

DXとAIが変革するゲーム技術の未来

東京理科大学 理工学研究科 情報科学専攻 客員准教授
株式会社 Cygames 技術顧問 / Cygames Research 所長

くらばやし しゅういち
倉林 修一

はじめに

あらゆる産業において持続的な成長を実現するためには、優れた研究開発とそれによるイノベーションが必要不可欠です。ゲーム産業においても、これまでにないゲームを開発するためには、研究により新しい知識を創造し、その知識を実践するという“知識創造とイノベーションの相互作用”を実現できる仕組みや組織体制が求められています。このイノベーションを加速する仕組みとして近年注目されているのが、デジタル・トランスフォーメーション（DX）と呼ばれる情報技術の社会実装です。DXとは、生活や産業の一連のプロセス全体をデジタル化し、社会にデジタル化の新たな価値を浸透させていく活動を意味します。このようなプロセス全体の変革を伴うDXを実現するためには、これまで情報通信技術（ICT）の利用者という立場にあった企業や個人が、技術研究や開発の主体として情報技術の浸透に貢献していくことが本質的に重要です。この流れはゲーム産業においても例外ではなく、ゲームを開発する企業が基礎技術研究所を設立しDXを含むさまざまな計算機科学の技術開発に取り組むようになってきています。そこで本稿では、筆者が所属するゲーム企業内の研究所であるCygames Researchの研究事例をご紹介します。DXとAIが作り出すゲーム技術の未来について論じます。

計算機科学から見たオンラインゲーム

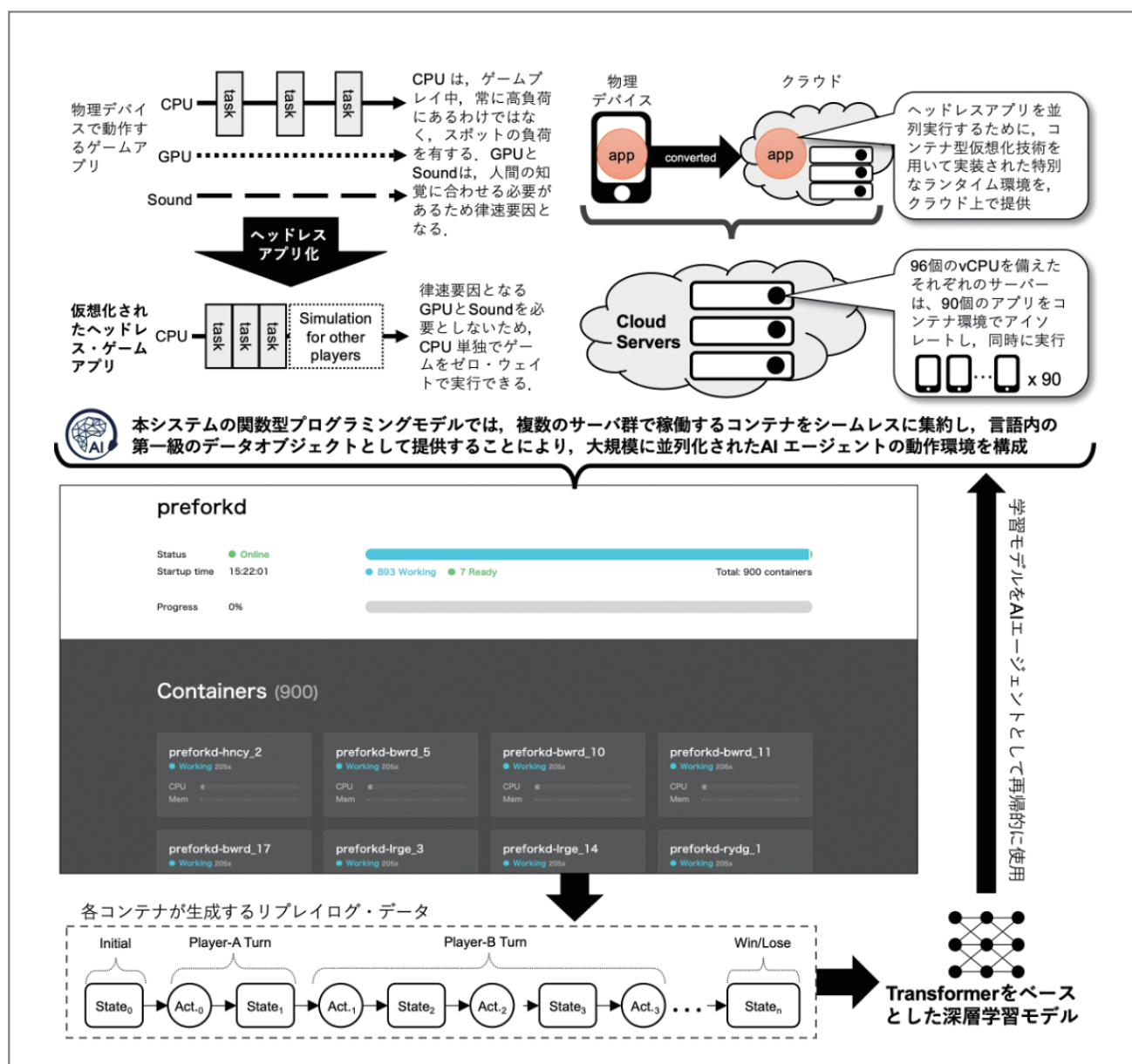
ゲームは質的にも量的にも大規模な情報システムであり、我が国のDXを考察する上で格好の題材だと言えます。具体的には、オンラインゲームの実装には、動画処理技術、コンピュータグラフィックス技術、サウンド技術、ユーザインターフェイス技術、分散コンピューティング技術、データベース技術、ネットワーク技術、ストレージ技術など、多様な計算機科学の分野の知見が導入されています。さらにオンラインゲームの運用では、2,700万人以上のユーザから、秒間28万回のリクエストを受け付け、秒間270万件のデー

タベースクエリを処理するという、極めてスケーラブルなバックエンドシステムも必要です。また、魅力的なオンラインゲームを提供するために、キャラクターやアイテム等のゲーム要素を毎月のように頻繁に追加し続けることや、ゲーム要素の組合せ爆発（例えば2,500の40乗）を起こすゲームシステム、すなわち、有限時間で最適解を発見することが困難なシステムの設計と構築が求められます。このように、実装に用いる技術の多様性、運用時に要求されるスケーラビリティの高さ、そして継続的な発展のためのNP困難なシステム要件（多項式時間で最適解を発見できないこと）という点で、ゲーム開発にはDXによるイノベーションの促進が強く求められているのです。

自動デバッグAIによるDX

Cygames ResearchではDXを「組織全体を貫くデータパイプラインと、イテレーション指向の組織の組み合わせにより、Software-Defined Companyを実装すること」と定義しています。ここで、Software-Defined Companyとは、人や書類ではなく、ソフトウェア間の通信で駆動する企業のことを意味します。すなわち、ソフトウェアで動く組織を作り上げる事がDXなのです。このようにソフトウェアで駆動する、ということの端的な事例が、当研究所が開発し実用化した自動デバッグAI【図1】です。自動デバッグAIは、スマートフォン用ゲームをエンド・ツー・エンドで仮想化して大規模に並列起動し、それらをAIエージェントが自動的にプレイすることができます。

本システムの基盤技術として、ゲームシステムをクラウド上で大規模に動作させるための、関数型プログラミングのパラダイムに基づく仮想化・並列化技術を開発しました。関数型プログラミングとはソフトウェアの「部品」としての関数ではなく、より数学的な意味での関数として、同じ引数を与えれば同じ結果が得られるという参照透過性を持つ関数の組み合わせでソフトウェアを実装するという考え方です。具体的には、クラウド上の仮想マシン／コンテナ／分散ストレージ

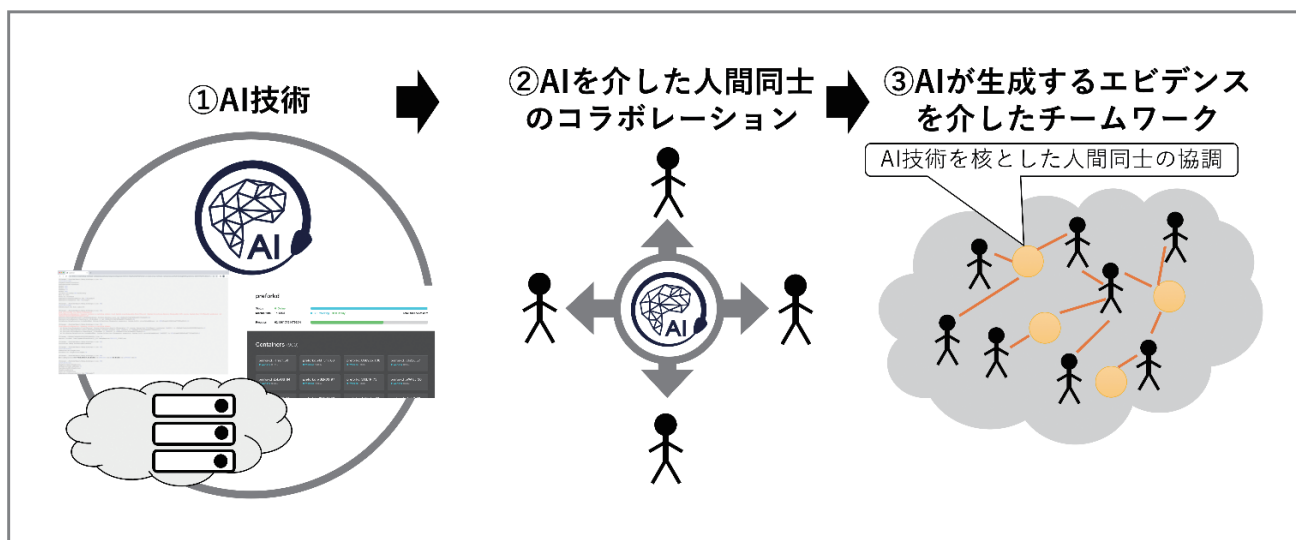


【図1】自動デバッグAI

を透過的に制御する独自の関数型プログラミング言語処理系を開発し、参照透過性と冪等性を持つ演算の繰り返しとして自動デバッグのプロセスをモデル化したのです。この関数型言語では、ゲームシステムを駆動させる処理を、稼働中のクラウド環境への冪等な関数適用の列として定義できるため、任意の関数適用の際にゲーム環境全体や一部を一時停止し、検証用スナップショットとして保存できます。これにより、96個の仮想CPUを有する1台のサーバ上で、90個のゲームプロセスをフルパラレルで動作させる高密度性を達成すると共に、Transformer等の時系列データの学習に適した深層学習モデルを用いてゲームのプレイ履歴を学習させられるようになりました。

この自動デバッグAIの実稼働として、運用中の

ゲーム開発環境を毎日ミラーリングし、その中でAIが自動デバッグを行うという「継続的デバッグ（Continuous Debugging, CDB）」を、ゲーム開発と並行して常時稼働させています。継続的デバッグの実用化のインパクトは大きく、1日8時間労働の人間に換算して約3千人分の作業に相当する、24時間で延べ2万時間以上の自動デバッグを常時稼働させ、ゲームの開発効率を大幅に向上させることができました。自動デバッグAIは、ゲーム内の不具合を見つけ出すと、その不具合が起きた状況と再現するために必要な操作履歴一式をまとめたレポートを自動生成します。このレポートが毎日ゲーム開発者に通知されるため、開発者はそのレポートに沿って不具合の改修を行うことができます。この自動デバッグAIの実用



【図2】AIを中核とした組織変革

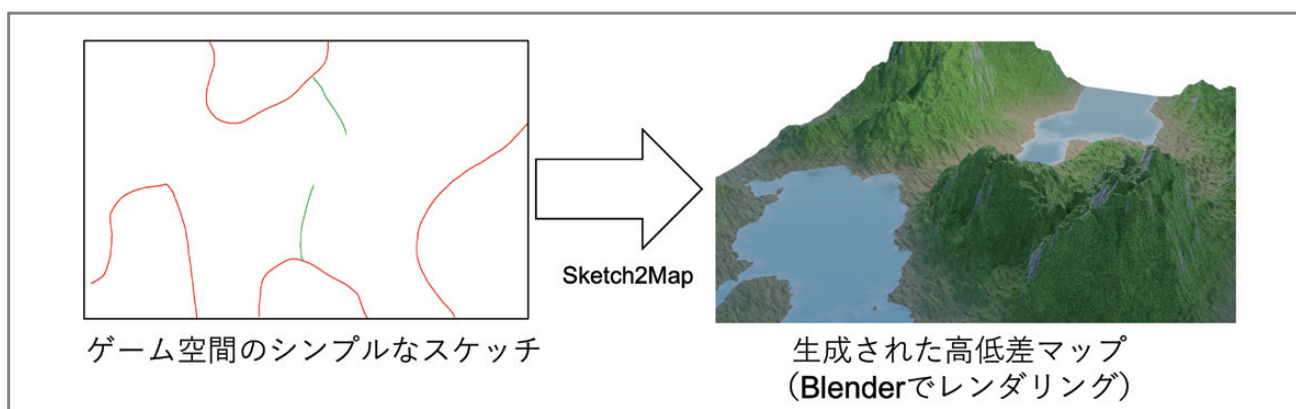
化は、単に個別の技術者を支援するだけでなく、自動デバッグAIが生成するデータやレポートを介した人間同士のコラボレーションを促進しました。不具合を改修するための議論を人間同士が行うときに、AIが生成したレポートのURLを識別子として使うようになったため、リモートワークでも綿密で具体的なコミュニケーションが実現できました。自動デバッグAIの実用化は、組織全体にAI技術を核とした自律・分散・協調の仕組みをもたらしたと言えます【図2】。

GANによるゲームデータの生成

ゲーム開発では、背景、マップ、キャラクターのパラメータなど、大量のデータを作る必要があります。従来、これらのデータは人手により作られてきましたが、現在ではAIを活用して自動生成、もしくは、人手の監修を介する半自動生成が行われるようになって

きています。さまざまなデータやパラメータを自動的に生成することができれば、ゲームの開発効率が上がるだけでなく、ゲーム中の要素をよりバリエーション豊かなものにすることができるため、この分野は活発に研究されています。この自動生成で活用される技術が、敵対的生成ネットワーク（Generative Adversarial Networks, GAN）というデータの自動生成に適したAIです。

まず、このGANを用いたオープンワールドゲーム用のマップ生成システムとして、当研究所が開発したSketch2Mapをご紹介します。Sketch2Mapは海岸線の輪郭や川の輪郭などの手描きのスケッチを入力するだけで、その輪郭と整合するリッチな3D地形マップを自動生成するAIです。このAIを使うと、ゲームデザイナーは、数分間で書けるようなラフなスケッチ画像から【図3】のような3Dの高低差や地形を持ったマップを得られます。Sketch2Mapの特徴は、手書



【図3】マップ自動生成AIのSketch2Map



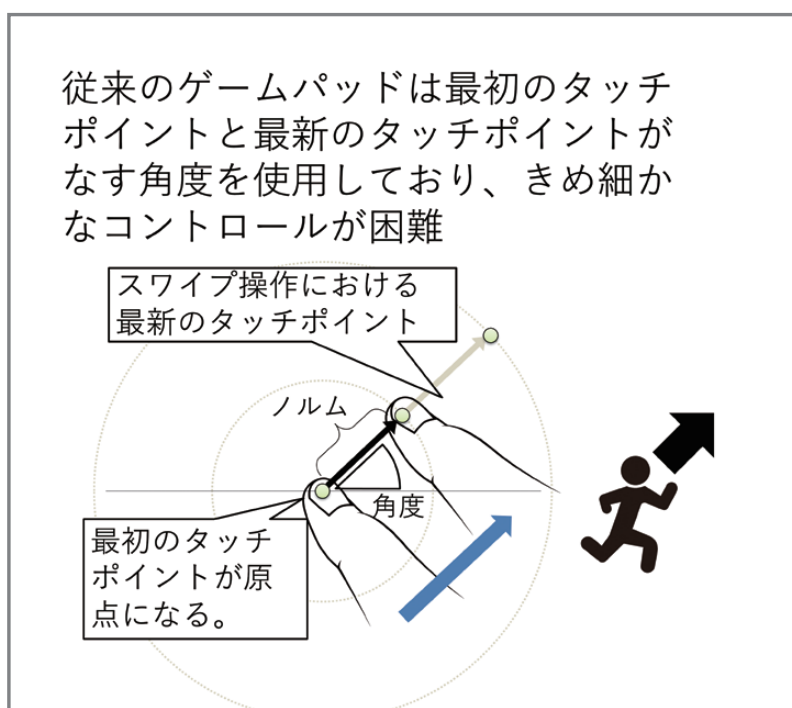
【図4】次世代バーチャルパッド Kinetics (スクリーンショット内のキャラクタモデルとしてUnity-Chan © UTJ/UCL を使用しています)

きスケッチを標高ビットマップに変換するcGANと、レベルアセットを生成する決定論的アルゴリズムで構成される2段階の生成モデルにあります。これにより、生成される3Dマップが多種多様なものになるようにしながら、生成されるデータの品質を高いレベルに保つことが可能になりました。

次に、GANの可能性を広げる基礎研究としてゲームの中でよく使われる乱数(サイコロを振って得られる

ような、恣意性のない値)をGANに学習させることに成功した研究をご紹介します。従来は、乱数を生成するアルゴリズム(乱数生成器)を設計・実装するためには、高度に専門的な数学の知識が要求されていました。しかし、当研究所が開発したLearned PRNG(学習型擬似乱数生成器)では、既存の乱数生成器が生成した乱数から、誰もがGAN学習として新たな乱数生成器を実装できるようになりました。さらに、こ

のLearned PRNGが生成した乱数が、信頼できる乱数であることを検証するために、米国商務省標準技術局(NIST)が開発したNIST Special Publication 800-22(NIST SP 800-22)と呼ばれる、乱数を統計的に検定するツールキットによる検定に全て合格できることも評価実験を通じて確認しました。このNIST SP 800-22は、15種類の検定法に基づく188個の試験からなる、厳密なテストスイートであり、この検定をパスした乱数生成器が、ゲームや一般のアプリ開発などで広く活用されています。Learned PRNGが生成した乱数列は、このNIST SP 800-22テストスイートに完全に合格したため、乱数という複雑な挙動を示すデータを、ニューラルネットワークが学習できる可能性を示唆することができました。また、ニュー



【図5】従来のバーチャルパッド

ラルネットワークのファインチューニング（特定の問題に適した処理ができるように微調整すること）を通じて、新しい乱数生成器を無尽蔵に作り出せるようになる可能性もあります。

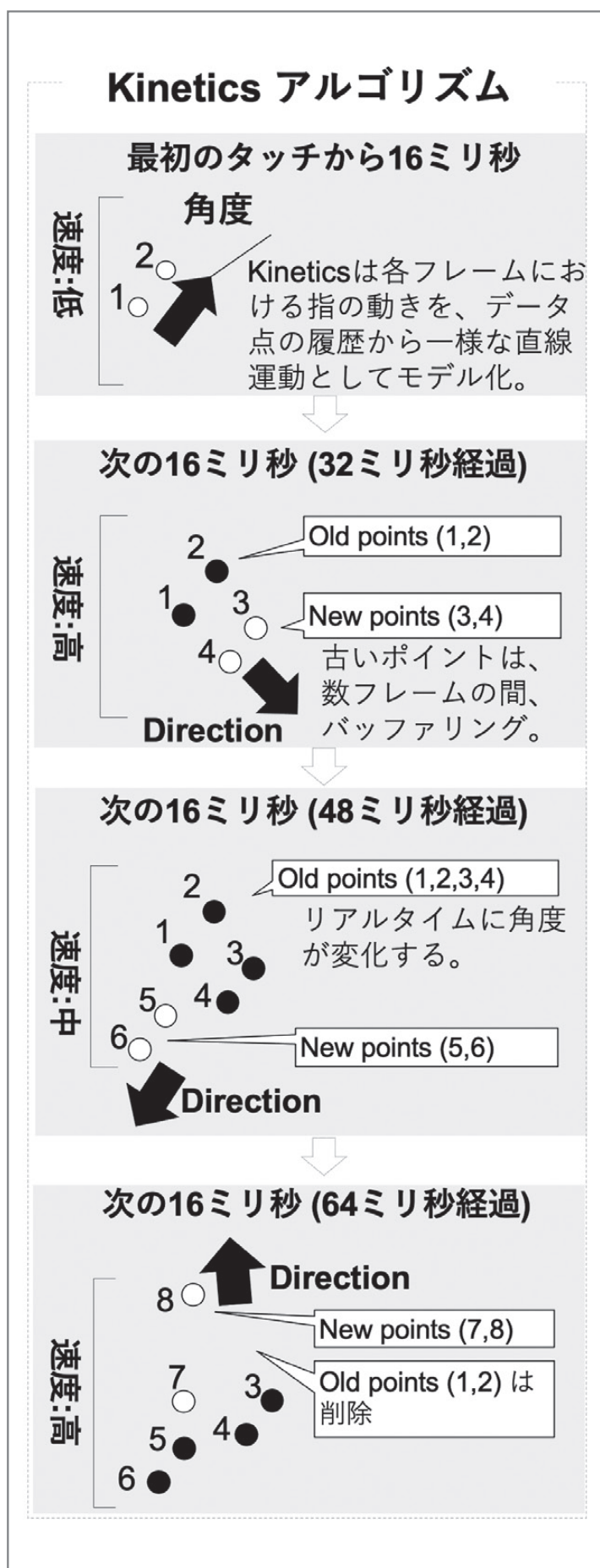
次世代ゲームパッドの研究

DXの根幹となる大規模データを活用するのは、必ずしも開発時だけではありません。現代のスマートフォンでは、利用者がタッチスクリーン上で指を軽く動かしただけで0.01秒以内に10個程度のタッチポイントを見つけ出す事ができます。単純計算では、1分間で6万個のタッチポイントが得られることになります。すなわち、スマートフォンの画面は、それ自体がビッグデータの発生源とみなすことができるのです。

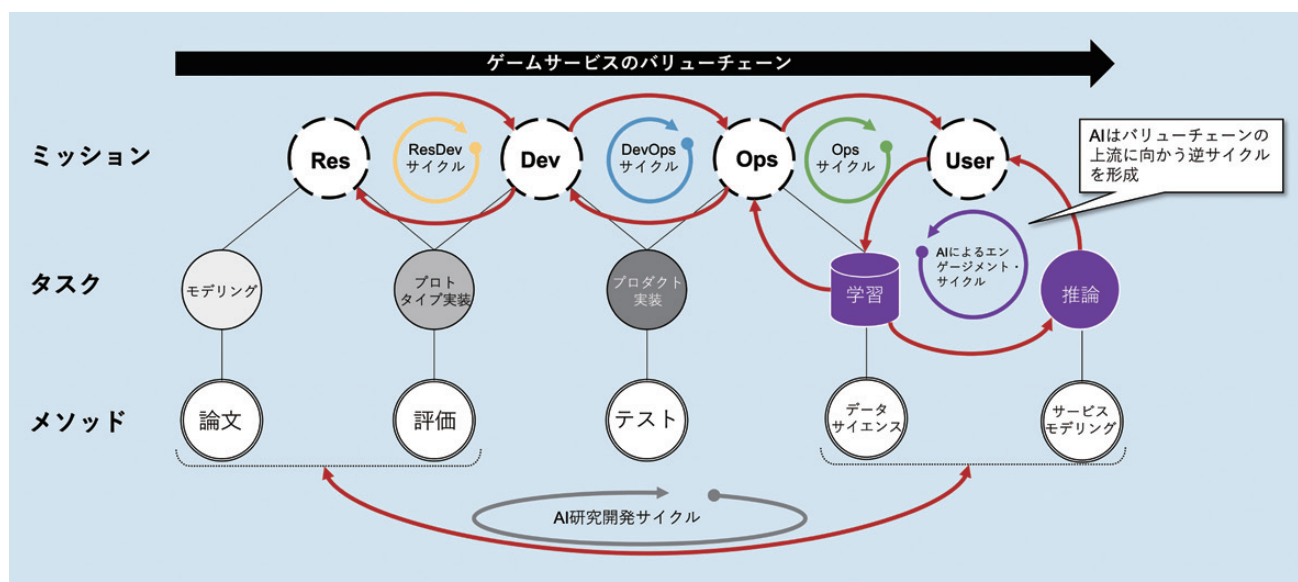
そこで当研究所では、瞬間的に入力されるビッグデータとも言えるこの6万個のタッチポイントを活用して、新しいバーチャルパッドである、Kineticsを開発しました【図4】。

従来の一般的なバーチャルパッドは、タッチの開始点と終点の2点がなす角度と距離を用いてキャラクタの移動方向と速度を算出します【図5】。一方、当研究所の新しいゲームパッドのKineticsはタッチスクリーンから得られる指の動きに対応するx座標とy座標の数列に、機械学習（AI）のテクニックの一つである線型回帰分析を適用することにより、既知のタッチポイントの傾向から、1フレーム後（0.015秒後）の未来の指の位置を予測することに活用しています。

具体的には、数フレーム（0.015秒間～0.075秒間）に検出されるタッチポイントの数列を対象として、最小二乗法的一种による回帰分析を用いて指が画面上で傾いている角度を推定し、さらに、数列の移動平均偏差の一種を算出することにより、指が傾けられる速さを推定しています。これにより、タッチスクリーン上で、あたかもアナログスティックがあるかのような操作を可能にしたのです。直感的には、Kineticsとは、指の動く方向の未来を毎フレーム予測し続けることにより、「ユーザが本当に指を動かしたい方向」を検知して、キャラクタの動きに変換する技術と言えます【図6】。



【図6】 Kineticsの基本アルゴリズム



【図7】 プロダクトまで貫通するDXサイクルの設計

Kineticsを利用するユーザは、キャラクターを操作するために指を大きくスワイプする必要がなく、あたかも指をジョイスティックのように画面の上で傾けるだけで、キャラクターを操作できます。Kineticsは、おおよそ、60ピクセル平方の、親指の爪と同程度の大きさの範囲だけで実装でき、指をそこから外に出す必要がありません。極めて小さい範囲で実装できるため、画面内のキャラクターや背景に干渉しにくい利点があります。このような、携帯端末側（エッジサイド）でのデータ利活用により、新しい操作性やゲーム性をもたらすことも、DXの重要な役割の一つです。

おわりに

本稿では、ゲーム産業におけるDX技術開発の事例として、自動デバッグAI、GANによるゲームデータ自動生成、エッジサイドにおける機械学習の応用について示しました。日本のゲーム産業は、世界中にコンテンツを提供していく中で、開発から運用までを貫通するDXを実践してきました。その知見は他の産業にも応用可能なものが数多く含まれているはずです。

【図7】に示すように、DXの要諦は、組織内でどのようにデータを流通させ、APIをどのように公開し、どこでデータを蓄積し、どのように学習し、どこでフィードバックをかけるかという、全体のパイプラインやサイクルをデザインすることにあります。DX時代の研究所の使命は、技術ロードマップを描くことではなくサイクルを設計することにあると言えるでしょう。自動デバッグAIの事例で示したように、人間同

士の信頼関係を基盤として、完全に自動化・機械化されたデータの流通基盤（パイプライン）を整備することがDXには必要不可欠です。本稿が読者の皆様のDXの実践のお役に立てば幸いです。なお、これらのDX技術や、組織全体に関わるDXの在り方については、東京理科大学大学院の選択必修講義「DX特論」でもご紹介しています。