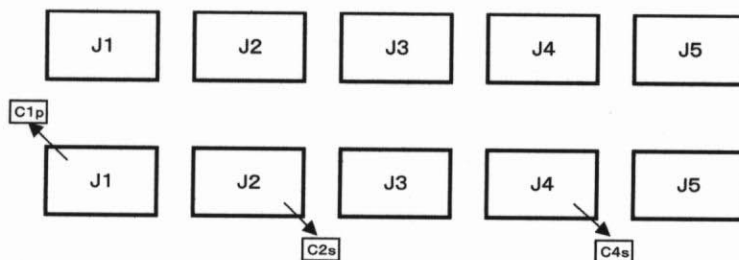
図 6-6 (a) の上段は基準階工程を 5 分割したジョブ工区を示したものである。これが 5 つに水平分割されたサイト工区に割付けられると、(b) の上段のようになり、ジョブ工区とサイト工区は 1 対 1 に対応している。 n 階で最も先行している工区は $S1$ であるが、1 タイムモジュールの後には n 階の $S1$ での作業は完了し、それ以外の S_i 工区のジョブ工区 J_k は一つ後続の $J(k+1)$ に変わる。ただし、 $S5$ 工区の工程は $(n+1)$ 階の $S1$ 工区に移ってそこが $J1$ ジョブ工区となる。しかし、同じ 5 つのジョブ工区を 6 つ以上のサイト工区にまたがって割付けると、(b) の下段に例示したように、上下階でジョブ工区が重なってくる。この場合、上下階の工程要素間にインターフェイスがない限り¹⁰、 $J1$ 工区と $J5$ 工区を合わせて一つのジョブ工区のように扱うことができる。この場合、全体は見かけ上 4 工程分割とみなせるため、其れにともなうサイト工区分割も 4 つに減らすことが可能である。

これを一般化したのが、同図 (c) の 5 つのジョブ工区を水平 3 サイト工区に割付けた例のようなケースである。ここでも、上下階の工程要素間に IF が存在しなければ、無条件にこれを 3 工区分割（サイト工区、ジョブ工区ともに）とみなすことが可能である。上下階の工程要素間に IF が存在する場合でも、タイムモジュール内で適切な工程順になるように IF を付加すれば、上下階を同一ジョブ工区とみなすことができる。上下階が 3 階以上にまたがる場合も同様である。

¹⁰ ジョブ工区分割のもととなる基準階工程は上下階の工程的関係を見逃している。しかし、2 章で述べたように、IF は構成材間の取合い（どちらが支えているか、勝っているか、仕上げ材か下地材か）によって一意的に決まるほか、作業上の都合（作業箇所へアクセスできるか、足場が用意されているか）によってきまるため、上下階の工程要素間でも IF の有無を検討しなければならない。一般的に言って構成材間の取合いが問題となることはなく、作業場の都合で IF が生ずるケースが多い。

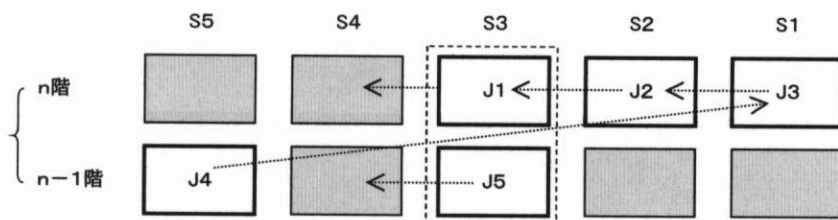
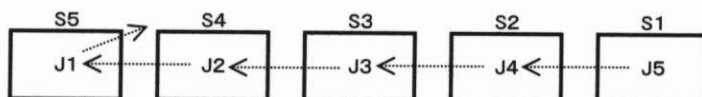
(a) 5ジョブ工区分割工程
(J1が先行工程)

J1から自由先行要素C1pを
分離, またJ2, J4から自由後続
要素C2s, C4sをそれぞれ分離



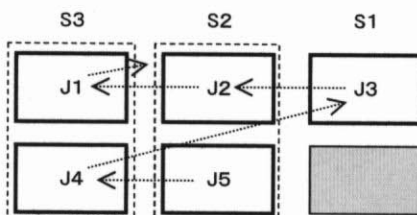
(b) 5ジョブ工区分割工程を
水平5サイト工区に割付け

n階



(c) 5ジョブ工区分割を
水平3サイト工区に割付け

n階
n-1階



(d) 自由先行要素, 自由後続
要素を独立ジョブ工区へ
ジョブ工区は見かけ上不变

n+1階
n階
n-1階

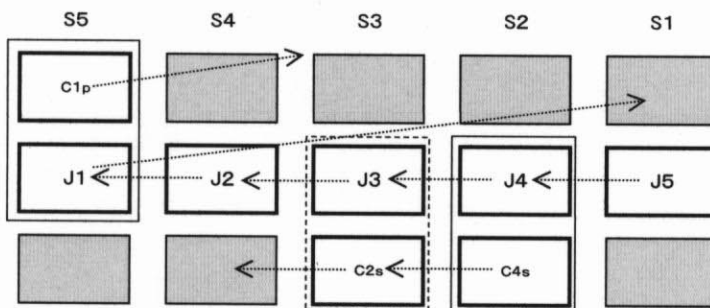


図 6-6

ジョブ工区分割された各工程には, それ自体に先行する工程要素が存在しない自由先行要素, および逆にそれに後続する要素を持たない自由後続要素が多く存在することになる。いま, これらの要素の一部を独立させ (図 6-6 (a) の下段), 自由先行要素, 自由後続要素を数工区分, それぞれ先行あるいは後続させることができる (同図 (d))。C_{1p}, C_{4s} はそれぞれ水平サイト工区分割数分ずつ先行あるいは後続させた例であるが, これらの場合にも, 上下階をひとつのジョブ工区とみなすことができる。

以上をまとめれば、ジョブ工区数は厳密な意味で基準階工程の分割数であり、工程同期化のためにはサイト工区分割数はジョブ工区の数だけ必要である。しかし、上下階のジョブ工区をひとつのジョブ工区とみなせる場合には、水平サイト工区分割数より多いジョブ工区分割（サイト工区数も実際は増える）を行い、より多くの資源を投入して同時作業に当たらせることによって工期短縮が可能となる。

次に、ジョブ工区の異なる工程要素間の順序関係について検討する。分割後のジョブ工区（サイト工区）間ではインターフェイスが消滅し、したがって工区が異なる工程要素間では同一作業チームの作業に属する工程要素間のそれを除き、順序関係はいっさい考慮しなくてよいとしてきたが、現実には、異なる工区間の工程要素間にインターフェイスがあると考えなければならない場合がある。図 6-7 にその例を示した。

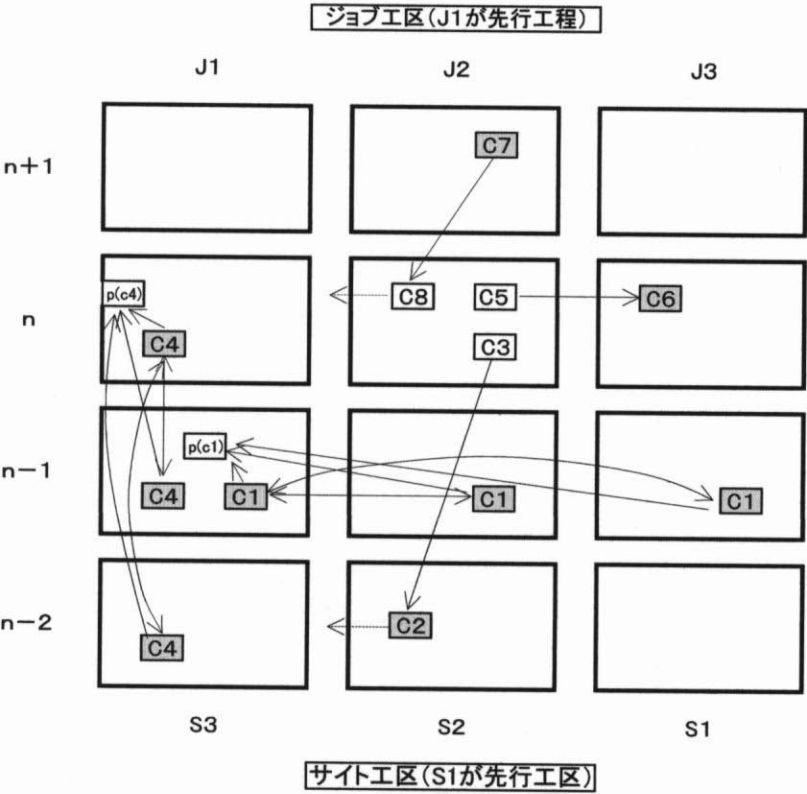


図 6-7

- 例 1 (C1) : 1フロア分のスラブコンクリートを3サイト工区分まとめて打設する場合
- 例 2 (C3→C2) : 下階で脱型した型枠を上階で転用する場合 (C2は自由後続)
- 例 3 (C4) : 鋼管コンクリートの充填コンクリートを3階1節で打設する場合 (C4は自由後続)
- 例 4 (C5→C6) : 大型型枠を解体 (C6) して後続サイト工区で転用組立 (C5) する場合 (C6は自由後続)
- 例 5 (C5→C6) : 大型型枠を解体 (C6) して後続サイト工区で転用組立 (C5) する場合 (C6は自由後続)

例 1, 例 3 は, 同一の工程要素について, 先行するいくつかのサイト工区の作業もまとめて行おうとするものである。この場合, 工程要素は自由後続要素である。また, 先行サイト工区の工程要素が後続するサイト工区の当該工程要素の直接先行工要素に IF を持つ¹¹。鉄骨建方のように, 自由先行要素を数サイト工区先行させ, 後続するいくつかのサイト工区分とまとめて施工する場合も同様である。

例 2, 例 4 は仮設の転用が関係するケースである。この場合, 先行するサイト工区の「解体」に対し後続するサイト工区の「組立」が IF を持つ。ジョブ工区分割を行わない状態での IF (「組立」に対して「解体」が IF を持つ) とは逆向きの IF であることに注意が必要である。また, 「解体」は一つのジョブ工区内では自由後続要素である。

例 5 は, 作業上の都合による順序関係であり, 先行するサイト工区に対し後続するサイト工が IF をもつ。以上の例に共通していることを整理すると, 次のようになる。

- ①異種工程要素が異なるサイト工区 (ジョブ工区間) で IF を持つとき, 先行する (番号の若い) サイト工区に後続する (番号の大きい) サイト工区が IF を持つ。特に仮設の転用がある場合, 先行するサイト工区の「解体」に対し後続するサイト工区の「組立」が IF を持つ。
- ②同種の工程要素をまとめて施工する場合, 後続する (最も番号の大きい) サイト工区に先行する (番号の若い) サイト工区が IF を持つ。
- ③上下階のいくつかのサイト工区, あるいは同一階のいくつかのサイト工区の位置関係, まとまりを指示するには, 水平サイト工区分割数をモードとした指標が有効である。

6. 3 IFM 上におけるジョブ工区・サイト工区への工程の割付け

タクト工程の IFM による割り付け→第 4 章参照

対話型

ジョブ工区・サイト工区への工程の割付けとは, ジョブ工区への工程分割とその工程をサイト工区へ割り付けることの二段階の手続きを指す。本節では, この操作を IFM 上で行うことの原理と手順について説明する。

6. 3. 1 ジョブ工区分割によるインターフェイスの消滅と生成

多工区分割工程では, 基準階工程をジョブ工区工程に分割し, それぞれのジョブ工区を独立した工事として扱えるようにする。そのため, ジョブ工区をまたがる工程要素間の工程関係はいったん消滅するが, 工区間を移動しながら作業を行う各職種については完全同期化が要求されるため, ジョブ工区をまたがって工程要素間に改めて順序関係を付加する必要がある。まず本項では, このような工程順序構造の変化を IF の消滅・生成として IFM 上で取り扱うことの原理について述べる。

図 6-8 は, 基準階工程を 3 ジョブ工区に分割した

¹¹ 当該工程要素は同時作業でもある。

対応するジョブエ区

インターフェイスマトリクス

1. 自エ区 2. 前エ区 3. 自エ区・前エ区 4. 前エ区以前

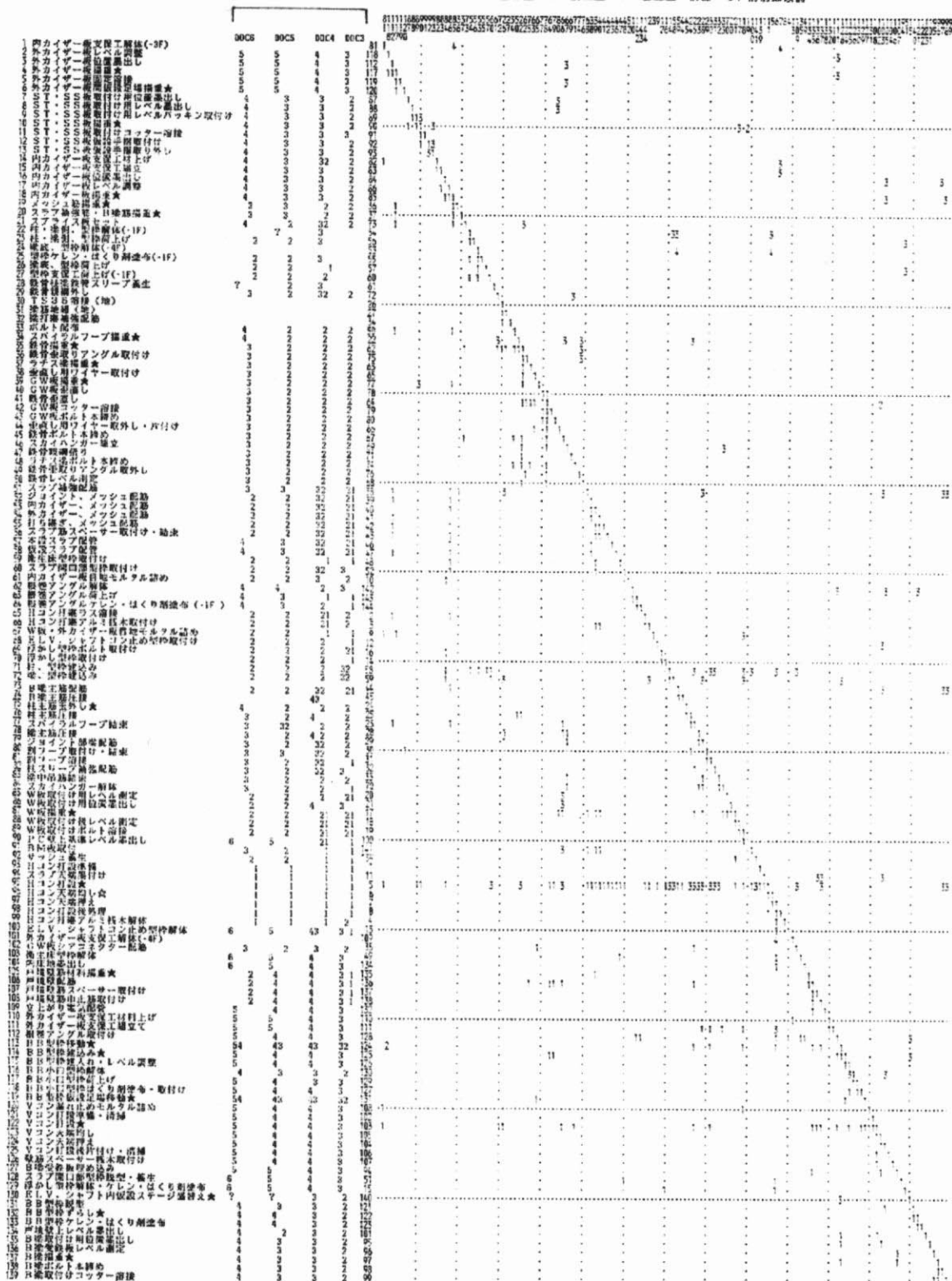


図 6-8 ジョブエ区への工程の割付 (3~6 工区の場合)

①単工区モデルに対応する基準階工程は、きりのよいところで始まり、かつ終わることを 前提に考える。すなわちその始点・終点を境にすでに「基準階」という単位でのジョブ工区分割がなされているということである。現場打ちのスラブが用いられている場合、通常その工程は床スラブが出来上がった状態での墨出し作業から始まり、スラブコンクリート打設（およびその養生）で終わる。

②基準階工程またはすでに工区分割されたジョブ工区工程から、その始点を含む工程要素の組 $\{c_i\}$ （ベクトル表示では）を取り出してこれらを数工区分遅れて始まる新たなジョブ工区に割り付けることができる。

まず、 $\{c_i\}$ の先行要素 $p(\{c_i\})$ と後続要素 $s(\{c_i\})$ からその共通部分 $p(\{c_i\}) \cap s(\{c_i\})$ を得る。これは誘引要素 $I(\{c_i\})$ と呼ばれる集合である¹²。この集合が、後続のサイト工区に移動する要素の全体を意味する。この操作は、IFM 上では、 $p(\{c_i\}) \cap s(\{c_i\})$ とそれ以外の要素のインターフェイスをすべて消去することに相当する。すなわち、IFM における工区分割とは、新たに別の工区に割り付けられる工程要素群の誘引要素を部分行列として取り出し¹³、それらと他の要素との間のインターフェイス（行列の非零要素）を消去する（0に書き換える）ことを意味する。

基準階工程がすでにいくつかのジョブ工区に分割されている場合でも、任意のジョブ工区を再分割することができる。この際、これらのジョブ工区群とすでに定義されたジョブ工区群との間の順序関係が逆転することはない。

逆に、すでに工区分割されたジョブ工区工程（またはその一部どうし）を合成して新たな一つのジョブ工区とすることもできる。ジョブ工区の合成は、新たなジョブ工区に含まれる工程要素群が再び同一作業空間の作業となったことを意味するから、一度消去されたインターフェイスを再度付加しなければならない。この操作を行うためには、単工区の基準階工程に対応する IFM の要素を参照することが必要となる。すなわち、ジョブ工区の分割は、基準階工程を含む任意のジョブ工区分割状態に対応する IFM のみにもとづいて行うことが可能であるが、ジョブ工区の合成は直前のジョブ工区分割状態の IFM と基準階工程（単工区モデル）に対応する IFM の両方を必要とする。基準階工程に対応する IFM は同一作業空間の中で組み立てられる構成材間のインターフェイスによって一意的に決められた工程順序をあらわしているため、その情報は保存されなければならないのである¹⁴。

終点を含むジョブ工区工程の一部を先行するサイト工区に移す操作も同様である。

③工区分割がなされた場合、同一の資源（職種・資機材）を同じ時間に重複して用いてはならない。重複が見られる場合には、適宜 IF を加えて調整する。

④基準階工程の中に仮設資材を転用することが含まれていれば、一般的にジョブ工区分割が必要となる。

一度に転用される仮設資材はひとつに限らないから、いま、 n 個のそれらの組を T_i ($1 \leq$

¹² 3 章参照。

¹³ 部分行列として並び替えることは必要ではない。

¹⁴ IFM の冗長性が消去されることなく絶えず保存されなければならないことも同じ理由による。

$i \leq n$) とあらわす。同じ仮設資材は、「建込み」と「脱型」や「組立」と「解体」といったように、工程要素としては必ず二度ずつ出現する。したがって、各仮説資材について、これらを T_{i1} , T_{i2} のようにあらわす。このとき、 T_{i2} は T_{i1} に対して必ずインターフェイスを持っていて、 T_{i1} に後続し T_{i2} に先行する工程要素の全体は、次に定義されるような誘引要素となる。

$$I(T_i) = S(T_i) \cap P(T_i) - T_i$$

$I(T_i)$ は T_{i1} に後続し T_{i2} に先行するものと、それぞれに先行しかつ後続するものに分かれるから、それらを I_{i12} , I_{i11} , I_{i2} とする。すなわち、

$$I(T_i) = I_{i12} + I_{i11} + I_{i2}$$

$S(T_i)$, $P(T_i)$, $I(T_i)$ のいずれにも含まれることのない工程要素は T_i の平行要素 L_i であるが、この中には $S(T_i)$ に先行するかまたは $P(T_i)$ に後続するもの、およびいずれにも先行・後続しないものの4通りがある。以上の関係を図 6-9 にあらわす。

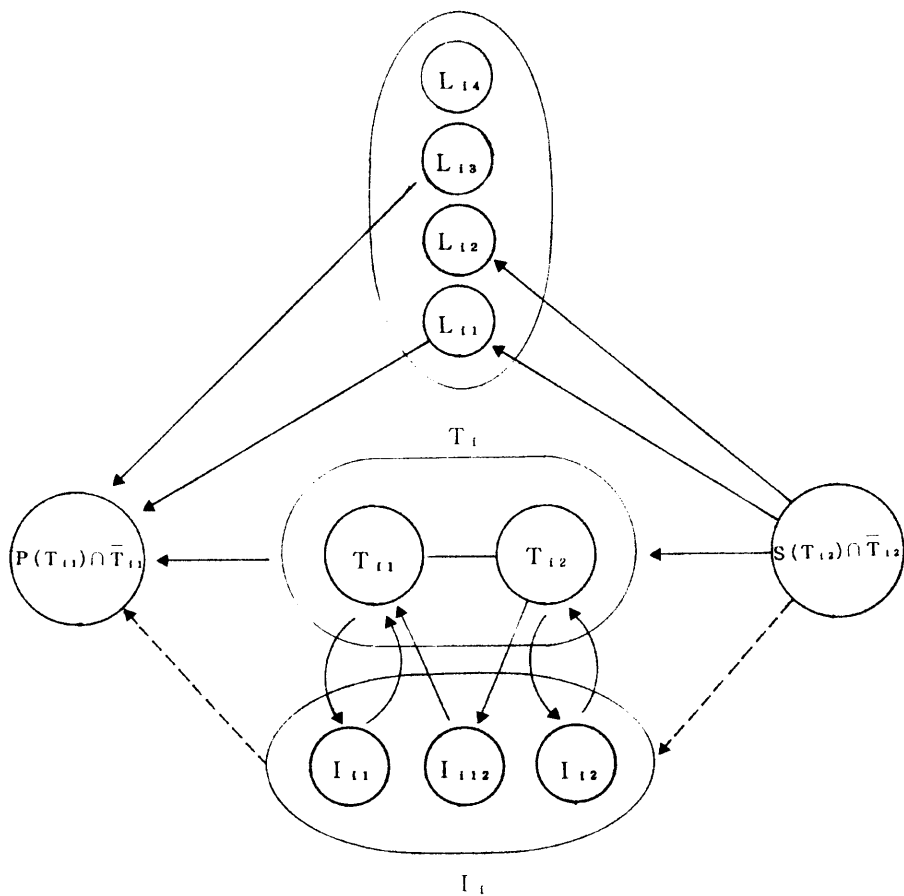


図 6-9

仮設資材 T_i の転用による工区分割とは、先行サイト工区に含まれる工程要素 T_{i2} が終了した後、後続サイト工区で T_{i1} が始められることを意味する。このとき、 T_{i1} 、 I_{i1} および T_{i1} の先行要素全体の集合（ただし T_{i1} を含まない）は先行ジョブ工区（後続サイト工区）に入り、 T_{i2} 、 I_{i2} および T_{i2} を含まない T_{i2} の後続要素全体の集合は後続サイト工区（先行サイト工区）に入る。平行要素 L_i および I_{i12} はどちらに入るか定まらない。

こうして、先行工区、後続工区間のインターフェイスは消滅するが、③により同一資源の重複使用は許されず、また前節で述べたように、転用される仮設資材のサイト工区間の工程順序関係もあって、IF がさらに追加される。このとき注意しなければならないのは、同一仮設資源間のインターフェイスがジョブ工区分割の前後で逆向きになることである¹⁵。一般的に言えば、 T_{i2} をタイムモジュール内の早い時点で施工しなければ、後続ジョブ工区の T_{i1} 以降の作業の進行に支障をきたす可能性がある。

⑤前節に示したように、施工順にリニアに番号を付けられたサイト工区には、基本的に水平・垂直の位置関係による違いはない。すなわち、しかし、足場や作業空間の確保という点で、垂直のサイト工区の位置関係による IF が付加されることがある。この場合、それらの IF は水平サイト工区分割数をモードとして決定される。

●部品化・プレファブ化による空間工区の独立分離

6. 4 多工区同期化工程の作成法

ここでは、戸境壁に大型型枠を用いた板状の SRC 造集合住宅躯体工事を例にとり、その単工区基準階工程を部分行列に分解して多工区同期化構工法計画を完成させる様子を手順を追って説明する。さらに、同一の部分構工法の組合せであることから同じ基準階工程を持つ工事でも、同じ投入資源水準を前提として¹⁶、基準階規模が異なれば異なった多工区同期化工程が導かれることを説明する。

基準階工程に含まれる工程要素、各工程要素の担当職種およびそれらの作業時間、および工程要素間のインターフェイスを表 6-1 および表 6-2 に示す。各職種の作業時間の合計は、歩掛りに基づいて投入資源水準（人数）を決めることにより、移動時間を含めてほぼ 1 日（＝タイムモジュール＝6. 5 時間を目安（最長 8 時間））分になるようにあらかじめ設定してある（完全同期化の前提）。この基準階工程から、適当なジョブ工区（＝サイト工区）分割を求めるという問題である。この場合、基準階工程の工程分割（ジョブ工区分割）を順次増やしながら、各ジョブ工区の工期の最大値、すなわち各職種の工期の最大値がタイムモジュールにおさまる工区分割数と工程要素の割付けを求めるという方法を探る。なお、C、D、E、F、G、S には組作業となっている。図 6-9 に初期入力 of 基準階工程 IFM を示す。

¹⁵ 「建込み」→「解体」が、「解体」→「建込み」の順に変わる。

¹⁶ 同一資源水準の連続的調達を前提とし、異なる基準階戸数の棟を何期かに分けて施工する際に典型的に現れるケースである。また、この場合施工速度が一定（例：2 戸／日）であることに注意。

(j)	要 素	職 種	時 間	先 行 要 素 (i)
A	KMH	筋工1	1.50	E R T
B	外カイザー	筋工1	0.75	E P R T
C	Hコン	筋工1 土工	3.25	B H I J N O P R S
D	Vコン	筋工1 土工	1.25	B E G K L Q R T
E	鉄骨建方	筋工2	2.50	
F	BB脱型	筋工2	0.75	D G K L R U
G	BB建込	筋工2	0.75	E L Q R T
H	内カイザー	筋工2	1.50	A E F R O
I	ST・SS	筋工2	1.00	A H
J	スラブ浮かし型枠	型枠工1	1.50	H
K	建込み・調整	型枠工1	2.00	G
L	根巻きアングル取付	型枠工1	0.75	
M	根巻きアングル解体	型枠工1	0.75	D F L
N	MOS柱・桁行梁建込 (下階の脱型含む)	型枠工2	6.50	A F R
O	内カイザー支保工	型枠工3	2.00	F K
P	外カイザー支保工	型枠工3	2.50	A
Q	戸境壁配筋	鉄筋工1	2.00	E R T
R	柱・梁ST・HP	鉄筋工1	3.50	E T
S	スラブメッシュ敷込み	鉄筋工1 鉄筋工2	1.00	B H R
T	鉄骨本締め	鉄骨工	2.50	E
U	Vコン養生		12.00	D

表 6-2

職 種 ジョブ	職 種	時 間	職 種 ジョブ	職 種	時 間
1	薦 工 1 (3 人)	6 . 7 5	6	型 枠 工 3 (2 人)	4 . 5 0
2	薦 工 2 (3 人)	6 . 5 0	7	鉄 筋 工 1 (4 人)	6 . 5 0
3	土 工 (4 人)	4 . 5 0	8	鉄 筋 工 2 (2 人)	6 . 5 0
4	型 枠 工 1 (4 人)	6 . 5 0	9	鉄 骨 工 (3 人)	5 . 0 0
5	型 枠 工 2 (6 人)	6 . 5 0			

ステップ1からステップ6までは基準階工程を十分に細かいジョブ工区に分割し、サイト工区への対応付けを準備する段階である。ステップ7，8はステップ5の段階，すなわち5ジョブ工区分割を水平サイト3工区(実際は5サイト工区)に割付けた手順をしめしたものであり，ステップ9はステップ6の段階，すなわち6ジョブ工分割を水平4サイト工区（実際は6サイト工区）に割り付けた手順を示したものである。

【ステップ1：単工区基準階モデル】

表1の基準階工程に対応するIFM，それを工程順に並び換えたもの，またそれをさらに到達可能行列として表現した行列を図6-10の(a)，(b)，(c)に示す。工区分割による同時作業がないため，基準階工程のサイクル工期（ t_1 ）はこの段階では33.1時間となっている。

【ステップ2：大型型枠（BB型枠），根巻きアングルを転用することによる2工区分割】

BB型枠（G，F），根巻きアングル（L，M）を転用することを前提に2工区分割を行う。このとき，

$$\begin{aligned} P_{i1} &= P(L, G) = \{E, (L), T, R, A, Q, (G)\} \\ S_{i2} &= S(F, M) = \{(F), (M), N, O, H, I, J, S, C\} \\ I_{i1} &= \{\phi\} \\ I_{i12} &= \{K, D, U\} \\ I_{i2} &= \{\phi\} \\ Li &= \{P, B\} \end{aligned}$$

すなわち、先行・後続各工区への割付けができない工程要素には、型枠建込み調整 (K)、Vコン打設 (D)、Vコン養生 (U) の誘引要素と、外カイザー支保工 (P)、外カイザー (B) の平行要素とがある。このうち、誘引要素についてはVコン養生に要する時間が長いので、後続ジョブ工区に含ませることにする。また、外カイザーについては、Vコン打設時の足場確保 (戸境壁を打つときに必ずしも必要というわけではないが) のために、仮に後続工区に入れておくことにする。この段階でサイクル工期 (t_2) は 24.4 時間となるが、Vコン養生 (夜間の無作業時間も利用可) を除けば 12.4 時間である。ただし、この時間は、同一の職種の作業時間が重ならないように調整した結果のものではない (以下のステップも同じ)。

【ステップ3：鉄骨建方、同本締め、柱梁ジョイント筋を分離した3工区分割】

鉄骨建方 (E)、鉄骨本締め (T) および柱・梁 HP・ST (R) を独立させ、後続ジョブ工区に移す。 $P(E, T, R) \cap S(E, T, R) = \{E, T, R\}$ であるから $I(E, T, R) = \{\emptyset\}$ 、したがって移動するのはこの3要素に限られる。

3工区分割後のサイクル工期 (t_3) は 18.0 時間、Vコン養生を除くと 8.2 時間 (Hコン打設終了が最長) となりほぼ所定のタイムモジュールに近づいている。

【ステップ4：Hコンの分離による4工区分割】

スラブ浮かし型枠 (J) と Hコン (C) を先行するジョブ工区に分離する。 $I(J, C) = \{\emptyset\}$ だから、J、C のみが新工区に属する。この段階ではVコン養生を除くサイクル工期 (t_4) は 6.5 時間 (柱・梁 HP・ST が最遅) で、タイムモジュール (1日=6.5～8時間) の範囲内に入っているが、まだ同一職種の作業時間の重複がある。

【ステップ5：KMHの分離による5工区分割】

柱・梁 HP・ST が分離したことにより、KMH (PCa 壁板) が新たに 2 番目のジョブ工区の始点となった。計画の自由度を確保しておくために、これを単独で別の工区に分離する。この段階のサイクル工期もステップ4と同じである。

【ステップ6：MOS柱、桁行き梁建込み、スラブメッシュ敷込みを分離した6工区分割】

いずれも、工区内の終点である早く始まる工区に移す。この段階でのサイクル工期も前ステップと変わりはない。

【ステップ7：水平3サイト工区への割付け】

ステップ5までの5つのジョブ工区を、もっとも作業が進んでいるHコン打設の工区を第一サイト工区 (=最終ジョブ工区) として、水平3工区に割付ける。

このステップではまずクレーンに関連のある工程要素に順序を与える (インターフェイスを付加する)。その他の職種のジョブ工区間工程順序に関しては、次のステップ8で順序を与える。その結果は、先行順に次のようになっている¹⁷。

クレーン1・鷹工1関係： $A \rightarrow C \rightarrow B \rightarrow D$

¹⁷ ただし、()内はこれ以前にすでにIFとして定義されているもの。

クレーン 2・蔦工 2 関係：E→F (→) G→H→I

サイクル工期は延びて 20.0 時間となるが、V コン養生を除けば最遅終了時間は V コン打設の 8.0 時間となっており、タイムモジュールの 1 日分に納まっている。

【ステップ 8：職種別作業時間のラップの解消＝水平 3 工区分割の完成】

最終的にことなるジョブ工区での作業時間の重複をなくすように、各職種が異なるジョブ工区で受け持つ工程要素にインターフェイスを付加して順序を付ける。付加された IF は次のとおりである。

型枠工 1：F (→) M

鉄筋工 1：R→S¹⁸

V コン養生を除くサイクル工期は 8.0 時間（最遅終了の工程要素は V コン打設）であり、タイムモジュールの範囲内にある。この例の水平 3 サイト工区分割は、結局、6 ジョブ工区分割を重ね合わせたものということになる。

【ステップ 9：水平 4 工区分割】

ステップ 6 の状態からジョブ工区間での同一職種の作業時間の重複を排除して、水平 4 工区分割を完成させる。ステップ 6 はステップ 5 よりもさらに工程に余裕を持たせ、作業空間の混雑を避けることができるように意図されている。

付加された IF は次のとおりである¹⁹。

クレーン 1・蔦工 1 関係：A→B→C→D

クレーン 2・蔦工 2 関係：E→F (→) G→H (→) I

土工関係：C (→) D

型枠工 1 関係：J→F (→) M→L (→) G (→) K

型枠工 3 関係：P→O

鉄筋工 1 関係：Q→S→R

鉄骨工関係：E (→) T

この結果、サイクル工期を 8.0 時間（最遅終了は V コン）とした、水平 3 サイト工区分割の多工区同期化工程が完成する。

以上のようにして完成された水平 3，4 工区分割の工程をバー・チャートの形であらわしたものが、図 6-11、6-12 である。また、同じ工程をタイムモジュール内の時間軸を入れた MAC のかたちであらわしたものが、図 6-12、6-13 である。

¹⁸ 基準階工程ではもともと IF が存在していたが、分割の途中でいったん消えていた。

¹⁹ () 内は既に存在していた関係。

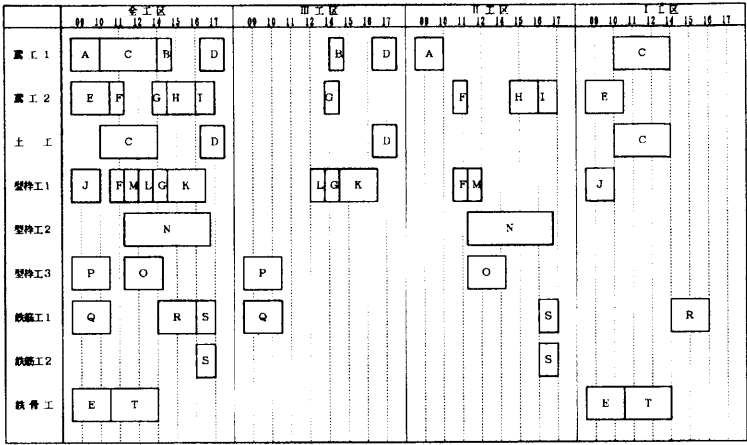


図 6-11

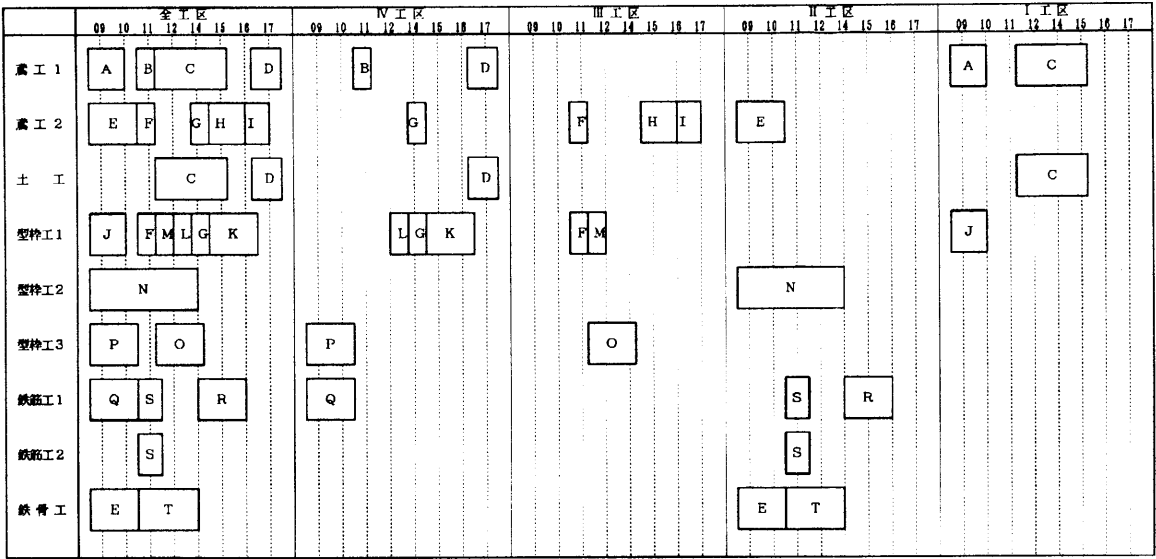


図 6-12

み か け の サイト工区	3	2	1
工 程 要 素	L Q G P B K D U	F M N O H I S	J C
(サイト工区)	(3)	(2)	(1)
工 程 要 素		E T R	A
(サイト工区)		(5 → M(2))	(4 → M(1))

み か け の サイト工区	4	3	2	1
工 程 要 素	L Q G P B K D U	F M O H I	N S	J C
(サイト工区)	(4)	(3)	(2)	(1)
工 程 要 素			E T R	A
(サイト工区)			(6 → M(2))	(5 → M(1))

図 6-13

これらからわかるように、水平 3 工区分割の例では 5 つのジョブ工区分割を同じく 5 つのサイト工区に対応させているが、1，4 サイト工区および 2，5 サイト工区が 3 サイト工区をモードとして重なっているために見かけ上は 3 つのサイト工区分割がなされているように見える。同様に、水平 4 工区分割の場合も、4 サイト工区をモードに 6 つのサイト工区が重なっているのである。

6. 6 本章の結論

【成果】

*工区分割を，基準階工程をあらわす IFM を部分行列に分割することによって解く手法を考案した。

*サイト工区分割とジョブ工区分割を区別して定義することにより，工区分割は基準階工程をジョブ工区に分節し，それをサイト工区に割付けることと，あきらかにした

*完全同期化という概念を導入し，職種の工区間移動をとまなうライン・バランシングの一般的解法を示した

*その際，従来のライン・バランシング手法のように最適解を求めることとせずに，完全同期化を満足する複数の解（工区分割数や工程の異なる複数の多工区同期化工程）を平易な論理で導き得ることを証明した

*工区分割後の工程を計算するために，基準階工程をあらわす IFM のインターフェイスを消去・付加して多工区同期化工程の IFM を作成するロジックを示した

*サイト工区分割数より多いジョブ工区分割を行って工期短縮化をはかるために，サイト工区分割数をモードとする多工区同期化 IFM の作成法を提示した

【第 6 章に関する主要既発表論文・講演】

- ・同期化された水平多工区分割工法の計画に関する研究（その 1）実測データに基づく工法の評価 （1985）
- ・同期化された水平多工区分割工法の計画に関する研究（その 2）工区数・工期・タイムモデルの関係 （1985）
- ・基準階 IFM からの n 工区分割同期化工法の導出法 （1990）
- ・Synchronised Multi-Site Scheduling (1990)

第7章 時間の関数としてのインターフェイス・マトリクス

7. 1 はじめに

本章では、ある時刻における IFM の状態を記述し、演算に応用する手法について述べる。

【目的】

*一つのサイト工区に着目した場合、インターフェイ・マトリクスの IF のうち、すでに消化された工程要素に対して直接後続要素が持っていた IF の意味は変質する。すなわち IFM を時間の経過によって書き換えることができる。

*このように定義された IFM によって、時々刻々変化する工程の進捗状況を表現し、それをリアル・タイムの工程計画に応用する方法を見出すことが本章の最終的な目標である

*なお、これについて最も効果的な IFM (t) の表現法と応用法は浦江真人氏の考案になるもので、これは同氏の学位論文の中核をなしている。本章では、これを別な側面から解釈することにより、一般化することを試みる。

【方法】

*方法1：工程消化にともなって IFM の次元を縮減する方法

*方法2（浦江氏の方法）：IFM をサイクル数 k と時間 t の関数 $R(k, t)$ として表現する方法

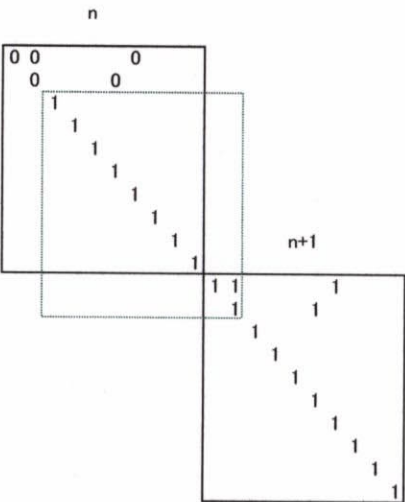
7. 2 方法1：IFM (t)

*あるサイト工区に着目する

*終了した工程要素および後続要素がそれに対して持つ IF を除き（要素の値を0とし）、IFM の次元を縮減する

*IFM (t) の要素は最終的に全てが0となる

*一方、後続するサイト工区で、先行する要素を持たない要素の集合は、それが間接的にでも先行工区に IF を持たない限り、時刻 t において開始可能



7. 3 時間を変数とした IFM (k, t)

7. 3. 1 時間を変数とした IFM の書き換え

7. 3. 2 時間とサイクル数を変数としたリアルタイムの工程演算法

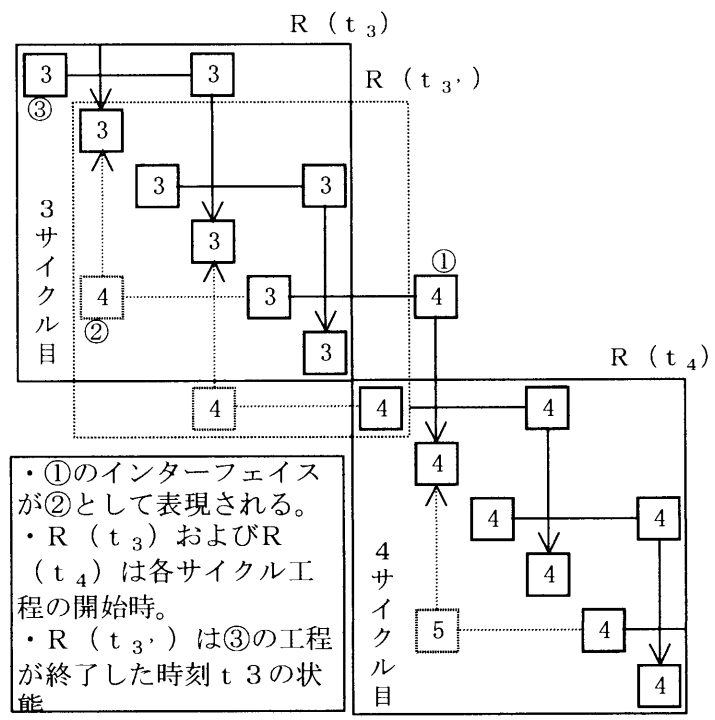
●インターフェイス・マトリクスによるサイクル表現

サイクル工程計画の演算では、インターフェイス・マトリクスのインターフェイスを同一サイクル内に限定しその値を1、それ以外のインターフェイスのないものを0で表わしていた。

ここでは、インターフェイス・マトリクスのインターフェイスの要素の値kを変えることにより繰り返しのあるサイクル工程を1つのマトリクスで表わす方法について述べる。

この場合、kの値はサイクル数となり階数や工区数等に相当する。また、サイクル間にもつインターフェイスは対象となる工程要素のサイクル数の値がサイクリックに代入される。kの値は工程の消化、すなわち時間の進行にともなって書き換えられるためここでのインターフェイス・マトリクスはサイクル数kと時刻tの関数 $R(k, t)$ となる。 $R(k, t)$ は時刻tでの工事の進捗状況（終了状況）をその時点からの1サイクル分の工程として表すマトリクスである（図7-1）。これは、インターフェイスを次サイクルのインターフェイスに置き換えていることを意味している。

図7-1 サイクル数を変数とした IFM の時間的变化



なお、このとき、インターフェイス・マトリクス $R(k, t)$ は上三角行列にオーダリングされている必要はない。ただし、ループは存在せず、すべての工程要素について同一要素のサイクル間のインターフェイスはリニアに並んでいるものとする。

このように、インターフェイス・マトリクスで繰り返しのあるサイクル工程を表現する場合には1サイクル工程分の要素数でサイクリックな表現が可能である。

工程の終了とともにインターフェイス・マトリクスは次の手順で書き換えられる。

- ①時刻 t におけるインターフェイス・マトリクス $R(k, t)$ において、工程要素 i の終了が報告される。
- ②要素 i にインターフェイスを持つ要素の値を置き換える。つまり、 $R(k, t)$ の i 行の非ゼロの要素 $(i, i), (i, j_1), (i, j_2), \dots$ の値を1だけ増す。このことは、 k サイクル目に持っていたインターフェイスを $k + 1$ サイクル目のインターフェイスに置き換えるということである。

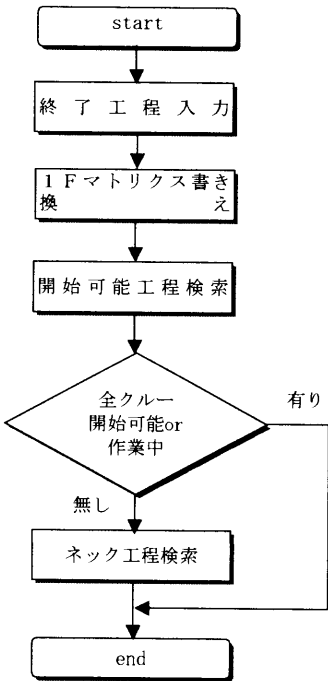
7. 4 リアルタイムの工程管理手法への IFM (t) の応用

多工区同期化工法のように複雑な工程や詳細な工程計画がなされている工事では、工程の進捗状況を常に把握している必要があるとともに、在来工法に比べその意義も大きい。なぜなら、在来工法は1つの工程の作業する時間が日単位（短くても半日）であるのに対して、多工区同期化工法では時間単位（短いものは15分）で計画されているからである。このため、各工程のフロートも短く工程の遅れが他の工程へ及ぼす影響が大きい。そのためにも、今どこで作業が行われているか直接現場にいかないで現場事務所等で知ることができれば管理者にとって非常に有効である。また、進捗の把握と同時に、次工程の表示や遅れ工程の警告等が表示できるような掲示板が現場にあれば詳細な工程のため作業者が工程を把握するのが困難な場合に手助けともなる。

このように、各作業者に次の作業内容を指示したり、作業の遅れを管理者や作業者が常に認識することにより、より緻密で状況に即したリアルタイムの工程の管理が可能になる。

次に、インターフェイス・マトリクスにサイクル数及び時刻を変数として組み込み、工事の進行過程でリアルタイムに工程統制を行うための演算手法について述べる。演算では工程の終了とともにマトリクスが書き換えられ、開始可能工程やネック工程が検索される。また、このマトリクスから次サイクル分の工程計画も作成される（図 7-2）。

図 7-2 インターフェイス・マトリクスによるリアルタイムの工程計画管理フロー



インターフェイス・マトリクスによるリアルタイムの工程統制の手法は次のとおりである。

1) 工程終了の処理

インターフェイス・マトリクスのインターフェイスの値を工程が終了する度に書き換える。マトリクス中の要素の値を書き換えることにより、工程の進捗状況を表わすことができる。

2) 開始可能工程の検索

開始可能工程とはある工程が終了したとき、次に開始することができる工程である。すなわち、終了した工程の直接後続作業が開始可能工程となる。開始可能工程は書き換えられたインターフェイス・マトリクスから検索される。

開始可能工程の検索は次の方法により行う。ここでは、その終了したクルーだけではなくすべての工程について検索対象としている。

書き換えられたインターフェイス・マトリクス $R(k', t')$ において他の工程要素にインターフェイスをもたない要素が開始可能工程である。つまり、工程要素 j にインターフェイスをもつ工程要素の $(i_1, j), (i_2, j), \dots$ がすべて (j, j) より大きいとき工程要素 j は開始可能工程である。これをすべての工程要素 j に対して検索する。

3) ネック工程の検索

あるクルー（職種）が他の工程の影響を受けて無作業状態になっていることをそのクルーが手持ちの状態にあるという。これは、前工程が未終了のためにその工程の終了を待っている場合である。ここで、現在作業中ではなく、かつ検索された開始可能工程の中に自クルーの工程がないクルーが手持ち状態にあるクルーである。そのクルーに対してクリティカルとなっている工程がネック工程である。

ネック工程の検索は次の要領で行う。

- ① 開始可能と作業中のすべての工程を終了とみなしインターフェイス・マトリクスを書き換える。
- ② 書き換えられたインターフェイス・マトリクスの開始可能工程を検索する。
- ③ 開始可能工程が手持ちのクルーに該当しているときこの工程要素が手持ちの工程要素となる。
手持ちの工程要素がない場合には①から③を繰り返す。
- ④ 手持ちの工程要素がインターフェイスにもつ工程要素で作業中または開始可能工程であるものを捜す。該当する工程要素があればそれがネック工程である。該当する工程要素がない場合は④を繰り返す。

このインターフェイス・マトリクスで取り扱う工程要素とは作業以外のものでもインターフェイスを持つものであれば構わない。また、ある時点での終了した状態の $R(k, t)$ は、次の時点で開始されるサイクルを表わしているため、これをもとに次サイクル工程や翌日工程を算出することができる。

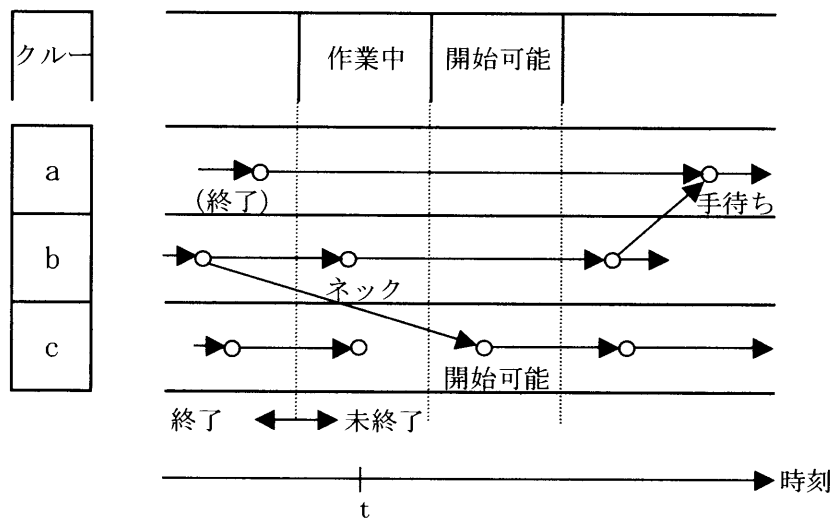


図 7-3 開始可能工程・ネック工程の検索モデル

4) 次サイクルについての工程計画

書き換えられたインターフェイス・マトリクスで、ある時刻以降の次サイクルの工程や翌日の工程の計画工程を作成する。これは、このインターフェイス・マトリクスがある時刻以降の 1 サイクル分の工程の作業順序（インターフェイス）を記録しているからである。これにより、実績に即したリアルタイムの工程計画が可能となった。

次に、進捗状況が記録されているインターフェイス・マトリクス $R(k,t)$ により、時刻 t 以降の計画工程を作成する手法を述べる。このときのインターフェイス・マトリクスは時刻 t から 1 サイクル分の工程要素の順序関係をインターフェイスとして保持している。

基本的な計画フローは前節のサイクル工程計画手法と同じである。次サイクルの工程計画はつぎの手順で行なわれる。

①インターフェイス・マトリクスのオーダリング

インターフェイス・マトリクスをサイクル順にオーダリングする。この場合、到達可能行列の計算は行なわない。すなわち、中身の仕事（対角線上）のサイクル数の大きさ順に要素を並びかえる（表 5.2 表 5.3）。

②作業時間の定義

習熟効果により歩がかりが向上した場合や作業者の人数が変化する場合に対応して必要作業時間を定義する。

③最早開始時間の演算

最早開始時刻およびフリーフロートの演算を行なう。これはサイクル工程計画の場合と同じアルゴリズムにより計算される。しかし、ここで演算に用いられるインターフェイス・マトリクスはサ

イクル数が要素に書き込まれている $R(k, t)$ である。このとき、要素 i のサイクル数 k_{ii} に持つインターフェイスのサイクル数 k_{ij} が同じ値の場合その要素のインターフェイスとして認識され演算の対象となる。これは、 k_{ii} より大きいものは次サイクル以降にインターフェイスを持っているもので、小さいものはもうすでに工程が終了している場合である。

表 5.4 および表 5.5 は次節で述べる適用事例での計算結果であるが対角線上の数値は各工程要素の作業時間である。

④作業時刻の調整

最早開始日程の演算結果よりバーチャートを出力する(表 5.6)。ここでクルー別に出力し、作業時間の重複を確認する。

同一クルー内で作業時間が重複する場合、これを調整し、必要があれば、インターフェイスや作業時間を変更し、計算を繰り返す。

⑤次サイクル工程の作成

以上の結果をもとに、時刻 t から 1 サイクル分の計画工程が作成される。また、次サイクル工程の第 1 日分は翌日の工程計画でもあるため、この手続きを毎日行なうことによりその翌日工程を毎日提示することも可能である。

7. 5 本章の結論

【成果】

- * 工事の進行状況に応じて、時刻 t における IFM の状態を書き換えられるようにした
 - * これにより、IFM のリアル・タイムでの工程計画・管理手法への応用が可能となった
- POS (Point of Sales) にならない POP (Point of Production) と名付けた

【第 7 章に関する主要既発表論文・講演】

- ・ 工程要素間のインターフェイスに基づいたリアルタイムの工事管理手法の開発 (1988)

第8章 垂直・水平の工区複合モードを組み込んだ多工区同期化工程の作成

8. 1 はじめに

8. 1. 1 ジョブ工区におけるサイト工区分割複合化が必要な理由とその解決策

本章では、水平・垂直に分割された繰返し工程を工程要素（のまとまり）ごとに工区規模を変えて複合し、高次の多工区同期化工程を作成する手法について述べる。

通常、多工区同期化工程では基準階単工区モデルから出発し、階毎の分節を前提に1階分をさらに水平工区分割する。その単位をサイト工区と呼ぶが、このとき、ジョブ工区分割（工程分割）の1単位は各サイト工区に対応しており、各ジョブ工区工程のサイクル工程も同一である。

このように一様なジョブ工区分割をした場合には、工程の単純化がはかれると同時に繰返し効果も大きいので生産効率が上がることはこれまで多くの事例で実証されている。しかし、条件によっては、このように一様な分割法に変えて、特定のジョブ工区工程を垂直・水平にまとめ、それぞれを異なる大きさとタイムモジュールを持つサイト工区分割に対応させた方が望ましい場合がある。

例えば次のような場合がある。

- 1) 既存構造物の解体工事や地下工事の都合により、階ごとに地上階の施工を進めることができず、水平に分割された一部の工区を先行して工事を進めなければならない場合。
- 2) 鉄骨柱の各階への分節や、スラブ・コンクリートの水平1工区分の打設など、1サイト工区単位の施工量・部材規模が小さすぎるか、そのために不必要な分割作業が強えられる場合。

このような場合には、適宜相当するジョブ工区に対応するサイト工区分割を大きくすることが必要となるが、そのときには各ジョブ工区の水平・垂直の工区複合数の最大をそれぞれ m 、 n として $m \times n$ の工区を含むインターフェイス・マトリクスの計算を行わなければならないため、計算に困難をきたす。

著者らはかつて、インデックス・テーブルと呼ぶ全階数の全水平工区分割数と工程要素群を両軸にとったマトリクスにより、この複雑な構工法計画問題を解く手法を提案した¹。その概要については次項にまとめて紹介するが、この方法にもやや複雑なところがあった。本章の次節以降では、第6章の成果にもとづき、工区間での工程要素のインターフェイスにもとづいたより簡便な手法を開発した結果について述べる。

8. 1. 2 インデックス・テーブルによる解法の概要

インデックス・テーブルの基本的なアイデアは次のようなものである。すなわち、IFMによって全工区にわたる全工程要素を一つずつ計画管理しようとするれば、工程要素の次元は繰返し数 r を基準階工程1サイクルに含まれる工程要素数 n に乗じた値となり、IFMは結局 $(n \times r)^2$ の大きさとなって事実上計算は不可能となってしまう。そこでこのような難点を回避するため、工区および工程要素群を両軸にとったインデックス・テーブル（以下 IDXT）を用いることとし、その要素としてあらわされた工区・工程を媒介に、工程要素間のIFをあらわす $n \times n$ の行列の計算を組み合

¹ 1988年、第4回建築生産と管理技術シンポジウム論文。この内容はすでに浦江真人氏の学位論文中に収められているが、比較のためその概要を「8. 1 はじめに」に含めて紹介することにした。

わせることによって全サイクル、全工程要素の工程を扱えるようにしたのである。

以下に、IDXT と $n \times n$ の IFM を用いて開始可能工程を検索する方法、およびそれにもとづいて次のサイクルに含まれる工程要素の工区を確定する方法、さらにマイルストーンを設定することによって適当な工区工程を求める方法を順次述べる。

●IFM, IDXT の定義

始点、終点が明確に決められた一種の工程要素の工程順序をあらわす IMF を、単工区モデルの IMF と呼び IMF1 と記す。工区境が空間的にも日期的にも確立した工区はすべて単工区モデルとして扱うことができる。工区間の IF をあらわす IMF として本研究では他に IFMW と IFMH を用いる。前者は上下の工区間の IF を表す行列、後者は隣接工区間の IF を表す行列である。隣接関係は、上下関係と異なって先行後続関係があいまいであるため、後述するように IF に関する演算は複雑になる。

インデックス・テーブル IDXT は縦軸、横軸にそれぞれ工区、工程要素をとり、その要素 (i, j) を、i 工区の j 工程要素が狩猟した場合に 0、それ以外の場合に 1 とした行列である。工事開始時には、と行く低の工程要素を欠く工区がある場合を除き、全ての要素が 1 となっており、工事の進行に伴って対応する工区・工程が 0 に書き改められていく。

●開始可能工程の検索

ここで言う開始可能工程とは、ある時点で未終了の工区・工程要素の組のうち、先行する工区・工程要素を全く持たないものを指す。これらは一つの工区・工程要素が終了するたびに計算し直される。

p 工区の q 要素が終了すると、それに直接後続する工区内の要素、または工区外の要素が開始可能な要素の候補として検索される。

RFS : ある時点で開始可能な工区全体の集合

IDXT (p, q) \rightarrow 0

IFM1 (q) : p 工区内の直接後続要素

IFMV (p, q) : p 工区の上階の直接後続要素

IMFV は上下工区の IF を表す行列

IFMH (p, q) : 隣接工区の直接後続要素の候補

IFMH は隣接工区区間の IF を表す行列

ただしここでの隣接工区は空間的相関関係だけに着目したもので (常に p 工区が先行と仮定)、IMFH (p, q) が実際 (p, q) に後続することを必ずしも意味しない。実際の後続要素 (f, g) を求めるのには、IDXT を用いる。すなわち、 $(f', g') \in \text{IFMH}(p, q)$ について、 $\text{IDXT}(f', g') = 1$ ((f', g') が未終了) ならば (f', g') は後続要素、 $\text{IDXT}(f', g') = 0$ (終了) ならば (f', g') は先行要素。

これら後続要素群の任意の要素について、それが工区内および先行工区に未終了の直接先行要素を持たない場合に限り、それは開始可能工程となる。ただし、 $\text{IMFH}^T(r, s)$ については、特定の工区毎に未終了の直接先行要素がないこと、あるいはすべてが未終了であること (その工区を後続とすること) を確認する。

$\forall (r, s) \in \text{IMF1}(q) \cup \text{IFMV}(p, q) \cup \{(f, g)\}$

$\text{IFM1}^T(s)$: r 工区内の直接先行要素

$\text{IFMV}^T(r, s)$: (r, s) の直接先行要素

$IFMHT(r, s) : (r, s)$ の直接先行要素の候補

もし、 $IDXT(r, x) = 0, \forall x \in IFM1^T(s)$

$IDXT(a, b) = 0, \forall (a, b) \in IFMVT(r, s)$

かつ $y_i, \{y_i | (y_i, z_j) \in IFMHT(r, s)\}$ について

$IDXT(y_i, z_j) = 0$ または $1 (\forall j)$

が同時に成り立つのであれば、 (r, s) は開始可能工程。 (p, q) の終了によって新たに開始可能になる (r, s) の集合 $RFS_{p,q}$ とあらわせば、 RFS は次のように更新される。

$$RFS = RFS - (p, q) + RFS_{p,q}$$

● サイクルへの工区番号の割付け

上述の開始可能工程を用いて、一連のサイクルに含まれる各工程要素に工区番号を割り付けていく方法について述べる。ここで言うサイクルとは周期的に施工される一群の工程要素の集合を指し、それぞれは時系列的に並べられるためサイクル番号を持つ。

工区境が揃えられているにしても、工程要素により工区分割が異なることがあるため (e x. 鉄骨を2層1節で施工する場合など)、周期性は工区分割規模の最小公倍数となって現れるが、ここでは計算の便宜のため1工区を単位としてサイクルを定義する。

各サイクルへの工区番号の割り付けは次の手順による。

- ① k サイクルにおいて作業中の工程要素を一つずつ終了させる。その都度、開始可能工程を書き換える
- ② k サイクル内で一度も終了していない工程要素を探して、①を繰り返す。また複数工区を1施工単位とする要素は、その工区数が繰り返されるまで終了させず次サイクルに繰り越す
このとき、未終了の工区がある工程要素については、(いずれの工区でも未着手の要素を除き) 必ず k サイクルを埋めなければならない。未終了の工区がありながら開始可能工区のない工程要素がある場合には、さかのぼって割り付けを変更する
- ③ k サイクルの開始可能工程の検索がすべて終わった時点で各工程要素につき $k+1$ サイクルの工区番号を選ぶ。その方法は①、②に同じ。

● 先行マイルストーン工程による工区工程の算出

開始可能工程を逐次検討しながら次サイクル工程に工区を割り付けていく方法によって全サイクル工程を求めることが可能であるが、その解は無数にある。ここでは先行のマイルストーン工程を設定することにより、そこに最短で至る経路を探す方法について述べる。

$\{(k, m)\} : \text{マイルストーンとなる先行工区・工程の集合}$

$\{(k', m')\} : \{IFM1^T(k, m)\} : \text{マイルストーン工程に各工区内で先行する要素の集合}$

これらの全てについて先行要素を順次求めてゆけばよいわけであるが、隣接工区の多くが未終了である可能性が強いため、工区間の先行後続を定義し難い。そこで、現時点までに終了している工区・工程をもとに経路を求める方法をとる。

- ① 現時点での開始可能工程 ($KK1$) の中に $\{(k', m')\}$ の要素 $(k'1, m'1)$ があれば、それを順次終了させ開始可能工程を書き直す ($KK1$)。 $KK1$ の中に $\{(k', m')\}$ の要素がなくなれば (あるいは初めからなければ)、 $KK1$ を全て終了させる
- ② そのときに開始可能工程 ($KK2$) を検索する。その中に $\{(k', m')\}$ の要素 $(k'2, m'2)$ があればそれを順次終了させ、 $KK2$ を書き換え、以下を計算する。

$$IFM1^T(k'2, m'2) \cap KK1 \quad (1)$$

$$IFMV^T(k'2, m'2) \cap KK1 \quad (2)$$

$$IFMH^T(k'2, m'2) \cap KK1 \quad (3)$$

$IFMH^T(k'2, m'2)$ は $(k'2, m'2)$ の直接先行要素を含む集合であって、直接先行要素そのものではないことに注意。(1)、(2)、(3)の結合が、 $(k'2, m'2)$ を開始するために終了させておくことが必要な先行要素の集合を含む集合 $F1$ である。 $KK2$ の中に $\{(k', m')\}$ の要素がなくなれば、 $F1 = \bigcup F_i$ とし、 $KK2$ を全て終了させる。

③同様に $KK3$ 、 $F2$ を求める。

④ $\{(k', m')\}$ の要素が洗い出されるまで繰り返す。

⑤ F_j の層の順に選ばれたサイクルがマイルストーンに至る最短経路である。

8. 2 工区の異なる工程要素間の IF による複合モードの多工区同期化手法

8. 2. 1 工区の異なる工程要素間の IF

a) 上下左右で隣接する工区間で同一工程要素を持つ工程関係

：異なるサイト工区の工程要素間の工程関係

上下左右に連続する工区単位について考えてみる。異なる工程要素については、これまで述べてきたように、ジョブ工区が異なればサイト工区も異なることになって、両者の間に IF は存在せず、したがって先行・後続関係もない。一方、隣接する工区間における同一工程要素のインターフェイスについては、上下階の関係と水平の隣接とで意味合いが異なる²。上下階では支持関係によるインターフェイスが存在することがあるため³、その場合、上階の要素が下階のそれに IF を持つ。水平の隣接関係ではどちらを先に開始してもよいが、計画上はこのような恣意性を積極的に許すことの意味はほとんどなく、また、他の工程要素との IF を考えるといつそう複雑になるから、サイト工区番号の若い工区（先行する工区）の同一工程要素が先行するということにしておく⁴。

a-1) 上階の工程要素は下階の同一工程要素が終了するまで開始することはできない。

a-2) 隣接工区間では、サイト工区番号の若い工程要素が先行し、サイト工区番号の大きい工程要素が後続する。

b) 自由先行要素と自由後続要素のジョブ工区移動と制限

：ジョブ工区工程内での工程要素の移動

各ジョブ工区の自由先行要素、自由後続要素については、それをさらに先行、あるいは後続するジョブ工区に移動することができるが、いくつかの制限がある。

² 6章参照。上下階と水平隣接階でこのような違いが出るのは、もともと単工区の基準階工程では上下階の重なりが無いものとしているからでもある。

³ 2章参照。

⁴ 異常作業が発生して順番を逆転しなければならない場合や、手待ち作業を発生させないために任意の工区で開始可能作業を検索し、作業を割り当てようとする場合（第7章参照）はこのかぎりではない。

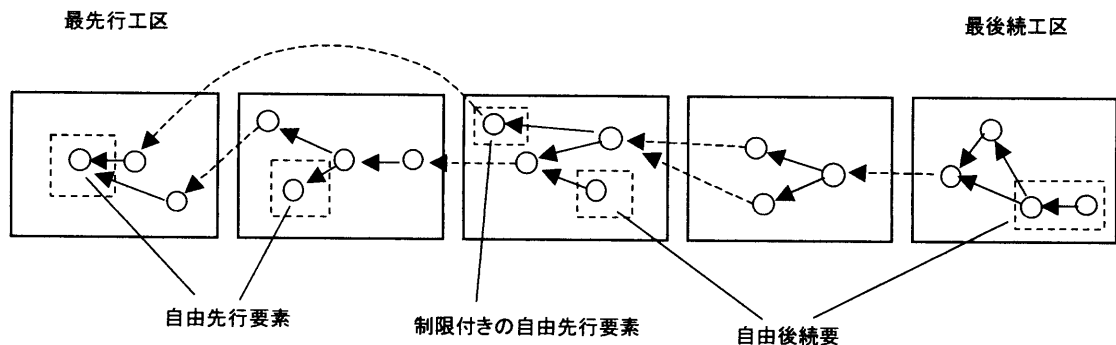


図 8-1

鉄骨建方のように、最先行工区の自由先行要素は、制限なく先行ジョブ工区に移動できる。最後続工区の自由後続要素も同様である。先行するジョブ工区を持つジョブ工区の自由先行要素は、それが単工区の基準階工程で先行ジョブ工区の工程要素に IF を持たない限り、自由に先行できる⁵。しかし、そうでない場合は、ジョブ工区分割以前に IF を持っていた直接先行要素のうち最遅終了のもののジョブ工区、またはその次の工区までしか、移動できない⁶。ここでは、一般的に成立可能な条件として、「次のジョブ工区」をとる。自由後続要素についても同様である（図 8-1）。

自由先行（後続）要素を先行（後続）するジョブ工区に移動させた場合、その直後（直前）の工程要素が新たにそのジョブ工区内で自由先行（後続）要素になることがある。この場合、これを先述の条件の範囲内で自由に先行させる事ができる。図 8-1 右端の 2 要素を例えばスラブ H コン打設およびスラブ型枠解体として読めば、スラブ型枠解体を数工区後に移動させ、その範囲でスラブ H コン打設を後続のジョブ工区に移動させることができる。

注意しなければならないのは、この操作が同一ジョブ工区内でのことであって、異なるサイト区間での工程要素の IF は無視していることである。これを含んで扱うには、（a）の条件を合わせて考えなければならない。（a）は異なるサイト工区の工程要素間の関係であり、（b）は同一サイトのジョブ工区間の工程要素の関係を扱ったものである。

以上を整理すると次のようになる。

b-1) 自由先行要素は、（a-1）、（a-2）の制限の範囲内で、ジョブ工区分割以前の直接先行要素のうち最遅終了のものの次のジョブ工区まで移動できる。

b-1')（上記の特殊な場合として）最先行ジョブ工区の自由先行要素は、自由に先行（ジョブ）工区に移動できる。

b-2) 自由後続要素は、（a-1）、（a-2）の制限の範囲内で、ジョブ工区分割以前の直接後続要素のうち最早開始のものの前のジョブ工区まで移動できる。

b-2')（上記の特殊な場合として）最先行ジョブ工区の自由先行要素は、自由に先行（ジョブ）工区に移動できる。

c) 仮設の取扱い

⁵ この際（a-2）も条件となっていることに注意。

⁶ 当該工程要素の作業時間による。

仮設は基本的に「組立」、「解体」の2工程要素に分解されており、また両者が同じジョブ工区に存在していないことが一般的である。仮設の転用がない場合、解体は一般に自由後続要素となる。仮設の転用は、6章で触れたように、同一構成材でありながら異なるサイト工区間の工程要素の関係であり、そのため同一ジョブ工区工程（基準階）のIFとは逆転した向きのIFを持つ（サイト工区間では逆転していない）。その転用サイクルが直接資材量に関係するため、これを工程要素間のインターフェイスとして取扱えるようにしておく必要がある。

仮設の転用には、任意の工区数先行するサイト工区から転用される場合と、その特別な場合として、直下階、またはさらにその下の階から転用される場合がある。後者の場合には、水平工区分割数をモードとしたIFの操作が可能である。

仮設の「組立」、「解体」の間には、仮設の利用期間がありこれに（同一サイト工区内では）誘引要素としての他の工程要素が含まれる。足場の場合には、それを必要とする工程要素がこれにあたる。RC造の躯体工事に用いられる型枠の場合には、仮設を直接必要とする作業はコンクリートの打設であるが、工程を考える上ではさらに型枠存置期間を考慮しなければならない。型枠存置期間には作業を必要としないが、これを担当職種と作業量がない工程要素として他と同等に扱うことができる。

c-1) 転用先のサイト工区の「仮設組立」は、転用元サイト工区の「仮設解体」および「仮設を要する作業」にIFを持つ。

c-2) 仮設がk階下の階から転用されるとき、転用先の「仮設組立」は水平サイト工区分割をk倍した数だけ先行するサイト工区の「仮設組立」および「仮設を要する作業」にIFを持つ。

c-3) 型枠存置期間は職種、作業時間のないダミーの工程要素として同一サイト工区の「コンクリート打設」にIFを持ち、「型枠解体」からのIFを持つ工程要素として扱う。

8. 2. 2 IFとしての取り扱い

階数、水平工区分割数mをモードとするルール。

階数をn ($1 \leq k \leq n$)、水平工区分割数m ($1 \leq i \leq m$) として

k階i水平サイト工区の通し番号は、 $s = m \times k + i$

仮設転用のモード：

k階下から転用する場合 $ct_{id}(s - k \times m) \leftarrow ct_{ia}(s)$

ただし、型枠存置期間 p_i については次のような関係がある。

(組立) \leftarrow (打設終了) \leftarrow (存置期間) \leftarrow (解体)

8. 3 複合サイト工区分割型多工区同期化工法の工程計画

8. 3. 1 ジョブ工程（工程要素）の垂直合成と水平合成

図8-2は4つの水平サイト工区分割数に対して6つのジョブ工区（工程要素）を持つジョブ工区工程の最先行工区を2階分垂直合成し、さらに最終工区を1階分水平合成した例を示したものである。最初のジョブ工区は鉄骨建方、最後のジョブ工区をスラブ・コンクリート打設としてみよう。

鉄骨建方を2層1節として2サイクル先行させると、後続の工程は鉄骨建方と干渉することなく、後続できる。同様に、各サイト工区のスラブ・コンクリート打設を各階の最終サイト工区のそれまで遅らせると、1階分のコンクリート打設をまとめて施工することができる。コンクリート打設の作業チームはこの場合4タイムモジュールでw1サイクルのところ、1タイムモジュール分しか

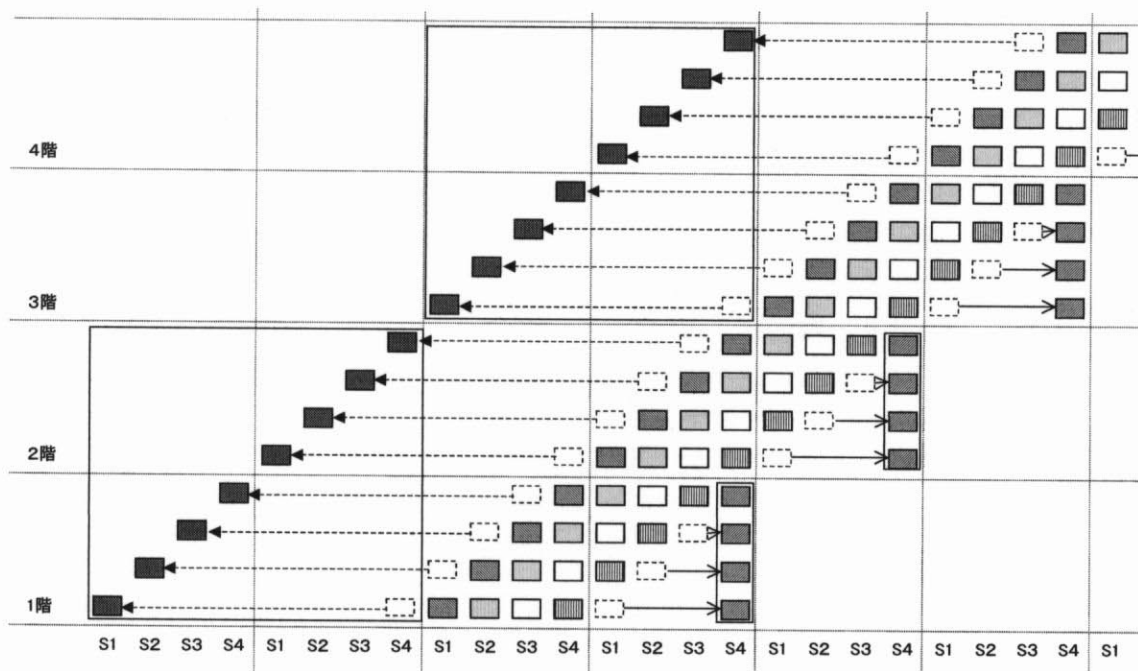


図8-2 ジョブ工区の垂直・水平統合

稼動していないので、完全同期化はなされていない。鉄骨建方でも本来同期化は保障されていないが、この例示では、柱は2層1節の長さを持つが、限られた本数を毎タイムモジュールに建てることとし、完全同期化がはかられたものとしている。

このように、単工区基準階工程から任意の工程要素のそれぞれについて任意の単位の垂直・水平サイト工区分割をもつ複合サイト工区分割型多工区同期化工程を作り出すことが可能である。寄り複雑な例としては、積層工法で適用される鉄骨の建方工程（図 8-3）がある。これは、柱長さを十分長く保っておきながら、歪み直しが容易なように各柱の接合位置を変えたものである。

8. 3. 1 IFM による複合サイト分割型多工区同期化工法の計画

8. 4 本章の結論

【成果】

*水平工区分割法を、垂直工区分割法、任意の水平・垂直工区分割の組合せによる工区分割法に拡張した

*行列の次元を大きくせずに IFM の演算を可能にする方法として、インデックス・テーブル IDXT を用いる方法と、ジョブ工区間のインターフェイスをサイト工区分割数のモードとして解決する方法の二つを示した

【第8章に関する主要既発表論文・講演】

- ・垂直工区分割工法の工程計画と評価 (1987)
- ・水平・垂直に工区分割された工区の工程順序化 (1988)