



ニューロ・チェーン：知能的なブロック・チェーン

Eukléia 総意
完全性と参加の証拠（P I I）
ワークフローの証拠

ニューロ・チェーン研究所

2017年12月3日

メール・アドレス: contact@neurochaintech.io

概要：本記事は、ブロック・チェーンのような分散されたシステムをベースにし、機械学習アルゴリズムによって可能となる新技術について記述します。ニューロ・チェーン技術は、ブロック・チェーンと機械学習の融合であり、以下の3つの項目を柱にしています。

- 意思決定者：ボットの鎖
 - 一連のルール：意思決定プロトコル（完全性と参加の証拠及び、ワークフローの証拠）
- ネットワーク及びメディア：現実的な通信チャネル（適応型通信プロトコル）と学習エコシステム

ボットは、ロボットの省略です。本記事のコンテキストでは、ネットワーク内の特定ノード上に動作する人工知能のことを指します。ボットは、ネットワークとプロトコルの上に実装された、機械学習アルゴリズムによって可能になります。ここで鍵となるのは、これらのボットが連結して動作することです。ボットの鎖がネットワーク上に分散されます。その他に、重要となることは、一つのボットは一つのノードに「設置」することです。ボット達がお互いに通信し、取引の検証を行い、セキュリティーや透明性を保証して、分散型インフラの問題（例：二重支払い問題、ビザンチン将軍問題等）を解決します。意思決定プロトコルは、2つの数学的ツールで構成されます：完全性と参加の証拠（P I I）。参加の証拠は、ネットワーク上の特定のボットが実現したエフォートを逆追跡します。このエフォートは、ボットのエントロピーとエンタルピー（価値の量/内部エネルギー）で測定されます。中心的な考えは、重要なボットが重要な要求に応え、より高い手数料を加担します。完全性の証拠は、ボットの信頼性を示します。現実的な通信プロトコルによって、企業や個人による大規模な導入を可能にします。

合計として、自律的に動作するボットの集まりは、意思決定の分散型エコシステムのメンバとして働きます。よって、システムは、集団の協業の人工知能のメリットを受けます。知能のシェアリングは、ワークフローの証拠プロトコルによって保証されます。この動的なシステムは、意思決定プロトコルのおかげで、自動的に素早くそれぞれの役割を果たすことができ、改ざんを防止するため、悪意のあるボットは、集団からの罰を受けることになります。

ニューロ・チェーン技術は、複雑な分散型アプリケーションを可能にします（例：トレーサビリティ問題、暗号化価値、知能的なアプリケーション、スマートプロパティ、ソーシャル・ネットワーク、

また、信頼性のある分散プラットフォーム)。ニューロ・チェーンは、ブロック・チェーンの新しい波であり、分散型経済の各制約を省いていきます。

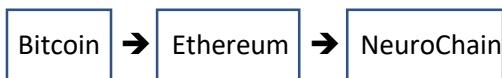
語源定義：ポットは、ロボットの省略であり、自動動作するタスクの上で稼働する、ソフトウェア・アプリケーションのことです。ニューロ・チェーンは、緻密的で複雑な特定のタスクを解決することを目的にした、ポットのつながりで構成されます。このようなアーキテクチャーは、**集団人工知能**のエコシステムを形成します。

ニューロ・チェーン：脳のニューロンの接続に因んだ、ニューロンの鎖、またはポットの鎖

クラウドジウス：価値の測定の基本標準

キーワード：ニューロ・チェーン、ポット・チェーン、ブロック・チェーン 2.0、粒子物理学、スマートブロック・チェーン、適応型通信レヤー、機械学習、ポット、分散アルゴリズム、エントロピー、エンタルピー、完全性と総意

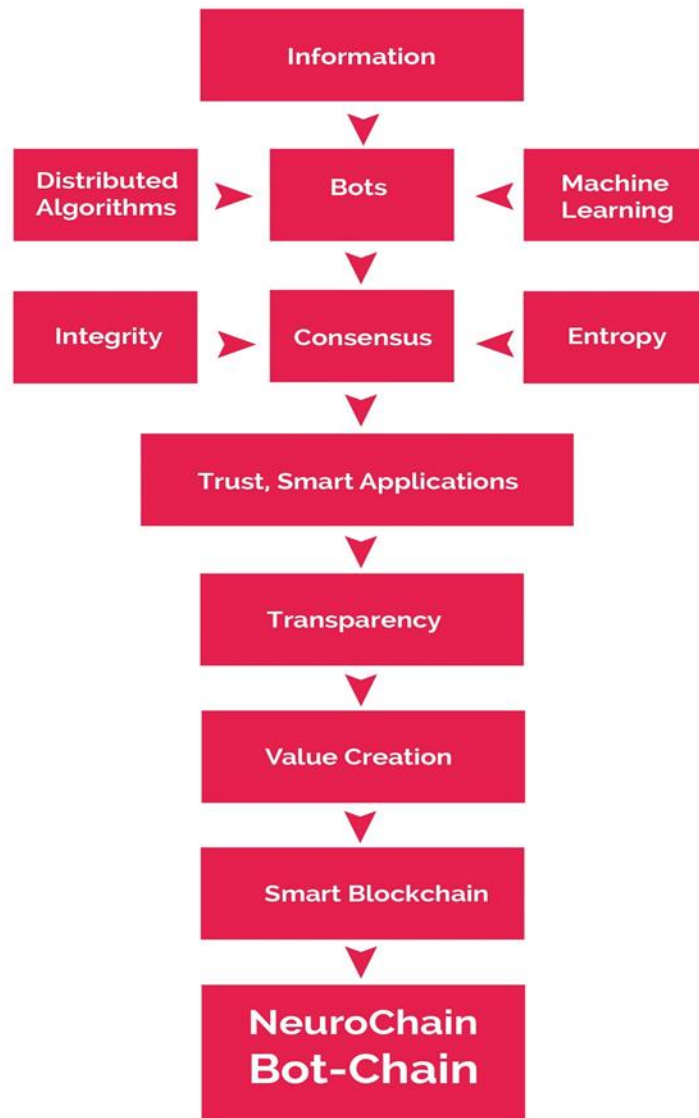
Technology evolution: 技術の進化

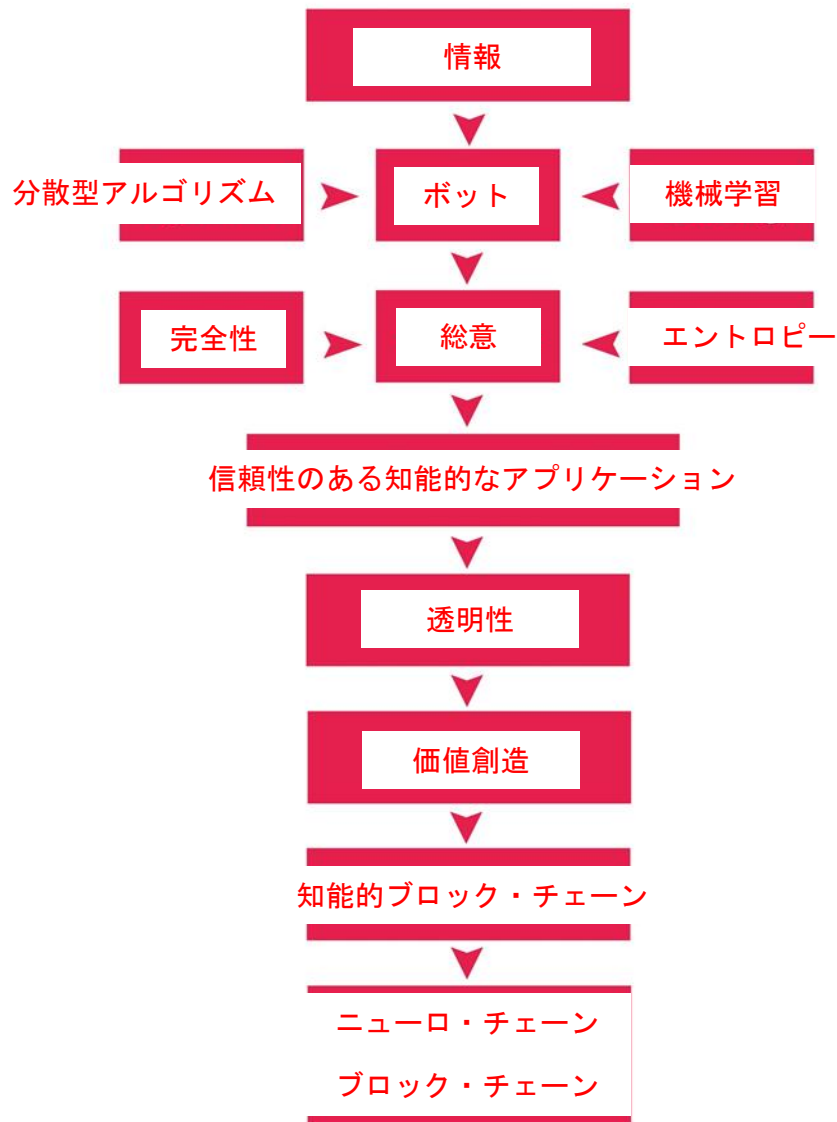


ビット・コイン → エテリウム → ニューロ・チェーン

ノート：ニューロ・チェーンは、現在進行中のプロジェクトです。集中的な研究と、実用最小限の製品の検証が、現在進行しています。したがて、この記事の改訂版と新しい成果は、以下のリンクから参照可能です：<https://neurochaintech.io>。ご質問やコメントがございましたら、こちらの連絡先まで、お問い合わせ願います：contact@neurochaintech.io。

Smart Representation: スマートな表現





上記図は、情報入力から価値創出までの、ロボットの解析と意思決定プロセスを示しています。

プロジェクトの始まり

ニューロ・チェーンは、ブロック・チェーンの潜在能力への長期的な関心と、現在のブロック・チェーンの制約から生まれたプロジェクトです。「ENI Editions」の手助けと編集への専念のおかげで、この関心は、2016年に出版された書籍の形で現れました。これは、ブロック・チェーン技術、その現状（ビット・コインやエタリアム）、制約、しかしこれらの上に、その素晴らしい可能性への絡み合った関心の、最初の表れです。提案されたプロジェクトは、現在拡大中のいくつかの定量的な分野の融合となります：粒子物理学、機械学習、分散アルゴリズム、意思決定プロセス。粒子物理学の視点は、ネットワーク本来の特性を説明するときに役に立ちます。機械学習アルゴリズムは、ノード（ロボット）の分析能力を改善します。熱力学のパラメータによって可能になった、高性能な総意達成方法は、巧みに分散され、取引量とリアル・タイム性（速度）の問題を解決します。定量的な融資は、よりフェアな分散メカニズムを実現します。

このような、バランスの取れた異分野の融合が、知能的な分散システムの始まりに導きました。ニューロ・チェーンの誕生です。

摘要：本文介绍一种基于例如区块链的分布式系统，由机器学习算法驱动。机器链技术（The NeuroChain technology）革命性的新型概念使得大规模分布式应用成为可能（可追溯性问题，数字化价值，智能应用，社交网络或者值得信任的分布式市场平台）。机器链新型的协议和实用的沟通渠道让合作者和领导者能广泛采用。由一条链分布式机器人组成的机器链构成了由集体和合作智慧为基础的决策和沟通生态环境。因此，机器链是一种智能或者说是智能区块链。

词源定义： Bot 是机器人“Robot”的简写，是一种自动运行的软件应用。NeuroChain 指派一条链的机器人来完成指定的合作任务。这种构造组成来一个集体智慧的生态结构。

Résumé: le document suivant décrit une nouvelle Blockchain basée sur de nouveaux protocoles de consensus et alimenté par du machine Learning et de l'intelligence artificielle. Cette technologie NeuroChain introduit de nouveaux concepts augmentés qui permettent la mise en place d'applications intelligentes complexes à grande échelle tels que la traçabilité, les crypto-Valeur, les applications autonome ou smart applications, les réseaux sociaux distribués et les applications métiers (réglementés et institutionnels). Les nouveaux protocoles de décision (**Proof of Involvement and Integrity & Proof of Workflow**) ainsi que la communication adaptative en fonction des performances vont permettre une adoption plus importante de NeuroChain par les entreprises et les particuliers. La NeuroChain est constituée par une chaine de Bots distribués (Bot-Chain) qui se sont accordés à structurer un écosystème de décision et de communication basé sur le concept d'intelligence artificielle collective.

Краткое содержание: в нижеследующей статье описывается новая технология, основанная на распределенной системе типа **блокчейн** и усиленная алгоритмами машинного обучения. Технология **Ботчейн** включает новые революционные концепции, которые делают возможным **масштабирование распределенных систем** (проблемы трассировки, криптоцены, умные приложения, социальные сети или надежные распределенные рыночные платформы). Новые протоколы принятия решений и прагматические каналы коммуникации Ботчейна открывают возможности по широкому применению технологии как корпорациями, так и людьми. Ботчейн, состоящий из цепочки распределенных ботов (роботов), позволяет создать экосистему принятия решений и коммуникации на основе коллективного объединенного интеллекта. Поэтому Ботчейн – интеллектуальный, или умный блокчейн.

Zusammenfassung: Das folgende Papier beschreibt eine neue Technologie, die auf einem verteilten System ähnlich Blockchain basiert und durch Machine Learning und künstliche Intelligenz angetrieben wird. Diese NeuroChain-Technologie stellt neue revolutionäre Konzepte vor, die den Einsatz von großformatigen intelligenten Anwendungen wie Rückverfolgbarkeit, Krypto-Währungen, intelligente Anwendungen oder Smart Application, verteilte soziale Netzwerke und Geschäftsanwendungen ermöglichen. Das neue Entscheidungsprotokoll sowie eine adaptive Kommunikation nach den Performances ermöglichen eine stärkere Annahme der NeuroChain durch die Unternehmen und die Einzelpersonen. Die NeuroChain besteht aus einer Kette von verteilten Bots (Robots), die vereinbart haben, ein Entscheidungs- und Kommunikations-Ökosystem auf der Grundlage des Konzepts der kollektiven Intelligenz zu strukturieren.

Contents 目次

プロジェクトの始まり	4
前書き	9
ニューロ・チェーンの運用	9
取引	10
ニューロ・チェーンの説明	12
プロトコル、総意、及び機械学習	16
総意：完全性と参加の証拠（P I I）	16
詳細には、アセンブリ構成内のボット数は 535 個のボットで始まり、ネットワーク運用 と合計重み付きエントロピーと完全性に応じて進化します。各アセンブリの検証の数は、 1638 ブロック（「黄金比」に関連する）から始まり、これはネットワークのパラメータ （ネットワークの完全性、フォークの確率、重み付き平均エントロピーの進化）に応じ て進化します。選出パラメータの動的プロセスは、アセンブリ構成のより頻繁なローテ ションを保証します。	22
分散型プロセス	22
機械学習	23
機械学習または人工知能という用語は、ボットの高度な抽象化を指します。様々な分類の アルゴリズムで構成されています。これらのアルゴリズムは、ボットによって様々なア プリケーションに使用されます：トレーサビリティの一貫性アルゴリズム、ボットが発信 した情報や取引の異常を検出のためのベイジアン・ネットワーク・アルゴリズム、情報交 換のための価格設定アルゴリズム、複雑なルール・ベース・アプリケーションなど。アル ゴリズムの認証は、IPFS プロトコルと暗号署名を使用します。その処理を以下に説明し ます。	23
ワークフローの証拠	24
現在の主要なブロック・チェーンとの比較	25
ニューロ・チェーンアプリケーション	26
ガバナンス	29
検疫されたボット	30
改正[フォーク]	30
ボット補償	30
ニューロ・チェーン相互エコシステム	30
PoC の最初の結果とパフォーマンス	31
結論	32
付録	33

前書き

過去10年にわたり、分散システムの登場とブロック・チェーン技術をベースにしたビット・コインの成功は、新しい交流と交換手段への関心と可能性を表しています。仲介者の排除とステークホルダー間の直接通信は、ネットワークにおける情報複製と検証プロセスを通じて、信頼性を向上させます。しかし、これらの分散技術、特にビット・コインのブロック・チェーンは、これらのプラットフォームの普遍的、つまりグローバルな特性を排除する、いくつかの本質的な欠点を持っています。つまり、各ブロック・チェーンは、そのプロトコルとその運用のために、独自のアプリケーションを持っています（主に暗号通貨）。最も有名で実績のあるブロック・チェーンは、暗号通貨プラットフォームであるビット・コインであり、取引発行者と取引承認者という、明確に分離した役割を持った不均衡なシステムから構成されています。ビット・コインは、当事者間の高い信頼性を確保「するため、ヘビー・プロトコルを採用し、トレーサビリティ、信頼できるソーシャルネットワーク、商用および知能的なアプリケーションなど、他のアプリケーションには、適していません。トレーサビリティは、食品業界、その他に責任、人的資源、商業取引といった商用的アプリケーションに応用可能で、バリューチェーンやプロセスの透明性を保証します。ニューロ・チェーンは、これらの透明性、認証、倫理の課題解決のために、開発されていますが、分散知能的なアプリケーションの支援も可能で、柔軟性を備えています。ニューロ・チェーンは、安全なチャンネルを介して通信し、アルゴリズムを介して協業する一連のボット（ロボットの略）からなる、新しいブロック・チェーンのプロトコルです。このプロトコルは、アルゴリズムによって適応可能な、様々な通信チャンネルを使用します。この適応性のある通信層の使用と、迅速かつ対話型の総意アルゴリズムの組み合わせが、セキュリティ、パフォーマンス、一貫性のすべての保証を維持しながら、ニューロ・チェーンの採用を容易にします。分散プロトコルの、障害許容アルゴリズムは、Paxos アルゴリズムに匹敵する、セキュリティと可用性をもたらします（Leslie Lamport, 2004）（Marshall Pease, 1980）。ニューロ・チェーンのアーキテクチャ[付録]は、様々な目的の取引の透明性を保証します：トレーサビリティ・プロセス、暗号値（暗号トーク）、知能的なアプリケーション（スマートシティ、スマート車両等）、ソーシャル・ネットワーク（ソーシャル・ボット）、分散プラットフォーム（商用アプリケーション等）。ニューロ・チェーンは本来、ビット・コインやエテリアムなどの、他のブロック・チェーンと協業可能で、グローバル環境に統合できるように、設計されています。

次の記事では、ブロック・チェーンであるニューロ・チェーンと、その概念証明におけるいくつかの結果と共に、説明することを目的とします。最初に、ニューロ・チェーンの運用と新しい総意コンセプトを紹介し、次に、応用可能ないくつかのアプリケーションを詳しく記述します。第2ステップでは、技術的な説明がその可能性に焦点を当てます。その後、様々な構成要素：総意、通信、機械学習が説明されます。最後にガバナンスの問題と最初の実験的成果が取り上げられ、結論で終わります。

ニューロ・チェーンの運用

全体として、ニューロ・チェーンは次のように動作します。ボットは、アルゴリズムの集合体であり、取引の実行と検証のツールでもあり、また価値とスマートなアプリケーションを構築するため通信手段です。つまり、レゴのように、異なった部分の組み合わせが、理解可能な概念を形成します。ニューロ・チェーンの最適なアーキテクチャは、ブロック・チェーンの適応動作を可能にします。取引は、通常の暗号化ブリック（暗号署名）、標準メンバーシップ情報、及び、（取引の有効を決める）適切な検証アルゴリズムを決定する、インタプリタから構成されます。すべての取引は、プールに集められます（検証後、かつ割り当てられた期間内に、取引ブロックを構成した各ボットが検証できた場合）[2]。その後、選出プロセスが始まります：最初に、Nブロックの検証用に、各ボットの次のパラメータ：重み付きエントロピー（GE Shannon, 1948, AN Kolmogorov, 1965）、完全性および評判（Myers, Zhu, and J. Leskovec 2012, Yang, Chen, and D. Agarwal, 2013）によって、委員会またはアセンブリ構成が設定されます。アセンブリ構成内ボットは、それぞれスコア（加重されたエントロ

ピーと完全性)に応じて、特定の数のビッグ・セレクターが割り当てられます。次に、ビッグ・セレクターは、ランダムに選択され[付録 5: 任意性と混沌としたプロセス]、それを持ったボットがブロック検証のために選出されることになります。他のボットは、関係するブロックを検証し、それをブロック・チェーンに統合することによって、選出されたリーダーに忠誠を示します。ニューロ・チェーンの一貫性を保証するために、最大累積加重エントロピー、および累積完全性(増加する状態量である)のブロック・チェーンが考慮されます。したがって、ボット集団は、単一の取引履歴に同意します(Nakamoto, 2009)。総意の詳細については、次のセクションで説明されます。

にもかかわらず、特定アプリケーションであるトレーサビリティについて、選出プロセスはチェーンでの参加をベースにし、リーダーは決定的に選出されます。これは、知能的なトレーサビリティアプリケーションの作成時に、リーダーが指定されたことを意味します。一般的に、リーダーはトレーサビリティアプリケーションの創始者と混同されます。その結果、検証プロセスは未支払いとなります。これは、証拠のチェーンが、ニューロ・チェーンの運用の基本的な基準である、価値と透明性を創出するためです(取引の検証は変わりません)。

このネットワーク上の価値創造は、強固な基盤を持ち、実体経済とのつながりを持つため、ブロック・チェーンは、真の価値を創出できるはずで、ニューロ・チェーンの「クラウドジウス」で定量的に表された価値の源は、主にシステムに注入された情報、検証、透明性によって生成されます。

様々なレベル(ボットとネットワーク)で異常や矛盾取引(Vandervort D)を検出できる、機械学習アルゴリズムによって、結果的に、セキュリティと一貫性が改善されます。このフィードバックは、各ボットのスコアに直接反映され、リーダーの選出に影響を与えます。

背景にあるアイデアは、システムがボットに、より高度な情報、完全性、透明性を提供しよう、動機づけます。実際に、ニューロ・チェーンでは、価値が情報によって表され、一般的に暗号化交換(クラウドジウス)と呼ばれる、暗号化価値といった、いくつかの形式で表示されます。ニューロ・チェーンによって作られた価値は、とりわけ、グローバル化によって生み出された、トレーサビリティと透明性の必要性に応えます。

取引

非対称暗号化プロセスは、ボット間の取引を安全にします。取引のシステムは次のようになります: 各ボットは、次のボットの公開鍵を使用して値を転送し、以前の取引にデジタル署名(ハッシュ)し、この情報を、現在の取引または価値に統合します。したがって、受信者は、署名をたどることで、価値の所有者の連鎖を検証することができます。ユニークな識別子は、ニューロ・チェーン(クラウドジウス)の生涯を通して作成された価値の、各「標準」単位を追跡可能にします。このメカニズムは、暗号化価値に適しています。

この構成では、すべての知能的なアプリケーションをサポートするために、インタープリタが追加されています。検証プロセスの一貫性を保証するため、特定の構造を持ったメタデータが、ボットに提供されます。つまり、インタープリタは、取引が有効となる条件を決定します。インタープリタは、入力によるパフォーマンスの割り当てと、通信層の適応にも使用されます。

この取引プロセスのアーキテクチャは、ボットの特定構造化を含み、知能的なチェーンの最初の部分を表します。図 1 は、ニューロ・チェーンの取引処理を示しています。検証プロセスは、インタープリタが加わった以外に、ビット・コインに類似します。

取引の重要な側面は、知能的なアプリケーションの複雑さ、アルゴリズムの複雑さ(A. N. Kolmogorov, 1965; Goldreich Oded, 2008)に応じて、**動的な料金**が加担されることです。ニューロ・チェーンのブロック・サイズと料金の設定の主な目的は、サービス拒否攻撃などの、ネットワークの直接攻撃を防ぐためです。

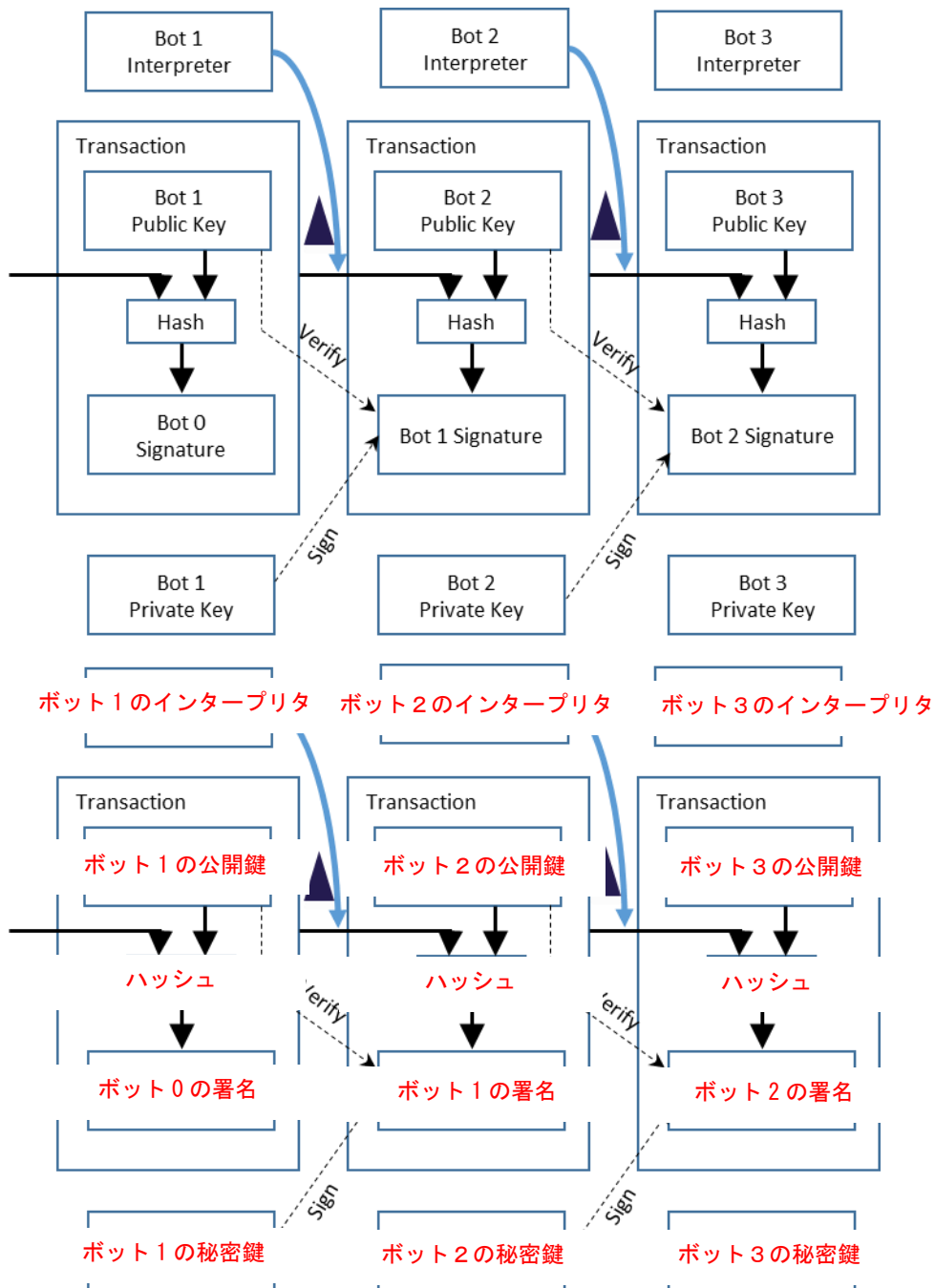


図 1：ニューロ・チェーンの取引処理の概略的なアーキテクチャ。取引の標準的な系列に、緻密な知的なアプリケーションを支援するため、インタプリタが追加されています（Nakamoto, 2009）。インタプリタは、取引が有効となる条件を決定します。

ニューロ・チェーンの説明

図 4 は、様々な機能を持つボットのトポロジを示しています。各ボットは五角形で表され、五角形の各面は、ボットの特定のタスクを表します。決定と総意は、分散プロトコルの永続性、正確性、活力を保証するための、ボットのコアとなります。通信ブロックは、様々なプロトコルで構成され、ブロック・チェーンの柔軟性と可搬性を実現します。五角形の学習側は、ブロック・チェーン運用上の事後フィードバックを可能にし、また、意思解析可能なソーシャル・ロボットや、外部アルゴリズムとリアルタイムな情報交換と解釈ができる、スマート・シティ・アプリケーションといった、特定の知能的な分析を可能にしたアルゴリズムで構成されます。ボットは、API を介して周囲の世界に接続され、様々なメディア（ウェブ、モバイルなど）に対応しています。また、他の既存のブロック・チェーンやエキゾチックな孤児ボット（独立したチャット・ボット）と接続し、自己の理解と解答を改善可能です。集約された構成は、多様かつ異なる用途、または知的なアプリケーションを可能にし、唯一の制約は想像力です。よって、ニューロ・チェーンは、知的なブロック・チェーンと言えます。

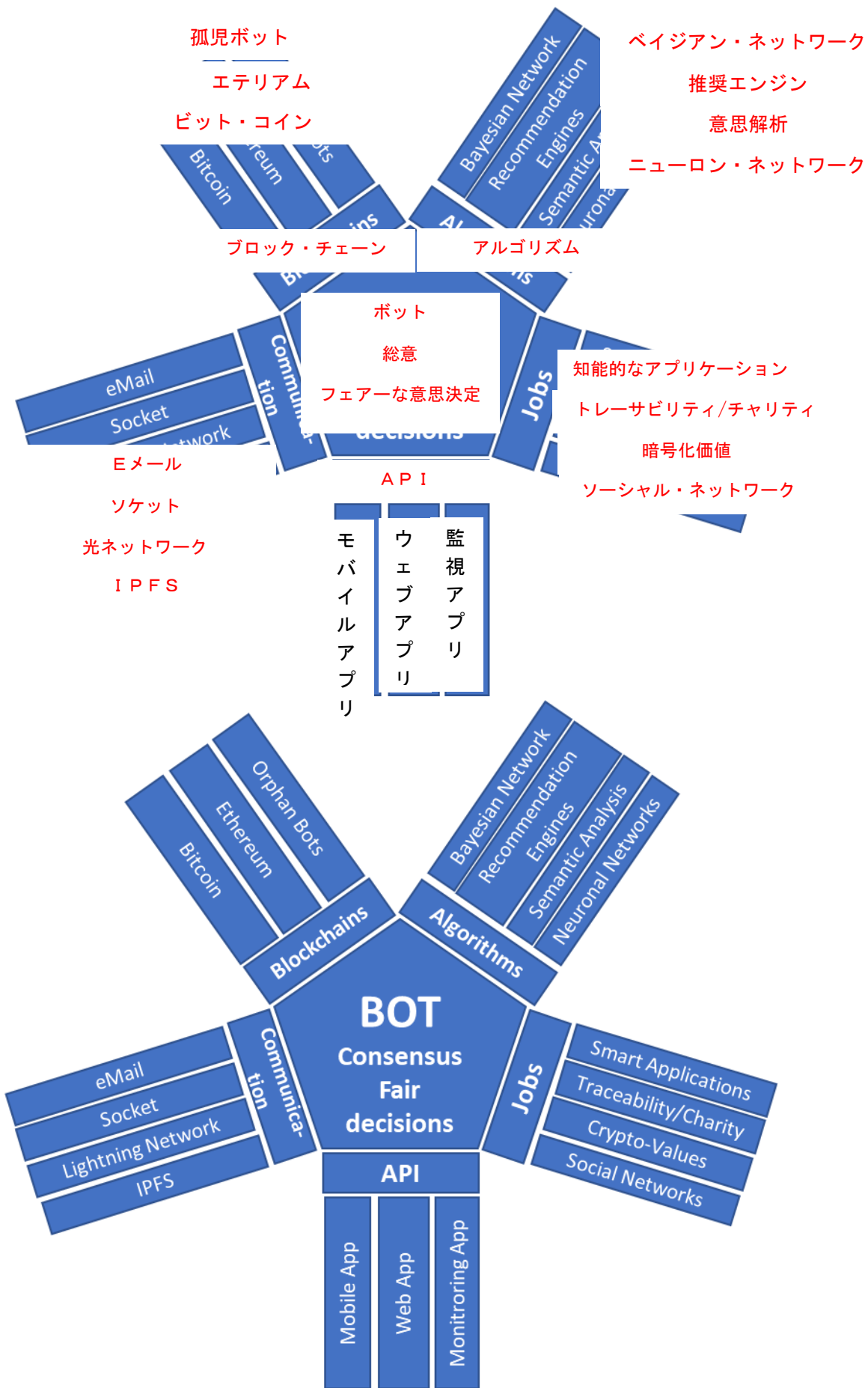


図 2：ニューロ・チェーンの概略的なアーキテクチャ。ボットの様々なタスクと、既存のシステムとの接続を示しています。適応可能な通信プロトコル、様々なジョブ、およびアルゴリズムに代表される知性を見ることができます。

ニューロ・チェーンを理解するためには、2 つの中心的概念を詳細に説明する必要があります：**分散されたボットと通信層**。

ボット：ボットは、分散システムのノードを表します。ボットは 2 つレベルの抽象階層で構成されています（小脳と脳に似ている）。第 1 レベルは異なるボット間の基本的な通信で構成され、第 2 レベルは、トレーサビリティ目的のための一貫性アルゴリズムのような、精巧なアルゴリズム等で構成されます。実際、知能的なアプリケーション、ソーシャル的な交流、またはスマート・ビジネス・アプリケーションのそれぞれのコンテキストで、異なったアルゴリズムを異なったタスクに使用できます。この場合、各ボットは、ネットワーク内の知的なエージェントとして機能し、その役割と仲間との交流に応じて動作します。アルゴリズムは、機械学習と人工知能から発行されます。分析と解釈のための適切なアルゴリズムは、インタプリタ（図 1）を通じて、明確に取引内で指定されています。

通信層：ニューロ・チェーンのもう一つの革新は、ボットの（推奨エンジンに基づいた）適応通信システムに関連します。実際、ジョブ実行のためにタスクとリソースが要求した、時間とリソースに応じて、柔軟でスケーラブル/進化可能な通信が採用されています。3 つの通信チャンネルが標準で利用でき（これは必要に応じて進化します）、TCP / IP モデルに基づいて、7 つのレイヤーで構成されます。また、ニューロ・チェーンの通信レイヤーが、ここに接続されています。

- A. SMTP (Simple Mail Transfer Protocol) (1995 年のメッセージサイズ宣言のための SMTP サービス拡張)：ビジネスネットワークとインターネット上で、メッセージを送信するための、標準的な通信プロトコルです。SMTP は、もともと 80 年代初頭に開発されたもので、世界で最も普及しているプロトコルの 1 つです。
- B. HTTPS (Hypertext Transfer Protocol 1.0) は、インターネット上で広く使用されている、コンピュータ・ネットワーク上の安全な通信のためのプロトコルです。HTTPS はトランスポート・レイヤー・セキュリティまたは、その前身であるセキュア・ソケット・レイヤーで暗号化された、接続内の HTTP (Hypertext Transfer Protocol) による通信で構成されます。HTTPS の主な開発動機は、訪問したウェブサイトの認証と、交換されたデータのプライバシーと完全性の保護です。
- C. IPFS：(Benet, 2016) IPFS は、すべてのコンピューティング・デバイスを、同じファイル・システムで接続することを目的としたピアツーピア分散ファイル・システムです。いくつかの点で、IPFS は World Wide Web と似ていますが、IPFS は 1 つの Git リポジトリ内でオブジェクトを交換する単一の BitTorrent 群として見るすることができます。つまり、IPFS は、高いスループットでコンテンツ・アドレス付きのブロック・ストレージ・モデルを、コンテンツ・アドレス付きハイパーリンクと共に提供します。これは、一般化された Merkle の有向非循環グラフ (DAG) を形成します。IPFS は、分散ハッシュ・テーブル、インセンティブ・ブロック交換、および自己証明の名称空間を組み合わせています。IPFS には、単一障害点はなく、ボットは互いに信頼する必要はありません。分散コンテンツ配信は、帯域幅を節約し、HTTP が苦勞する DDoS 攻撃を防ぎます。ファイルはハッシュで識別されます。ファイルは BitTorrent ベースのプロトコルで配布されます。他のユーザーがコンテンツを見ることで、ネットワーク上の他のユーザーにコンテンツを提

供します。IPFSは、IPNSという名称サービスを持っており、それはPKIに基づいたグローバル名称空間で、信頼チェーンの構築を支援し、他のNSと互換性があり、DNS、onion、.bitなどをIPNSにマッピングできます。

ここに示した3つの通信チャンネルは、相補性の一例に過ぎず、ニューロ・チェーン・ネットワークの3つの主要な特性：セキュリティ、柔軟性、スケーラビリティ、およびトレーサビリティを達成するために挙げられました。以下の表1は、これらの通信プロトコルの、特性の分布を示します。通信の面では、ニューロ・チェーンの特質は、特定のタスクを実行するために、必要なパフォーマンスとセキュリティに応じた適応性です。たとえば、トレーサビリティ・チェーンの通信チャンネルは、インフラと入力（ボリュームと速度）に従って進化します。SMTPなどのメッセージング・プロトコルは、セキュリティと柔軟性の理由から企業に適していますが、専用の「通信ポート」は、ブロック・チェーンの価値創造に、高速通信が必要な場合に使用されます。ポットのアーキテクチャは、単一障害点（SPoF）を抑えるために最適化されています。それぞれの部分は、最大限の独立性を保証するために開発されています。

通信レイヤにおける重要な改善軸は、情報伝達のためのLi-Fiプロトコルの使用です。光に基づくこの技術は、Wi-Fiと比較してより安全で（セキュリティの侵害）、高いレベルの性能を提示します。この研究の軸は、セキュリティとパフォーマンスのを保証し、大規模な採用を確実にする目的で、ニューロ・チェーンにとってとても重要なのです。

	Volume	Velocity	Security	Adoption
HTTP				X
HTTPS			X	X
SMTP				X
SMTP with TLS			X	X
IPFS	X		X	
FTP	X	X		
FTPS	X	X	X	

	ボリューム	速度	セキュリティー	導入
HTTP				X
HTTPS			X	X
SMTP				X
SMTP with TLS			X	X
IPFS	X		X	
FTP	X	X		
FTPS	X	X	X	

Table 1 : Intrinsic characteristics of different communication protocols.

表1：様々な通信プロトコルの本質的な特徴

上述のように、ポットは、ネットワーク上の価値創造（暗号値、透明性、証明書など）や、知能的なアプリケーションの精巧な動作をで実行するために、あるレベルの自律性を可能にする様々なアルゴリズムを使用しています。ポットは、通信プロトコルを適応させることで、Web、モバイル、またはハイブリッドなど、様々なプラットフォームでサポートできます。またポットは、ビット・コインやエテリウムのような、他の既存のブロック・チェーンと接続することも可能です

。図 6 は、プラットフォームに応じて、ニューロ・チェーンの様々な通信プロトコルをまとめたものです。プラットフォーム間のリンクの太さは、必要な通信性能を表しています。

ニューロ・チェーンのこの側面の説明の後、ニューロ・チェーンを、生物学でのニューロンに例えることができます：ポットが神経細胞体であり、通信層が軸索に相当します。実際、ニューロ・チェーン研究所では、ポットを使った人工ニューロン・ネットワークの実現のを目指した、活発な研究が開始されております。アイデアとして、必要な解析に従って、ポット間の適応的な接続を作ることです。いわゆる、深い分散型学習です。

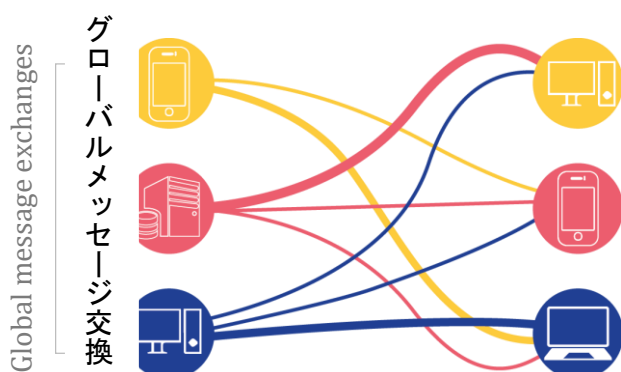


図 3：ポットは自動的に（推奨エンジン）通信プロトコルを変更可能で、帯域幅（量、速度、セキュリティ）を最適化することができます。ポットの頭脳が、最適なチャンネルを選択するために、開発されています。

プロトコル、総意、及び機械学習

再度強調すべき点は、ニューロ・チェーンの背後にある技術は、適応通信プロトコル、進化的かつ低遅延の総意、および関連する機械学習アルゴリズムです。適応型通信プロトコルは、ニューロ・チェーンの柔軟な運用を可能にし、ネットワークのセキュリティと性能の向上をもたらします。また、企業や人々（大衆市場）による、大規模なプロトコルの導入を促進します。実際、適応型通信レイヤーは、企業によるニューロ・チェーンの使用を、容易にするだけでなく、ポットとのやり取りも簡素化します。様々なプロトコルは、様々なプラットフォームとサポート（スマートフォン、ベースコンピュータ、IoT など）からのアプローチを可能にします。

ニューロ・チェーンが提案した低遅延の進化型総意は、パフォーマンスの最適化のため、機能的なアプリケーション（トレーサビリティまたは知能的なアプリケーション）に応じて、適切な総意アルゴリズムが選択できます。ポットの目標達成のため、トレーサビリティと最大重み付きエントロピー総意の決定要因が設定されます。様々な総意については、次のセクションで詳述します。

最後に、ポットの頭脳である機械学習アルゴリズムは、ポット間の取引や通信の適格性評価と定量化に、様々な要因を加えることができます。また、これらのアルゴリズムは、予想と予測、および異常検出を可能にします。このポットの特性は、ネットワークを単一のエンティティとしての分析可能にし、ポットの集団人工知能を利用して、ブロック・チェーンの完全性を保証できます。このニューロ・チェーンの特性については、次のセクションで詳細に記述されます。

総意：完全性と参加の証拠（P I I）

ニューロ・チェーンでは、取引検証とトレーサビリティをベースにしたアプリケーションによる透明性が、真の価値（貴重な価値）を創造し、ネットワーク上に循環する情報や、配布される認証書類（IPFS）の形として現れます。

ブロック・チェーンの検証と統合のプロセスを報償することによって直接創造された価値は、ポット間の信頼性を高め、これはある意味、ネットワークへの透明性の統合と言えます。したがって、ポッ

トを通して移動する情報の測定とその重みを、総意達成と公正な決定の際に、考慮に入れるべきです。また、ポットの完全性のフィードバック要素を総意プロセスに含めることも重要です。その測定は、異常検出、システムプロセス、および一貫性アルゴリズムに基づき、ポットのやり取りの一貫性と堅牢性を評価します。総意は、**参加と完全性の証拠 (PII)** と呼ばれています。

総意のプロセスは次のとおりです。

リーダーは、ネットワークへのポットの参加と完全性に基づいて、選出されます。ポット完全性の程度は、一方では、ネットワーク内で交換される情報や取引に依存し、他方では、実際の貢献度（認証）とその信頼性に応じて、動的に割り当てられた、各ポットの透明度スコアに依存します。スコアは、直接的な差別尺度を得るため、ネットワーク上で正規化されます。

この総意の基礎は、情報とその測定です。最も適切な情報の尺度の1つは、「エントロピー」です。Claude E. Shannon は、1948 年の「通信の数学的理論」という記事で、この情報理論の状態量を導入しました。Shannon にとって (CE Shannon, 1948)、情報は任意性を示し、システムの予測不可能性の程度を表します。したがって、事象の不確実性は、情報の測定手段とみなします。エントロピーの妥当性を説明するために、粒子（電子および陽子）の集合を想像し、各粒子はその位置とエネルギーで定義されます。エントロピーは、システムの「混雑さ」の尺度（熱的攪拌）であり、システムの状態の数、または微視的な構成に関連します。

n の要素を持ったソース B について、実現可能性が p_i の場合、エントロピー H は、次のように定義されます (C. E. Shannon, 1948)。

$$H_b(B) = - \sum_i^n p_i \log_b(p_i)$$

ビット単位の情報に相当するため、通常、対数は底を 2 にします ($b = 2$)。その他の場合は、自然対数を使用すべきです。最大エントロピーの考慮は、ベイジアン推論で事前分布を決定するために便利です。重要なことは、状況に応じて、Shannon のエントロピーを推定するための、様々な方法があるということです。以下では、例としてエントロピーの経験的推定方法を詳述します。ニューロ・チェーンには、2 つの推定方法が利用可能です：「Dirichlet」、「Bayesian」、「MillerMadow」 [<http://strimmerlab.org/software/entropy/>]（機能したライブラリ）。また、Kullback-Leibler 発散、chi-squared、相互情報、chi-squared 独立統計のような、様々なラメータが、ニューロ・チェーンの意思決定プロセスに、影響を与えられます。例えば、「ChaoShen」エントロピー推定 (A. Chao and T-J. Shen, 2003) は、ポットが低稼働し、まれな取引を処理する場合に最適です。

さて、各粒子がポットを表し、粒子の様々な状態がポットの取引状態を表すと想像してください。この時、エントロピーは、ポットのやり取り及び参加のレベルに相当します。エントロピーの使用の主な利点は、特定のポットの他のポットとのやり取りを測定可能な上に、他のポットが特定のポットとどのようにやり取りするか（または特定のポットが、ネットワークに取ってどのように考慮されるか）も測定可能です。

計算されたエントロピーはまた、取引の強さまたは価値で正規化されます。つまり、取引の強さは、ネットワークにとっての、その本質的な価値の増加を表します。たとえば、暗号通貨では、強さは、ネットワーク内のすべてのポット単位の交換で正規化された、取引の大きさ（保持する暗号化通貨の大きさ）を示します。これは、各ポットの取引の内部頻度と程度、いわゆるエンタルピー (P. W. Atkins, 1998)、つまりポットの（理論的な）内部エネルギーの状態量を反映しています。前述の粒子群に戻って、陽子の内部エネルギーは、電子の内部エネルギーとは全く異なっています（おおよそ 1836 倍高い）。したがって、対象の環境におけるそれぞれの影響力は、まったく異なります。

ポットのエンタルピー $B_i W_B$ は、そのポットがシステム内で交換したすべての価値の累積関数です。これは、取引中に**所有された豊かさ**の尺度です。

よって、重み付きのエントロピー $H_{w,B}$ は、

$$H_{w,Bi} = W_{Bi} \cdot H_{Bi}$$

したがって：

$$H_b(B) = - \sum_i^n w_i \cdot p_i \log_b (p_i)$$

ニューロ・チェーン・ネットワークの重み付きエントロピーは、ボットの微視的状态を反映しています。これは、各ボットの他のボットとのやり取り**レベル**と**強さ**を表します。また、**ボットの遺産**（ステーク）によって正常化された、**活動**の概念に関連します。

また、ネットワークの**巨視的**な状態も反映しているため、ブロック・チェーンの最後のバージョンの署名を表します。**完全性スコア (SoI)** と共に、それはブロック・チェーンの履歴の信頼できる署名となり、ボットに堅牢な総意プロトコルを提供します。
総意プロセスを図4に示します。

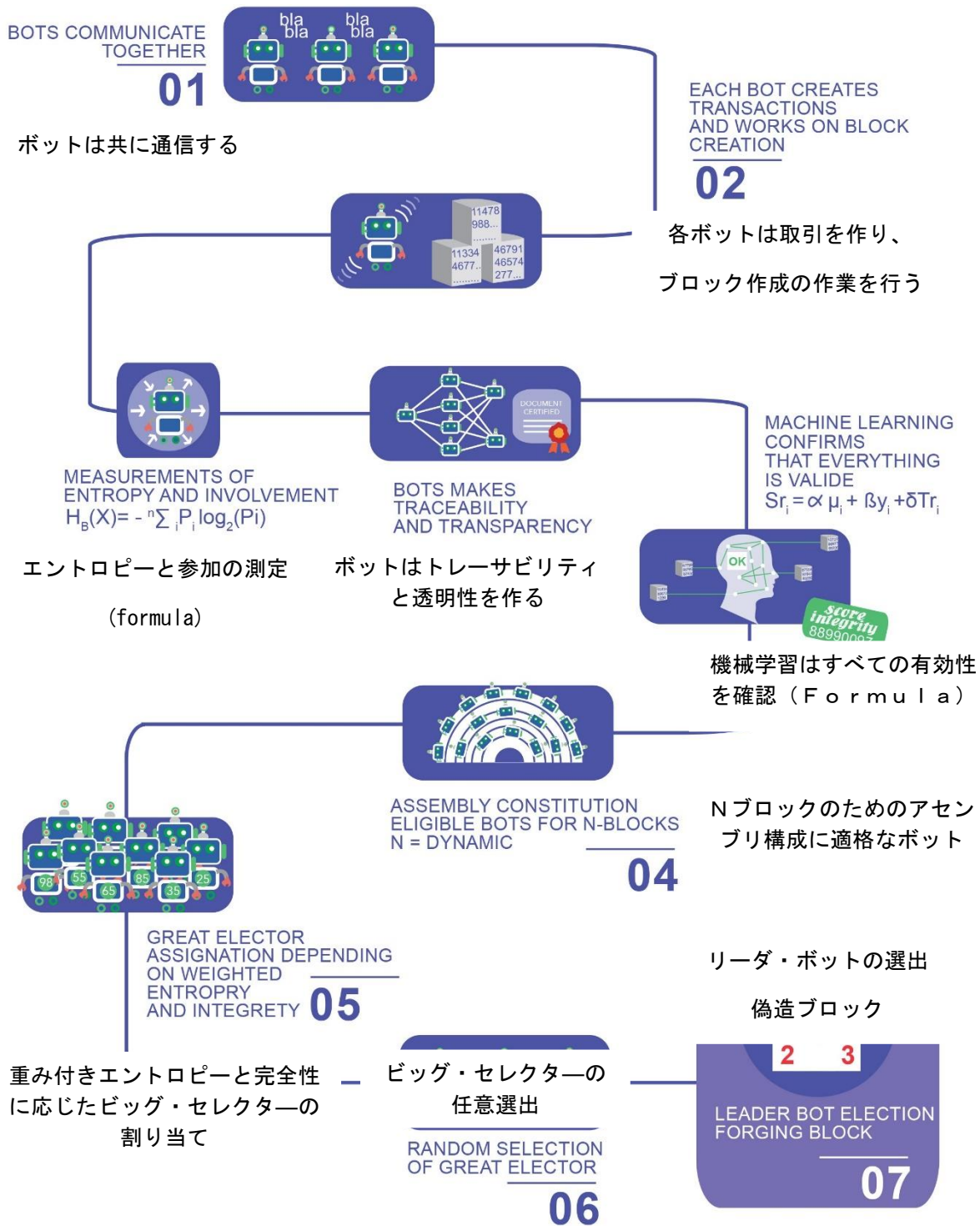


図 4：ニューロ・チェーンの総意を達成するための様々ステップ。エントロピー計算と完全性スコアの評価を示します。リーダーの選出は、これらの2つの指標に基づき、これらの指標は、ブロック・チェーンのユニークなバージョンの署名としても機能します。

このアプローチを説明するために、一例を以下に示す。

サイズ $N = 5$ で、有向非巡回グラフ $G = (B, E)$ でモデル化された、任意のニューロ・チェーンを考えます。ここで、 $B = \{1, \dots, 5\}$ はボットを表し、 $E \subseteq B \times B$ はボット間の端線を表します。

ボット間のやり取りは括弧内の数値で与えられます：

このネットワークのグラフ的な表現は、図 5 に示します。

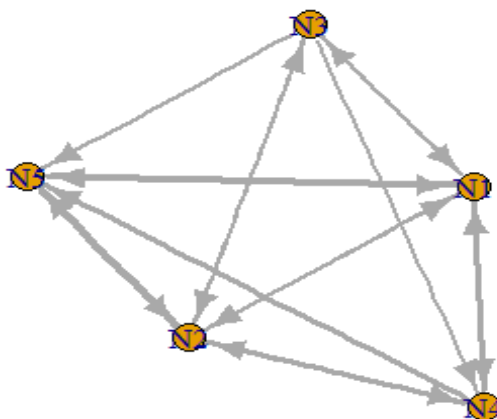


図 5 : サイズ $N = 5$ のグラフ $G = (B, E)$ の図式的表示

プログラム :

```
library(entropy)
library(igraph)
N <- 5
for (i in 1:N){
  freqs = table(Data_noeuds[,i])/length(Data_noeuds[,i])
  E = entropy.empirical(freqs, unit = "log2")
  #E = info(freqs)
  cat( "\n the Entropy of Bot =", i, " est ", E)
}
```

各ボットの分散を表したエントロピー計算は、以下のようになります。

ボット 1 のエントロピーは : $H_1 = 1.996792$

ボット 2 のエントロピーは : $H_2 = 1.932915$

ボット 3 のエントロピーは : $H_3 = 1.944623$

ボット 4 のエントロピーは : $H_4 = 1.255537$

他のボットとの最大取引を含むため、ボット 1 が最大エントロピーを示し、また、ネットワーク内の最小の交換を含むため、ボット 5 が、最小エントロピーを示します。

ここに示した例では、エンタルピーを考慮していません。重み付きエントロピーの全体的な統計的解析は、[付録 4 : 重み付きエントロピーの統計解析]に記載されています。

参加と完全性の証拠の第 2 パラメータは、「完全性スコア」と呼ばれ、ボットに完璧さを目指すように（ボットは本来、正直である）動機づけます。上記で展開されたように、重み付きエントロピーは、ボットとその環境とのやり取りを強調し、 Sr_i は、次のモデルによって示された、関連する様々な測定値の連結である、Bot 固有の特性を表します。

そのモデルは、以下の項目をベースにします。

- **Bot の評判** : Bot の評判は、ネットワークがボット B_i に対して経験したことを反映した、ユニークなグローバル的信頼価値を表します。また、コヒーレンスおよび異常検出アルゴリズムによって、 B_i の完全性を反映しています (Gelman, Carlin, Stern, & Donald B Rubin, 2003)。
したがって、評判 μ_i は B_i が原因の、ネットワーク上の**不規則性**の割合を表します。たとえば、不正行為は、 $Sr = 0$ を持ち、結果として除外されることにつながります。評判のための強力なアルゴリズムの 1 つは、動的閾値を有するベイジアン・ネットワークです。
- **真の価値創造** : 価値創造 γ_i は、取引を通じてネットワークに注入された新しい情報に関連しています。この新しい情報は、オブジェクトや概念や知識のトレーサビリティ・チェーンに代表されます。
- **透明性** : ニューロ・チェーンでは、透明性が検証された取引と、共有された認定済みの支援と文書によって表されます。透明性 Tr_i は、ネットワークを介して、 B_i が発行した認定済みの情報のレベルを反映しています。認証のために、ニューロ・チェーンは拡張 IPFS プロトコルを使用します。

よって、完全性のスコアは、上記の 3 つのパラメータを持った線形回帰としてモデル化されます (Myers, Zhu, & and J. Leskovec, 2012) :

$$Sr_i = \alpha\mu_i + \beta\gamma_i + \theta Tr_i$$

ここで、 α 、 β 及び θ はシステムによって学習・調整される係数で、 $\alpha + \beta + \theta = 1$ を満たします。初期の運用では、すべての係数は同じ重みを持ちます。

そして、参加と完全性の証拠 (P I I) は、2 つのパラメータの線型的合計となります。

$$PII_i = \omega Sr_i + \kappa H_{w,i}$$

ここで、 ω と κ はネットワークが学習すべき**動的なパラメータ**です。

アセンブリ構成 : ボットの重み付きエントロピーと、その完全性を考慮して、所定数の検証ブロックからなるサイクルによって、アセンブリが構成されます (プロセスは以下で詳述されます)。したがって、各ボットは、グローバル取引プールから、不均衡な取引のブロックを構成します。ブロックには、ブロック・チェーンの最後のブロック情報と、関連する情報が含まなければなりません。これは、すべてのブロックに、最新版の元帳 (システム) が必要なためです。完全で有効な取引ブロックを持った各ボットは、ブロック・チェーンに書き込むための、任意な決定の資格を持つこととなります。

選出プロセス

再度、選出プロセスを説明するために、粒子の例を思い出しましょう。核物理学では、荷電した超熱粒子 (高エネルギー粒子) を分離するため、磁気トラップが用いられます。ニューロ・チェーンでは、ネットワーク内で最も参加し、統合されたボットを識別するために、グローバル重み付けエントロピーと、完全性 (PII) をベースにした動的フィルタが使用されます。適格なボットの構成を選出するため、加重エントロピーおよび完全性スコアの動的閾値が設定されます。選出された各ボットには、その PII に比例した、複数の「偉大な」セクター (動的セキュア・トークン) が割り当てられます。大規模な選出プールから、検証のために、ボットが任意に選出されます。すべて検証は支払われ

ます。任意の選出は、分散された真の乱数生成[付録 5：任意性およびカオスプロセス]に基づいています。

詳細には、アセンブリ構成内のポット数は 535 個のポットで始まり、ネットワーク運用と合計重み付きエントロピーと完全性に応じて進化します。各アセンブリの検証の数は、1638 ブロック（「黄金比」に関連する）から始まり、これはネットワークのパラメータ（ネットワークの完全性、フォークの確率、重み付き平均エントロピーの進化）に応じて進化します。選出パラメータの動的プロセスは、アセンブリ構成のより頻繁なロテーションを保証します。

現在の選出プロセスの主な特徴は、セキュリティ、柔軟性、及びスケーラビリティです。実際に、プロトコルの進化は、ネットワークの運用に関連するパラメータに依存します。ネットワークは、どのように動作し、適応するかを学習していきます。

このプロトコルは、ポットに、リーダーに選ばれる可能性が低い（実際の既存総意に依存）にもかかわらず、「ブロック検証のため、運に頼らず」、アセンブリ構成に参加することを動機づけます。

ニューロ・チェーン総意の哲学は、リーダーになるには、ポットがネットワークに参加する必要があるということです。また、高い完全性を示さなければなりません。柔軟性、スケーラビリティ、自己適応の特性により、プロトコルは迅速かつ効率的な進化可能です。総意は、価値創造と透明性のための好循環と競争を創造します。実際に、取引検証とトレーサビリティ・チェーンによってもたらされる透明性は、完全性スコアに基づいた総意一部なのです。ニューロ・チェーンで創造された価値の例は、トレーサビリティ・シンク、暗号化価値作成ツール、価値交換または暗号化交換プラットフォームの基盤、及び知能的なアプリケーションの評価ソース等です。

これらの概念をよりよく理解するために、作業の証拠の総意は、基本的なマイニングに基づき、外来/外的作業の源泉であるのに対して、参加と完全性の証拠の総意は、ブロック・チェーン内で創造された、情報の本質的価値、透明性、完全性に基づいています。作業証拠ブロック・チェーンの検証は、ネットワークで消費された合計計算パワーであるのに対して、ニューロ・チェーンの検証は、ブロック・チェーンで作成された、合計重み付きエントロピーと完全性です。

ニューロ・チェーンの情報回復基準は、「クラウドジウス」です（エントロピーを初めて記述した Rudolf Clausius に因んだ）。クラウドジウスは、ネットワーク内を循環する取引フローの基本単位です。例として、暗号化交換のアプリケーションの場合、クラウドジウスは、ポット間で交換される価値を表します。クラウドジウスの価値は、ネットワークに応じて進化します。

分散型プロセス

要約すると、ネットワーク内のステップは以下のようになります。

1. 新しい取引が作成され、検証のため、すべてのポットにブロードキャストされます。
2. 一定数のブロック検証のために、参加と完全性の証拠に応じて、アセンブリ構成が指定されます。
3. 各ポットに割り当てられたビッグ・セレクタを分散的な任意な方法で選択することで、対象となる構成ブロック検証用のリーダーが選ばれます。
4. すべてのアセンブリ構成は、有効で認証済みの取引ブロックのみを受け入れます。
5. アセンブリ構成は、前のブロックの受け入れの証拠である、IDまたはハッシュを持った、新しいブロックを作成していきます。

取引の検証プロセス（標準的な暗号検証に加えて）は、その応用分野（トレーサビリティ、暗号化価値、知能的なアプリケーション）の性質によって異なってきます。すべてのポットは、発生する可能性があるすべての検証プロセスに備え、ツールキットまたはアルゴリズムボックスを持っています。すべてのポットは、（最大加重エントロピーと完全性を持つ）最長のチェーンを参照し、それを拡張していきます。

機械学習

機械学習または人工知能という用語は、ボットの高度な抽象化を指します。様々な分類のアルゴリズムで構成されています。これらのアルゴリズムは、ボットによって様々なアプリケーションに使用されます：トレーサビリティの一貫性アルゴリズム、ボットが発信した情報や取引の異常を検出のためのベイジアン・ネットワーク・アルゴリズム、情報交換のための価格設定アルゴリズム、複雑なルール・ベース・アプリケーションなど。アルゴリズムの認証は、IPFS プロトコルと暗号署名を使用します。その処理を以下に説明します。

- **一貫性アルゴリズム**：トレーサビリティは、オブジェクトのライフラインの連続性、またはブロック・チェーンに統合された概念に基づいています。トレーサビリティ・チェーンの送信者は、オブジェクトの本質的な情報最と、最初の取引でのボット集団とのやり取りに関する情報を指定します。アルゴリズムは、様々な取引、固有のメタデータ、および検証のための関連した認証ドキュメントの整合性を確認します。また、このアルゴリズムは、時間の一貫性、及び取引順序を検証します。たとえば、オブジェクトが一連のステップで、一つの段階をスキップした場合や、プロセスの特定のステップで停滞した場合に、アルゴリズムはそれらを疑わしいオブジェクトとして検出します。
- **ベイジアン・ネットワーク・アルゴリズム**：(Darwiche Adnan, 2009; Gelman, Carlin, Stern, & Donald B Rubin, 2003) は、高次元で不均衡なデータのために適応された、確率的なグラフィカルモデルであり、条件付き確率計算に基づいています。確率的なグラフィカルモデルでは、各ノードは変数（または観測対象）を表し、ノード間のリンクは、これらの変数間の、起こり得る因果関係や相関を表します。これをニューロ・チェーンに適用すると、ノードはボットを表し、リンクは、ボット間の様々な取引を表します。結果として、作成されたグラフは、方向性を持ち、ベイジアン・ネットワークは、有向非循環グラフ (DAG) に属します。異常検出アルゴリズムのしきい値は、ネットワークに応じて、動的に設定されます。次の例を考えてみましょう：2つのイベント A、B があり、両者がイベント C を引き起こせるとします。そして、イベント B がイベント A に影響を与えるとします。この状況は、図 6 に示すベイジアン・ネットワークでモデル化できます。

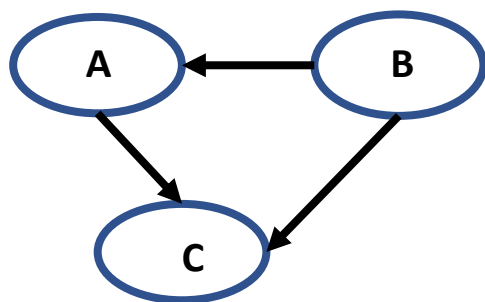


図 6：ボット間のやり取りを示す有向非循環グラフ

したがって、確率確率関数は次のようになります。

$$Pr(A, B, C) = Pr(C/A, B) \cdot Pr(A/B) \cdot Pr(B)$$

ベイジアン・ネットワークでは、観測されない変数の推定、パラメータ学習、構造学習など、3つの主要な推論アプリケーションがあります。これらのアプリケーションは、事前確率と尤度と共に、あるデータとパラメータに対する基本ベイジアン解析をエコーします。このアプローチは、分散型のボット・ネットワークと全体的に整合しています。

異常検出のプロセスでは、各ボットの原子レベルの取引に関連する、他の検証が考慮されません。

プログラム：

```
library(bnlearn)

bn.hc <- hc(data_N)

plot(bn.hc, main = "Hill-Climbing", highlight = c("flowtype"))

fittedbn <- bn.fit(bn.hc, data = data_N)

loglikelihood <- logLik(fittedbn, clust_CFL, by.sample = TRUE)

plot(loglikelihood)
```

背後にあるアイデアは、アルゴリズムがボット間の取引から学習して、モデルのパラメータを計算することです。次に、新しい取引ごとに、やり取りの新しい確率が計算されます。**不規則な動作**をした、悪意のある取引が検出されます（許容しきい値に依存）。

- 意思解析：意思解析は、ソーシャル・ボットのようなソーシャルアプリケーションで主に使用されます。これらのアルゴリズムは、ユーザーがボットを介して、お互いに交流するのに役立ちます。感情分析、実体認識、及び辞書構成は、人間のユーザとボット、またボット間のコミュニケーションを容易化する手続きの例です。
- ルール・ベース・システム (Giarratano&Gary Riley, 1998)：は、条件とルールを扱う強力なエンジンです。この種のシステムは、人工知能の基盤要素の一つです。ニューロ・チェーンでは、ルールベースのシステムが、スマートビジネスアプリケーション用に、知識の格納・組み換えを行い、アプリケーションに応じて、適切な方法での情報解釈に使用されます。
- レンド検出アルゴリズム：ニューロ・チェーンでは、ボットがチェーン内の暗号化価値の交換の権限があります。これらの値を評価するために、ネットワークに基づき、取引アルゴリズムも提供されます。新世代の分散型価格設定と予測アルゴリズムが、開発される予定です。したがって、トレーディングプラットフォームは、システム上に現れます。

上述のアルゴリズムは、様々な目的を達成するために、ボットに提供される多種のアルゴリズムの代表例です。

ワークフローの証拠

このセクションでは、セキュリティと認証が保証された、分散システムにおける、機械学習アルゴリズムの実行プロセス、つまりワークフローについて説明します。アルゴリズムは、ボットから発行され、IPFS プロトコル（分散ストレージ）に格納されます。アルゴリズムのハッシュは、IPFSのアドレスを表します。ワークフローは、入力データ、命令（決定的アクション）、伝播、例外、および定義によって定義されます。ワークフローの証拠は、ボットによって発行され、ネットワークによって完全性が保証された、分散ワークフローを定義します。更に詳しく説明すると、完全性のワークフローは、ボットの署名と IPFS プロトコルに依存します。アルゴリズムの発行者は、システム内の構造化されたワークフローに応じて、取引を IPFS アドレスで署名します。ワークフローの証拠の利点は、ネットワーク全体で、様々な方法と補完的データが提供されていくため、集合的な人工知能の第一歩を創出することができます。この目標を達成するために、ニューロ・チェーン・ネットワークは IPFS プロトコルに、ますます関わっていく予定です。したがって、ニューロ・チェーンは自己一貫性を持つこととなります。

次の図 7 は、ニューロ・チェーンのワークフローを示しています。

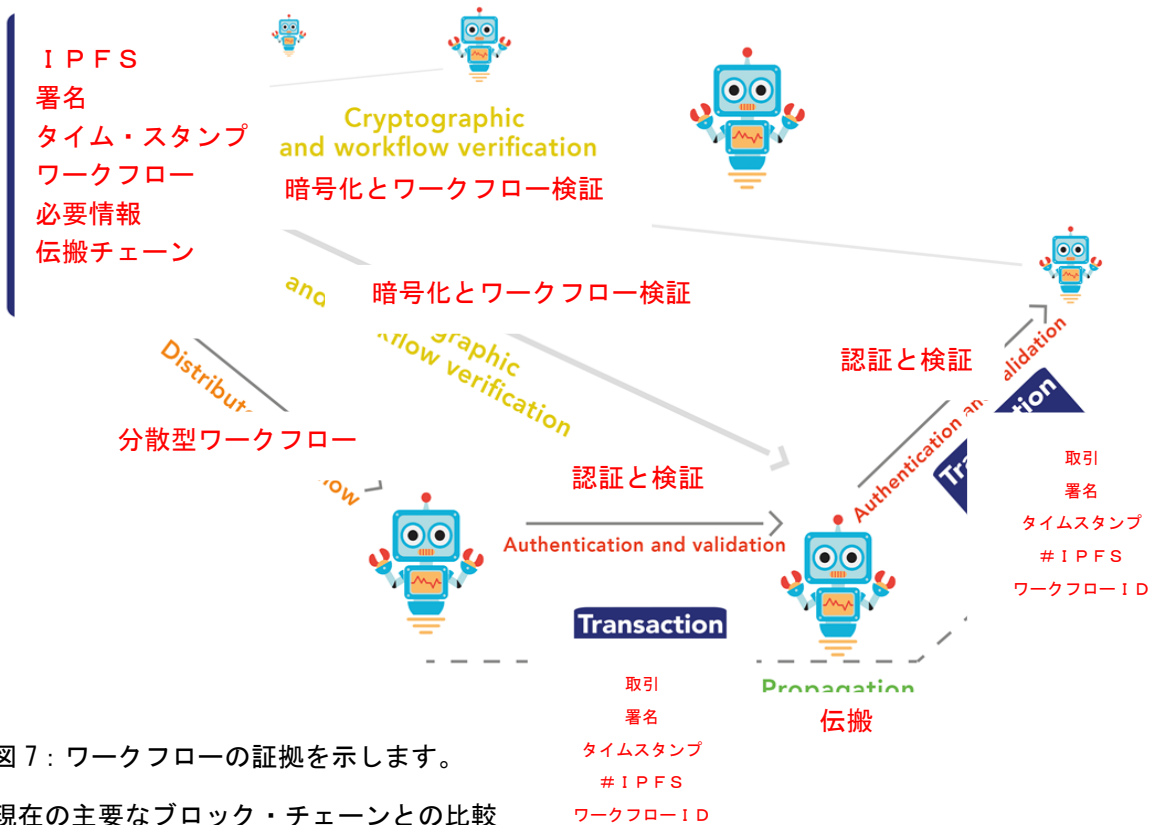


図7：ワークフローの証拠を示します。

現在の主要なブロック・チェーンとの比較

下の表3は、ニューロ・チェーンとビット・コインとエタリアムの標準ブロック・チェーンの比較を示しています。このベンチマークは、いくつかの特性とパフォーマンスに関する、ニューロ・チェーンの進化を示しています。

特性	ビット・コイン	エタリアム	ニューロ・チェーン
コイン	ビット・コイン	エテル	クラウドジウス
応用	金融的取引	スマート契約	知的なアプリケーション
			高レベルの抽象化
意思決定	体系的	非体系的	アセンブリ構成
			フェアな決定
総意アルゴリズム	ワークの証拠	現在：ワークの証拠	参加と完全性の証拠 (P I I)
		将来：ステークの証拠	重み付きエントロピーと評判スコアリング
取引パフォーマンス	7tx/sec	~25tx/sec	数千 tx/sec (最適)

ブロック間隔	10 分	15 秒	~3 秒（最適）
ブロックサイズ	1MB/動的	動的	動的
技術	分散ネットワーク	分散ネットワーク	分散ネットワーク+機械学習
アプリケーション	暗号化通貨	基本的なスマート契約	緻密なアプリ、暗号化価値、認証されたデータ・バンク、ソーシャル交流、スマートIoT、ビジネス・アプリ
通信プロトコル	静的	静的	動的で適応可能

表 2：本質的な特性および性能に関するニューロ・チェーンと他のブロック・チェーンとの比較。

ニューロ・チェーンアプリケーション

ブロック・チェーンにおいて、バリデーション、トレーサビリティ、透明性、完全性によって、真の価値が生成されているため、ニューロ・チェーンは、様々なアプリケーションに応用できます。

暗号化価値（交換可能な価値）：最初の直感的なアプリケーションは、「クラウドジウス」と呼ばれる暗号化価値です。クラウドジウスは、透明性と完全性を通じて、ニューロ・チェーンで作成された価値に直接リンクし、ブロックの検証とトレーサビリティのための情報の統合が報償されます。この標準的な価値創造は、ボットに透明性と相互コミュニケーションを促進する手段でもあります。

ニューロ・チェーンでは、暗号化交換プロセスにより、特定の価値に対する供給と需要に完全に結びついた取引や貿易が可能になります。

標準クラウドジウスの 2 つの妥当性評価があります：内因的および外因的。内因的な妥当性は、価値と概念の暗号化交換によって駆動されます。クラウドジウスの外因的な妥当性は、外来標準（ビット・コイン、フラット等）と関連しています。ニューロ・チェーンでは、標準的なリファレンスはネットワークです。

トレーサビリティ・チェーン：以下の段落では、物流におけるニューロ・チェーン内のプロセスをよりよく理解するために、運用トレーサビリティ・チェーンが説明されています。トレーサビリティ・チェーンは、決定的なルールまたは条件を実行し、取引フローをトリガする知能的なアプリケーションとして考えることができます。関連するブロック・チェーンは、トレーサビリティ・チェーンの様々な部分を表すボットで構成されています。発信者が各トレーサビリティ・チェーンを開始します。すべてのトレーサビリティ・チェーンがブロック・チェーンに共存します。異なるチャネル間のやり取りにより、ブロック・チェーン内の検証プロセスが増加し、メンバーと最終受益者の信頼と透明性が高まります。

トレーサビリティ・チェーンごとに、ブロック検証のリーダーが決定論的な方法で選出されます。チェーンの発行者は、チェーンの完全性を維持することに、最大の関心を持つと仮定して、リーダーとして指定されます（単純なプロトコルの運用において（附属書 1：トレーサビリティ・チェーン））

。

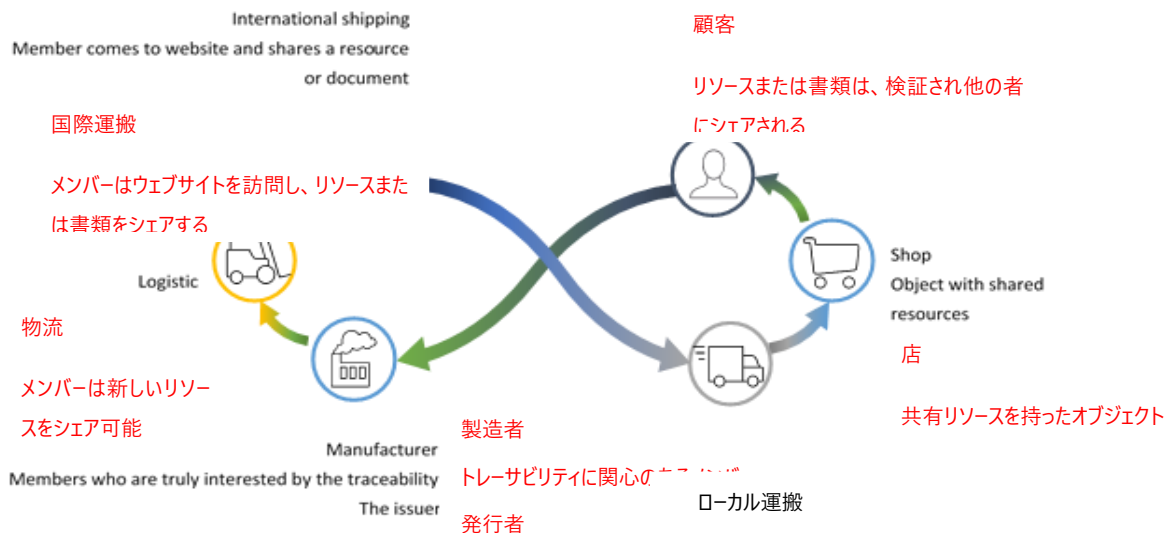


図 8：様々な部分と参加者を持ったトレーサビリティ・チェーン（ボットの機能的な表現）

すべてのボットは、取引が有効で、一貫性があり、認証されたドキュメントと一致するかどうかをチェックします（ファイルストレージは分散型 IPFS プロトコルによって保証されています）。方法と、一貫性と追跡のアルゴリズムは、リーダーから発行されます。チェーン内のすべてのボットは、リーダーに忠誠を示し、ネットワーク内で取引を広めるよう動機付けられています。

図 8 は、単一チェーンが、6 つのボットで構成されると仮定の、トレーサビリティ・チェーンのキネマティクスを示しています。各ボットはチェーンで独自の役割（プロデューサー、トランスポーター、卸売業者、ディストリビューター、最終的な顧客）を担います。ボット間のリンクと透明性を確実にするために、チェーンのオブジェクト、概念、懸念事項を交換するための、自律的なコミュニケーションが確立されています。プロデューサーによって、オブジェクトまたは概念「A」が生成され、検証されると、取引のようなメッセージがブロック・チェーンのすべてのボットに送信されます。そして、すべてのボットは、取引の一貫性と有効性を確認するために（リーダーが開始した方法論に従って）検証プロセスを開始します。取引がボットに到着すると、異常検出アルゴリズムも実装され、取引を検証し、悪意のあるオブジェクトや一貫性のないフローを検出します。さらに、「IPFS」プロトコルに基づくアルゴリズム認証文書は、ボットが文書（証明書またはサポート文書）を提供するときに利用可能です。この取引に関する総意が達成されると、取引はブロック・チェーンに組み込まれます。その後、オブジェクトがキャリアによってサポートされれば、新しい取引が発行され、同じ検証プロセスが再開されます。これは最終顧客まで繰り返されます。

このプロセスにより、プロデューサーから顧客へのオブジェクトとコンセプトのトレーサビリティが確保されます。ブロック・チェーンは、チェーン内の様々なアクター間のリンクを提供し、認証文書に追加された検証プロセスの重複によって、取引の合法性が保証されます。したがって、透明性のある価値がブロック・チェーンに作成されます。

指名されたトレーサビリティのリーダーには、2 つの主な動機があります：第一に、商業目的のための透明性とトレーサビリティです。第二に、チェーンによって引き起こされた取引の手数料です。したがって、システムは自己一貫性があります。このシステムは、IoT プロトコルへの接続もサポートしています（たとえば、入力データ用）。

知能的なアプリケーション：知能的なブロック・チェーンであるニューロ・チェーンは、スマートな複雑なアプリケーションを作成する能力を提供します。スマートアプリは、情報や取引の流れを生成する、複雑で精巧な条項や状況を受け入れられる、知能的な契約の進化です。ルールを効率的に解釈するためや、ロボットとのやりとりを効率的に行うために、ルール・ベース・システム（Gupta

、Forgy、Newell、&Wedig、1986) とアルゴリズムを活用します。知能的なアプリケーションのニューロ・チェーンは、完全型チューリングです。

知能的なアプリケーションは、方法論（アルゴリズム）、アプリケーション領域、および標準化された入力データによって定義されます（このメカニズムは以下で展開されます）。

革新的で知能的なアプリケーションの例は、スマート・シティ・アプリケーションに代表され、規則、保護、透明性（ボットの協業によるスマートな開発）[スマート都市のためのスマートビルディング]に関する総意を達成するため、アルゴリズムが活用されます。

ソーシャル・ネットワークまたはソーシャル・ボット：ニューロ・チェーンの分散アーキテクチャと通信により、ボット間のソーシャル・インタラクションの確立が可能になります。柔軟な通信プロトコルは、連続的に適応された交換を誘発します。ソーシャルボットは、仲介なしにボット間で交換された有効かつ関連する情報を、ネットワーク上に生成します。意思解析やエンティティ認識などの機械学習アルゴリズムは、コミュニケーションを理解し、最終的なユーザーの価値を抽出するために使用されます。たとえば、知能的なソーシャル・ボット・アプリケーションを使用して、タスクまたはオブジェクトを、ネットワーク全体または特定のプライベート・ネットワークに、通知または推奨し、履歴交換に基づいた取引を開始できます。別のアプリケーションは、ボット（所有者）のヒントと推奨事項を認定・サポートし、統合コンサルティング・プラットフォームを形成します。その結果、これらの共有されたヒントの関連性が、劇的に向上します。

認定されたデータ・リポジトリ：ネットワークにおける検証の冗長性は、文書の暗号署名を提供するIPFS（附属書 2：通信プロトコルの抜粋）などの特定のプロトコルを使用することによって、高いレベルの信頼と証明を達成します。文書のスタンプは、デジタル署名の一意性と耐久性を保証するために、ブロック・チェーンに保存されます。このドキュメントのリポジトリは、トレーサビリティ、実現の証明、配信証明などの様々なアプリケーションに使用されます。

スマート IoT：ボットは、スマートフォンや他のプローブを使用して IoT の役割を果たすことができます。情報は、安全なプロトコルで解釈され、伝播されます。ニューロ・チェーンは、柔軟で持続可能な総意プロトコルを通じて、大量の取引をサポートしています。ボットのソーシャル的機能は、ネットワーク内で交換される情報を評価するためにも使用できます。また、情報を収集するための異種ハードウェアもサポートしています。ボットが提供する様々なアルゴリズムは、IoT によって収集された大きなデータを分析することができます。

ニューロ・チェーンにおける IoT は、トレーサビリティなどのいくつかのアプリケーションの一部であり、インタラクティブなエコシステムの情報源でもあるため、重要な役割を果たします。

ビジネスアプリケーション：機械学習アルゴリズムを活用した分散型ボットは、会計、税務、ビジネス分析、保険、認証またはサポート機能など、単純で反復的で規制ありまたは規制なしの企業アプリケーションを実行できます。ここでは、ボットは情報フローとアクションを生成する、自律エージェントと見なされます。総意アルゴリズムはまた、決定プロセスを助け、不適切な事象を回避します。

図 9 は、ニューロ・チェーンの様々な実現可能な機能と非網羅的な機能を示しています。

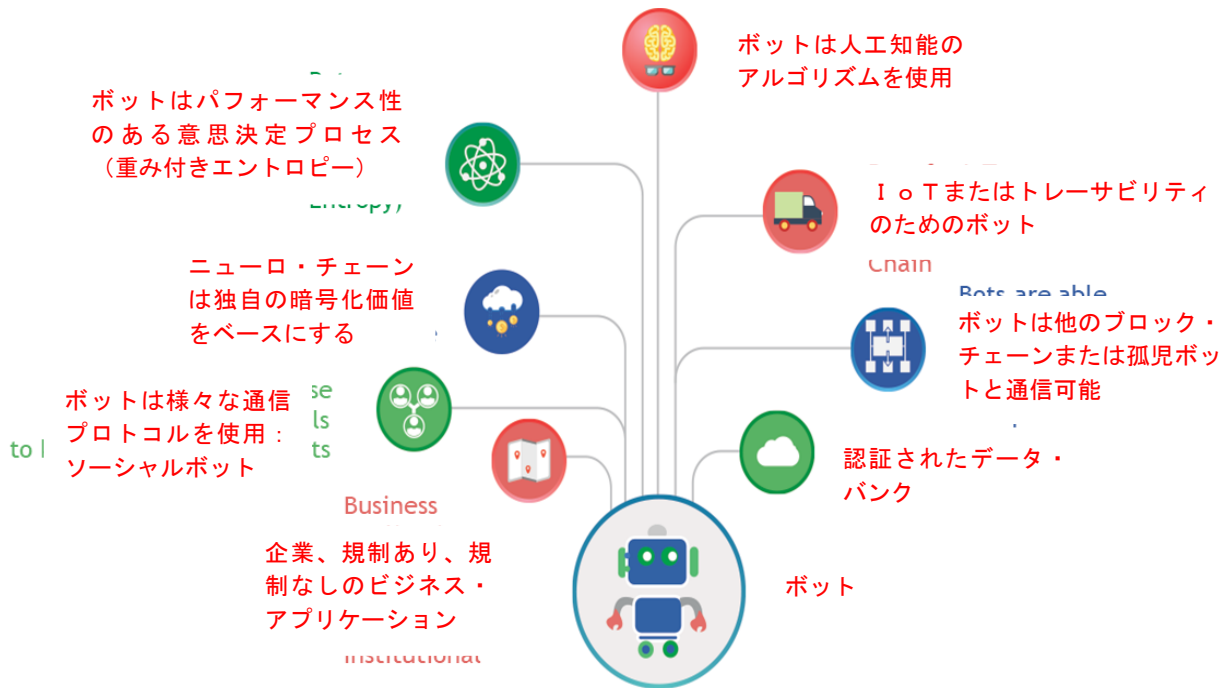


図 9：ニューロ・チェーンの様々な側面

ガバナンス

ニューロ・チェーンのガバナンスは、ロボットのやり取りを管理する民主的なプロセスです。この仕組みは、ネットワークの総意達成を可能にし、行動の流れをトリガします。完全性と参加の証拠は、各ロボットの参加、価値の保持、および完全性を反映する 3 つの独立したパラメータに基づいた、自己矛盾のない合意アルゴリズムです。総意アーキテクチャーは、偶発性を誘導し、総意をアルゴリズムによって完全に予測できないようになっています。言い換えれば、予測は確率的に不確実です。

機械学習アルゴリズムに起因する各ロボットへのフィードバックは、事実上、ロボットの完全性を保証します。ロボットのネットワークと総意への反作用は、不正なロボットの隔離や孤立化を引き起こします。

ガバナンス総意は、様々な調整パラメータによって修正可能で、柔軟性があります。次の表 3 は、動的パラメータとその決定プロセスへの影響を示しています。

パラメータ	エインパクト
通信相	ネットワークのパフォーマンス
完全性のスコア S_r $[\alpha, \beta, \theta]$	評判、価値創出、透明性
$PII = [\omega, \kappa]$	重み付きエントロピーと選出プロセス
PII の動的閾値	アセンブリ構成
ブロック数/サイクル	パフォーマンスとターン・オーバー
ビッグセクタの数	選出プロセス

表 3：この表は、様々な調整パラメータと、それらのネットワークと総意への影響を示します。

上記の様々なパラメータは、プロトコルの極限的な柔軟性を反映しながら、セキュリティ、生存率、および訂正（すべてのパラメータが他のパラメータとバランスを取る）をすべて保証します。アセ

ンブリ構成に基づく選挙プロセスは、リーダーの選定において高いレベルの任意性を保証するだけでなく、ネットワークのエントロピーと完全性の進化に従って進化する、ブロックの動的検証のためにアセンブリが選出されるため、プロトコルの実行において高いパフォーマンスを保証します。委任期間中のネットワークの進化は、アセンブリにおいて一定の更新性を保証します。

検査されたボット

特定の状況では、特定のボットによって検証された、異常なまたは予期しない取引について、ネットワークはこれらの取引を停止し、混乱したボットを（完全性スコアを介して）隔離する権限を持ちます。このプロセスは、**2 番目の異常**しきい値に達したときにシステムでトリガーされます（整合性計算）。このプロセスは、ネットワーク内の取引フローを誘導し、ボットの状態はブロック・チェーンレジスタに統合されます。

改正 [フォーク]

標準的な運用では、ニューロ・チェーンはフォークを経験しません。これは、アセンブリ内でボットが、検証の時間制限と強く協力しているためです。フォークがあった場合、ブロック・チェーンは最大加重エントロピーと完全性を持つ最長チェーンを採用します。

ボット補償

ニューロ・チェーンでは、インセンティブや価値創造は、選出プロセスに関連する検証プロセスを通じて行われます。この選出では、システムは、**評判スコアの評価**を通じて、その完全性を評価し、エントロピーとエンタルピーの計算を通じて、ボットの参加を検証します。換言すれば、ボットの報酬のために、システムは、透明度のレベル、ネットワークに注入される関連する認証情報のレベルを測定します。また、知能的なアプリケーションは、アルゴリズムやワークフローの複雑さに応じて、報酬を計算します。

報酬は、実体経済とインフレの指標に基づいて計算されます。有望な指標の例は、ネットワーク内の入札依頼、暗号通貨（ビットコインとエテリウム）、ネットワークの本質的なパフォーマンス、および経済インフレです。これは、実際のインパクトが異なると、時間の経過とともに資産または価値の進化を反映します。

投機とバブルを避けるために、NeuroChain と実際の経済を関連付けることが重要です。

ニューロ・チェーン相互エコシステム

ニューロ・チェーンは、周囲のあらゆるエコシステムと通信し、相互作用するように設計されています。まず、ニューロ・チェーンは、ビットコインやエテリウム、または他のブロック・チェーンと互換性のある標準ブロック・チェーンです（特にスマート契約に関する）。また、API を介してボットは、ウェブや知能的なアプリケーションと交換することができます。その他に、孤児の外因性のボットと相互作用し、それらをニューロ・チェーンの環境に持ち込むことができます。

通信レイテンシ

ニューロ・チェーンのソフトウェアは、すべてのプラットフォーム（パソコン、モバイル、または開発された IoT）でサポートされるように、設計されています。総意アルゴリズムは低遅延であり、**擬似リアルタイム**で多数の取引を検証することが可能です。適応型通信レイヤーは、プロトコルの高性能と高弾力性を保証します。同じ CPU 内および複数の CPU 上の並列化タスクは、パフォーマンスを向上させます。最後に、オープン・ソースのビッグデータ技術（Elastic Search、Neo4J、Spark 等）を使用すると、ネットワーク内での、やり取りや様々な要求の遅延が大幅に減少します。一般に、これらのオープンソースのビッグ・データ・ソリューションは、リアルタイム、並列化、およびストレージの問題を解決します。

PoC の最初の結果とパフォーマンス

提案された概念を検証し、ブロック・チェーンに関連するいくつかの性能パラメータを測定するために、様々な通信プロトコル、抽象化層および新しい総意アルゴリズムを有する分散アーキテクチャが開発されました。適応通信レイヤがテストされ、有望なパフォーマンスを示しています。様々な機械学習アルゴリズム[異常検出、一貫性、意味解析等]が個別に検証され、分散アーキテクチャ用に最適化されています。主な分析は、衝突と時間的進化のための、重み付きエントロピーの統計解析を用いた、総意アルゴリズムに焦点を当てています。統計解析は[付録 4：加重エントロピーの統計解析]に示されています。ネットワークに統合された重み付きエントロピーは、システムで高いパフォーマンスを発揮し、低い遅延を示しています。[]

新しい総意の優れた特徴の 1 つは、関連するパラメータの相殺です。換言すれば、エントロピー、エンタルピーおよび完全性は、ネットワーク動作中に逆相関します。これは、ボットの取引レベルが高く、エントロピーが増加すると、このアクティビティ状態は、各取引状態の強度を測定するエンタルピーによって重み付けされることを意味します（標準動作では、エントロピーの増加は、エンタルピーの減少を誘導）。同様な分析が完全性にも適用可能で、これは、ボットの強い活動（エントロピー増加）が、完全性スコア（ ΔSOI ）の変化を引き起こすためです。これらのパラメータの振る舞いを説明するために、ボットがエントロピーを増やすために仲間と不正にやり取りしようとしていると想像してください。エンタルピー（通常は弱い取引）と完全性スコアに影響します。逆に、アクティビティの低い（エントロピー減少）ボットは取引を強制すると、エンタルピーが増加し、完全性スコアが安定します。

総意達成に関連するパラメータは、ネットワーク活動の尺度でもあり、その意味では、時刻 t でのブロック・チェーンの写真を表すと言えます。

結論

ニューロ・チェーンは、拡張されたブロック・チェーンであり、ブロック・チェーンの持続可能性と分散性を活用し、また機械学習、人工知能、ビッグ・データ・ソリューションの適応性と予測性を利用します。熱力学に基づく新しい総意アルゴリズムは、取引量のスケーラビリティを可能にし、したがって、多種多様な知能的なアプリケーションに応用可能です。

ニューロ・チェーンは、ワークフローの証拠によって完全性が保証された、様々な実装アルゴリズムのおかげで、非常に複雑なビジネスおよびソーシャル・アプリケーションを可能にします。分散されたアプリケーションは、その環境とやり取りが可能で、適応型エコシステムに容易に実装できます。

知能的なブロック・チェーンの時代が始まりました。

付録

付録 1：トレーサビリティ・チェーン

トレーサビリティ・チェーンでは、2 つの動作状態を区別する必要があります：標準動作状態と悪意のある取引のケース。

標準的な操作では、2 つのケースを分離する必要があります。

1. 単一のトレーサビリティ・プロセスにおけるニューロ・チェーン：このケースでは、1 つの証拠のチェーンのみが考慮されます。取引は、異なるポットによって発行され、それらポット間の因果関係を持つ事象としてみなされます。取引の一貫性を保証するために、Lamport クロック因果関係 (Lamport, 1978) と同様のクロック因果関係が、システムに含まれています。各ポットには、一貫性アルゴリズムと認証アルゴリズムを含む一連のアルゴリズムがあります。前述のように、すべてのポットは取引を検証し、ブロック・チェーンへの組込みは、発行者によって決定されたリーダー (完全な権限を持つ) によって行われます。総意によって検証された各取引は、取引プール (一時データベース) から削除されます。証拠チェーンでは、最初の取引 (アルファ取引) は、オブジェクトのライブ・ラインの意味で、トレーサビリティ・チェーンのリーダーまたはオリジネータによって発行され、結果として、ポット間で一貫した取引が生成されます。
1. 相互作用トレーサビリティにおけるニューロ・チェーン：この場合、トレーサビリティ・チェーンは他のチェーンとやり取りします。各トレーサビリティ・チェーンにおける各リーダーの権限 (検証の権利) は、チェーン間の相関関係に応じて、他のリーダーと共有されます。相関測定は、両チェーンでシェアされたポットまたは共有オブジェクトに依存した、微妙なパラメータです。チェーン間の相関関係を理解するための、その他の側面は、トレーサビリティが応用された製品の経済圏に関連します (正の相関関係のみが考慮されます)。この相関はまた、エントロピーの概念に関連します。他の側面について、ニューロ・チェーンは前述のように機能します。

技術的側面では、チェーン内の各オブジェクトまたはコンセプトは、生存期間中に一意の ID を持ちます。オブジェクトに関連するすべての情報は、このオブジェクトの ID スタンプにメタデータとして追加されます。ネットワーク内のより良い情報の構成と一貫性を保つために、オブジェクトと取引の標準的なデータ構造が採用されています。

付録 2：通信プロトコルの概要

1. メール・ソフトウェアは、一般的に SMTP を送信のために、POP3 (Post Office Protocol 3) または IMAP (Internet Message Access Protocol) プロトコルを受信のために使用します。
2. その使用期間の長さにかかわらず、現段階では、主流の手段として SMTP に対等する代替技術はありません。ここで、このプロトコルがどのように動作するかを少し記述します。最新の電子メール・クライアントのプログラムは、すべて SMTP をサポートしています。電子メール・クライアントに SMTP の設定として、SMTP サーバーの IP アドレス (電子メールを受信する POP または IMAP サーバーのアドレスとともに) が格納されます。Web クライアントは、そのコンフィグレーションに、SMTP サーバーのアドレスが含まれ、PC クライアントには、ユーザーが独自のサーバーを指定できるようにする、SMTP 設定が用意されています。

3. 物理的な SMTP サーバは、専用で電子メールのトラフィックだけを処理するよう設定できますが、少なくとも POP3 や、他のプロキシ・サーバ機能と組み合わせられることが多いです。SMTP は TCP / IP 上で動作し、標準的な通信には TCP ポート番号 25 を使用します。SMTP を改善し、インターネット・スパム対策に役立てるため、標準化団体は、プロトコルの特定の機能をサポートするため、TCP ポート 587 を設定しました。Gmail などの一部のウェブ・メール・サービスでは、SMTP に非公式の TCP ポート 465 を使用します。SMTP は、「ストア・アンド・フォワード」と呼ばれるプロセスを使用して、電子メールをネットワーク上移動させます。SMTP は、メール転送エージェント (MTA) と緊密に連携して、インターネット上の適切なコンピュータと、電子メール受信トレイに送信します。

SMTP は、通常、電子メールのクライアント・アプリケーション内に統合され、4 つの主要コンポーネントで構成されています。

- メール・ユーザー・エージェント (MUA) と呼ばれる、ローカル・ユーザー端末のツール
- メール送信エージェント (MSA) と呼ばれるサーバー
- メール配達エージェント (MDA)
- メール転送エージェント (MTA)

SMTP は、電子メールのフローを管理するメールサーバー間の電子メールの通信を簡素化する、一連のコードを提供します。単純に、メッセージの様々な部分を、別のサーバーが理解できるように、様々なカテゴリに分割します。メッセージを送信すると、そのメッセージはテキスト文字列に変換され、文字列は、各セクションの目的を識別できるコードワードで区切られています。

これをニューロ・チェーンに適用して、ポットの特定のコンテキストでの通信に、SMTP プロトコルが使用されます。シンプルさと導入の容易さのために、企業やビジネスのコミュニケーションに適しています。このプロトコルを採用することで、脆弱性の原因となる新しい通信ポートを開く必要がないため、ビジネス上の重要なセキュリティの問題も解決できます。

4. HTTP : デフォルトのポートは TCP 80 ですが、他のポートも使用できます。このプロトコルは、コンピュータが互いに通信するための、標準的な方法を提供します。

クライアントとサーバーは、個々のメッセージを (データのストリームではなく) 交換することによって通信します。クライアント (通常は Web ブラウザ) から送信されるメッセージは要求と呼ばれ、サーバーから応答として送信されるメッセージは応答と呼ばれます。この通信は図 xxx に示されています。

HTTPS は、インターネット上で広く使用されているコンピュータ・ネットワーク上の安全な通信のためのプロトコルです。HTTPS は、Transport Layer Security またはその前身である Secure Sockets Layer で暗号化された接続内の HTTP (Hypertext Transfer Protocol) による通信で構成されます。HTTPS の主な開発動機は、訪問したウェブサイトの認証と、交換されたデータのプライバシーと完全性の保護です。

HTTPS は、通信相手のウェブサイトと関連ウェブサーバーの認証を提供し、「man-in-the-middle 攻撃」から保護します。さらに、クライアントとサーバー間の通信の双方向暗号化を提供し、盗聴や通信内容の改ざんから保護します。

結論として、このプロトコルは、交換における高レベルのセキュリティとスピードが必要な場合の、ポット通信に適合しています。

5. 現在、インターネットはハイパーテキスト転送プロトコル (HTTP) に基づいています。HTTP は、場所のアドレッシングに依存し、IP アドレスを使用して、要求された情報をホストしている特定のサーバーを識別します。これは、情報が要求される度に、ホスト・サーバーまたは CDN 内のサ

ーバーから、情報をフェッチする必要があることを意味します。図 10 は、IPFS の分散アーキテクチャを示しています。

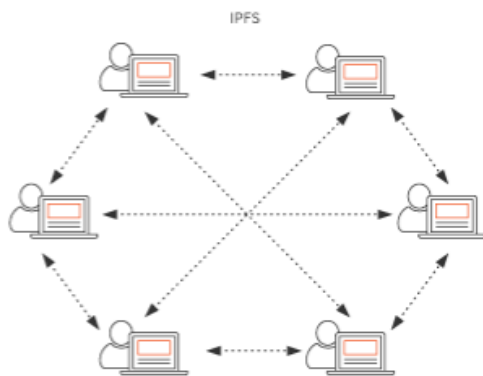
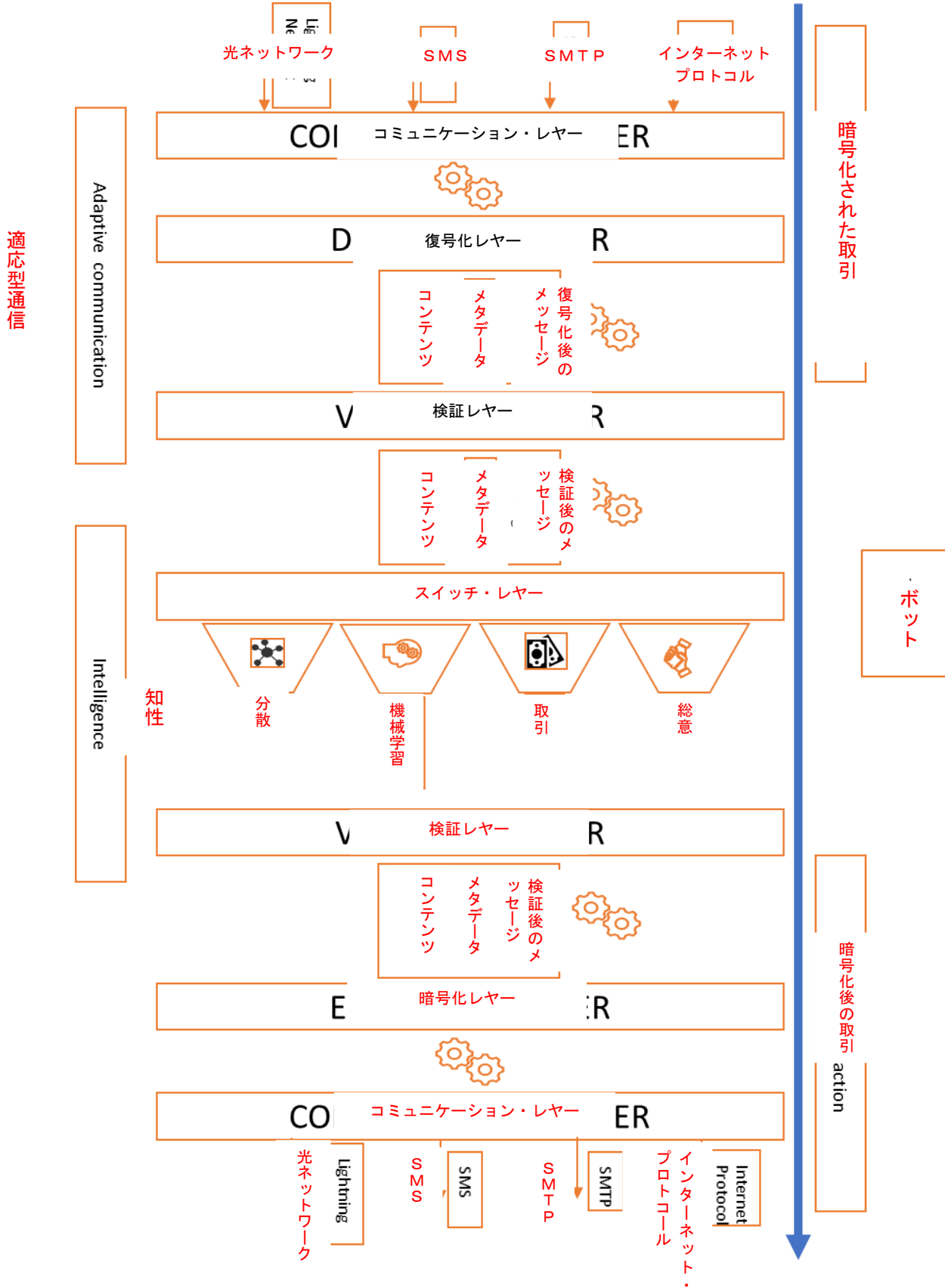


図 10：IPFS アーキテクチャの概略図。IPFS は、文書の暗号署名の分散データベースです。

付録3：ニューロ・チェーンの技術アーキテクチャー



付録 4：重み付きエントロピーの統計的解析

下記の解析は、ネットワークの特性に依存した、エントロピー計算の統計的分析に焦点を当てています。この解析は、システム内の重み付きエントロピーの進化に関する、直感的な感覚を与えます。

(The following programs/algorithm is not translated)

```
library(entropy) # Pour calculer l'entropie
library(igraph)

##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats ':
##
##   decompose, spectrum

## The following object is masked from 'package:base':
##
##   union

library(ggplot2)
```

- エントロピー計算関数

```
info <- function(CLASS.FREQ){
  freq.class <- CLASS.FREQ
  info <- 0
  for(i in 1:length(freq.class)){
    if(freq.class[[i]] != 0){ # zero check in class
      entropy <- -(freq.class[[i]] * log2(freq.class[[i]])) #I calculate the entropy for each class i here
    }else{
      entropy <- 0
    }
    info <- info + entropy # sum up entropy from all classes
  }
  return(info)
}
```

任意に生成されたN個のボット間の取引の解析は、次のようになります。

- 特定のボットと交換する全てのボットのIDを含むN個のベクトルを、任意に構成します。
- 取引の数をシミュレーションするため、各交換に乱数を適用します。
- ラウンドごとのボットの数、確率的でなければなりません。

```

N = 10000 # nombre du noeud dans le système
list.of.samples = lapply(1:N, function(x) {
  Subsize = sample(80:120) # choisir la taille du subgroupe aléatoirement
  Size = sample(200:300) # Choisir aléatoirement Le nombre d'échanges pour chaque noeud
  subgroupe <- sample(1:N, size = Subsize ) # construire le subgroupe de taille Subsize
  sample(subgroupe,size=Size, replace = TRUE)}) # construire le vecteur d'échanges
echange_length <- lapply(list.of.samples, function(x) length(x)) # vecteur contenant le volume d'échanges pour
chaque noeuds dans le système
echange_tot <- sum(unlist(echange_length)) # Le volume d'échange totale dans le système
echange_fraq <- unlist(echange_length)/echange_tot # la fraction d'échanges pour chaque noeuds

```

- 下記は、ポット 5 の交換ベクターの例です。

```

list.of.samples[5]
## [[1]]
## [1] 9997 6113 2355 3864 9555 2356 9948 7630 8077 1023 5955 9541 5141 9555
## [15] 4593 2884 8077 5623 2373 9105 7654 5231 5717 952 2017 6113 6109 9247
## [29] 4783 2884 9986 7691 7691 8077 2884 950 7635 7691 2634 7206 2017 3518
## [43] 952 9613 7206 4571 3473 9105 9247 5973 6250 8192 3144 1742 9971 5717
## [57] 6652 3864 9986 5993 5353 9189 7955 8077 3823 2373 2884 2355 2884 9971
## [71] 1023 756 5231 2884 2373 2355 5231 2356 7117 8116 5165 4200 952 3144
## [85] 3376 1742 1473 5567 4526 5353 1798 4783 4282 1968 3970 407 6336 4323
## [99] 2017 9541 6336 4200 9971 189 2355 5567 2758 6113 1023 2634 3376 7630
## [113] 7206 3970 3473 5623 9541 6349 5353 6349 2777 952 1742 952 2467 8116
## [127] 2956 5973 3518 3144 2884 6336 4323 3823 7654 6250 7117 4930 5165 7206
## [141] 7955 3144 4571 2017 6113 5993 3376 6128 9613 9559 6250 2777 5229 1817
## [155] 5955 4575 5973 6128 1023 5567 7630 1798 5165 189 9971 6109 4200 3093
## [169] 2373 3864 5353 3376 2017 9559 5231 5229 4571 9948 8681 1473 4282 4571
## [183] 7691 4783 4561 9555 8116 7117 952 9849 2758 1798 2634 8592 5623 952
## [197] 5141 9971 5955 3473 952 8681 5973 407 5955 5717 5623 5567 7117 1968
## [211] 3093 5973 4602 9971 5353 5231 8681 6652 7955 950 9541 6250 3186 952
## [225] 3144 8357 5850 9189 5353 2758 5231 8116 4593 5567 1968 1798 1798 3093
## [239] 5229 8681 3376 1023 1473 6336 6113 1473 2017 8192 7117 9105 3864 6109
## [253] 3473 3861 3864 952 6109 5850 5993 5231 4200 950 7630 9948 4602 4783

```

```
## [267] 1817 9555 4593 6109 3823 9189 2467 5993 6652 3186 7117 2467 7206 2355
## [281] 9541 3093 3140 3861 1742 1798 4526 9189 7206 5231 6652 1742 1023 3823
## [295] 1968 2634
```

そして、サブグループ内の、各ボットのエントロピー $E = - \sum P_i \cdot \log(P_i)$ を計算します。
その後、エントロピーは交換の比率で重みづけられます。

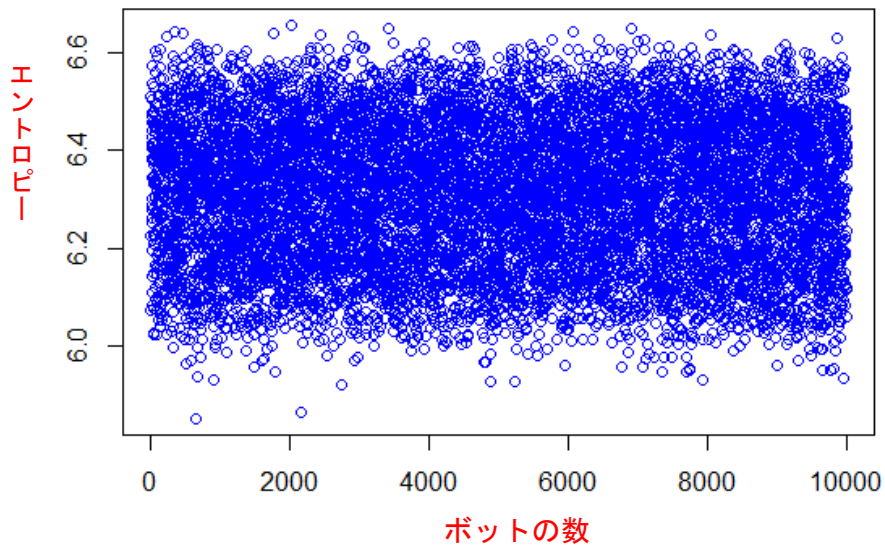
```
E_vect <- vector("numeric", N)
E_pondere <- vector("numeric", N)
for (i in 1:N){
  freqs = table(list.of.samples[i])/length(list.of.samples[i])
  E_vect[i] <- entropy.empirical(freqs, unit = "log2")
  E_pondere[i] <- E_vect[i]*echange_fraq[i]
}
```

- システムの図式的な表現

```
plot(E_vect,main = "Bots Entropies", ylab = "Entropy", xlab = "Bots", col = "blue")
```

```
#title(main = "Bots Entropies", ylab = "Entropy", xlab = "Bots")
```

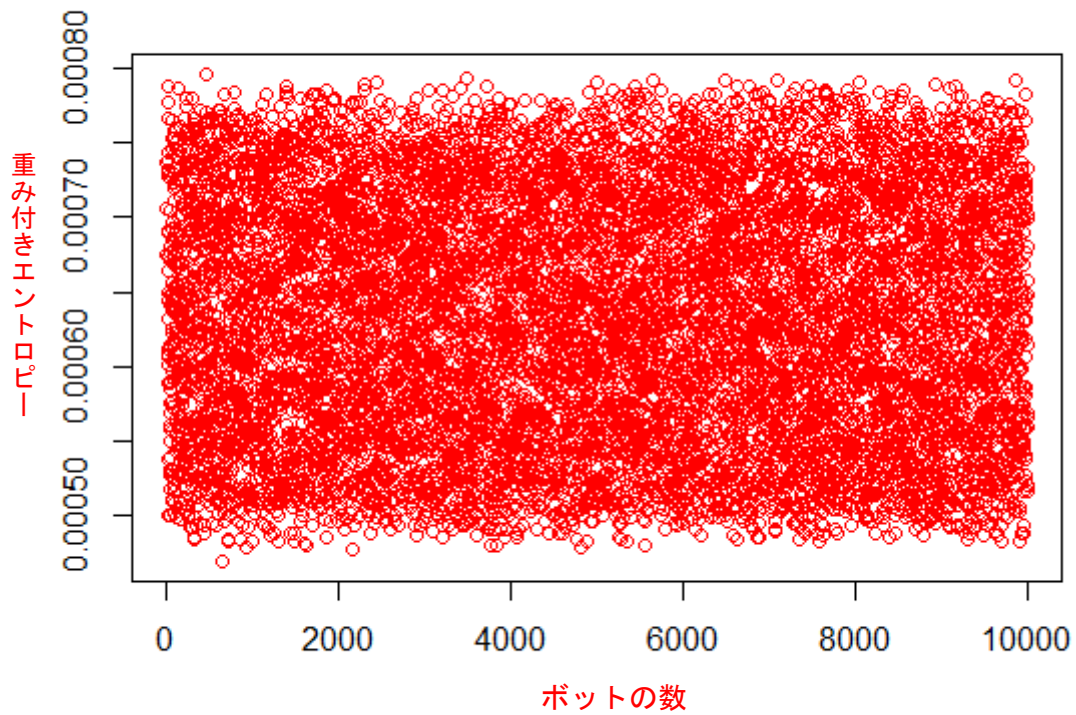
ボットのエン트로ピー



取引または交換の量を考慮に入れると、重み付きエン트로ピーは次のようになります。

```
plot(E_ponder, main = "Bots weighted Entropies", ylab = "Weighted Entropy", xlab = "Bots", col = "blue")
```

ボットの重み付きエン트로ピー



静止的な状態（時間進化なし）で、ネットワーク内のボットの数、各ボットのサブグループのサイズ、また取引の量が、ネットワークのエントロピー衝突へ感性に影響を与えます。

1. 衝突への感性は、サブグループのサイズに依存

```
N = 5000
M = 100
collisions.count = vector("numeric", M)
for (i in 1:M){
  collisions.mean = vector("numeric", 30)
  for (j in 1:30) {
    list.of.samples = lapply(1:N, function(x) {
      Subsize = sample((110 - i) : (220 - i)) # choisir la taille du sousgroupe aléatoirement

      Size = sample(200:300) # Choisir aléatoirement Le nombre d'échanges pour chaque noeud

      sousgroupe <- sample(1:N, size = Subsize ) # construire le sousgroupe de taille Subsize

      sample(sousgroupe,size=Size, replace = TRUE)}) # construire le vecteur d'échanges

    echange_length <- lapply(list.of.samples, function(x) length(x)) # vecteur contenant le volume d'échanges pour
    chaque noeuds dans le système

    echange_tot <- sum(unlist(echange_length)) # Le volume d'échange totale dans le système

    echange_fraq <- unlist(echange_length)/echange_tot # la fraction d'échanges pour chaque noeuds

    E_vect <- vector("numeric", N)
    E_pondere <- vector("numeric", N)
    for (k in 1:N){
      freqs = table(list.of.samples[k])/length(list.of.samples[k])
      E_vect[k] <- entropy.empirical(freqs, unit = "log2")
      E_pondere[k] <- E_vect[k]*echange_fraq[k] }

    collisions.mean[j] = N - length(unique(E_pondere))
```

```

}

collisions.count[i] = N - round(mean(collisions.mean))

}

print("Le nombre de collisions obtenue en dimiuant la taille du sousgroupe\n")
collisions.count
## [1] 4974 4974 4976 4974 4976 4976 4977 4978 4978 4978 4980 4979 4982 4980
## [15] 4979 4981 4982 4981 4983 4983 4983 4985 4984 4984 4985 4985 4986 4987
## [29] 4987 4987 4986 4988 4988 4988 4989 4988 4989 4989 4989 4990 4991 4991
## [43] 4991 4992 4991 4992 4991 4992 4992 4992 4993 4993 4994 4993 4993 4994
## [57] 4994 4994 4994 4994 4995 4994 4995 4995 4995 4995 4995 4996 4996 4996
## [71] 4996 4997 4996 4996 4997 4997 4997 4997 4998 4998 4997 4997 4997 4998
## [85] 4998 4998 4998 4998 4998 4998 4998 4998 4998 4999 4998 4998 4998
## [99] 4998 4999

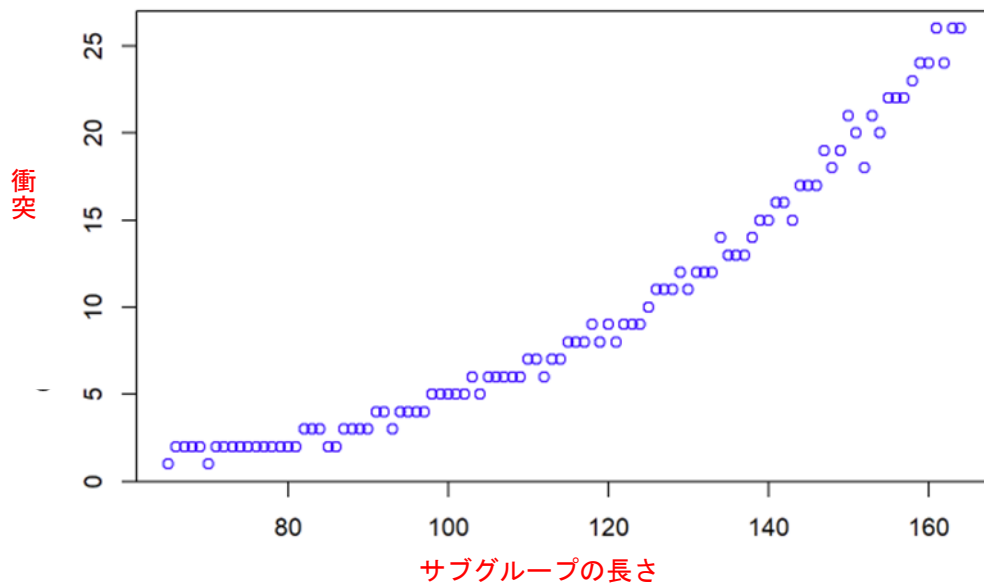
# vecteur taille moyenne des sousgroupe
mean_size = vector("numeric", length(collisions.count))

for (i in 1:length(collisions.count)){
  mean_size[i] = mean(sample((110 - i):(220 - i)))}

plot(mean_size,N - collisions.count, main = "l'évolution des collisions en fonction de la taille du sousgroupe ", yla
b = "nombmre de collisions", xlab = "taille du sousgroupe", col = "blue")

```

サブグループの長さに対する衝突への感性



グラフからはっきりしたように、衝突への感性はサブグループ（複数のポットがある特定のポットの周りに集中する）の長さと共に増加します。

2. 衝突のへは感性は、取引量に依存

```
N = 5000
M = 75
collisions.count1 = vector("numeric", M)
for (i in 1:M){
  collisions.mean = vector("numeric", 30)
  for (j in 1:30) {
    list.of.samples = lapply(1:N, function(x) {

      Subsize = sample(100 : 200) # choisir la taille du sousgroupe aléatoirement

      Size = sample((200 - 2*i):(300 - 2*i)) # Choisir aléatoirement Le nombre d'échanges pour chaque noeud

      sousgroupe <- sample(1:N, size = Subsize ) # construire le sousgroupe de taille Subsize

      sample(sousgroupe,size=Size, replace = TRUE)}} # construire le vecteur d'échanges

      echange_length <- lapply(list.of.samples, function(x) length(x)) # vecteur contenant le volume d'échanges pour
      chaque noeuds dans le système
```

```

echange_tot <- sum(unlist(echange_length)) # Le volume d'échange totale dans le système

echange_fraq <- unlist(echange_length)/echange_tot # la fraction d'échanges pour chaque noeuds

E_vect <- vector("numeric", N)
E_pondere <- vector("numeric", N)
for (k in 1:N){
  freqs = table(list.of.samples[k])/length(list.of.samples[k])
  E_vect[k] <- entropy.empirical(freqs, unit = "log2")
  E_pondere[k] <- E_vect[k]*echange_fraq[k] }

collisions.mean[j] = N - length(unique(E_pondere))

}

collisions.count1[i] = round(mean(collisions.mean))

}

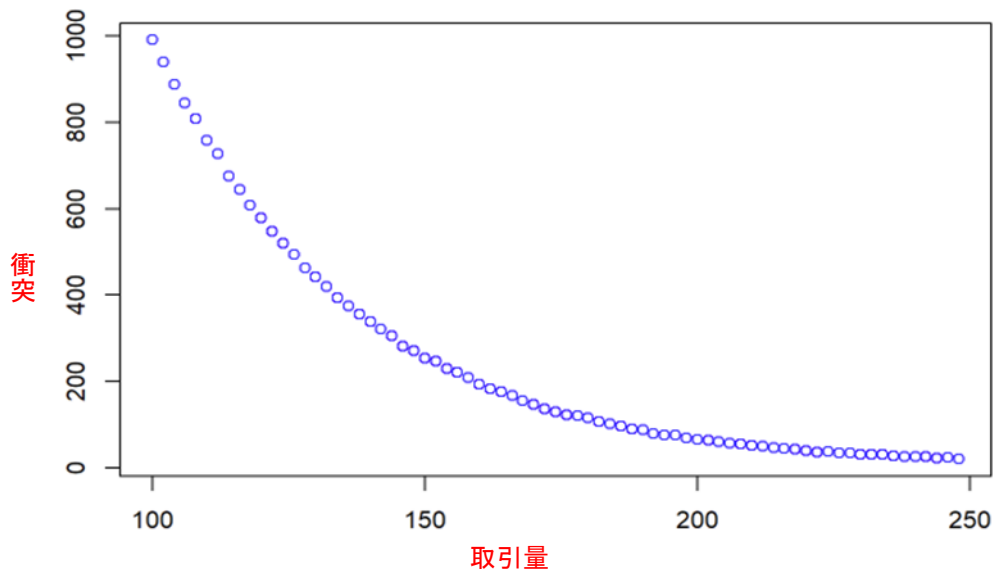
print("Le nombre de collisions obtenue en dimiuant le volume d'échanges par noeuds\n")
## [1] "Le nombre de collisions obtenue en dimiuant le volume d'échanges par noeuds\n"
collisions.count1
## [1] 20 23 22 26 26 26 28 31 30 30 34 34 37 36 40 43 44
## [18] 47 49 51 55 57 60 63 66 68 75 76 79 87 89 96 101 106
## [35] 116 120 122 130 136 146 155 167 176 183 193 209 221 229 246 253 271
## [52] 281 306 321 339 356 375 393 419 442 463 494 519 547 578 608 645 676
## [69] 727 759 808 845 888 939 991
# vecteur taille moyenne des sousgroupe
volume_echange = vector("numeric", length(collisions.count1))

for (i in 1:length(collisions.count1)){
  volume_echange[i] = mean(sample((200 - 2*i):(300 - 2*i)))}

```

```
plot(volume_echange,collisions.count1, main = "l'évolution des collisions en fonction du volume d'échange par noeud ", ylab = "nombmre de collisions", xlab = "volume d'échange moyen par noeuds", col = "blue")
```

取引量に対する衝突への感性



取引量が増加すると、衝突の数は劇的に減少します。

3. 衝突への感性は、ネットワークのサイズに依存

```
N = 7000
collisions.count2 = vector("numeric", 30)
n = 0
for (i in seq(1000,N,200)){
  collisions.mean = vector("numeric", 20)
  for (j in 1:20) {
    list.of.samples = lapply(1:i, function(x) {

Subsize = sample(100 : 200) # choisir la taille du sousgroupe aléatoirement

Size = sample(200:300) # Choisir aléatoirement Le nombre d'échanges pour chaque noeud

subgroupe <- sample(1:i, size = Subsize ) # construire le sousgroupe de taille Subsize
```

```

sample(subgroupe,size=Size, replace = TRUE)) # construire le vecteur d'échanges

echange_length <- lapply(list.of.samples, function(x) length(x)) # vecteur contenant le volume d'échanges pour
chaque noeuds dans le système

echange_tot <- sum(unlist(echange_length)) # Le volume d'échange totale dans le système

echange_fraq <- unlist(echange_length)/echange_tot # la fraction d'échanges pour chaque noeuds

E_vect <- vector("numeric", i)
E_pondere <- vector("numeric", i)
for (k in 1:i){
  freqs = table(list.of.samples[k])/length(list.of.samples[k])
  E_vect[k] <- entropy.empirical(freqs, unit = "log2")
  E_pondere[k] <- E_vect[k]*echange_fraq[k] }

collisions.mean[j] = i - length(unique(E_pondere))

}

collisions.count2[n] = round(mean(collisions.mean))
n = n + 1 }

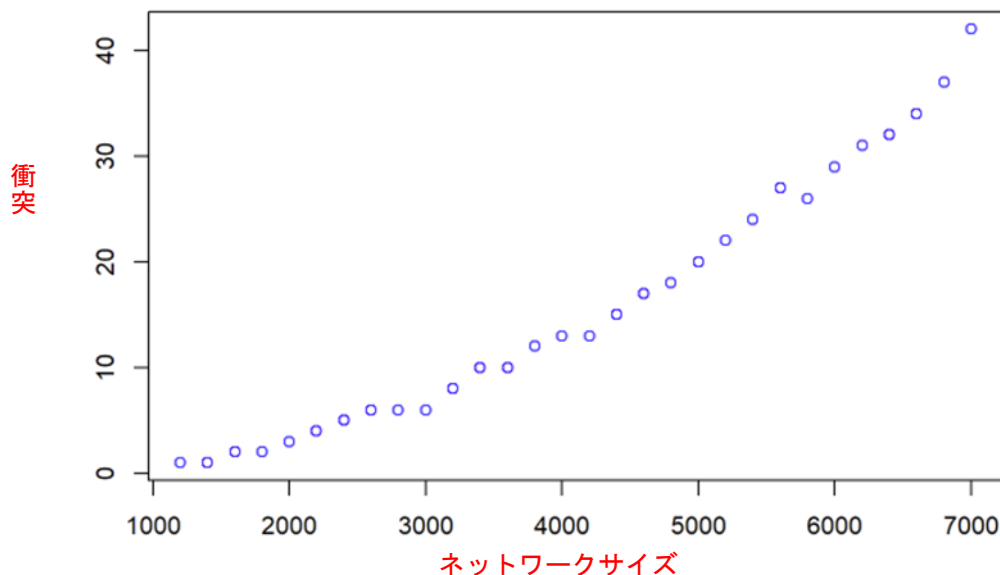
print("Le nombre de collisions obtenue en augmentant le nombre de noeuds dans le système \n")
## [1] "Le nombre de collisions obtenue en augmentant le nombre de noeuds dans le système \n"
collisions.count2
## [1] 1 1 2 2 3 4 5 6 6 6 8 10 10 12 13 13 15 17 18 20 22 24 27
## [24] 26 29 31 32 34 37 42
# vecteur taille moyenne des sousgroupe
Taille_Systeme = vector("numeric", length(collisions.count2))

for (i in 1:length(collisions.count2)){
  Taille_Systeme[i] = 1000 + i*200}

```

```
plot(Taille_Systeme,collisions.count2, main = "l'évolution des collisions en fonction de la Taille du Système ", ylab = "nombmre de collisions", xlab = "La taille du Système", col = "blue")
```

ネットワークサイズに対する衝突への感性



上図に示すように、ネットワークのサイズ（システム内のロボットの総数）の増加に応じて、衝突が増加することがわかります。しかし、増加の曲線は、サブグループのサイズによる増加と比較して小さいです。

結論：ここでの統計的解析は、2つの仮定の下で行われました：システムの**定常性**と取引の任意性。実際には、分散ネットワークを運用する場合、システムは定常ではなく、取引はランダムですが、決定的であり、場合によっては非線形です。しかし、非現実的な仮定の下でありながらも、この解析は、ネットワークサイズ、取引量、またはサブグループサイズなどのネットワーク・パラメータとして、重み付けエントロピーの進化を関数化でき、ある程度の結果を与えました。

分析の結果、重み付けされたエントロピーの衝突は、ネットワークのサイズと共に増加し、サブグループのサイズの増加によって、より大きい変化量で、増加することが示されています。取引の量に関しては、衝突がその増加すると共に減少します。

付録5：任意性とカオス的プロセス

本文を読むときの正当な質問は、ニューロ・チェーンとランダム・プロセスの関係は何かです。答えは簡単です：ニューロ・チェーンの総意アルゴリズムでは、アセンブリ構成でのリーダー選出は、無作為サンプリングに基づく確率論的プロセスです。この任意性の性質は、信頼性とシステムの予測不可能性を保証するための原点です。

コンピュータサイエンスでは、すべてのプログラムが決定的（命令のシーケンス）であるため、任意性（Gregory J. Chaitin 教授）の達成は困難です。乱数を生成するには、擬似乱数ジェネレータと真

の乱数ジェネレータの 2 つの方法があります。両者のアプローチには、異なった特徴を持っています (Dr Christopher Wetze)。

擬似乱数ジェネレータ：擬似は、プロセスがランダムではなく、厳密に言えば、サイコロの程度という意味です。擬似乱数ジェネレータは、数学または事前計算テーブルに基づくアルゴリズムであり、ランダムに現れる数列を生成します。アルゴリズムの一例は線形合同法です。秘密鍵と公開鍵を生成する際に、暗号化技術によって利用されているといった、様々な用途があるため、ここ数年、アルゴリズムの研究が活発的に行われてきました。

真の乱数ジェネレータ：この場合、物理現象から任意性を抽出し、それをコンピュータ環境に接続します。これは、コンピュータに接続されたサイコロに似ています。科学者は、コンピュータへの接続が容易な実用的な物理現象を使用します。実際には、物理的な現象はマウスの動きのわずかな変化や、キーストローク間の時間のように単純です。しかし、これらの情報源には限界があります。良好な物理現象の 1 つは、放射性物質の減衰する時点であり、完全に予測できないものです。また、簡単にコンピュータに接続することもできます。もう一つの現象は、通常のラジオで簡単にとらえられる大気騒音です。また、エントロピー・ベースの方法もあり、例えば、ウェブサーバのエントロピー・プールが受信する、ウェブページのヒットのような、様々なソースを収集します。

次の表は、両方のアプローチの違いを示しています。

特性	疑似の乱数ジェネレータ	真の乱数ジェネレータ
効率	優れた	悪い
決定的か	決定的	非決定的
周期性	周期を持つ	周期を持たない

カオス的なプロセスと量子事象

ランダムなように見えて、実はそうではない他のプロセスがあります。最も有名なのは、カオス的なプロセスです。カオス的なプロセスは、決定的な非線形過程であり、次の 2 つの主な特徴を示します。

- 初期条件に対する強い依存性
- 高い周期性を持つ、つまり、システムが特定の位置で出会ると、無限に、この位置で再び出会うこととなります。

カオス的なプロセスの良い例は、いわゆるバタフライ効果で、初期条件の小さな変化が、システムに大きな影響を与えます。

いくつかのケースで、カオス的なプロセスは、任意性を持つかのような振る舞いと、モデル化の容易さから、乱数生成のための強力な方法になり得ます。ランダム・プロセスは、非常に複雑なカオスプロセスとして考えることができるため、ランダム・プロセスとカオス・プロセスの境界線は非常に微妙です。

一例として、大気ノイズを用いて乱数を生成することができます。しかし、これらの数を予測するには、非線形システムをモデル化するため、大気各粒子の位置と速度を知る必要があります。この難しさは、簡単に理解できません。

量子力学は、数学を使って原子レベルと準原子レベルで、宇宙を表現する物理学の分野です。小準原子レベルの粒子は、ランダムに振る舞いますが、これらの事象の原因については、何も知られていません。結果として、これらの事象は、**本質的に非決定的**であると考えられます。この場合、量子事象は任意性を生成するために、使用可能です。

最終的には、主にその定義に依存しますが、任意性は、人間やより大きなものが予測できない特性を指す、と言えます。宇宙は、その存在自体が、ランダムではないからです。

各ボットに対して一貫したランダムプロセスを行った後、このプロセスを n 個のボットに分散します。そのためには、数学的関数を使用して、ネットワークの総意を達成する方法が強力になります。以下は、暗号化技術から導かれた1つの提案された方法です。

$$Ran_n = \sum_n A_{al} \text{ mod } n$$

ここで、 Ran_n はネットワーク上の乱数で、 A_{al} は各ボットの乱数です。 Ran_n は、すべてのボットで、同じ値を取ることが期待されます。

結論として、任意性は選出プロセスにとって非常に重要であり、また、使用される分散型ランダム選出の方法は、ニューロ・チェーンが要求するセキュリティのレベルと複雑さに応じて使い分けられます。

参考文献

- A. Chao and T-J. Shen. (2003). サンプル中に見えない種がある場合のシャノンの多様性指数のノンパラメトリックな推定. *環境・生態統計*, DOI: 10.1023/A:1026096204727.
- A.N. Kolmogorov. (1965). 情報量の概念の定義に対するこれらのアプローチ. *Problemy Peredachi Informatsii* 1, 3-11.
- Benet, J. (2016). *分散されたウェブ*. MIT Licence.
- C.E. Shannon. (1948). コミュニケーションの数学理論. *ベルシステム技術ジャーナル*, vol. vol. 27, p. 379-423 and 623-656.
- Darwiche Adnan. (2009). ベイジアンネットワークによるモデリングと推論. *Cambridge University Press*, ISBN 978-0521884389.
- Dr Christopher Wetze. (n.d.). *ランダムに行動できますか？*
- Gelman, A., Carlin, J. B., Stern, H. S., & Donald B Rubin. (2003). ベイジアンデータ解析の基礎：第5章階層モデル. *CRC Press*. ISBN 978-1-58488-388-3., 120.
- Giarratano, J. C., & Gary Riley. (1998). *あエキスパート・システム*. PWS Publishing Co. Boston, MA.
- Goldreich Oded. (2008). 計算の複雑さ：概念的な視点. *Cambridge University Press*.
- Gupta, A., Forgy, C., Newell, A., & Wedig, R. (1986). ルールベースのシステムのための並列アルゴリズムとアーキテクチャ. *ACM Digital Library*.
- (n.d.). *Hypertext Transfer Protocol 1.0*.
- Lampport, L. (1978). 時間、クロック、および分散システムにおけるイベントの順序. *Communications of the ACM*, vol. 21, no 7, p. 558-565.
- Leslie Lampport. (2004). 非同期コンセンサスの下限.
- Marshall Pease. (1980). 障害が存在する場合の合意に達する. *コンピューティング・アソシエーション学会誌*, vol. 27, no 2.
- Myers, S. A., Zhu, C., & and J. Leskovec. (2012). ネットワークにおける情報拡散と外部からの影響. *SIGKDD*, 33.
- Nakamoto, S. (2009). ビット・コイン：ピアツーピア電子式現金システム. 4.
- P. W. Atkins. (1998). *Éléments de chimie physique*. De Boeck Université.
- Prof. Gregory. J. Chaitin. (n.d.). *任意性を探る：アルゴリズム情報理論*.
- (1995). *メッセージサイズ宣言のためのSMTP サービス拡張*. RFC1870.

Vandervort D, G. D. (n.d.). ビット・コインのようなコミュニティ通貨を設計する際の問題. *Brenner M, Christin N, Johnson B, Rohloff K,*

Yang, J., Chen, B., & and D. Agarwal. (2013). ソーシャルデータ較正による共有者評判の推定. *SIGKDD*, 59.