

Title	プレイヤーの困惑状態を検知してヘルプを提供するシステム
Author(s)	鈴木, 克磨; シュエ, ジュウシュエン; 池田, 心
Citation	ゲームプログラミングワークショップ2025論文集, 2025: 154-157
Issue Date	2025-11-07
Type	Conference Paper
Text version	publisher
URL	https://hdl.handle.net/10119/20616
Rights	社団法人 情報処理学会, 鈴木 克磨, シュエ ジュウシュエン, 池田 心, ゲームプログラミングワークショップ2025論文集, 2025, 154-157. ここに掲載した著作物の利用に関する注意 本著作物の著作権は情報処理学会に帰属します。本著作物は著作権者である情報処理学会の許可のもとに掲載するものです。ご利用に当たっては「著作権法」ならびに「情報処理学会倫理綱領」に従うことをお願いいたします。 Notice for the use of this material: The copyright of this material is retained by the Information Processing Society of Japan (IPSJ). This material is published on this web site with the agreement of the author (s) and the IPSJ. Please be complied with Copyright Law of Japan and the Code of Ethics of the IPSJ if any users wish to reproduce, make derivative work, distribute or make available to the public any part or whole thereof. All Rights Reserved, Copyright (C) Information Processing Society of Japan.
Description	第30回ゲームプログラミングワークショップ (GPW-25), 箱根セミナーハウス, 2025年11月14日-16日



プレイヤーの困惑状態を検知してヘルプを提供するシステム

鈴木 克磨^{1,a)} シュエ ジュウ シュエン^{1,b)} 池田 心^{1,c)}

概要：本研究は、デジタルゲームにおいてプレイヤーが操作に迷った際の「困惑状態」を検知し、適切なタイミングでヘルプを提示する支援システムの構築を目的とするものである。従来のゲームチュートリアルは、プレイヤーが興味や必要性をまだ感じていない段階で一度に提示されることも多く、時間あるいは量的に負荷が高い内容のときはスキップされてしまうこともある。そのような場合、プレイに必要な情報を得られず、しばらくしてそれが必要になってから情報の所在が分からずに困難な状態へと陥ってしまう。そこで本研究では、プレイヤーの行動ログから順調なゲームプレイでは現れにくい遷移や操作の停滞を検出することで困惑状態を認識し、あわせてプレイヤーの「やりたいこと」を推定することで、文脈に即したヘルプを提示する統合的な枠組みを提案する。初期の対象領域としては困惑が生じやすく、実務アプリケーションとも似た構造を持つ RPG のメニュー操作に着目し、ログ設計・収集からプロトタイプの開発、被験者実験による支援効果の評価までを行う計画である。最終的には、ゲーム UX の改善や開発支援に資する知見を提供することを目指す。

キーワード：ヘルプシステム、困惑検知、意図推定、ゲームチュートリアル、プレイヤー支援

A System for Detecting Player Confusion and Providing Assistance

KATSUMA SUZUKI^{1,a)} CHU-HSUAN HSUEH^{1,b)} KOKOLO IKEDA^{1,c)}

Abstract: This study aims to develop a support system that detects a player's state of confusion during gameplay and provides contextually appropriate assistance at the right timing. Conventional game tutorials are often presented all at once before players feel a need or interest, and when the content is too lengthy or demanding, players tend to skip it. Consequently, they may fail to acquire the necessary information for gameplay and later fall into difficulty when that knowledge becomes relevant. To address this issue, we propose an integrated framework that identifies confusion by detecting unusual transitions or operation stagnation that rarely occur during smooth gameplay, based on player action logs. In addition, by inferring the player's intended goal, the system can deliver context-aware assistance tailored to the current situation. As an initial target domain, we focus on menu operations in role-playing games (RPGs), which are prone to confusion and share structural similarities with business application interfaces. Our research plan includes log design and collection, prototype development, and an experimental evaluation of support effectiveness. Ultimately, this work aims to contribute insights that enhance game user experience (UX) and support the design of more adaptive and user-friendly game systems.

Keywords: Assistance System, Confusion Detection, Intent Inference, Game Tutorial, Player Support

1. はじめに

多くのデジタルゲームにおいては、プレイヤーの理解を

促進するためのチュートリアル機能やヘルプ機能が標準的に実装されている。これらはゲームのシステムや操作方法、ルール、メニューの使い方などをあらかじめ伝えることで円滑なプレイを支援することを目的として実装されている。しかしながら、実際にゲームがプレイされている様子を観察していると、図1のようにプレイヤーが提示されたチュートリアルを読み飛ばす様子もしばしば見られる。

¹ 北陸先端科学技術大学院大学
Japan Advanced Institute of Science and Technology,
Nomi, Ishikawa 923-1211, Japan
a) katsuma@jaist.ac.jp
b) hsuehch@jaist.ac.jp
c) kokolo@jaist.ac.jp

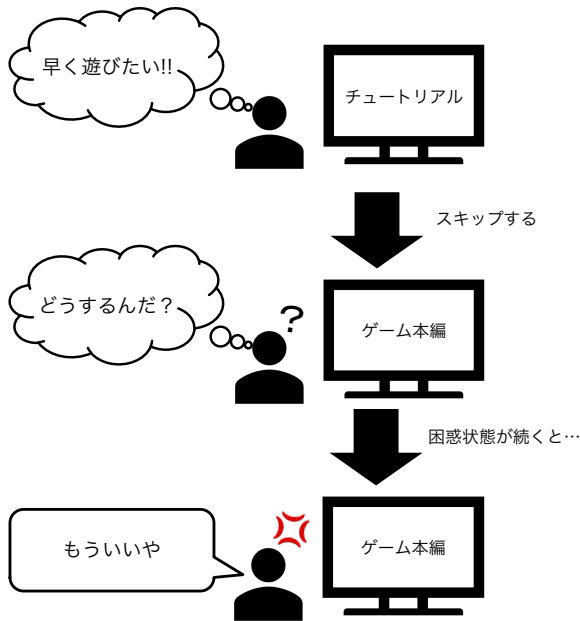


図 1 チュートリアルを飛ばすプレイヤーの例

このような現象の背景には、プレイヤーはゲームプレイのテンポ感や没入感を損なうことを嫌い、状況に関係なく提示されるチュートリアルなどの説明を煩雑なものとして扱う傾向があると考えられる。結果として、プレイヤーが思うようにゲームを進められない状況に直面した際には、必要な手がかりを得られないまま操作に迷い、ゲームの進行が停滞したり、最悪の場合には離脱に至ったりする事態が発生する。

そのような事態の具体的な例として、一般的なシステムと RPG での事例の 2 つを挙げる。一般的なアプリケーションにおける事例としては、ユーザーがテキストファイルを編集し、PDF ファイルとして保存を行い、内容を確認しようとする場面を考える。文章の編集後に出力操作を実行したものの、ファイルが意図した場所に保存されておらず、どこに出力されたのか分からないまま探し続け、作業が中断してしまうといった事例である。2 つ目の RPG における事例は、プレイヤーが新たに入手した装備アイテムをキャラクターに装備しようとする場面を考える。プレイヤーはメニューを開き、キャラクターのステータス画面を確認するが、装備の変更が行えないかもしれない。実際には装備の付け替えはキャラクター画面でなく、アイテム一覧から行う仕様であった場合、そのことに気付くまでに時間を要し、結果的にプレイの流れが途切れてしまうといった事例がありうる。

このようにプレイヤーが提示されたチュートリアルやヘルプを読まずに進めた結果、その後の場面でつまづくことは少なくない。したがって、従来のように一括して情報を提示するチュートリアルだけでは、プレイヤーが必要とするタイミングで適切に支援することが難しい。本研究で

は、チュートリアル の在り方を模索し、プレイヤーが実際に困惑したタイミングを検知して、その状況に応じた情報を動的に提示することにより、より柔軟で効果的な支援の枠組みを構築することを目指す。

2. 困惑について

2.1 困惑の定義

本研究では、困惑状態のうち「プレイヤーが明確な目標を有しているにもかかわらず、その達成に必要な操作系列を誤解あるいは未把握であるために適切な行動選択ができず、意味のない操作あるいは操作の停止が継続している状態」のみを解決すべき状態であるとして支援の対象とする。ここでの「操作系列」とは、UI 上の選択や入力、メニュー遷移など、ゲームにおける操作レベルの行為を指す。一方で、物語の進行や情報の意味解釈、次に何をすべきかといった意味レベルの判断に起因する困惑は対象外とする。

例として、RPG で新しい町に到着後、「聞き込みを雑に行ったために次に何をすればよいかわからないプレイヤー」を考える。結論から述べると、支援の対象外である。なぜならば、操作レベルではキャラクターとの会話という行為を実行できているからだ。ここでの目標到達を阻害する要因は「誰と話すか」「何の情報が有益か」という意味レベルの問題に起因しているためである。このようなケースであっても、例えば仲間キャラクターが「さっきの噂話にならないか？もう一度酒場に行ってみようぜ」などと自然なヒントを出すような支援がありうるだろうが、意味レベルでの支援はゲームの本質的な面白さを阻害する恐れもあるため、本研究では対象としない。

逆に、RPG をプレイ中にセーブを実行したいが「どのメニューからセーブできるかが分からないプレイヤー」は、操作の特定というレベルで失敗しているため、操作レベルでの困惑として支援の対象となる。このような場合、ゲームの本質的な面白さが現在進行形で阻害されているわけであるから、それを解消するような支援は多くの場合有益であると考えられる。

また、この定義は状態遷移モデルとしても定式化できると考える。ゲーム内で目標とする状態をノード $v \in V$ として、その状態から行える操作系列をエッジ $e \in E$ とする。これらからなるグラフを $G = (V, E)$ とする。ユーザーが到達を意図する目標ノード $v_g \in V$ に対して、現在ノード v_t から v_g への有効経路が存在するにも関わらず、局所的なループやバックトラック、無効な入力などを繰り返し、目標ノードに辿り着けないことを「困惑」と見なすことができる。

2.2 困惑の事例

本節では、定義の妥当性と取り扱う範囲を示すため、RPG のメニュー UI を中心に扱う事例と扱わない事例を併記す

る。また表 1 では、代表的な困惑事例を「目的／困惑」という形で要約する。

2.2.1 要点の整理

表 1 に示す代表的な困惑事例それぞれについて、なぜ取り扱うのか、取り扱わないのか明記する。

- **C1: 対象とする.** 入手したアイテムを確認したいという明確な目標を有しているが、その所在がメニュー内のどのタブに分類されているかを特定できず、複数の画面を往復してしまう事例である。例えば「龍眼」というアイテムを得たプレイヤーが、それを武器が含まれているタブ、防具タブ、アクセサリタブなどを探すものの見つからず、実は食料品が含まれているタブの中にあった、というようなケースである。この困惑は UI 構造の把握に失敗している点で、操作レベルにおける困惑といえる。すなわち、プレイヤーは操作手段そのものを誤っているわけではないが、目的に対応する UI 要素を同定できず、結果として適切な操作系列を構築できていない。
- **C2: 対象とする.** 装備を変更したいという目標は明確であるが、そのための方法がわからないというケースを考える。プレイヤーはキャラクターのステータス画面から直接装備を変更できると想定して操作を試みるかもしれない。しかし、場合によってはアイテムリスト側から装備品を選択し、対象キャラクターに付け替える操作手順が求められることもあるだろう。そうすると、プレイヤーは誤った画面で試行を繰り返すことになる。この困惑は、ゲームシステムの操作構造に対する想定と実際の UI 設計と乖離していることに起因しており、操作レベルにおける構造的誤解として位置づけられる。
- **C3: 対象としない.** キャラクターの能力値やスキル効果を確認したいという目標は明確であるが、表示される解説文が専門的または抽象的であり、どのような効果を持つ能力なのかを理解できない事例である。この困惑は、UI 上の操作や遷移は正しく行われている一方で、提示されたテキストの意味を理解するのに失敗している。すなわち、操作レベルでは問題が生じておらず、意味レベルにおける情報解釈の困難として位置づ

けられるため、本研究の支援対象ではない。

- **C4: 対象としない.** パズルゲームにおいてハイスコアを更新したいという目標を持つが、どのようなプレイ方針や行動選択を取れば得点が伸びるのかが分からない事例である。この困惑は、操作自体や UI の利用に問題はなく、ゲームルールやスコア算出の仕組みに関する理解・戦略判断の不足に起因している。すなわち、意味レベルにおけるプレイ最適化の問題であり、操作レベルの困惑を対象とする本研究の支援範囲からは除外される。
- **C5: (現段階では) 対象としない.** 対象ドメインが異なるため本研究の範囲外である。本事例はゲーム外の一般システムにおける困惑であるが、内容としては、ユーザーがキーボード配列を変更したいという目標を持ちながら、設定メニュー内のどの項目から変更できるかを特定できず、複数の項目を行き来してしまう困惑であり、これは操作系列の特定に失敗しているという点で、C1 や C2 と同様に操作レベルの困惑と言える。したがって、将来的にゲーム外ドメインも対象とする際の比較参照例として位置づける。

3. システムの要素技術

本研究が目指す最終的なシステムは、プレイヤーの困惑状態を検知し、それを解消するためのヘルプを適切なタイミングで提示するものである*1。その実現には、プレイヤーが困惑していることを正確に把握し、さらにその時点でプレイヤーが意図している行動を推定して支援する必要がある。これらが不十分であると、プレイヤーが困っていないのに無駄なヘルプを出したり、やりたいと思っていない内容に関するヘルプを出したりしてしまうためである。この過程を支える要素技術は主に 2 つであり、1 つ目は困惑状態を検出する「困惑検知」、2 つ目はプレイヤーの行動目的を推定する「意図推定」である。

また、本システムは幅広いデジタルゲームに適用できるよう、プレイヤーの操作ログ全体を対象にしたいと考えて

*1 本来、チュートリアルシステムとヘルプシステムおよび UI 設計はすべて密接に関係しており、総合的に改善を図るべきものであるかもしれない。しかし本研究では、適切なタイミングのヘルプシステムだけでも一定の効果を挙げることはできると考える。

表 1 代表的な困惑事例（扱う／扱わないを含む）

ID	ドメイン	目的	困惑	取り扱い
C1	RPG	入手したアイテムを確認したい	アイテムが分類されたタブのどこにあるのかが分からない。	○
C2	RPG	装備を変更したい	想定していたメニューから装備変更できずに行き詰まる。	○
C3	RPG	キャラクターの能力を知りたい	複雑な解説でどのような効果を持つ能力なのかが分からない。	×
C4	パズル	ハイスコアを更新したい	どのようなプレイをすれば得点が伸びるのかが分からない。	×
C5	システム	キーボード配列を変えたい	どの設定項目で変更できるかが分からない。	△

いる。しかし、研究の初期段階ではRPGにおけるメニュー操作に限定し、「順調なプレイでは現れにくい遷移や停滞」を検出するプロトタイプの実装を目標としている。

3.1 困惑検知

まず必要な技術は、プレイヤーの操作ログ（ないし、環境によってはプレイヤーの映像、声、生体情報など）を用いて、プレイヤーが困惑していることを検知するための機能である。ゲームの謎解きや作戦立案などの良い意味での悩みなのか、操作法などゲームの流れを阻害する悪い意味での困惑なのか、その判断も含めて困惑検知は行われるのが望ましい。一般に、何らかの時系列データが通常とは異なる挙動を示していることを検知する技術は工場の異常検知、サーバーの攻撃検知など、さかんに研究されてきた。以下に、本研究に用いることができそうなアプローチの候補をいくつか示す。

- **ルールベース.** プレイヤーの操作ログから短期間の往復回数や滞在時間、連続した無効な操作回数などを計測して、閾値を超えた場合に困惑とするアプローチ。メリットは軽量であること、デメリットは閾値を設定する際に開発者の負担が大きいことである。
- **教師あり学習.** ログから抽出した特徴（遷移回数や最短経路比、操作頻度など）を用い、SVMやランダムフォレスト等の分類器で困惑状態を推定するアプローチ。メリットはルールベースでの検知よりも表現力が高いことや、閾値設計の負担を減らせること、デメリットはアノテーションが必要となり、幅広いゲームで適用するためには大量の訓練データを要し、特定のゲームに特化すれば新機能が追加された場合に再学習が必要になるかもしれないことである。
教師あり学習により「通常」と「異常（意図外行動）」を識別できれば、プレイヤー行動の困惑検知に応用可能である。例えば Hwoij ら [1] は、ネットワーク管理プロトコル (SNMP-MIB) から得られる通信ログを対象に、ルールベース分類器を用いて正常通信と攻撃行動を高精度に分類する手法を提案している。

3.2 意図推定

本研究で対象としたいような困惑が検知できたとして、それだけではどのような支援を行えばよいかは決定できない。例えば、1章で挙げたように、新たに入手した装備アイテムがあり、そこでプレイヤーが困惑しているとしよう。プレイヤーはそのアイテムを装備したくて方法が分からないのかもしれないし、そのアイテムの性能や背景ストーリーを見たいのかもしれないし、そのアイテムを加工したいのかもしれない。目的ごとに、やるべきこと、開くべきメニューは異なる。もし例えばプレイヤーがキャラクターのステータス画面で腕の場所を何度もクリックするような

ら、それはおそらくそのアイテムを装備したいのだろうと想像できる。あるいは、魔石など加工用アイテムをドラッグアンドドロップしているなら、それは加工がしたいのかもしれない。逆に、別のアイテムについて性能や背景ストーリーを5分前にも10分前にも見ているようなら、そこで悩んでいるわけではないという判断ができる。このように、現在の行動およびこれまでのさまざまな操作ログから、今何がしたくて困惑しているのかを推定する必要がある。意図推定は、ヒューマンロボットインタラクション、自動運転・運転支援、医療・福祉支援、教育・学習支援などさまざまな分野で研究されてきた。

- **ルールベース.** メニューに限定すれば、ゲーム内で直前に起きたイベントを参照することでルールを書くこと自体は可能だが、近年一般的なゲームのイベント量は膨大であるため、ルールも相当量必要となり、現実的なアプローチではない。
- **時系列分析.** 操作ログを系列データとして扱い、プレイヤーがどのメニューや操作に到達しようとしているかを推定するアプローチ。メリットは逐次推定が可能であること、デメリットはデータ量が多く必要になることである。

LSTM (Long Short-Term Memory) などの時系列分析を用いた手法は、ログ列の前後関係を学習し、次に発生するイベントを確率的に推定できる点で有効である。例えば Chen らによる LogLS (Dual LSTM) モデル [2] では、システムログを前後方向から同時に学習し、過去・未来双方の文脈から異常状態を推定している。このような双方向系列モデルは、プレイヤー行動の文脈を把握し、「次に何をしようとしているか」を意図として推定する応用に有効であると考えられる。

4. おわりに

本稿では、デジタルゲームをプレイしているときに見られる困惑について着目し、いくつか事例を紹介したのち、それらについて考察した。また、困惑を推定するための技術やアプローチについても取り上げた。今後はそれらの方法を踏まえて、困惑検知システムの作成に取り組むとともに意図推定も並行して進めていくことで、実際にプレイヤーが困惑している状況で適切なヘルプを提供するシステムのプロトタイプ実現を目指す。

参考文献

- [1] Hwoij, A., Al-kasassbeh, M. and Al-Fayoumi, M.: Detecting Network Anomalies using Rule-based machine learning within SNMP-MIB dataset (2020).
- [2] Chen, Y., Luktarhan, N. and Lv, D.: LogLS: Research on System Log Anomaly Detection Method Based on Dual LSTM, *Symmetry*, Vol. 14, No. 3 (online), DOI: 10.3390/sym14030454 (2022).