

構造マトリクスとは - 概念の説明

外山 啄之

構造マトリクス：Structure Matrix とは、

定量的表現によるモデル構築の技法である。業務のロジックとデータ両者が繋がる仕組み全体を俯瞰し、必要に応じ部分に立ち入り、詳細な仕組みを把握できる ---- 透明性を備え、かつ因果要因をたどれる可視性 Visibility with Causal Traceability ---- を特徴とする、

構造マトリクスは、ドイツ染料会社(当時の E.G. Farben 社内のヘキスト部門)の化学技師 Otto Pichler が、化学反応やその収率と流れを扱う技術系の人々と原価を取り扱う事務系の人々との共通の対話基盤を確立するために考え出したものである。経営管理

の構造を上位と下位の区画からなるマトリクス表にモジュール化して表し、複雑な事象を扱う技法である。これにより、マクロとミクロの2つの視点を満たすアイデアとして、1930年代後半に提案され、命名されたとされる。これによると、縦横の関係で事象を表現し、また表現されたものを縦横で把握でき、多元な要素が絡み合う複雑な事象の表現に適している。コンピュータで扱うには、Mother Table の各要素が下位の Children Table にリンクし、Mother Table で計算の関数とその過程を図示的に表現し、因果の関係を演算可能とする。また、階層的な扱いも可能で、巨大複雑事象への適用と発展が広く見込まれる。

構造マトリクスの原理

Fig.1 マトリクスとベクトルの積 に示したマトリクスとベクトルの内積は、Fig.2 縦積・横和 をベースにした表現 に示すように上辺部、左辺部、中央部からなる図として表すことができる..ここで、これらの記号はスカラー値であるとする。

Fig.2 の表現では、上辺部の要素数と、中央部のマ

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} M1 & M2 & 0 & 0 \\ 0 & 0 & M3 & 0 \\ 0 & 0 & 0 & M4 \end{pmatrix} \times \begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix}$$

Fig.1 マトリクスとベクトルの積の従来の表現

ここで使われている記号をスカラー値に付けられた記号とみなして、Fig.3 にあるように $X=C$, $Y=D$, $Z=E$ としてみよう。すると、上辺部と左辺部を関連付けることができ、左辺部の記号に内容が随伴して:上辺部に回ると考えることができる。

そこで、上辺部の2つの矢印が付いた記号(先決変数)から始めて縦積・横和を行い、左辺の X を算出する。

この X (中間変数)が上辺に回って、縦積・横和(この場合は、横がないので縦積のみ)を行い、左辺部 Y を算出する。

同様な操作で中間変数 Y が上辺に回り左辺部に Z

トリクスの、横方向の要素数が一致するため要素の対応を特に意識する必要がなく、上辺部と中央部で縦方向に掛け(縦積) その結果同志を横方向に足す(横和)の演算—縦積・横和—をも表していると理解することができる。

		上辺部			
		A	B	C	D
左辺部	X	M1	M2	0	0
	Y	0	0	M3	0
	Z	0	0	0	M4

中央部

Fig.2 縦積・横和 をベースにした表現

を得る。 Z は上辺部には無い(後決変数)ので計算はここで終る。

これを数式として読み出したのが Fig.4 である。

このように、上辺にあって、左辺にないものを先決変数として、両辺に存在する中間変数を計算し、左辺にあって上辺に無いもの(後決変数)を順次計算していけばすべての左辺の要素を計算できる。

これらの行あるいは列の配置とは関係なく、上辺の先決変数から始まって縦横で影響する因果関係を把握できることになる。

ただし、上辺部・左辺部の各々における記号はユニークであり、ループを含まない一方向の順次計算

として進める。ループ処理については、さらに発展形がある。

この、基本となる計算アーキテクチャーはドイツ

の原型の正方マトリクスを基本とする方式からさらに発展し、ソフトウェアの開発と相まって、1980年、外山によって確立された。¹⁾

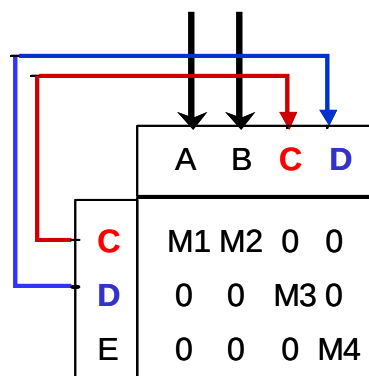


Fig. 3 上辺部と左辺部の関係付け

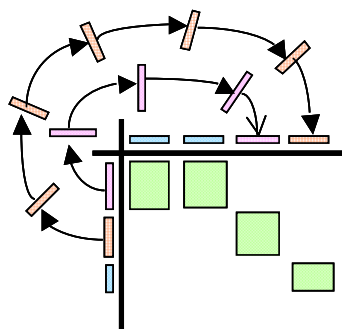
ここで、Fig.5のように上・左辺部の各要素をベクトルと考え、左辺部の縦ベクトルは上辺部に回って横ベクトルになると考える。アーキテクチャーは上・左辺部をベクトル、中央部の各要素はマトリクスへと拡張する。さらに、Fig.6のように上辺部、中央部、下辺部の各要素をマトリクス(2次元のデ

$$\begin{aligned}
 C &= M1 \times A + M2 \times B & \dots (1) \\
 D &= M3 \times C & \dots (2) \\
 E &= M4 \times D & \dots (3) \\
 &= M4 \times M3 \times C \\
 &= M4 \times M3 \times M1 \times A + M4 \times M3 \times M1 \times B \\
 &= \underbrace{M4 \times M3 \times M1}_{Ma} \times A + \underbrace{M4 \times M3 \times M1}_{Mb} \times B \dots (4)
 \end{aligned}$$

Fig.4. Fig.3 から計算順の自動確立

ータブロック)に拡張し内積関係を保つ式(1), (2), (3)とすることも可能である。これは、テーブル相互の演算をする Table Processor とすることができる。

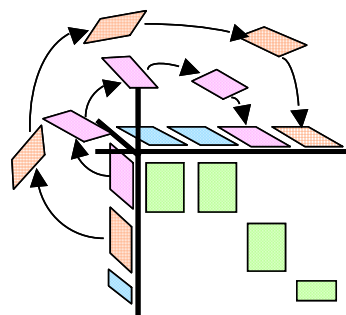
さらに、Fig.6の式(4)の **Ma**, **Mb** に示すように、内積関係を圧縮し感度マトリクスを取り出し、間接的な影響度合いを直接的に把握することも可能となる。



上・左辺部の要素 → ベクトル
中央部の要素 → マトリクス

Fig.5 各要素の複数化 その1

この拡張に対応して、中央部は記号のマトリクスと考え、記号の集まりのテーブルである「Mother Table」と数値データのテーブル群[Children Tables]で構成し、しかも演算による因果関係を表す技法と考えることができる。この一群のテーブル処理を可能とするパッケージの表示画面上に、経営の因果関係をモデルとして投影することができ、さらにそのモデルを操作して想定する事象に対応する結果が得られれば、'複雑な経営現象のモデルに対応する「透明



全要素 → マトリクス に拡張
すなわち Inter Table Processing を可能化

Fig. 6. 各要素の複数化 その2

性を備え、俯瞰的な可視性」が実現される。すなわち、Mother Table で全体の計算の因果関係を知り、その個々の要素である Children Tables に入って数値データとその関連を知ることができることによりコンピュータ内の論理とデータに因果関係をも追求できる「透明性を備えた可視性」をもってアクセスでき、想定した複数のケースについて計算し、経営策定や差異分析が可能となる。

タイプ記号

構造マトリクスにおいてモデル化を容易にする重要な要素に、適用業務の詳細部分を論理記述する技術範囲の増加と共に 20 種類程度に達した。ある発展段階の例を Tab.1 に示す。

記号全使用量の
80%を
占める

タイプ 記号	意味・機能・使用法
A	Add or All=1の意味、主に上辺ブロックの合計に使用する。
B	Bundle, Bit or Bool代数の意味; 上辺ブロックの数値の集約、順序変更、選択等に用いる。記号Bに続いて固有名を指定する。数値:0,1を登録可能。
C	Conditionの意味、記号Cに続いて固有名を指定し、条件式を登録する。
D	Dummyの意味;何も計算しない。複数の入力ブロックを参照する計算タイプ記号 P, C, *, /, %, Xと同一行に使用し、計算の同期化、およびMother Table上で同時使用の明示記号として使用する。
E	Each or Eigen (Unit Matrix)の意味: 対角線上の数値が全て 1 の働きをする。後続の括弧内に倍数を指定可能。記号Eに続いて固有名を指定し対角線に数値を設定し保持できる。
G	Group分配の意味: 記号Gに続いて固有名を指定する。縦に加えて数値1になる比率値を保持する。
H	Horizontal Addの意味; 上辺ブロックが2次元テーブルの場合、第2軸、すなわち奥行き方向に加える。記号Aと方向が異なる。
M	1次元Matchingの意味; 上辺ブロックおよび左辺ブロックに随伴する各項目ブロックの項目名の文字列を指定し一致する位置に、数値 1 が存在するブロック (タイプ記号Bの如く) が仮想的に生成され計算される。 2次元Matchingの意味; 上辺ブロックが2次元テーブルの場合、第2軸、すなわち奥行き方向に随伴する各項目ブロックの項目名の文字列について、マッチングを行なう
P	ProductionRuleの意味; If, Then, Elseの条件式、実行文を登録し実行可能。Ifはタイプ記号Cとほぼ同じ、実行文は四則演算可能。
Q	Quota割当ての意味; 階層構造の各分岐ごとに異なる配賦基準を与えトップ数値を配賦する。記号Qに続いて固有名を指定し組織構造を上辺ブロックにトップ数値と配賦基準値を登録する。
R	Reciprocalの意味; 記号Rに続いて固有名を指定し、その中央ブロックに登録した数値を逆数に変換した数値し演算する。
S	Standard Useの意味; 記号Sに続いて固有名を指定しその中央ブロックに数値を登録する。数値に特別な扱いはしない。
T	Transposeの意味; タイプ記号Tに続いて指定した別のブロック名で数値テーブルを引出し、縦横を転置し演算する。 Turnの意味; 上辺ブロックが2次元テーブルの場合に使用し軸を入れ替え演算する。
V	Volute渦の意味; 記号Vに続いて固有名を指定し親子のツリー構造を登録し展開による必要量を算出する。
X	User Exitの意味; 記号Xに続いてユーザが開発したプログラム固有名を指定し計算実行する。
*	上辺ブロック相互のMultiplyの意味; 辺ブロックと乗算する。タイプ記号*に続いて直上の上辺ブロック以外の固有名を指定し、直上の上辺ブロックと乗算する。
/	上辺ブロック相互のDivideの意味; 記号/に続いて、直上の上辺ブロック以外の固有名を指定し、直上の上辺ブロックを割算する。
%	Prorate割掛の意味; 記号%に続いて直上の上辺ブロック以外の固有名を指定し、その数値の比率を算出し直上の上辺ブロックと乗算する。
@	線形Loopの意味; タイプ記号@に続いて固有名を指定して親子のツリー構造を登録し、収斂するならば展開による必要量を算出する。

法としてのタイプ記号がある。タイプ記号は適用業

Tab.1 主なタイプ記号の一覧表²⁾

その基本的な考えは、中央ブロック名の先頭文字 (キャップ文字ともいう) を「タイプ記号」として付加して、中央ブロックに保持するデータの持つ性格付けを担わせ、同時にユーザに記号の意味を明確に伝えることである。さらに、この考えを発展させ具体的なデータの保持の有無とは異なる観点から関数的な役割をするものもある。表 1 に主なタイプ記号を列挙し、その働きについて説明する。これらタイプ記号の具体的な使用法については、外山論文(参考文献)を参照されたい。

しかし、経営管理モデルの大部分が、この記号のみで記述できることは驚異的である。これは、詳細に眺めると、それらの大部分は四則演算に帰するも

ので複雑に構成されていることによる。

その理由は、財務計算であれば、加減算で大部分が構成され、縦積・横和で吸収される。原価計算では、加減算、掛け算に加えて割り算が大部分を占めるが、割り算はソフト上の工夫で逆数の掛け算に帰し縦積・横和で吸収できることによる。さらに、親子の関係、オプション項目の抽出、関係箇所とのリンク等もマトリクス演算に帰すことができるためである。

注目すべきは、M: Matching が使われる頻度が全体の 80% を占めることである。これは、複雑なモデリングの各段階において、多元的要素を共通項で抽出し纏め、あるいは共通で取り扱うべき多元的場所に集積することを扱っていることを示している。

構造マトリクスの連鎖

複数の構造マトリクスを相互に連鎖するには、次の Fig.7 にも示すような要件を考える必要がある。

- ・複数の構造モデルの自由な結合・組替・組外し
- ・予期しなかったモデルの連鎖
- ・モデル間の任意の箇所間での大量データの簡潔な受渡し
- ・専門・担当を異にする人々間の Collaboration
- ・連鎖相手のモデル内容を詳しく知らなくても簡単に連鎖可能

このためには、変更に耐えうる自由度の高いタフな連鎖構造が必要であり、構造間→（左辺ブロック

→上辺ブロック間）→項目間へとリンクの関係を順次にテーブルを経て受け渡し最後にデータ受け渡し実現している原理をFig.8に示す。

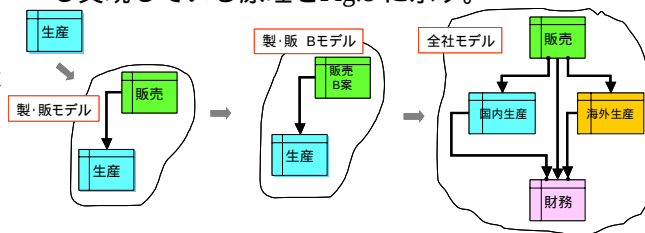


Fig. 7 構造マトリクスの連鎖要求の発生

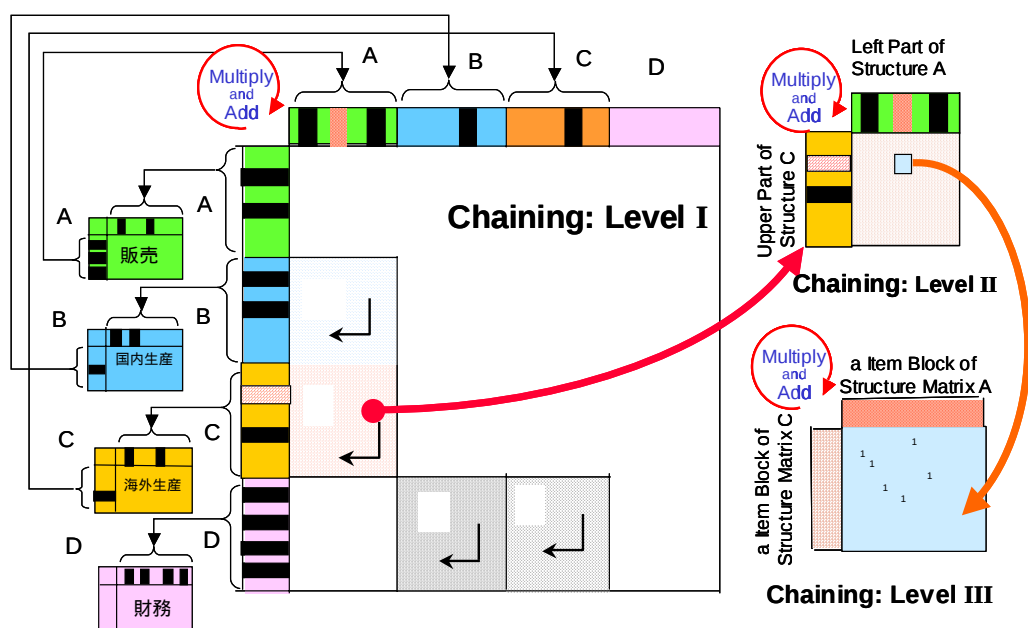


Fig.8 連鎖のアーキテクチャーと受渡の関係の設定

おわりに

このようなアーキテクチャーを構成することによって経営管理システムと情報処理システムにまたがる分野で、次のようなシステム要件を満たす Platform としての道を開いたと信ずる。

- ・縦横の関連発送からモデリングへ
- ・俯瞰から部分へ到達 Visibility
- ・複雑で多元的関連のモデル化
- ・エンドユーザーの関与できるモデル
- ・Black Box の回避、Programming の回避
- ・数値データの由来の追求 Causal Traceability
- ・柔軟でスピーディな変更への対処
- ・モデルの共有化と巨大連鎖およびその組み替え（プログラミングでは一般的方法では、ほとん

ど不能）

- ・項目の追加・変更への対処によるプログラム見直しの回避（当内容については省略）

これによって、MIS の悲願ともいえる 3 要件

- ・企業人の関与できるスパンを拡大する
 - ・企業人の相互理解の共通基盤を提供する
 - ・理解し得るならば、到達しうる時間を短縮する
- について、論理とデータが明確に繋がる輻輳した分野の解決手段に見通しを拓いたものといえよう。

これまで、複雑なシステムを分析する場合、データ中心に進め、適用業務のアルゴリズムには距離を

置いてきた。これに対しても一石を投ずるものである。ようやく仕組みをユーザのものとし、共有化し、ユーザ管理下におき、ユーザ自身によるモデル開発・論理確認・修正により大規模経営シミュレーションを運営可能としつつある。

また、これまで、業務の縦割りの弊害が叫ばれて久しいが、この分野に対しても、道具を提供するものである。

さらに、これから途方もなく複雑な分野に挑む生命科学に対してもヒントを与えるものとなる。

21世紀は、複雑で多元なシステムに挑むために、Node Oriented から Link Oriented に Paradigm Shift が望まれる。

注意：

もともと、原価が絡むとモデルが複雑となって専門家以外は立ち寄りがたくなる。-----例えば、経営管理で原価低減、サービス原価取り扱い、原価見積もり、制度原価システムの構築など---- この Platform 上の表現も、複雑度が高くなる。

この発展的解決のためには、アクティビティと原価の構造を明確な論理で表現できる新しい理論と技法が不可欠である。

もともと、構造マトリクスは複雑な原価システムの表現から出発しその一般適用性を追求して発展してきた。その内容については、当コンファレンスの外山 別発表論文「非金銭活動と金銭価値を精緻な構造で結ぶ基本——複雑で多元的原価作用因の求源可能なモデル化理論とその拡大」を参照頂きたい。

参考文献

- 1)外山啄之、飯川雅孝、構造マトリクスによる経営管理システムへのアプローチ、オペレーションズ・リサーチ、Vol.27 No. 7 1982 p402-408
- 2)外山啄之 他、構造マトリクスにおけるタイプ記号の諸性質とビジネス構造の表現に関する一考察、経営情報学会誌、Vol.9 No.1 June 2000
- 3)外山啄之、非金銭活動と金銭価値を精緻な構造で結ぶ基本、第1回横幹連合コンファレンス, Nov. 2005
- 4)Takayuki Toyama, Keiichi Endoh, Table- Driven- Tables -Processing for DSS beyond spread sheet, ISDSS, 3-4/1996, p355-380
- 5)Takayuki Toyama, Business Structure Modeling Based on Structured Matrix, Dr. Paper, Tokyo Institute of Technology, March 2000