

意識処理と無意識処理の相互作用のモデル化と 並列論理型言語によるシミュレーション

(財) 新世代コンピュータ技術開発機構

岡 夏樹 oka@icot.junet

1. はじめに

人間の認知機構は主に次の2種類の方法でモデル化されてきた。神経回路網類似のネットワークに基づくモデル化と逐次的な記号処理に基づくモデル化とである。前者は入力からの自動的な処理や連想的な処理を説明でき、後者は意識的な問題解決・推論を説明できる。それらの中間段階の処理のモデル化の必要性は指摘されている[辻井]が、十分なものはなかった。著者らは、それら中間段階の処理とそれを伴った問題解決・推論を、意識処理/無意識処理の観点から包括的に説明することを目指し、以下に述べるような認知モデルの作成に着手した。

2. C/Uモデル

意識処理/無意識処理については種々の観点があるが、著者らはまず次のような処理のモデル化から始めることにした。以下ではこのモデルを「C/Uモデル」(Conscious/Unconscious Model)と呼ぶ。

- (1) 入力処理の比較的後の段階は意識的な/無意識的な予測や期待の影響を受ける。
- (2) 意識に浮かぶ知識は外界からの入力や先に意識された内容に関連するものに無意識的に絞られるので、意識的な問題解決・推論は、それに必要な関連する知識だけを使って効率良く行なわれる。
- (3) 意識的に行なっていたことが、熟練により自動的に行なわれるようになる。

図1にC/Uモデルおよびその周辺の説明図を示す。外界からの入力は、《意識処理》の影響を受けない《入力モジュール》(これはC/Uモデルの範囲外) [Fodor] でまず処理され、その結果がC/Uモデルの《非記号的無意識処理》の部分に入力されることにする。ここでの処理は《意識処理》、《記号的無意識処理》および《感情、欲求、気分など》(これもC/Uモデルの範囲外) の影響を受けて進む(1)。その

過程で必要なら意識に対して注意を要求し、入力処理の結果の一部は意識に上る。

図1のC/Uモデルの上半分は記号処理を行なう。このうちの特殊な部分として《意識処理》があり、残りは《記号的無意識処理》である。C/Uモデルの下半分と対比するときは、両者をまとめて《記号的処理》と呼ぶ。なお、意識処理の一部であると思われるイメージの扱いは、ここでは除外する。

《記号的処理》は並列に行なわれているが、ある時点ではそのうちの最大で1つが《意識処理》される。どの処理が意識されるかは種々の注意の要求により時分割的に切り替わっていく。《意識処理》からアクセスできる知識は《非記号的無意識処理》により活性化されているものに絞られるので、意識的な問題解決・推論は、それに必要な関連する知識だけを使って効率良く行なわれる(2)。このように《非記号的無意識処理》は入力処理だけに限らず、連想的な知識の活性化に使われる。

《意識処理》の特徴は短期記憶を使って、やり直し、判断の保留など比較複雑な処理ができることにある。時分割的な処理も短期記憶により可能になっている。一方《記号的無意識処理》では、決定的な処理が自動的に行なわれる。たとえば熟練により活性化の結果に選択の余地がなくなり決定的になった処理(3)はこの一例である。このような処理でも通常と違うことが起こり失敗したら、意識に対して注意が要求され、《意識処理》されることになる。

意識/無意識という言葉はいくつかの意味で使われる(例えば[Klatzky][坂野])。まず、自発的意識(意識がある、意識不明などの言い方に対応)は、C/Uモデルにおいては《意識処理》が行なわれている/行なわれていないことに対応する。いま進行中の情報処理のモニターとしての意識は、《意識処理》の内容に対応する。何を意識していたか報告できるのは、短期記憶に最近の意識の内容が残っているからで

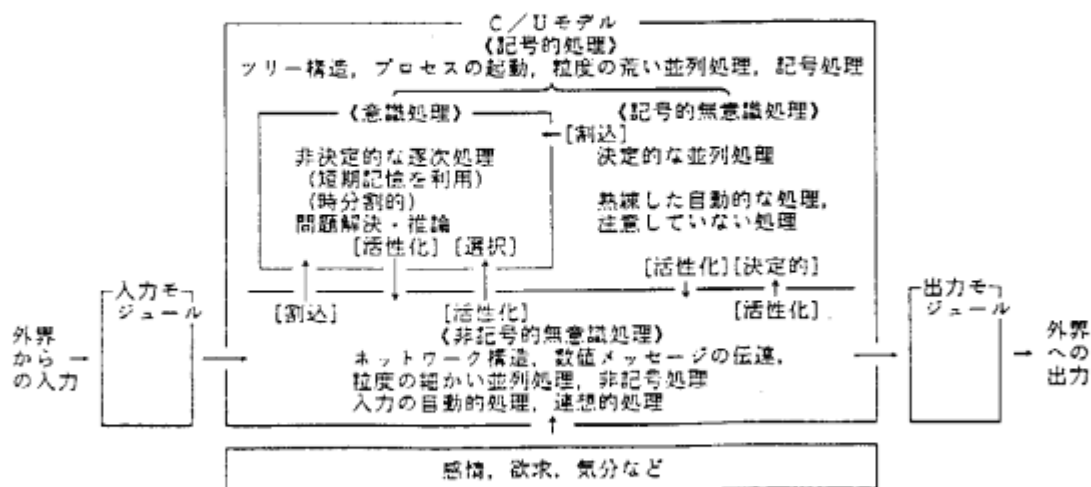


図1 C/Uモデルの構成

ある。認知的意識（記憶内容についての意識）は、まだ本モデルでは扱っていない。一般に無意識と呼ばれるもののうちで、熟練した技能や注意の向けられていない過程は《記号的無意識処理》に対応し、意識処理の下に伏在する非言語的な処理は《非記号的無意識処理》に対応する。その他に《感情、欲求、気分など》や《入出力モジュール》での処理も無意識と呼ばれるものの一つである。

3. 並列論理型言語によるシミュレーション

以下では並列論理型言語の1つとしてGHC[上田]を採用し、その特徴を生かした自然なシミュレーション法を示す。記号処理（図1のC/Uモデルの上半分）と非記号処理（下半分）との対比をGHCでは以下のように表現することにする。記号処理はAND/ORツリー構造におけるプロセスの起動で表現し、非記号処理はネットワーク構造における数値のみのプロセス間通信で表現する（図2）。

《記号的処理》と《非記号的無意識処理》とはAND並列で実行する。《非記号的無意識処理》としてのプロセス間通信は、GHCのプロセス記述言語としての性格を利用してストリーム通信として簡潔に記述できる。この部分については、ほぼコネクショニストモデルをGHCでシミュレートしていると考えてよい。

一方《記号的処理》自体もAND並列で実行される。《意識処理》はある時点ではそのうちの1つを逐次実行するが、さらに種々の制込要求（たとえば《非記号的無意識処理》からは入力処理プロセス実行の要求、《記号的無意識処理》からは割り当てられた処理の失敗によるやり直しの要求など）が出される。《意識処理》はそれらを時分割的に、物理的には1個のプロセッサで実行する。時分割的実行は短期記憶を利用して近似的に実現される。熟練による自動化は、他のプロセッサへのプロセスの割り当てにより表現する。

《意識処理》において、《非記号的無意識処理》により活性化された知識のうちの1つが意識に浮かぶことは、GHCのコミット操作（ガードの実行が成功した節のうちから1つを選ぶ操作）とそのまま対応させることができる。すなわち、《意識処理》の候補節のガード部で、その節に対応するプロセスの活性度を検査することにより、活性化された知識に相当する節のガード部は成功し、それらの中から1つの節が選

ばれ、この選択のやり直しはできない(choice nondeterminism)。このようにGHCでは基本的には後戻りはしないのであるが、意識処理の引数の1つとしていくつかの状態を持ち回ること（短期記憶に相当）により、必要なら近似的にバックトラックすることが可能となる。

GHCの基本的な機能にどのような機能を付加したかという観点で以上をまとめると次のようになる。コミット操作とAND並列の実行（《記号的処理》に相当）を基本にして、プロセス間通信（《非記号的無意識処理》に相当）を候補節の非記号的な絞り込みのために使い、短期記憶を伴ったメタプログラム（《意識処理》に相当）により、近似的バックトラック等の比較的複雑な処理や外部との逐次的コミュニケーションの手段を実現した。

4. おわりに

小さいいくつかの例題についてモデルの試作・動作実験を行なっているが、紹介は別の機会に譲る。なお、学習機能は現在のところまだなく、知識はあらかじめ全部与えている。

今後の課題は、中規模の例題のシミュレーションによりC/Uモデルの妥当性・有効性を検討しモデルを改良すること、C/Uモデル構築に必要な知識獲得・学習方式を検討すること、意識処理と無意識処理に関連した他の現象も説明できるようにモデルを拡張することなどである。

<参考文献>

- [Fodor] Fodor, 伊藤 他訳, 精神のモジュール形式, 産業図書, 1985.
- [Klatzky] Klatzky, 川口 訳, 記憶と意識の情報処理, サイエンス社, 1986.
- [上田] 上田 他, 並列論理型言語GHCとその応用, 共立出版, 1987.
- [岡] 岡, 意識処理/無意識処理の認知モデル, 日本ソフトウェア科学会第4回大会, 459-462, 1987.
- [坂野] 坂野, 意識とはなにか: フロイト=ユング批判, 青木書店, 1985.
- [辻井] 辻井, 知識の表現と利用, 昭晃堂, 1987.

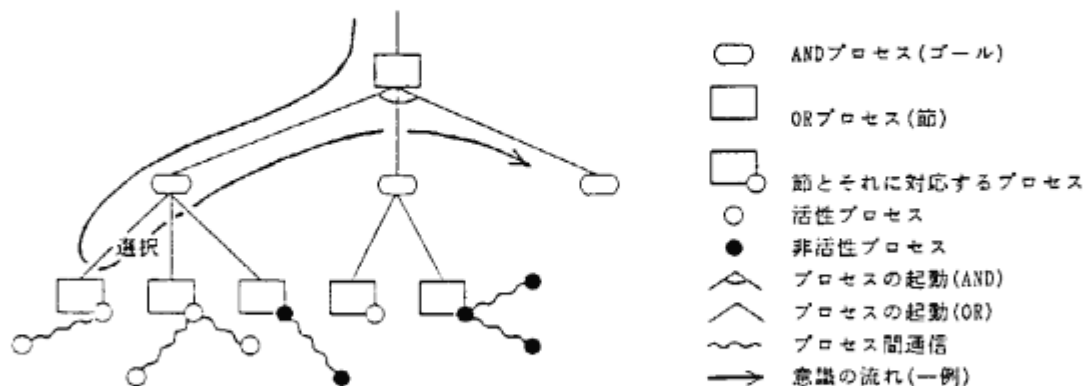


図2 C/Uモデルの計算木