

1. はじめに

汎用人工知能の実現は PC クラスタを想定しているのですが、そのとき、大枠の領野(意志、思考とか行動とか)は PC で、その通信は TCP/IP です。その PC 内部で、複数の領野・・・これは学習によって増えていくことが予想されます、・・・その通信は共有メモリかスレッド・プロセス間通信です・・・その通信ネットをプログラム(プロセス)からの依頼で動的に作り上げるという仕様が必要です。プロセス間通信を人間が手でコーディングするのは簡単ですが、プロセスの指定で造るとなると、結構めんどいです。規約造らねばなりません。通信するプロセスと作業するだけで通信しないプロセスに分類するとかの作業もあると思います。

「山へ登る」という意志が発生(意志領野で)したとして、その時、何が汎用人工知能に起きるかというようなことを考えて行く必要があります。思考領野に意志領野から「山へ登る」というコマンドが発行されます。意志領野には、どういう具体的な行動が起きれば満足いくかというような評価項目が発火するでしょう。それは思考領野との相互関係にある評価項目群です。思考で、満足をキーに、今までの経験とか行ける場所とか、日時とかの評価が成されます。思考の結果、「城山に登ろう」と言うことになったとして、そのコマンドが意志システムに通知されます。そこで、満足ならば決定事項としてコマンドがキープされます。意志領野は直ちに思考領野へコマンドを発行します。思考領野では、コマンドに関連する経験イメージ群から関連するデータを発火します。その思考の結果として、過去の経験イメージから意志のコマンドに最も評価の高い項目を選択して、結果、城山へ行く準備していくことになるわけです。行動と思考の領野もやりとりします。そのときに重要なことは、領野を越えてやりとりするのはコマンドだけであること。データは領野内部にとどまり、そこで、評価をうけて、パターンマッチングのみを行うということです。例えば、準備といってもどの手段で城山に行くかといったこととか、何時行くかといったこと、持って行く物は何かといったことの決定があります。それらの行動の選択はまず、意志領野へ決定事項の選択とか計画立案とかのコマンドを発行し、意志領野で、優先順位を決めて思考領野にコマンドで指示することになります。思考では、行動領野へのコマンドを発行して、例えば地図を取りに行くというような行動を指示します。

領野をまたぐデータはコマンドで、各領野で最適(意志領野と関連が深い、感情とか評価の領野の支持のもとに最適解を決定します)な処理を選択しながら段々に具体化した行動へと収束していくことになります。意志から行動へと繋がるアンサンブル処理が汎用人工知能の基盤となります。

2. 領野の構成

汎用人工知能は領野群と領野内のサブ領野の集まりで実現されます。各領野やサブ領野は、固有の機能を実現しています。

領野には次のものがあります。

- (1) 世界座標空間
- (2) 身体座標空間
- (3) 感情・評価機構
- (4) 意志
- (5) 視覚
- (6) 聴覚
- (7) 体性感覚
- (8) 言語
- (9) 思考
- (10) 行動

2. 1 世界座標空間領野

外界に広がるオブジェクトの配置関係を時間と空間の世界座標を1つ定めて管理します。配列の組み合わせであり配列のエントリはオブジェクト名や関連情報を記号化したものになっています。オブジェクト本体の画像や動画像、解析結果の塊データはそれぞれが分担する領野に留め置き、名前でこの世界座標空間から参照されます。汎用人工知能の世界を認識するための基盤となる領野です。

最初に基準となる配列を2次元配列と3次元配列で用意し、そこにオブジェクトの配置をしていき管理します。2次元配列が基本で、時間的に変化するオブジェクト配列イメージは配列（マップ）で表現しこの2次元配列に添付して管理します。オブジェクト自身の変化は画像の時間変化として所定の領野で管理します。3次元配列も部分空間として2次元配列に添付して管理します。

位置関係がさらに細密化していくと新たに配列を作り細密化のスケールを提示して細密の配置を管理します。

SLAMにより部分空間を切り出し、作業座標空間を作り、身体の姿勢変化に対応していく空間とします。作業空間からは定時に世界座標空間にオブジェクトの配置関係を反映します。

2. 2 身体座標空間領野

身体の姿勢を基準に世界座標空間の一部を座標変換して持ちます。作業用の座標空間とか視覚野の3次元認識機能の座標空間もここで管理します。内容は世界座標空間と同じ構成をしています。

この空間のデータは論理 SLAM (変換行列ではなく、大体の傾向としての配置をもって、対応しながら行います) によって世界座標空間領野に反映されます。

身体の姿勢を管理します。そのため、体性感覚領野をサブ領野に持ちます。

2. 3 感情・評価機構領野

全ての感情と評価の基本はメリット (優位性の獲得) とデメリット (障害の発生) です。そこへ、相同性と相異性によって感情・評価の接地が行われ、交差法によって分類され、生まれるコマンドデータの集まりです。コマンドの接地は各領野の生データへのコンセプト名による参照です。ディープラーニングやセット処理 (交差法)、強化学習などの学習機構によって配列とリストとステートマシンによってデータエリアは表現されます。ステートマシンによる評価グラフは、例えば、幸福感という評価項目からメリットセンサーへのパスになります。評価のためのパターンマッチングは各領野の生データによります。そうして、それらの生データにはコマンドが添付されます。感情・評価機構領野のデータエリアはデータエリアの名前によって、意志領野などから参照され、関連する領野の生データでのパターンマッチングして感情の発生とか評価項目の結果をコマンドの形で得て、評価を必要とする領野に発行することになります。

「リンゴを食べる」というイメージがあったとして、それは思考領野にコマンドが発行されます。「リンゴ」は感情・評価機構領野にメリットとして評価するというコマンドがあります。その「リンゴ」がメリットであるのは、生活体験から感情・評価機構領野と視覚、体性感覚領野の生データとして記録されています。そうして、「メリットのあるものを食べる」ということで、「食べる」=「get」となり、最終的にメリットであるという評価になります。

評価のクラスターとしての調停場などのコマンド群が格納されて管理し一括評価されます。評価項目セットを調停場に連想させ、コマンドが調停場に入ると自動で評価プロセスが起動して、結果を調停場に返します。調停場では評価結果を解析して最適な方策を決定します。調停場は配列のリストになっています。配列は、調停場に入れるプロセス結果とその基となるコマンドを持ちます。そうして評価プロセスの結果もその配列に入れます。

2. 4 意志領野

コンセプト（画像とか音声とかのオブジェクトとか属性のイメージ）に番号を生成して振りつけます。

そうして、行動を起こす震源をコマンドで保持します。また、コマンドの発行のプランニングを感情・評価領野の支持のもと実現します。意志には規範のコマンドも保持していて、規範に則った行動を行うように常に監視します。

外界をモニタし、感情・評価機構領野による外界の快適さや発した感情を管理します。

2. 5 視覚領野

2次元静止画の大きさ向きを正規化した配列とその時間シーケンスを解析し、解析結果と解析前の結果を保持します。静止画は閉曲線とその中の部品群を基盤にオブジェクトを認識し、その正規化画像を重ね合わせて保持していきます。そうして、オブジェクトの相同性と相異性を弁別し、クラスタリングして特徴データとオブジェクト認識データとパターンマッチングデータを構成していきます。オブジェクトは3次元空間に配置され、世界座標空間に反映する基本データになります。

パターンマッチング用配列は、線分分布とか特徴図形分布を配列要素に保存することで、配列の近傍検索でパターンマッチングを実現します。もっとも近い要素の多い図形が求められる図形であるというアルゴリズムでパターンマッチングします。

画像とパターンマッチング配列とで、一つのイメージを構成し、イメージの時系列（リスト）で、動画像を表現します。

また、思考領野との共同作業として、コマンドへのイメージの埋め込みがあります。コマンドの引数に相当するイメージはこの視覚領野にあります。そのため、コマンドを自在に構成せきるとは、視覚領野でのイメージの組み合わせが自在にできるというものです。

視覚を基盤とする言語の獲得と表出もコマンドをサポートすることで、公差法とステータマシンの組み合わせで実現します。

コンピュータで画像解析をするとき、一度に一画素しか処理できません。これを、画像全体を一度に解析するという人間のような視覚解析を実現するという方法があります。FPGAで、画像の各画素に1群のプロセスを割り当て、画像全体を一気に解析するのです。

2. 6 聴覚領野

フーリエ変換と聴覚データの両方を解析し、解析結果と解析前のデータを保持します。他の領野からのコマンドでの指示でパターンマッチングして、必要なデータをコマンドで返すということを行います。配列とコマンドの塊が保持するデータです。

音源の3次元配置を解析で得ることになります。その結果は視覚領野との共同で、世界座標空間に反映されます。

音素を獲得します。それは聴覚データの重ね合わせと重ね合わせのクラスタリング（公差法）で実現します。あとはグラフ（ステートマシン）を駆使して単語や文、文章を学習していき、言語を獲得します。

2. 7 体性感覚領野

身体の大きさ、姿勢、動作の速さ、感覚の身体分布を身体座標空間の支持のもと解析し、結果を保持します。体性感覚領野は、身体座標空間領野のサブ領野として存在し、メモリを身体座標空間領野と共有します。

手足、胴体などの身体部品の長さや太さ、空間配置をセンサー群から得ていきます。そうして身体座標空間の基盤を構成します。体性感覚領野は身体座標空間領野のサブ領野になっています。

2. 8 言語領野

コマンド生成を管理します。コマンドの骨格をもとに、動詞と格、オブジェクト・属性を整理して記録します。コマンドのパラメータは記号であり、本体は各担当領野の中で、交差法により **and,or** セット操作によって造られていきます。動画像の、どのイメージ部分が「歩く」という意味を持つかというようなことを記録していきます。

また、コマンドを深層表現とし、自然言語である表層表現と対応を取り、ステートマシンとして機能するデータエリアをもって、言語認知と言語表出を行います。

2. 9 思考領野

教示や汎用人工知能自身の試行錯誤による行動のトレースをステートマシン（記号世

界のコマンドセット)で保持し、飛び飛びにあるステートマシンを繋いで、一つの行動(思考行動)に纏めることを行います。ステートマシンの選択や結果の項目評価を感情・評価領野の支持のもと行いつつ思考制御していきます。各時点で、項目が必要になれば、直観システムというプロダクションシステムによって候補の検索と評価を自動で行います。直観システムの結果は調停場という評価空間に展開します。調停場のデータの本体は本体データの所属する領野にあり調停場は名前で参照するだけです。

5円玉を認識する視覚領野での解析では、5円玉の形状とオブジェクトの認識しかしません。「5円玉の属性として穴がある」という風な解析能力はありません。「5円玉の形状を配列で表現する」だけです。その解析では、「円オブジェクトとその真中に穴オブジェクトがある」と認識します。

思考領野で、5円玉が10枚あって持ち運びたいというコマンドが発生したとします。そのとき連想するのは、5円玉に穴オブジェクトがあることから紐を通せばまとめて持ち歩けることです。そのように、オブジェクトとその属性を操作することが思考となります。

2. 10行動領野

思考と同じく、ステートマシンによって管理します。教示や汎用人工知能自身の試行錯誤の手足の動きを記録して、意志領野と思考領野からのコマンドの発行によって動きます。

身体座標空間領野との関係も深いものがあります。身体座標空間領野のデータは記号によって行動領野に伝えられます。コマンドです。この意志の下、ステートマシンでの予測とセンサー値による行動結果の確認を行いつつ、ステートマシンの状態の変化パスを修正して行動を完遂します。

3. 思考の実現と行動

汎用人工知能の実現するメリットの主な物は「思考」の実現にあります。ここでは、いくつかの例を挙げて思考のプロセスを検討してみます。

窓の外を見ていたら、雨が降ってきました。雨が降ってきて困ること（デメリット項目）を今日の生活から思い出します。「母が美容室へ出かけた」という項目を思い出し（エピソード記憶（思考領野にあるコマンドと、視覚野にある動画像で構成する）から）、傘を持っていかなかったことを発見（直観システム）し、調停場に置きます。このときデメリットである評価を感情・評価領野から得ます。デメリット解消の状況を直観システムによって調停場に展開します。評価として傘が有れば良いという項目が展開します。思考のステートマシンで、汎用人工知能が傘を母に届けるというコマンドを起動し、その実現性を感情・評価領野で評価します。「よし」ということで、傘を届ける行為を意志領野にコマンドで思考領野から発行します。

部品のピッキングや衣類の折りたたみなどの精密な制御が必要な行動は思考が働きます。それは、ピッキングですとロボットの指と部品の配置がマッチする関係は明確に定義できます。今あるロボットの腕と指の配置から指と部品形状がマッチングする点まで、腕と指を移動させるのはステートマシンによる思考の下に行われます。ステートマシンはとびとびに作られていますが、そのステートマシンから次のステートマシンへの移行は、ある種の大枠のステートマシンの制御によります。そのステートマシンの選択というものは学習とシミュレート（予測）による思考が為すものです。

衣類の折りたたみも同じです。腕と指の動きと布の形状の変化は断片的に定義されてるだけです。その断片を埋めて、制御の筋を通すのは思考です。

4. 人手によるブートシステムの構築

汎用人工知能は決められたデータのみを使うので、汎用人工知能に渡すものは汎用人工知能のデータエリアを用意していくことで、人手で構築した部分と汎用人工知能が自律・自己組織化能力で構築した部分がシームレスに機能することが保証されます。コンセプトも例示データを汎用人工知能に提示すれば、記号で返してきます。そんな仕組みで、人手で構築した部分システムも汎用人工知能は飲み込んでいきます。

技術者と汎用人工知能の間のデータの融通を滑らかに行えるようにするため、データエリアには規則が必要になります。

- (1) コンセプトの番号を定める（汎用人工知能が自律で生成する）。
- (2) 領野の番号を定める。基本領野の名前であります。
- (3) サブ領野の番号を定める。領野内のデータエリアのタイプやプロセスによって番号を名前として定めます。

データエリアのタイプは次のものが考えられます。

- (1) 配列
- (2) リスト（セット）
- (3) グラフ（ステートマシン、コマンド）

5. おわりに

領野を越えて通信するデータはコマンドのみであることが分かります。画像データなどの重たいデータエリアはそのデータを獲得した領野に留め置き、利用されるように設計できます。データエリアを獲得する学習機構とか利用していくパターンマッチングの処理は各データエリアが所属する領野で行うことになります。

サブ領野は共有メモリのプロセス環境であり、サブ領野は動的に汎用人工知能が自律・自己組織化能力として設置できることが必須になります。

汎用人工知能は、真つ新たな状況では無力です。ある程度人手で能力を構築する部分が多数あるでしょう。人手で構築された能力と自律・自己組織化能力で獲得していく能力がシームレスに移行できなくてはなりません。そのため、パターンマッチング能力などのオントロジー機構は明確に規則として明文化しておかねばなりません。

おわり