

「デジタルゲームにおける人工知能技術・プロシージャル技術入門」

三宅 陽一郎
y.m.4160@gmail.com
 Twitter: miyayou

2010.10.25
 東京工芸大学



自己紹介

1999年京都大学総合人間学部基礎科学科卒業。
 2001年大阪大学理学研究科修士課程物理学専攻修了。
 2004年東京大学工学系研究科博士課程(単位取得満期退学)
 同年、株式会社フロム・ソフトウェア入社。



デジタルゲームにおける人工知能の研究・開発

[講演] 全講演資料を公開しています。

- 2006年
 CEDEC2006
 「クロムハウズにおける人工知能開発から見るゲームAIの展望」
<http://blogai.igda.jp/article/33936286.html>
- 2007年
 AOGC2007 招待講演「人工知能が拓くオンラインゲームの可能性」
<http://www.bba.or.jp/AOGC2007/080/>
 CEDEC2007 招待講演「エージェント・アーキテクチャーから作るキャラクターAI」
<http://blogai.igda.jp/article/33936286.html>
- Korea Game Conference 2007招待講演
- 2006年～2007年
 IGDA日本、ゲームAI連続セミナー「ゲームAIを読み解く」全6回
<http://blogai.igda.jp/article/33936286.html>
- 2008年
 CEDEC2008 招待講演「ゲーム開発のためのプロシージャル技術の応用」
http://cedec.cesa.or.jp/2008/archives/archive_1.html
 DiGRA JAPAN 公開講座「Spore におけるゲームAI技術とプロシージャル」
<http://www.digrajournal.org/modules/mydownloads/viewcat.php?cid=10>
 IGDA日本 GDC報告会「GDCに見る最新AIとプロシージャル技術」
<http://blogai.igda.jp/article/33936286.html>
- 2009年
 IGDA日本 GDC報告会「これからのゲームAIの作り方」
<http://www.digrajournal.org/modules/mydownloads/>

自己紹介

- [特別論文]
 人工知能学会誌 Vol. 23 No. 1 (2008年1月) 「ゲームAI特集」
 「デジタルゲームにおける人工知能技術の応用」(三宅)
- [報告書]
 デジタルコンテンツ協会
 2007年度 第3章「ゲームAI」
 「デジタルコンテンツ制作の先端技術応用に関する調査研究報告書」
 2008年度 第3章「プログラミングAI」
 「デジタルコンテンツ制作の先端技術応用に関する調査研究報告書」
 ゲームAIの情報について、総合的にまとめてあります。
<http://www.dcaj.org/report/index.html> よりPDFダウンロードできます。
- [インタビュー]
 大学からゲームメーカーへ——AI研究で広がるステキなゲームの世界とは？
http://gamez.itmedia.co.jp/games/articles/0901/08/news129_3.html
<http://gamez.itmedia.co.jp/games/articles/0901/09/news075.html>
- (ブログ) y_miyake のゲームAI千夜一夜 (IGDA日本)
<http://blogai.igda.jp/>

ゲームAIセミナー群

第4回 Halo2 有線状態機械
 第3回 ChromeHounds マルチエージェント
 第2回 F.E.A.R. プランニング
 第1回 Killzone 知識表現(世界表現)

社内セミナー (週1回/～190回)
 IGDAゲームAI連続セミナー (2ヶ月1回/全6回)

FROM SOFTWARE
 igda Japan
 CEDEC 2007

研究・勉強の仕方

- (1) 言葉を大切にする
 - (2) コツコツと知識を積み重ねる
 - (3) 小さなことを積み重ねていると、他人には追いつけないとんでもないところへ行けてしまう。
- (By イチロー)

本講演の主旨

- ① 「デジタルゲーム」を人工知能から眺める。
- ② 過去から現在、そして未来へ向かう「デジタルゲームの流れ」を体感する。

本授業の指針

4つの言葉を覚えて帰しましょう

- (1) 自律型エージェント (Autonomous Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

ここから眺めると、「デジタルゲームのAI」がよく見えます。
勉強するときに大切なのは、言葉を大切に、一日の中で一つ一つ覚えて行くことです。

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

4つの言葉を覚えて帰しましょう

- 第一章 自律型エージェント (Autonomous Agent)
- 第二章 プロシージャル (Procedural)
- 第三章 メタAI (Meta-AI)
- 第四章 マルチエージェント (Multi-agent)

質疑応答

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

- 第一章 自律型エージェント (Autonomous Agent)
- 第二章 プロシージャル (Procedural)
- 第三章 メタAI (Meta-AI)
- 第四章 マルチエージェント (Multi-agent)

質疑応答

第1部 ゲームとは何か？

ゲームとは何か？

ゲームとは何でしょうか？
考えたことがありますか？

今日は一緒に考えてみましょう。

ゲームの種類

ゲームには
どんな種類のものがあるでしょうか？

ゲームの種類

ゲームの種類	詳細な分類	さらに詳細な分類
① アナログゲーム	カードゲーム ボードゲーム ギミックゲーム	ウォーゲーム 積み木ゲーム シミュレーションゲーム トレーディングカードゲーム



カタン



スリードラゴンアンティ



人狼



?

ゲームの種類

ゲームの種類	詳細な分類	さらに詳細な分類
② デジタルゲーム	商業ゲーム インディーズ・ゲーム シリアスゲーム カジュアルゲーム ハードコアゲーム オンラインゲーム オフラインゲーム	カードゲーム シミュレーションゲーム アクションゲーム 戦略ゲーム アドベンチャーゲーム 恋愛シミュレーション・ゲーム パズルゲーム(落ちゲー) コンストラクション・ゲーム RPGゲーム スポーツゲーム 音ゲーム



シムシティ4
(コンストラクション)



ハーツオブアイアン2
(シミュレーション)



パラッパラッパー(音ゲー)

ゲームの種類

ゲームの種類	詳細な分類	さらに詳細な分類
③ 代替現実ゲーム (ARG)	ネット カード 現実の施設 ファックス	Perplex City I love bees あんたがた



ネット、架空サイト、2ch、
TV番組、電話、ファックス、
セブンイレブンのサービスなど、
電話ボックス、看板、など、
現実にあるギミックを使用して、
ヒントを点在させながら、
謎解きをさせるゲーム

?

ゲームの種類	詳細な分類	さらに詳細な分類
① アナログゲーム	カードゲーム ボードゲーム ギミックゲーム	ウォーゲーム 積み木ゲーム シミュレーションゲーム トレーディングカードゲーム
② デジタルゲーム	商業ゲーム インディーズ・ゲーム シリアスゲーム カジュアルゲーム ハードコアゲーム オンラインゲーム オフラインゲーム	カードゲーム シミュレーションゲーム アクションゲーム 戦略ゲーム アドベンチャーゲーム 恋愛シミュレーション・ゲーム パズルゲーム(落ちゲー) コンストラクション・ゲーム RPGゲーム スポーツゲーム 音ゲーム
③ 代替現実ゲーム (ARG)	ネット カード 現実の施設 ファックス	Perplex City I love bees あんたがた
④ スポーツ	球技・アスリート	サッカー、走り高跳び...
⑤ 伝統的ゲーム	身体を使う	おにごっこ、かくれんぼ、 缶蹴り、タッチおに

ゲームの種類	詳細な分類	さらに詳細な分類
① アナログゲーム	カードゲーム ボードゲーム ギミックゲーム	ウォーゲーム 積み木ゲーム シミュレーションゲーム トレーディングカードゲーム
ゲームには、実にいろいろな種類のゲームがある。		
③ 代替現実ゲーム (ARG)	ネット カード 現実の施設 ファックス	Perplex City I love bees あんたがた
④ スポーツ	球技・アスリート	サッカー、走り高跳び...
⑤ 伝統的ゲーム	身体を使う	おにごっこ、かくれんぼ、 缶蹴り、タッチおに

ゲームを定義できるか？

- ① ルール ② プレイヤー ③ フィールド
...

我々はゲームという概念をよく知っているつもりである。
しかし、厳密にゲームを定義することは、
それほど簡単なことではない。

ゲームという広大なフィールドは
研究するに値する深さと広がりを持っている

ゲームを研究する(研究者)



『遊びと人間』
ロジェ=カイヨフ

『Rules of Play』
Katie Salen,
Eric Zimmerman

『Ludology』
(ゲーム学)
Gonzalo Frasca

『フロー体験』
チクセントミハイ

ゲームを研究する(研究者)



『遊びと人間』
ロジェ=カイヨフ
(1958)



遊びの快楽を4つに分類

アゴーン(競争)、アレア(偶然)、
ミムクリ(模倣)、インクリンクス(賭)

人類学から『遊び』を研究する

ゲームを研究する(研究者)



『フロー体験』
チクセントミハイ



人が心地よくなる体験とは何か？ (フロー体験)



<http://game.watch.impress.co.jp/docs/20080410/ps3on.htm>

ゲームを研究する(研究者)



『Ludology』
(ゲーム学)
Gonzalo Frasca
(1999)



ゲームを物語論(narratology)ではなく
ゲームそのものとして扱うべき、という考えから、
ゲーム学(ludology)という言葉を作って、
ゲーム学の端緒を作る。

※ludoは、ラテン語で「遊ぶ」(play)を意味する

ゲームを研究する(研究者)



『Rules of Play』
Katie Salen,
Eric Zimmerman
(2002)



可能性空間

(=ユーザーのアクションがゲーム空間内で
影響・意味のある範囲)

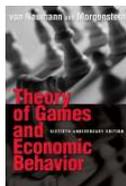
マジックサークル

(= ゲームの面白さを味わう行為のループ)

ゲームで研究する 世の中の現象をゲームとして捉えて研究する

経済

フォン・ノイマン、
モルゲンシュタイン



生物進化

メーナード・スミス



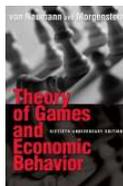
ESS (安定な進化戦略)

ゲームで研究する

世の中の現象をゲームとして捉えて研究する

経済

フォン・ノイマン、
モルゲンシュタイン



経済活動をゲームとして捉えて、
ミニマックス定理など、
ゲーム原理上の定理を用いて、
経済現象を定式化した、革命的な仕事。

ゲームで研究する

世の中の現象をゲームとして捉えて研究する

生物進化

メーナード・スミス



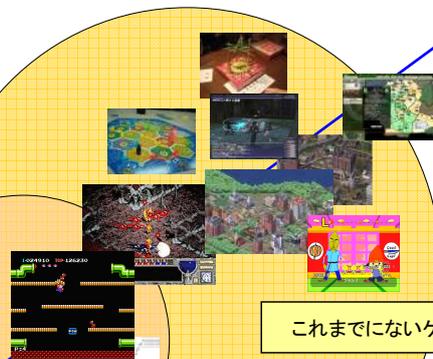
ESS (安定な進化戦略)

進化をゲームとして見立てて、
生物がどのような戦略のもとに進化して
行くかを説く。

ゲームを創作する



ゲームを創作する



これまでにないゲームを作る

ゲームを創作する

(ゲームはゲームを製作する者によって進化する)



ゲーム作りはプロ開発者に限らない。
誰が行ってもよい。面白ければ勝ち。

ゲームを創作する

学問としてのゲーム

ゲーム研究者、ジャーナリスト、ゲームファン
「ゲームとはこういうものだ」と定式化する。



芸術(創作)としてのゲーム

ゲーム開発者
「ゲームとはこういうものだ」と固定観念を壊して、
新しいものを創造して行く

ゲームを創作する

(ゲームはゲームを製作する者によって進化する。)

ゲームは進化する。
ゲームを作る人の手によって。
ゲームをプレイする人によって。
ゲームを作り出す社会によって。

=
ゲーム開発者になるとは、
ゲームの進化の流れの中に身を置いて、
ゲームの進化に貢献する人間となること。

ゲーム作りはプロ開発者に限らない。
誰が行なってもよい。面白ければ勝ち。

ゲームの未来

ゲームの進化の果てには、
何があるのでしょうか？

...誰もわからない。

100年、200年後のゲームは、
どうなっているのか？

ほっておいても進化しない。
ゲーム開発を続けることで、
過去から未来へつないで行く。

第一章まとめ

- ① ゲームを研究する。
- ② ゲームで研究する。
- ③ ゲームを創作する。

「ゲーム」という概念の中には、人間の知識、生活にとって、
奥深い認識、まだ発見されていない機能が隠されている。

たとえ、ゲーム開発者にならなくても、
ゲームから物事を捉えること、
ゲームとして物事を捉えること、
ゲームを実際に作ってみることは、
新しい世界の見方を教えてくれる。

第一部まとめ

- ① デジタルゲームは既に単なるピコピコゲームではない。
30年をかけて、技術と共に進化されて来た。
- ② デジタルゲームは既に科学的/哲学的研究の
対象である。(学会: DiGRA デジタルゲーム学会)
- ③ ゲーム開発もそういった研究の成果を取り入れて、高
いレベルでデザインして行く時代になりつつある。

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

4つの言葉を覚えて帰しましょう

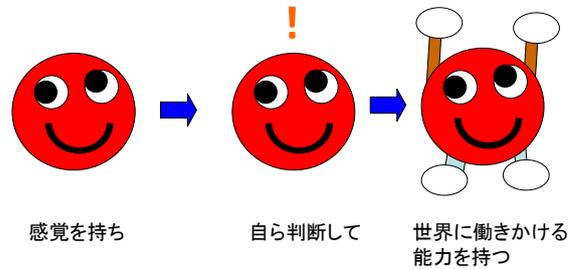
- 第一章 自律型エージェント (Autonomous Agent)
- 第二章 プロシージャル (Procedural)
- 第三章 メタAI (Meta-AI)
- 第四章 マルチエージェント (Multi-agent)

質疑応答

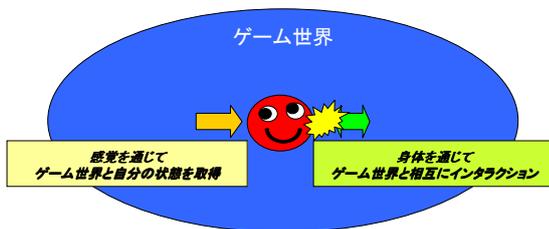
第1章 自律型エージェント (Autonomous Agent)

エージェントとは？

- ① 環境に対して情報を集める感覚(センサー)を持つ
- ② 自ら判断する能力を持つ。
- ③ 環境に対して働きかけることができる能力を持つ。



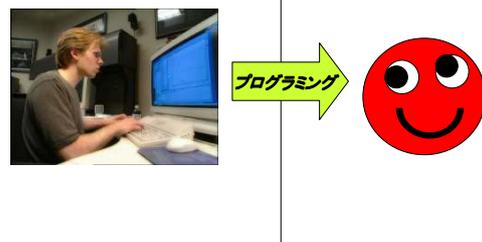
ゲームにおけるエージェント



エージェント = キャラクターAI

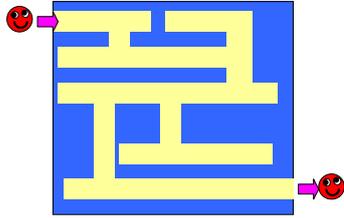
エージェントを作ろう！

プログラマーとエージェント



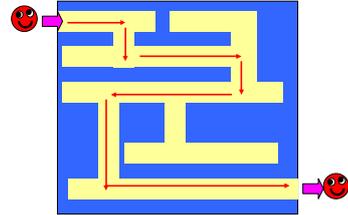
エージェントの自由な移動

Q1: 与えられた迷路を抜けることが出来るAIを考えてみよう!
どのように実装すればよいか?



エージェントの自由な移動

A1: プログラマーが迷路を解いて答えを実装する

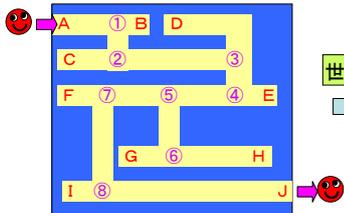


- プログラム:
- | | |
|------------|-------------|
| ① 5m直進 | ⑦ 10m直進 |
| ② 右折 | ⑧ 左折 |
| ③ 行き当たりまで直 | ⑨ 行き当たりまで直進 |
| ④ 左折 | ⑩ 左折 |
| ⑤ 行き当たりまで直 | ⑪ 直進 |
| ⑥ 右折 | ⑫ ゴール |

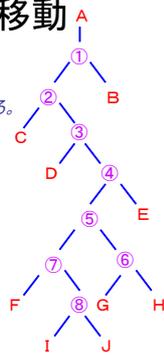
エージェントの自由な移動

A2: AI自身に迷路を解かせる

「迷路内の移動=グラフ上の移動」と問題が簡単になる。
グラフ検索のアルゴリズムでAIを動かすことが可能になる。



世界表現



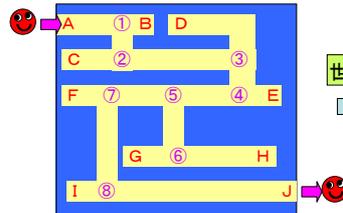
最大3分岐の合流しない迷路は
2分木(binary tree)と等価である

ゲーム世界をAIの制御のために表現したものを世界表現という。

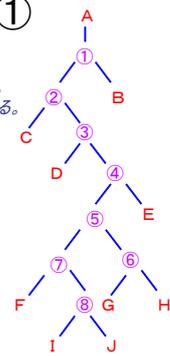
NPCの自由な移動 ①

A2: AI自身に迷路を解かせる

「迷路内の移動=グラフ上の移動」と問題が簡単になる。
グラフ検索のアルゴリズムでAIを動かすことが可能になる。



世界表現



最大3分岐の合流しない迷路は
2分木(binary tree)と等価である

迷路を世界表現(この場合はグラフ表現)して
探索アルゴリズムによって、AI自身に問題を解かせる

Counter Strike



プログラマーとエージェント

ハードコーディング型実装

プログラマーがステージごとに考えた思考を実装する



プログラミング
実装



プログラマーが答えを解いて
それをAIに実装する

プログラマーとエージェント

自律型エージェント実装

どんなステージでもエージェント自身が考えて実装する



プログラミング
実装



プログラマーはAI自身が問題を解くように実装する

AI自身が答えを解く

現代のエージェント技術の流れ

ハードコーディング型実装



自律型エージェント実装

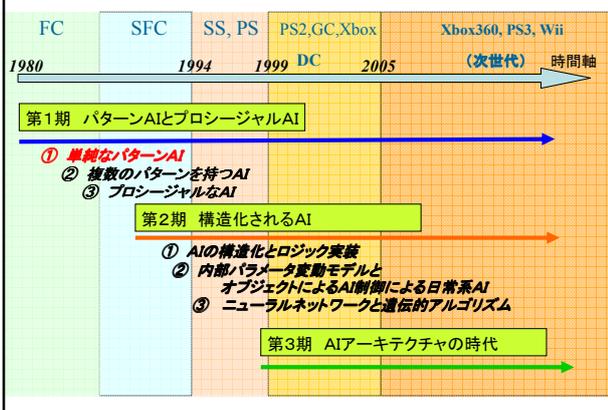
デジタルゲームのエージェントは、「プログラマーがステージについて考えたことを実装する」方式から、「エージェント自身に考えさせる」という方式に以降しつつある。

自律型エージェント (autonomous agent)

だんだんと AIがプログラマーの手を離れて自律する

ここでエージェントの実装のされ方の歴史を追ってみましょう！

ゲームにおけるエージェントの歴史



第1期 ①単純なパターンAI



同じパターンをくり返すだけ

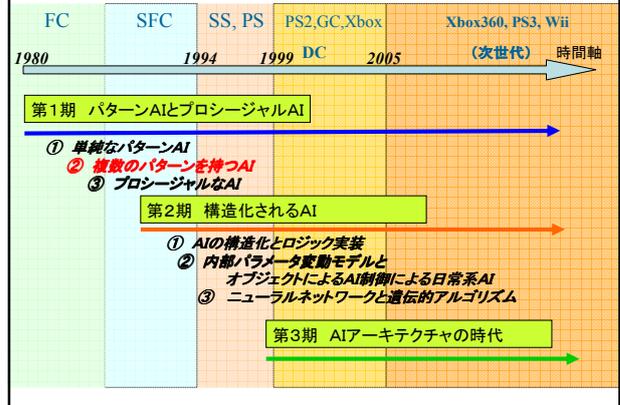
(例) スペースインベーダー (1978)



プレイヤーの動きに関係なく、決められた動きをする



ゲームにおけるエージェントの歴史



第1期 ②複数のパターンを持つAI

Interactive



あらかじめ決められた行動を、状況によって使いわけるAI

「プリンス・オブ・ペルシャ」など、スプライトアニメーションを用意する必要がある場合、数パターンに限られる。

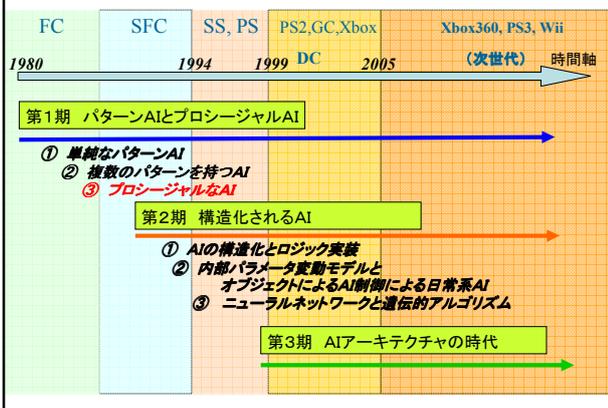
(例) プリンス・オブ・ペルシャ



「プリンス・オブ・ペルシャ」など、スプライトアニメーションを用意する必要がある場合、必然的にこういった制御となる。

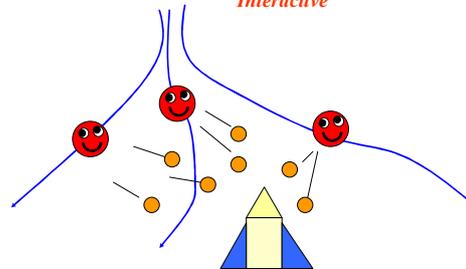


ゲームにおけるエージェントの歴史



第1期 ③プロシージャルなAI

Interactive



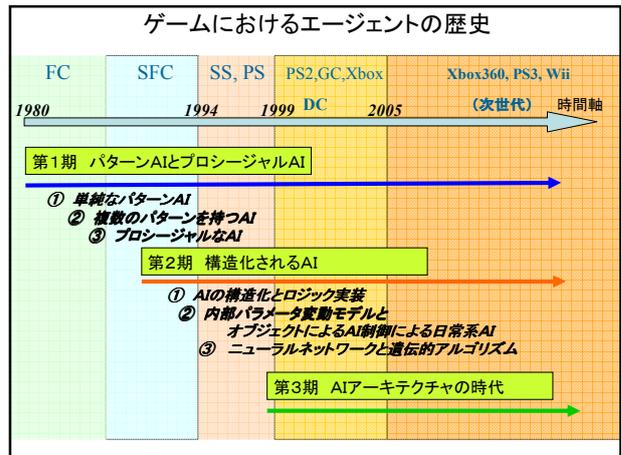
シューティングゲームなど、機体の軌道や弾道を、逐次的に関数の計算で行なう。
(例) 数値列を用意する場合もある。

(例) ゼビウス?



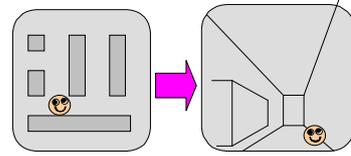
ゼビウス

道原雅博氏 あと面白い機體なんですけれど、ゼビウスには非常に簡単なAIが組み込まれています。
「プレイヤーがどれくらい強いのか」というのを判断して、出てくる敵が強くなるんです。強いと思った相手には強い敵が出てきて、弱いと思った相手には弱い敵が出てきます。そういったプログラムが組み込まれています。
ゲームの難易度というのは「初心者には難しく、上級者には簡単」ということが、ひとつの難易度で調整をやっていくと起きちゃうので、その辺を何とか改善したいな、ということでそういうことを始めてみたのなんですけれど、
お蔭で割合にあまり上手くない人でも比較的長くプレイできる。
うまい人でも最後のほうに行くと結構ハママックに陥れる。そういった感じになっています。
— ゼビウスをマスター —
<http://spiffire.client.jp/shooting/zeibuss2.html>



第2期 構造化されるAI

アセンブラからC言語への移行
2Dから3Dが主流へ → AIにとって爆発的な情報量の増大
80年代のAI技術の盛り上がりゲームへスピノフ



グリッド上のロジック 俯瞰制御 → 無数のレイキャスト(射線計算) 主観制御

2D・3DのAIをムービーで比較



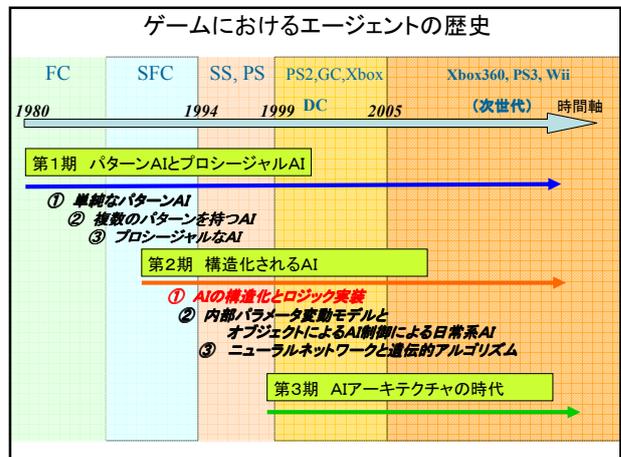
マリオ → Halo

第2期 構造化されるAI

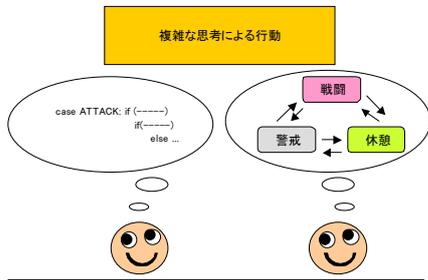
AIが処理すべき膨大な空間情報
→ 結局、対処出来ませんでした (複雑すぎる、1994年から10年の課題となる)。

暫定的解決策

- ① AIの行動範囲をある限定した自由に移動できる空間に限定すること。
- ② ステージを単純化すること(例えば、空の上ならAIは動き放題である)。
- ③ あらかじめ、地形に沿った運動をプログラミングしておく (固定パスを与えておくなど)。

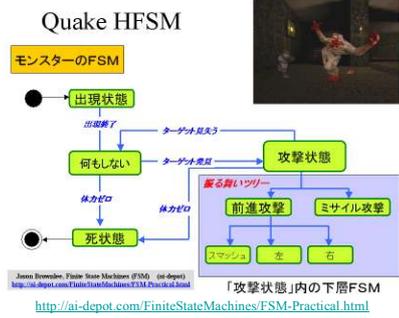


第2期 ①AIの構造化とロジック実装

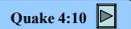


複雑な環境下(処理すべき情報が多い)で、キャラクターを制御する仕組みを入れる。

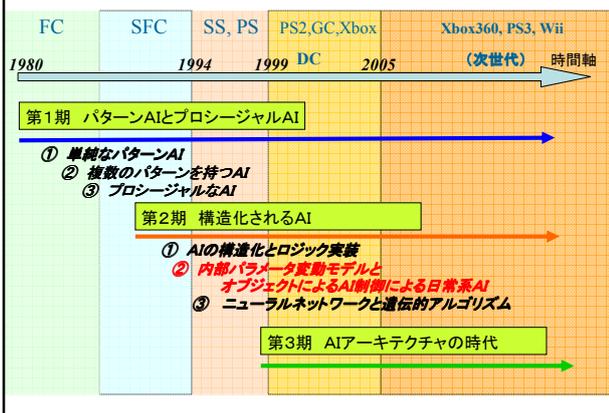
(例) Quake HFSM



状態遷移図を用いる



ゲームにおけるエージェントの歴史



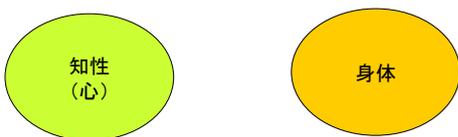
第2期 ②内部パラメータ変動モデルとオブジェクトによるAI制御による日常系AI



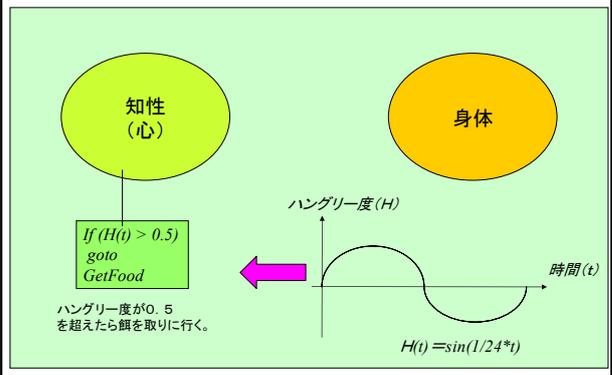
戦闘のAIから日常のAIへ

戦闘のAI(極限状態) < 日常のAI(煩雑)

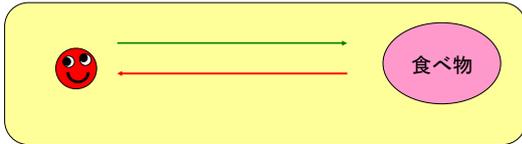
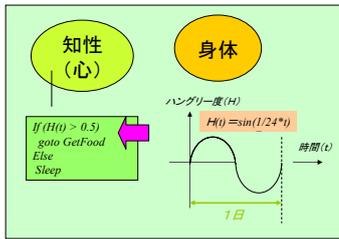
自律型AIの作り方



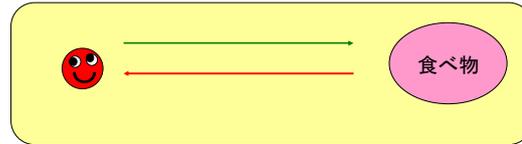
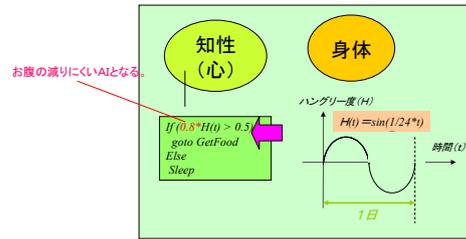
自律型AIの作り方



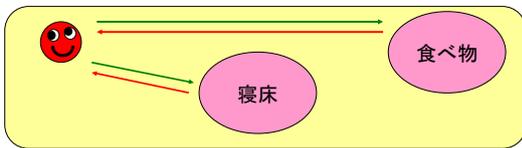
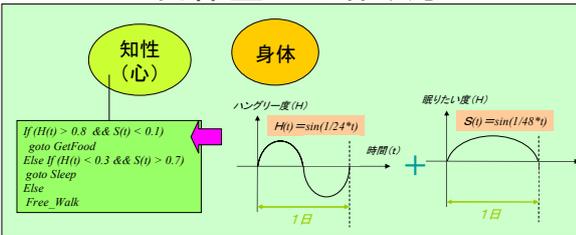
自律型AIの作り方



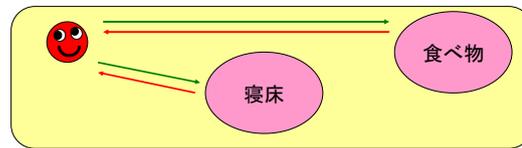
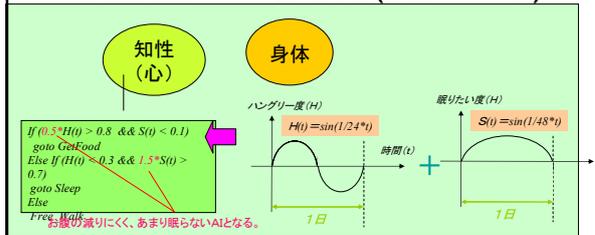
自律型AIの作り方(個性付け)



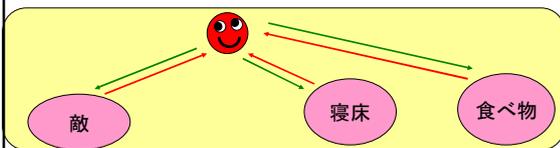
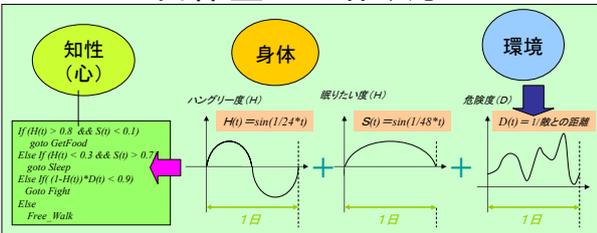
自律型AIの作り方



自律型AIの作り方(個性づけ)



自律型AIの作り方



こういったデモに3Dモデルを被せると...

② The Sims 3 (Maxis, EA)



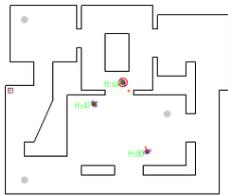
ムービー

Trait ... 特性、性格パラメーター
Wish ... 将来の目標

Sims Getting Smarter: AI in The Sims 3 (IGN)
<http://pc.ign.com/articles/961961065p1.html>

GDC09 資料 <http://cmpmedia.vo.llnwd.net/o1/vault/gdc09/slides/Combined.ppt>

Mat Bucklandの自律型エージェントのデモ



目標を持ち、目標ごとに内部状態に応じて変動する関数を持つ

- ① 攻撃する
- ② 移動する
- ③ 回復アイテムを取る
- ④ 武器を取る
- ...

意思決定: 全てのゴールのうち、最大値を持つゴールを選択して行動する。

(例)③

$H(i) = 0.8 * (1 - \text{現在のHP}) / \text{回復アイテムまでの距離}$

デモ



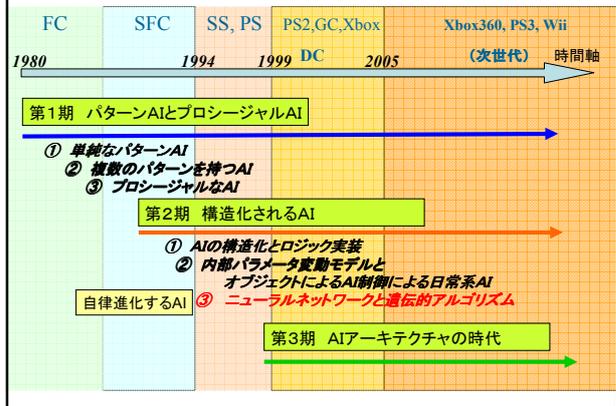
オライリー・ジャパン

「実例で学ぶゲームAIプログラミング」

(Mat Buckland 著、松田真一訳)

<http://www.oreilly.co.jp/books/9784873113395/>

ゲームにおけるエージェントの歴史



(例) 決して多くない

ニューラルネットワーク

1996年 BATTLECRUISER: 3000AD (3000AD)

1997年 がんばれ森川君2号 (muumu)

2000年 Colin McRae Rally 2.0 (Codemasters)

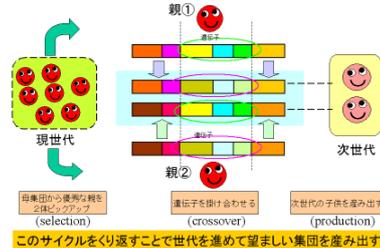
2001年 Black & White (Lionhead Studio)

遺伝的アルゴリズム

1998年 アストロノーカ (muumu)

第2期③ニューラルネットワークと遺伝的アルゴリズム

遺伝的アルゴリズムの仕組み



遺伝的アルゴリズムとニューラルネットワークがゲームAIの視野の中に入って来た。

(例) アストロノーカ

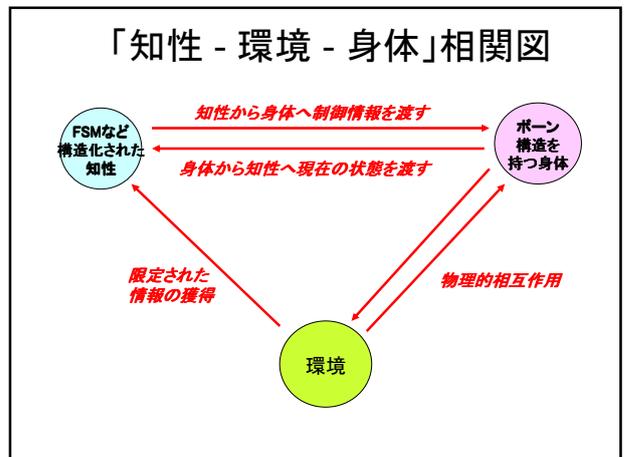
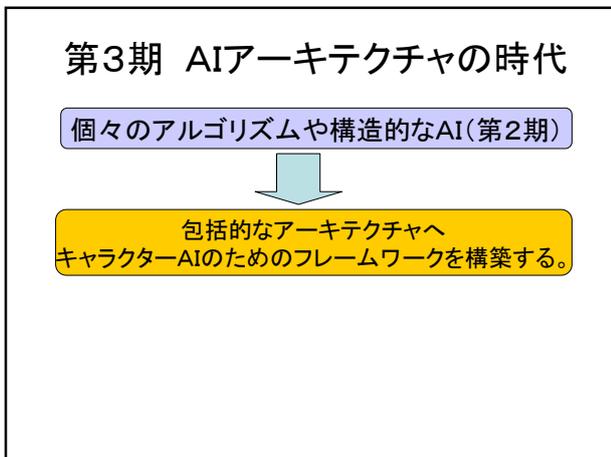
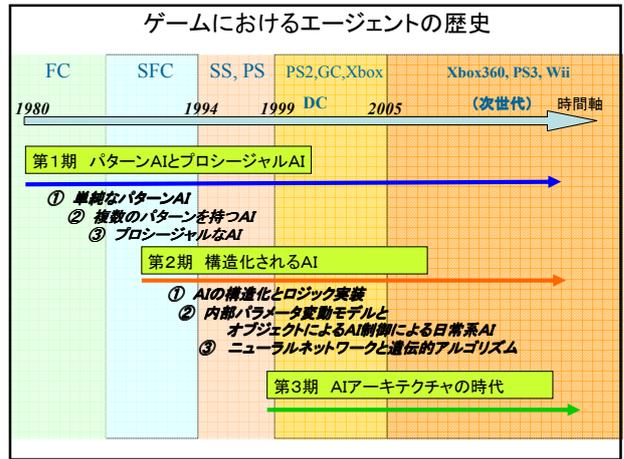
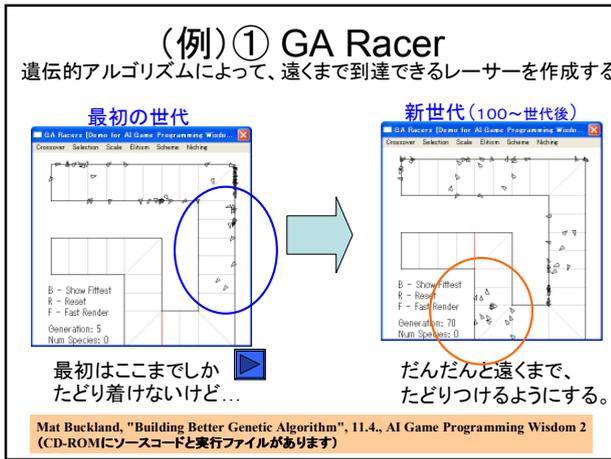
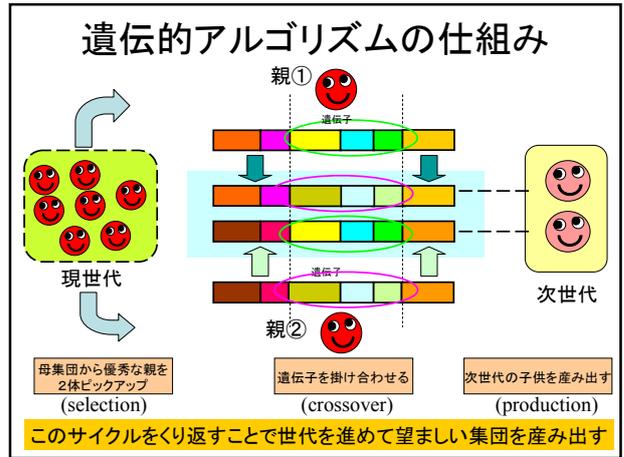
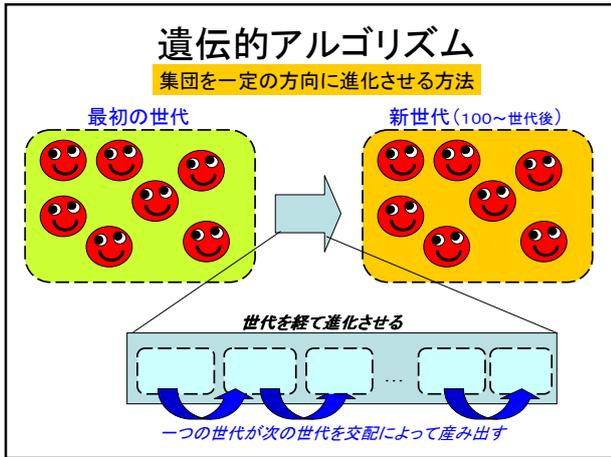


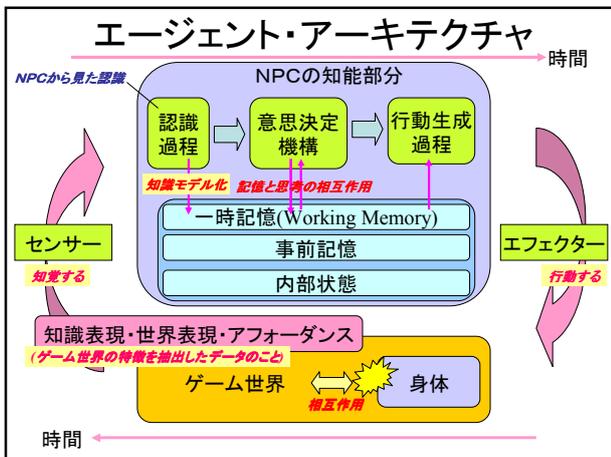
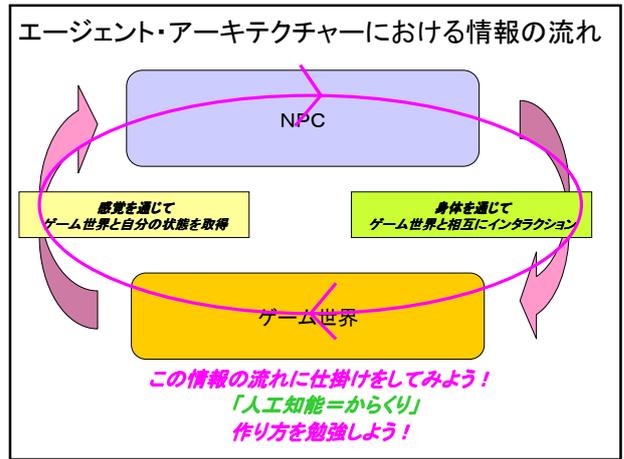
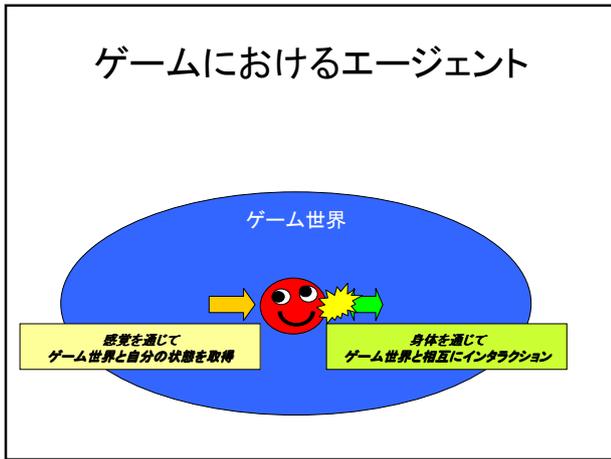
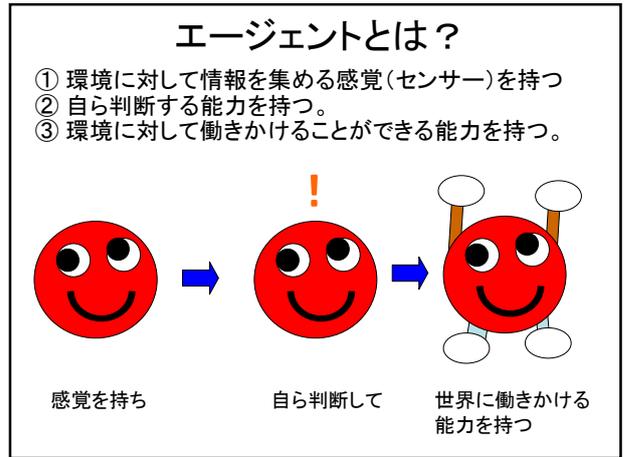
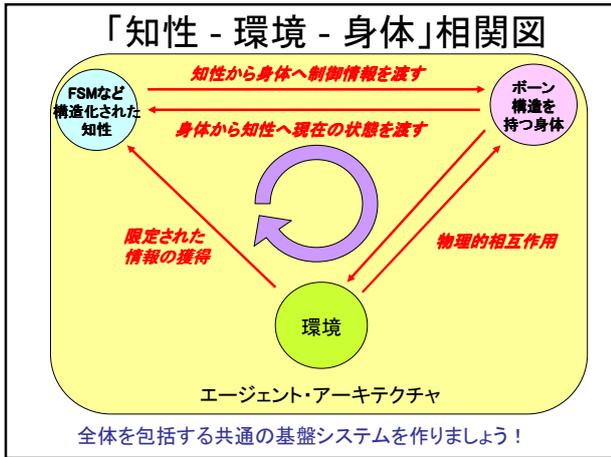
MuuMuu, プレイステーション用ソフト「アストロノーカ」(Enix, 1998)

<http://www.muumu.com/games/astro/>

参考文献

森川幸人,
「テレビゲームへの人工知能技術の利用」,
人工知能学会誌vol.14 No.2 1999-3
<http://www.1101.com/morikawa/1999-04-10.html>

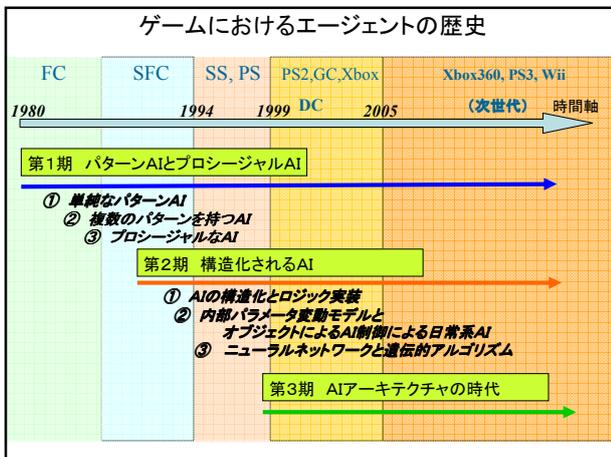
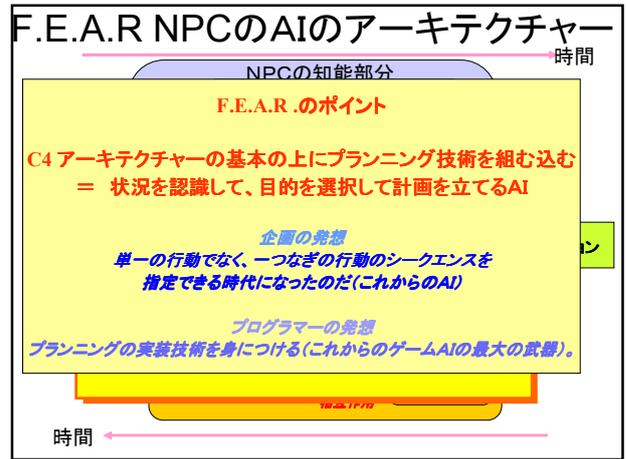
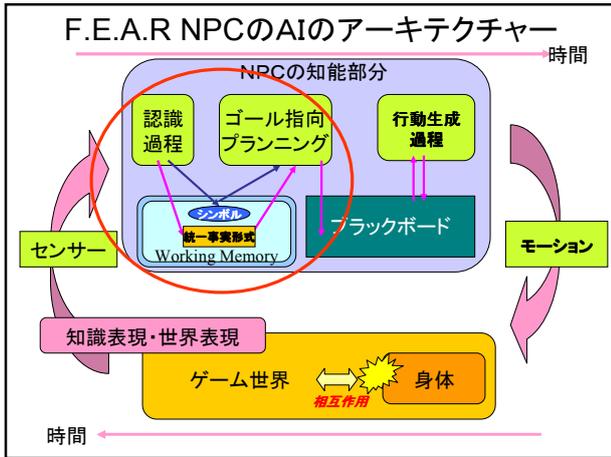




F.E.A.R

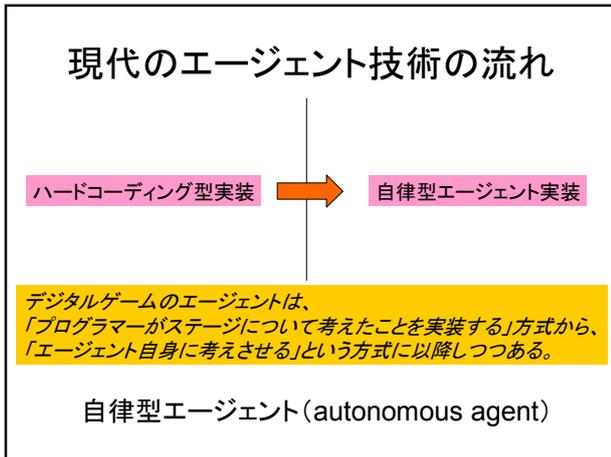
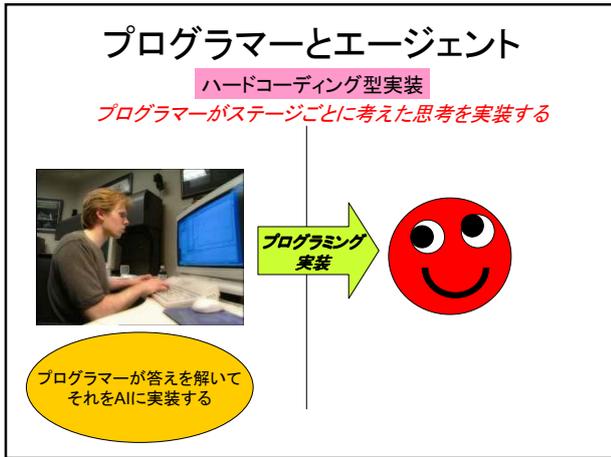
内容: 閉鎖空間の中のホラーFPS
開発元: Monolith Production
出版: SIERRA
Hardware: Windows, PS3
出版年: 2004年

FPSとホラーを、映画的な演出によって結びつけたエポックメイキングな名作。長年発展させて来たAI技術の本領が発揮され、開発者、プレイヤーから高い支持を集める。



第1章 自律型エージェント (Autonomous Agent)

まとめと考察



- ### 本授業の主旨
- 4つの言葉を覚えて帰しましょう
- (1) エージェント (Agent)
 - (2) プロシージャル (Procedural)
 - (3) メタAI (Meta-AI)
 - (4) マルチエージェント (Multi-agent)

コンテンツ

- 第1部 ゲームとは何か？
- 第2部 デジタルゲームにおける人工知能技術
 - 4つの言葉を覚えて帰しましょう
 - 第一章 自律型エージェント (Autonomous Agent)
 - 第二章 プロシージャル (Procedural)
 - 第三章 メタAI (Meta-AI)
 - 第四章 マルチエージェント (Multi-agent)
- 質疑応答

第2章 プロシージャル (Procedural)

プロシージャルとは？

プロシージャル(Procedural)
= 計算による、連続した操作による

デジタルゲームでは？

ゲーム空間、デジタル空間において、
自身の連続操作によって自律的な力を持つファクター

(例) 自動生成、自律的なAI

プロシージャル・コンテンツ・ジェネレーション
(Procedural Contents Generation = PCG)

プロシージャル制御
(Procedural AI)

プログラマーとゲーム世界

ハードコーディング型実装

プログラマーがステージの動作を定義する



プログラミング
実装



プログラマーが世界や
オブジェクトの動作を
全て定義する



プログラマーとゲーム世界

プロシージャルな世界(自律的に変化する世界)

ゲーム世界が自身が固有の法則をもって自律的に変化する



プログラミング
実装



現代のプロシージャルの流れ

ハードコーディング型実装

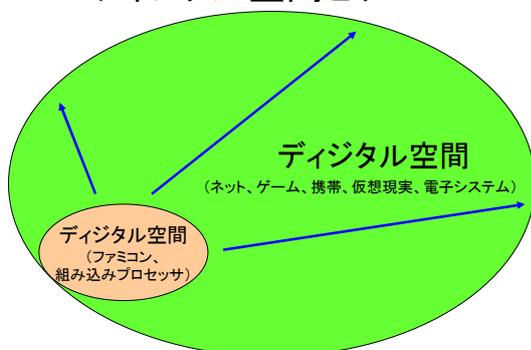


プロシージャル実装

デジタルゲームの世界は、
「プログラマーがステージのオブジェクトごとに法則を実装」方式から、
「普遍的な変化・発展の法則を世界に実装する」
という方式に以降しつつある。

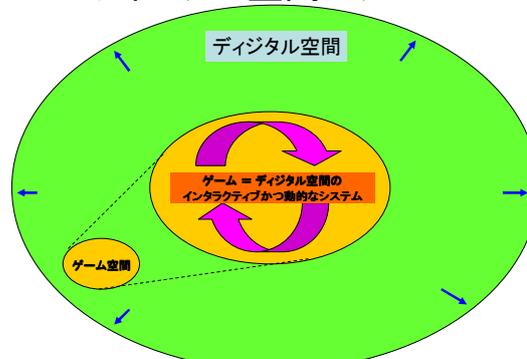
自律変化・発展する世界 (Procedural World)

デジタル空間とゲーム

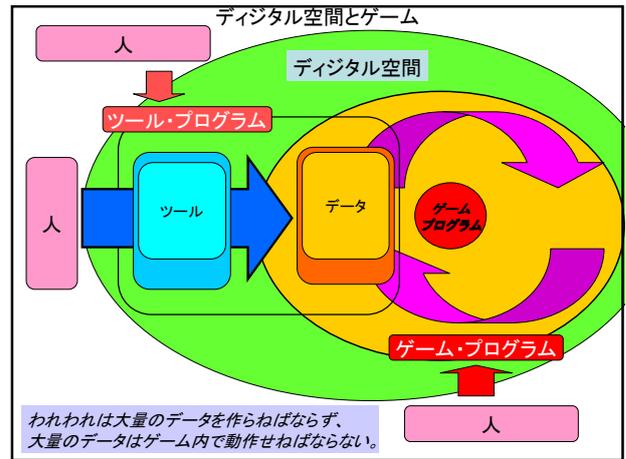
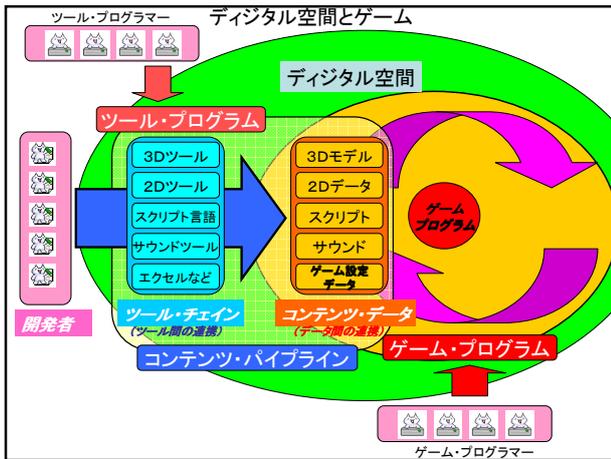


この30年で我々が触れることが出来るデジタル空間は格段に増大した

デジタル空間とゲーム



ゲーム空間も格段に増大した



拡大するゲーム開発に対してゲーム開発者は何が欲しいか？

①ゲームデータを生成・発展・消滅させる力
②ゲームを動作させる力

①人の手でデータを生成する
②人の手で動作を定義する

マンパワー

①計算によってデータを生成する
②計算によって動作を定義する

プロシージャル

どちらかを選ぶかはゲームの主に規模による = 次世代機からは需要が急増

導入コスト: 人件費(コスト) 導入コスト: ソフトウェア・システムを
整備するまでの投資(2~3年)

問題点: 開発の規模に比例して 問題点: 高度な技術レベル(簡単なものもある)
人手が必要 技術を使いこなすノウハウの必要性

巨額の開発費があれば... 中途半端ではケガをする

プロシージャル・コンテンツ・ジェネレーション

デジタル空間で草原を作りなさい

①アプローチ1
人の手によってモデリングを行う

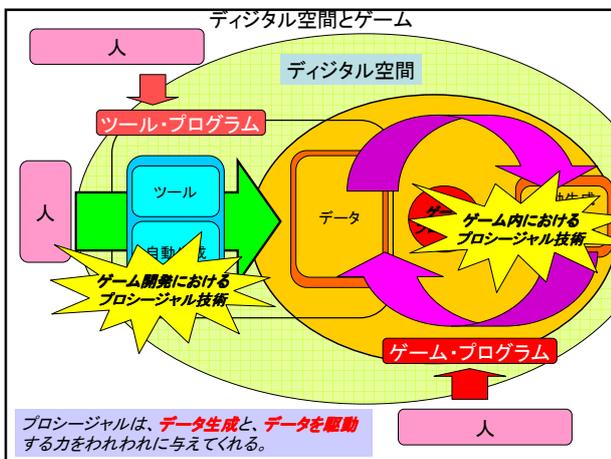
②アプローチ2
アルゴリズム
プログラムによってモデルを自動生成する

プロシージャル

プログラムの演算によってモデルを生成する

コンテンツ自動生成 (Procedural Contents)

2003, Yoshichiro Miyake



プロシージャルの歴史: ダンジョン自動生成

1980

Rogue Clone
ローグライクゲーム

Rogue 1980

NetHack 1987

不思議のダンジョン 1993~

Diablo 1996

テイル・オブ・エモンズ 1987~

Hellgate: London 2006

ブルードラゴン 2007

TOEO 2006

マビノギ 2004

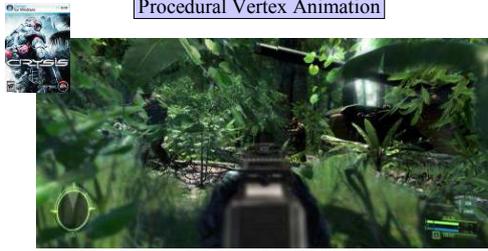
FF11 2007

CODED ARMS 2005

MMOにおける応用

Crysis における植物アニメーション自動生成

Procedural Vertex Animation

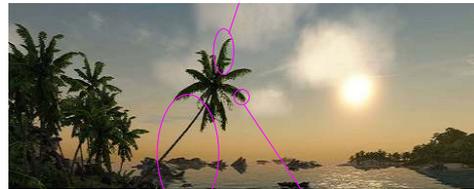


Tiago Sousa, "Chapter 16: Vegetation Procedural Animation and Shading in Crysis", GPU Gems 3, 2007

Finding Next Gen CryEngine2, [http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2\(Siggraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2(Siggraph07).pdf)

風に揺れる森 in Crysis

葉全体(leaves)



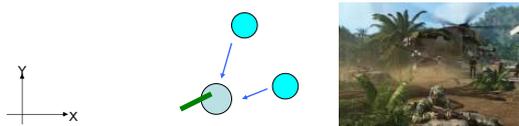
全体(entire) 葉(leaf)

風と葉の硬さからモデルの曲がり具合を計算する

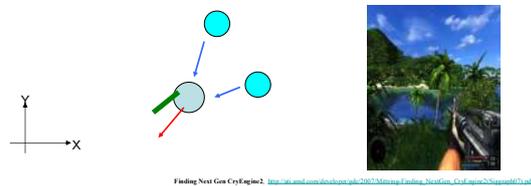


風に揺れる森 in Crysis

① その樹木に対する幾つかの風源の効果を足し合わせる

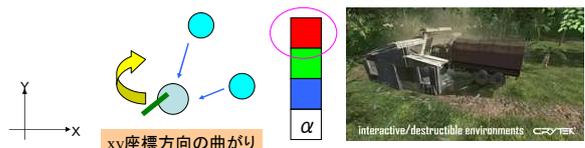


② 世界座標(xy)上の風ベクトルから、全体の曲げ方を決める。

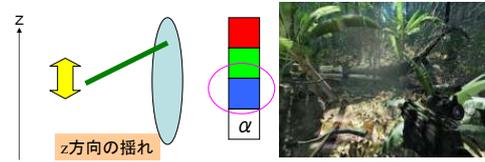


風に揺れる森 in Crysis

③ 葉のまとまりに対する曲がりを、風の強さだけから計算する



④ 一枚の葉の揺れ(z方向)を、風の強さだけから計算する



デモ



Procedural Vegetation Animation in Crysis

Tiago Sousa, "Chapter 16: Vegetation Procedural Animation and Shading in Crysis", GPU Gems 3, 2007

References

- (1) SpeedTree <http://www.speedtree.com/>
- (2) Procedural vertex-animation(OceanWater) http://b0rken.dk/hornet/shaders/proc_water.html
- (3) Tiago Sousa, "Chapter 16: Vegetation Procedural Animation and Shading in Crysis", GPU Gems 3, 2007
- (4) natural motion, <http://www.naturalmotion.com>
- (5) E3 2006: Indiana Jones Eyes-on, <http://ps3.ign.com/articles/705/705489p1.html>
- (6) Finding Next Gen CryEngine2, [http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2\(Siggraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2(Siggraph07).pdf)

Figures on the pages are from these references.

DUNIA ENGINE 2.0

DUNIA Engine デモ

- (I) Growing Vegetation
- (II) Tree Generation
- (III) Day and Night cycle
- (IV) Realistic Weather System

[参考] データ自動生成技術を大胆に導入したFPS、「Far Cry 2」の光と(4gamers)
<http://www.4gamer.net/games/047/G004713/20080222019/>

プロシージャルなゲームエンジン②

Frostbite Engine

- ① オブジェクト生成
(We can generate objects with procedural techniques
-Then use rules to deform / destroy / modify / move them
-Better interactivity)
- ② Semi-procedural surface shader
- ③ Procedural shader
- ④ Procedurally distributed on the fly

GDC 2007 Frostbite "Rendering Architecture and Real-time Procedural Shading & Texturing Techniques"
[http://developer.amd.com/assets/Anderson-Tatarchuk-FrostbiteRenderingArchitecture\(GDC_07_AMD_Session\).pdf](http://developer.amd.com/assets/Anderson-Tatarchuk-FrostbiteRenderingArchitecture(GDC_07_AMD_Session).pdf)
 GDC 2007 "The Importance of Being Noisy: Fast, High Quality Noise", N. Tatarchuk
[http://developer.amd.com/assets/TatarchukNoise\(GDC07-D3D_Dev\).pdf](http://developer.amd.com/assets/TatarchukNoise(GDC07-D3D_Dev).pdf)
 SIGGRAPH 2007 Johan Andersson "Terrain Rendering in Frostbite using Procedural Shader Splatting"
[http://ati.amd.com/developer/gdc/2007/Andersson-TerrainRendering\(SIGgraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Andersson-TerrainRendering(SIGgraph07).pdf)

プロシージャル - 固定された世界から変化する世界へ

デジタル・ワールド

固定モデル 固定テクスチャー 固定アニメーション 固定AI 固定インタラクション

プロシージャル・コンテンツ・ジェネレーション

テクスチャー生成	Procedural Texture
モデル自動生成	Deformable Model
アニメーション生成	Animation Synthesis
即応インタラクション	物理
認識・判断するAI	Procedural AI
街生成	City Generation
会話・ストーリー生成	Story Generation
人工市場	Artificial Societies
究極 地球シミュレーター	Earth Simulator

(世界シミュレーターという意味で)

プロシージャルの歴史: CG

1980 1990 2000

1975 フラクタル幾何学

1987 SIGGRAPH フラクタルイメージ

フラクタル幾何学を基にしたプロシージャルなモデル生成

3Dモデル生成

- 地形自動生成
- 植物自動生成
- 雲自動生成

2Dモデル生成

- テクスチャ自動生成

1997, 2002

- Terragen
- SpeedTree
- natFX MAX
- パリン・ノイズ
- Pro FX

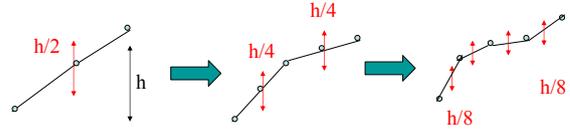
1988 1994 2003

②グラフィックス自動生成

- I. 地形自動生成
- II. 植物自動生成
- III. 雲自動生成
- IV. パスデータ自動生成

地形自動生成

中点変位法



スケーリングに応じて、振れ幅を変えていく



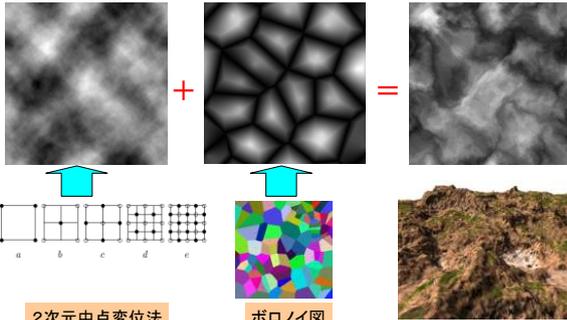
フラクタルイメージ - 運動とプログラミング (ハードカバー)
ハルグオスターバインゲン編著、チートニー・ザウベ編著、山口昌弘翻訳
ソフトバンク・サイエンス・フィクション (1998)



中点変位法によるフラクタル曲線の描画。
http://oddlabs.com/wiki/wiki.php?title=Midpoint_Displacement

地形自動生成

ノイズ法(濃い=低い、白い=高い)



2次元中点変位法

ポロノイ図

Jacob Olsen, Realtime Procedural Terrain Generation
http://oddlabs.com/download/terrain_generation.pdf

Ken Musgrave



<http://www.kenmusgrave.com/>

Age of Empires III における地形自動生成



西川善司, 「3DゲームファンのためのAGE OF EMPIRESエンジン講座(後編)にだわりの影生成と算術合成されるディテール、次回作はXbox2?」, GAME Watch, 2005

References

- (1) Jacob Olsen, Realtime Procedural Terrain Generation
http://oddlabs.com/download/terrain_generation.pdf
- (2) Ken Musgrave
<http://www.kenmusgrave.com>
- (3) Terragen (Planetside Software)
<http://www.planetside.co.uk/terrigen>
- (4) Introversion Software, "Procedural Content Generation",
GameCareerGuide.com, 2007
http://www.gamecareerguide.com/features/336/procedural_content.php
- (5) 西川善司, 「3DゲームファンのためのAGE OF EMPIRESエンジン講座(後編)にだわりの影生成と算術合成されるディテール、次回作はXbox2?」,
GAME Watch, 2005

Figures on the pages are from these references. m

②グラフィックス自動生成

- I. 地形自動生成
- II. 植物自動生成
- III. 雲自動生成
- IV. パスデータ自動生成

植物自動生成

Since 1968 A. Lindenmayer

研究分野としては
人工生命、CGIにまたがる

L-system 文法規則

構成要素 F, +, -, [,]
規則 $F \rightarrow F[-F]F[+F][F]$

0世代 1世代 2世代

$F \Rightarrow F[-F]F[+F][F] \Rightarrow F[-F]F[+F][F][F[-F]F[+F][F]] F[-F]F[+F][F][F[-F]F[+F][F]] F[-F]F[+F][F]$

解釈

F - まっすぐ進む
+ 30度左回転 - 30度右回転 []内は個別のスタック



植物自動生成

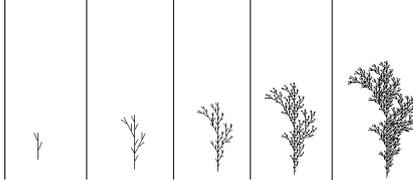
Since 1968 A. Lindenmayer

L-system 文法規則

構成要素 F, +, -, [,]
規則 $F \rightarrow F[-F]F[+F][F]$

0世代 1世代 2世代 3世代 4世代 5世代

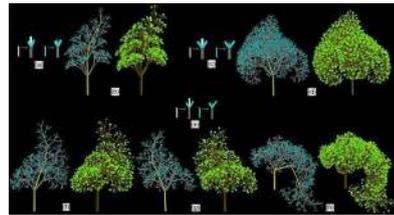
$F \Rightarrow F[-F]F[+F][F]$



Simulating plant growth by Marco Grubert <http://www.acm.org/crossroads/xrds8-2/plantsim.html>

The Sketch L-System: Global Control of Tree Modeling Using Free-form Strokes

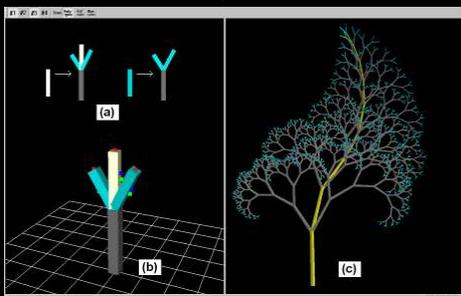
Takashi Iijiri, Shigeru Owada, Takeo Igarashi.



L-system を用いて簡単な操作で木のモデルを作成するツール

The Sketch L-System:
Global Control of Tree Modeling Using Free-form Strokes
<http://www.ai.is.s.u.tokyo.ac.jp/~ijiri/SketchLSystem/index.html>

デモ



SC06_SketchLSystem

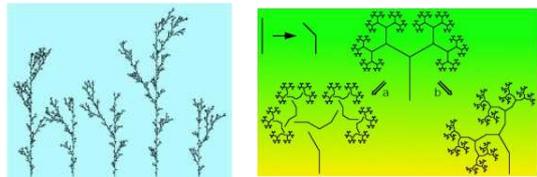
The Sketch L-system

Takashi Iijiri, Shigeru Owada, Takeo Igarashi
The Sketch L-System: Global Control of Tree Modeling Using Free-form Strokes
<http://www.ai.is.s.u.tokyo.ac.jp/~ijiri/SketchLSystem/index.html>

発展: 確率的 L-system

構成要素 F, +, -, [,]

規則 $F \rightarrow F[+F]F[-F]F$ 0.33
 $F \rightarrow F[+F]F$ 0.33
 $F \rightarrow F[-F]$ 0.33



規則を変えればいろいろな様相が自動的に生成される

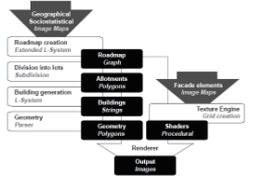
植物以外に使えないだろうか? じっと見る

Simulating plant growth by Marco Grubert <http://www.acm.org/crossroads/xrds8-2/plantsim.html>

L-system による街の自動生成

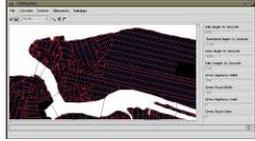
City Engine(central pictures)

<http://www.centralpictures.com/ce/>



Yoav I H Parish, Pascal Müller

<http://www.centralpictures.com/ce/tp/paper.pdf>



George Kelly, Hugh McCabe,
A Survey of Procedural Techniques for City Generation

L-system による街の自動生成

City Engine(central pictures)



Yoav I H Parish, Pascal Müller

<http://www.centralpictures.com/ce/tp/paper.pdf>

References

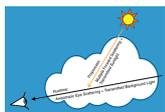
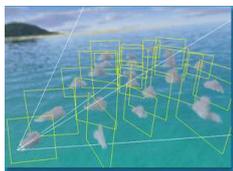
- (1) Simulating plant growth by Marco Grubert
<http://www.acm.org/crossroads/xrds8-2/plantsim.html>
- (2) L-System Wiki
<http://en.wikipedia.org/wiki/L-system>
- (3) The Sketch L-System:
Global Control of Tree Modeling Using Free-form Strokes
<http://www-ui.is.s.utokyo.ac.jp/~jiji/SketchLSystem/index.html>
- (4) Yoav I H Parish, Pascal Müller
<http://www.centralpictures.com/ce/tp/paper.pdf>
- (5) City Engine(central pictures) <http://www.centralpictures.com/ce/>
- (6) George Kelly, Hugh McCabe,
A Survey of Procedural Techniques for City Generation
<http://www.gamesitb.com/SurveyProcedural.pdf>

Figures on following pages are from these references.

②グラフィックス自動生成

- I. 地形自動生成
- II. 植物自動生成
- III. 雲自動生成
- IV. パスデータ自動生成

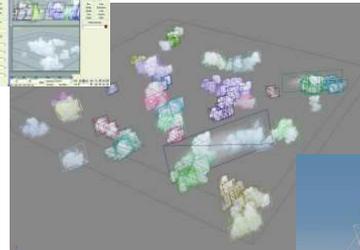
Microsoft Flight Simulator 2004



ビルボードからの様々なレンダリング効果を重ねてリアリティーのある空を再現 (Skywork、GDCで発表)

Mark.J.Harris, REALISTIC CLOUD ILLUMINATION
<http://www.markmark.net/clouds/index.htm>

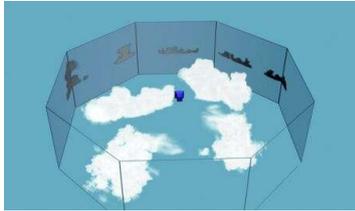
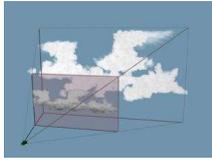
Microsoft Flight Simulator 2004



Niniane Wang, Realistic and Fast Cloud Rendering
<http://igt.akpeters.com/papers/Wang04/>

Mark Harris の方法を発展させる

Microsoft Flight Simulator 2004



Niniane Wang ,Realistic and Fast Cloud Rendering
<http://jgt.akpeters.com/papers/Wang04/>

8つのリング状の背景とビルボードによる写し込み

デモ



Niniane Wang ,Realistic and Fast Cloud Rendering
<http://jgt.akpeters.com/papers/Wang04/>

Microsoft Flight Simulator 2004

References

(1) hugo.elias Cloud Cover

http://freespace.virgin.net/hugo.elias/models/m_clouds.htm

(2) Kim Pallister, Generating Procedural Clouds in real time on 3D HW

<http://www.intel.com/cd/ids/developer/asm-na/eng/segments/games/20534.htm?page=11>

(3) David S. Ebert , Volumetric Procedural Implicit Functions

<http://www.csee.umbc.edu/~ebert/cloud>

(4) Mark.J.Harris, REALISTIC CLOUD ILLUMINATION

<http://www.markmark.net/clouds/index.htm>

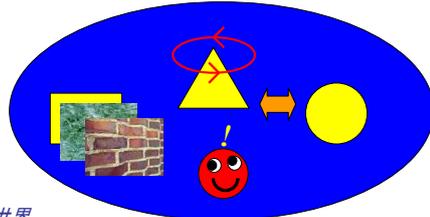
(5) Niniane Wang ,Realistic and Fast Cloud Rendering

<http://jgt.akpeters.com/papers/Wang04/>

(6) Perlin Noise

<http://m> Figures on the pages are from these references.

プロシージャル – 固定された世界から変化する世界へ デジタル・ワールド



固定した世界

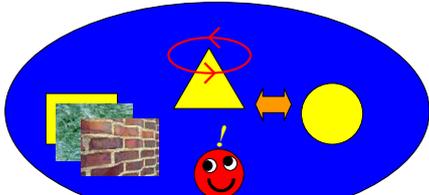
固定モデル 固定テクスチャー 固定アニメーション 固定AI 固定インタラクション

変形するモデル 生成するテクスチャー 生成するアニメーション 認識・判断するAI 即応インタラクション (物理)

変化する世界

Procedural World

プロシージャル – 固定された世界から変化する世界へ デジタル・ワールド



固定した世界

固定モデル 固定テクスチャー 固定アニメーション 固定AI 固定インタラクション

変形するモデル 生成するテクスチャー アニメーション生成 認識・判断するAI 即応インタラクション (物理)

変化する世界

Procedural World (全てが計算される世界)

プロシージャルとは？

プロシージャル(Procedural)
= 計算による、連続した操作による

デジタルゲームでは？

ゲーム空間、デジタル空間において、自身の連続操作によって自律的な力を持つファクター

プロシージャル・コンテンツ・ジェネレーション (Procedural Contents Generation = PCG) プロシージャル制御 (Procedural AI など)

(質はともかく)プロシージャル技術は、デジタル空間でコンテンツを自動生成、自動制御する力を与えてくれる

Sporeにおけるプロシージャル・ユーザー・アクション

簡単な操作で3Dクリーチャー・モデルからアニメーションまでを作成する

ユーザー作成コンテンツ (User Generated Contents, UGC)

キャラクターを作成するエディター 生物を作成するエディター 惑星(天体)を作成するエディター 宇宙船を作成するエディター

生物、建物、惑星、主要なものは全てユーザーが生成可能

シミュレーション
(Sim Everything !)

単細胞から宇宙までの進化シミュレーションゲーム

単細胞フェーズ 創生フェーズ 文明化フェーズ ギャラクティックフェーズ

「Spore」はプロシージャル技術の結晶

グラフィック、オブジェクト、サウンド、あらゆる分野においてプロシージャル技術を応用する次世代のゲーム

- ① テクスチャ (SIGGRAPH2007)
- ② オブジェクト配置 (SIGGRAPH2007)
- ③ モデル生成 (SIGGRAPH2007)
- ④ モデル変形 (SIGGRAPH2007)
- ⑤ アニメーション生成 (SIGGRAPH2008)
- ⑥ 音楽プロシージャル (GDC2008初公開)

Procedural Brushing Demo

3rd spore demonstration at SIGGRAPH 07 (YouTube)

Pd とは？

演奏者にあたるAI

メトロノーム

音を鳴らすところ

EAPd

伝染病

Plague Outbreak - World of Warcraft - September 2005
<http://www.mmocrunch.com/2007/12/04/top-5-most-memorable-events-in-mmorpg-history/>

Spore におけるプロシージャル技術

UGC (User Generated Contents, ユーザー作成コンテンツ) に PCG (Procedural Contents Generation, 自動生成技術) を応用することで、ユーザーにコンテンツを産出す力を与え、小さな労力で、大きなコンテンツを作り出す面白さを実現した。

References

- (1) カーネギー・メロン大学の [Andrew J. Willmott博士のHPページ](http://www.cs.cmu.edu/~ajw/s2007/) のSIGGRAPH2007のコーナーにまとめられています。
<http://www.cs.cmu.edu/~ajw/s2007/>
- (2) [E3 2006 #011] E3 2006最大の話題作? ウィル・ライトの「Spore」
<http://www.4gamer.net/news/history/2006.05/20060511195155detail.html>
- (3) Steve Capell et al., Interactive Skeleton-Driven Dynamic Deformations
<http://grail.cs.washington.edu/projects/deformation>
- (4) Development of Spore
(このサイトによると、デモシーナーたち、著名なフラクタル関係の技術者を Maxis は集めているようです)
http://en.wikipedia.org/wiki/Development_of_Spore
- (5) Spore の Procedural Music については、WEB に動画などがアップされているので、[Spore Procedural Music](#) などで検索してみよう!

Figures on the pages are from these references.

プログラマーとゲーム世界

ハードコーディング型実装

プログラマーがステージの動作を定義する



プログラマーが世界やオブジェクトの動作を全て定義する



プログラマーとゲーム世界

プロシージャルな世界(自律的に変化する世界)

ゲーム世界が自身が固有の法則をもって自律的に変化する



現代のプロシージャルの流れ

ハードコーディング型実装



プロシージャル実装

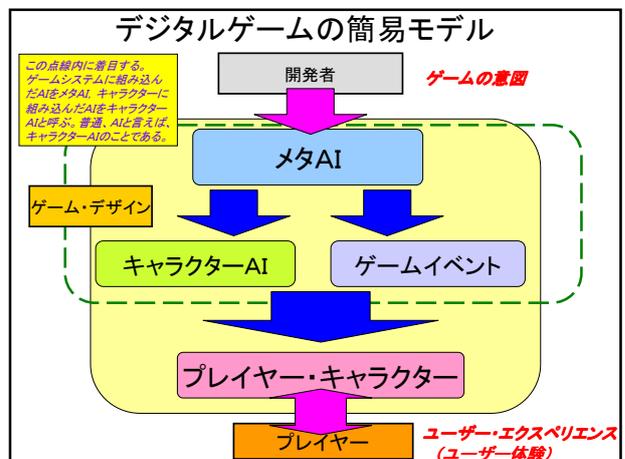
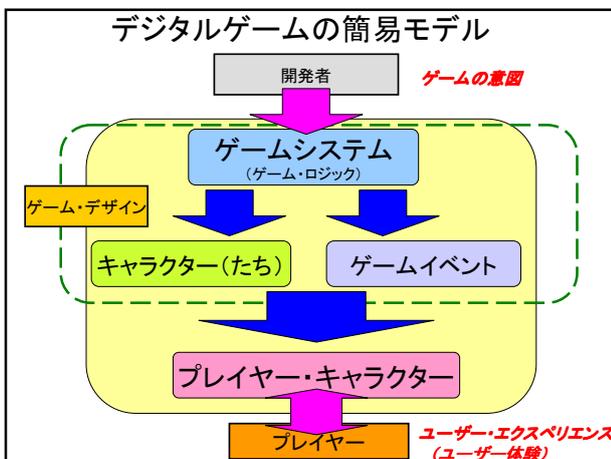
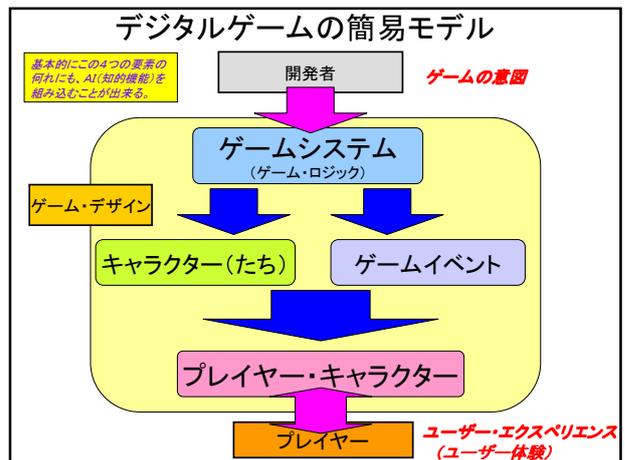
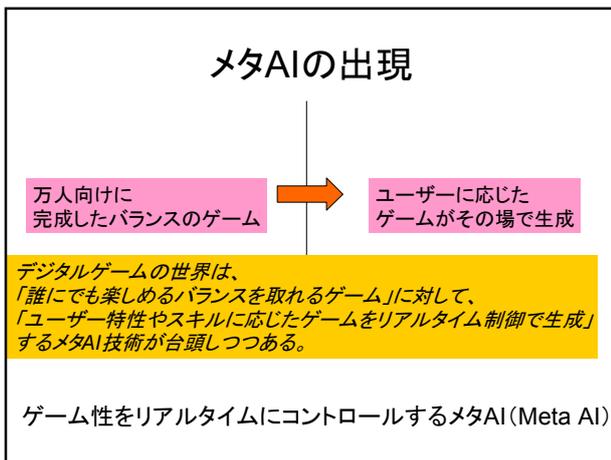
デジタルゲームの世界は、「プログラマーがステージのオブジェクトごとに法則を実装」方式から、「普遍的な変化・発展の法則を世界に実装する」という方式に以降しつつある。

自律変化・発展する世界 (Procedural World)

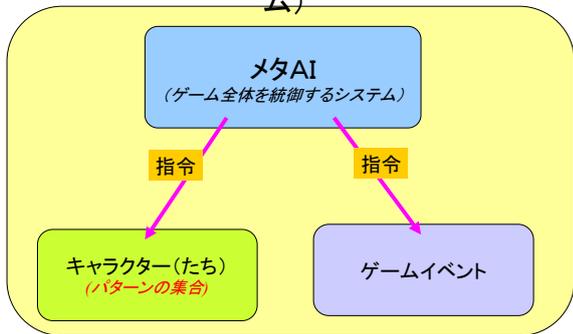
本授業の主旨

4つの言葉を覚えて帰しましょう

- (1) エージェント (Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

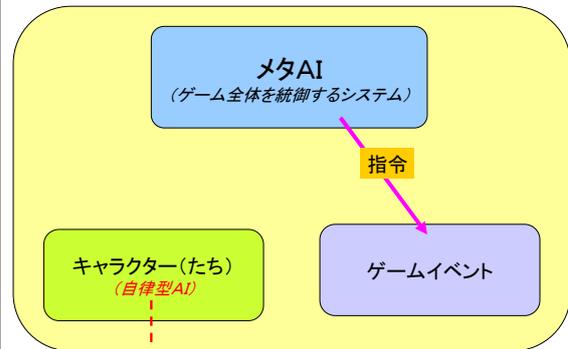


かつてのゲームAIの全体像(フルフレームム)



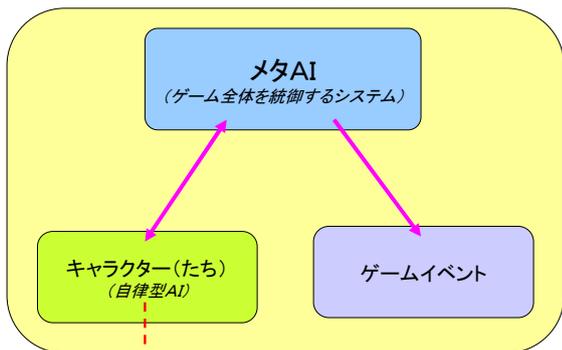
ゲームシステムはゲームステージの一部として、キャラクターが制御される(お化け屋敷AI)

最近のゲームAIの全体像(フルフレームム)



自分で考えて、自分で行動するAI

最新のゲームAIの全体像(フルフレームム)



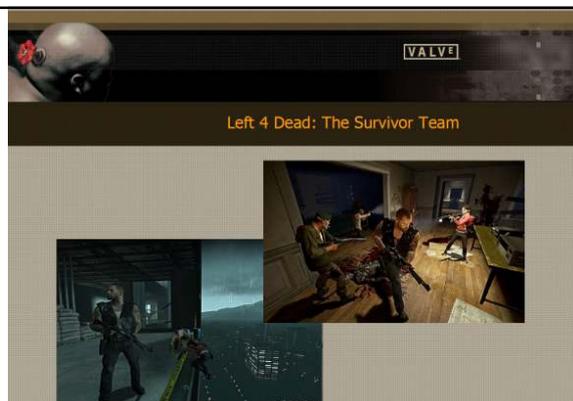
自分で考えて、自分で行動するAI

Left 4 Dead



内容: 4人チームでゾンビと戦うオンラインゲーム
 開発元: Valve Software
 出版: Valve Software
 Hardware: PC, Xbox360
 出版年: 2008年

オンラインで4人でチームを組んでゾンビと戦う。
 AI Directorによって、毎回、敵の出方が変化する。



Michael Booth, "The AI Systems of Left 4 Dead,"
 Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

Counter Strike から L4D へ: くり返し遊べる協力ゲーム体験

From COUNTER-STRIKE to LEFT 4 DEAD: Creating Replayable Cooperative Experiences

構築された非予測性

パーティの感情:
 総合的な緊張感



モンスターの
 配布

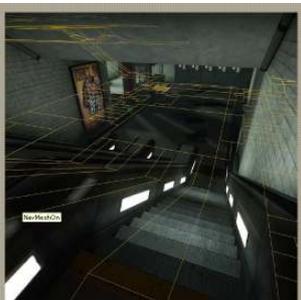
プレイヤーのインテンシティ(緊張感)を計算→ リラックスした後にモンスターを放つ!

Michael Booth, "The AI Systems of Left 4 Dead,"
 Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

ナビゲーションメッシュによる オフライン・リアルタイム空間情報解析

➤ The Navigation Mesh

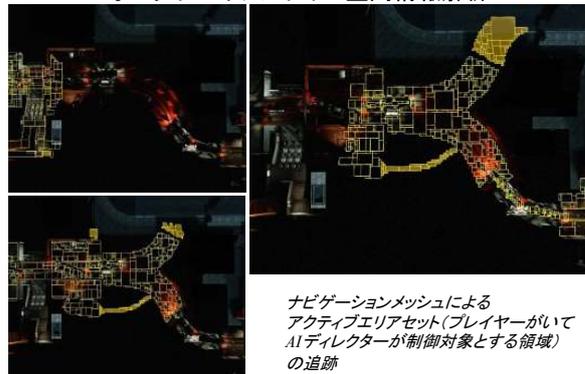
- Represents "walkable space"
- Originally created for Counter-Strike Bot pathfinding
- Useful for general spatial reasoning and spatially localized information
 - Has area A ever been seen by actor B?
 - Is area X potentially visible by area Y?
 - Where is a spot near the Survivors, not visible to any of them?
 - How far have we traveled to reach this area?



その領域をプレイヤーは見たか？あるエリアは他のエリアから見えるか？ など。

Michael Booth, "The AI Systems of Left 4 Dead," Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

ナビゲーションメッシュによる オフライン・リアルタイム空間情報解析



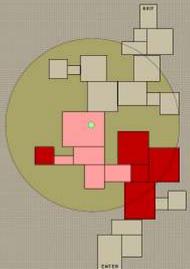
ナビゲーションメッシュによる
アクティブエリアセット(プレイヤーがいて
AIディレクターが制御対象とする領域)
の追跡

Michael Booth, "The AI Systems of Left 4 Dead," Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

Goal: Promote Replayability
Procedurally Populated Environments

➤ Where to spawn

- Behind Survivors
 - Only select valid areas in the AAS that are at or behind the Survivor team's "flow" distance
 - 75% of Mobs come from behind, since wanderers and Special/Boss Infected are usually engaged ahead of the team

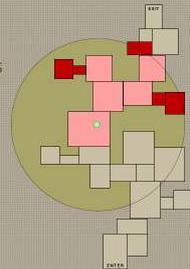


Michael Booth, "The AI Systems of Left 4 Dead," Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

Goal: Promote Replayability
Procedurally Populated Environments

➤ Where to spawn

- Ahead of Survivors
 - Only select valid areas in the AAS that are at or greater than the Survivor team's "flow" distance

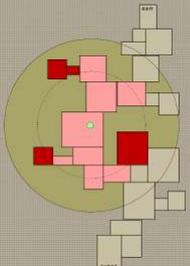


Michael Booth, "The AI Systems of Left 4 Dead," Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

Goal: Promote Replayability
Procedurally Populated Environments

➤ Where to spawn

- Near Boomer Vomit Victim
 - Only select valid areas in the AAS that are near the Boomer Vomit Victim's "flow" distance

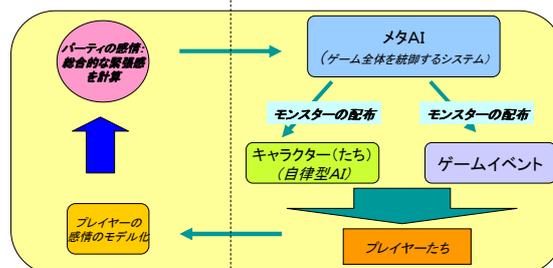


Michael Booth, "The AI Systems of Left 4 Dead," Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

Counter Strike から L4D へ： くり返し遊べる協力ゲーム体験

From COUNTER-STRIKE to LEFT 4 DEAD: Creating Replayable Cooperative Experiences

構築された非予測性



From COUNTER-STRIKE to LEFT 4 DEAD: Creating Replayable Cooperative Experiences

Counter Strike から L4D へ: くり返し遊べる協力ゲーム体験

構築された非予測性

① カウンターストライクは今なおロングセラー →何故?
→分析→オンラインゲームでの緊張感と緩和の緩急

②プレイヤーに緊張の波を作りたい→敵の出現でコントロール

③ゾンビ恐怖ゲームが最適!

参考文献

- (1) 良質なゲームを生み出すプレイテストのアプローチ Valveの実験心理学博士が明かす、プレイテスト手法のあれこれ(GameWatch)
http://game.watch.impress.co.jp/docs/news/20090329_80132.html
- (2) Valve語る、「Counter-Strike」から「Left 4 Dead」へ
協力プレイ、リプレイ性、AIディレクターの秘密(GameWatch)
http://game.watch.impress.co.jp/docs/news/20090327_80051.html
- (3) Valve Software、「Physical Gameplay in Half-Life2」 (GameWatch)
物理エンジンが生むリアリティーとゲームとしての面白さのバランス
<http://game.watch.impress.co.jp/docs/20060325/valve.htm>
(実験段階)
- (4) Left 4 Dead Interview (Eurogamer)
<http://www.eurogamer.net/articles/left-4-dead-interview>
- (5) なんのためにボスキャラがいるのか? (4gamers)
「Left 4 Dead」に見る、プレイヤーをCo-opに誘う仕掛け
<http://www.4gamer.net/games/035/G003579/20090327043/>

参考文献

昨年までのValve の発表資料です。

- (1) How To Go From PC to Console Development (GDC 2008)
http://www.valvesoftware.com/publications/2008/GDC2008_CrossPlatformDevelopment.pdf
- (2) GRID Broadband Content Delivery Platform (GDC 2008)
http://www.valvesoftware.com/publications/2008/GDC2008_StylezationWithAPurpose_TF2.pdf
- (3) Source Engine 解説
<http://source.valvesoftware.com/SourceBrochure.pdf>

メタAIの歴史①

岩谷徹氏:
ファンと一般のユーザーを満足させる方法のひとつは人工知能AIのような考え方です。

プレイヤースキルをプログラム側から判断して、難易度を調整していくというものです。これを私はセルフゲームコントロールシステムと呼んで10年以上前から開発に使っています。

— International Game Designers Panel —
http://game.watch.impress.co.jp/docs/20050312/gdc_int.htm

<http://spitfire.client.jp/shooting/xevious2.html>

メタAIの歴史①

遠藤雅伸氏 あと面白い機能なんですけれど、ゼビウスには非常に簡単なAIが組み込まれています。

「プレイヤーがどれくらいの腕か」というのを判断して、出てくる敵が強くなるんです。

強いと思った相手には強い敵が出てきて、弱いと思った相手には弱い敵が出てきます。そういったプログラムが組み込まれています。ゲームの難易度というのは「初心者には難しく、上級者には簡単だ」ということが、ひとつの難易度で(調整を)やっていくと起きてしまうので、その辺を何とか改善したいな、ということでそういったことを始めてみたのですけれど、お陰で割合にあまり上手くない人でも比較的長くプレイできる、うまい人でも最後のほうに行くまで結構ドラマチックに楽しめる、そういった感じになっています。

— ゼビウスセミナー

<http://spitfire.client.jp/shooting/xevious2.html>

メタAIの歴史②

「エージェント・アーキテクチャーから作るキャラクターAI」
(CEDEC2007、三宅)
<http://www.igda.jp/modules/mydownloads/visit.php?cid=2&lid=78>

...その他、多数の使い方がある。

Left 4 Dead のメタAIの新しい点

Left 4 Dead は、メタAIを、調整と言う意味ではなく、

- ① ゲームの面白さを作るために、**積極的に用いた**
- ② ユーザーをモデル化してプロシージャルにユーザーの状態を解析して対応した。

という2点で、これまでのメタAIの使い方とは違う。

メタAIを使う場合の注意

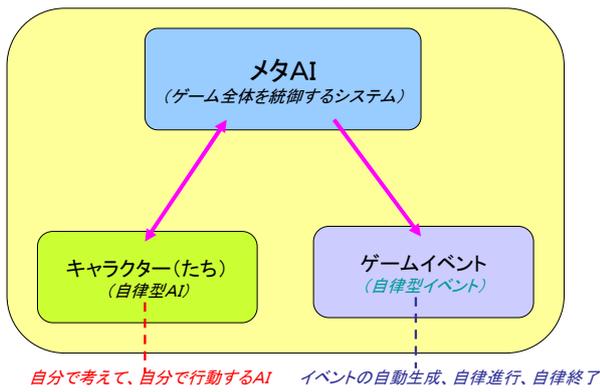
個々のAIは**ローカル**な変化に対応し、
メタAIは、**グローバル**な状況と時間的変化を解析する。

メタAIは、個体のAIがある程度できた後でこそ、
最大の威力を発揮する。

個体のAIの不備は、グローバルなAIで埋めようとしても
埋まらない。そして、ユーザーのファースト・インプレッションは、
個々のAIの振る舞い、ゲーム全体としては、メタAIで決まる。

メタAIを使う場合は、
まず、個々のAIの知性がある程度完成させることが必要。

これからのゲームAIの全体像(フルフレーム)



イベントの自動生成はあり得るか？

- アルゴリズム(プランニング、イベントグラフなど)
- バス検索技術によるNPCの非局在化
(これまでのイベントのように場所を限定する必要がない)

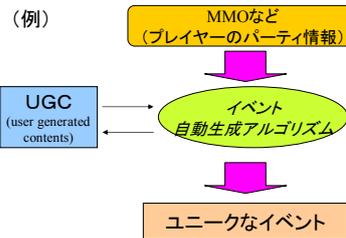
テクニック

① ある程度、内容が決まったイベントで、トリガー条件を満たされれば、発生する。
(例) ユーザーが作る家の密度が増すと、ドラゴンが出て来て焼き払おうとする。

② プレイヤーに善悪度や評判度というパラメータを振っておき、その数値によって街の人々の対応や、イベント発生率、商店での価格が変化する。
(例) Fable (Lionhead Studio)

③ シナリオ生成、あらかじめ決められたシナリオの役割に、プレイヤーやAIを当てはめる。
(例) パーティで一番強いプレイヤーの一人が、他のプレイヤーから魔物に見えてしまうイベントなど。MMOの中で、「宿敵」を自動選別して、マッチングさせる。

④ プランニング、イベントの単位を、(前提条件、イベント、結果)という形で複数階層し、それを連鎖アルゴリズムで、条件に応じてつないで行く。
(例) Façade (Michal Metas, CMU, Oz Project)



参考文献

- (1) FABLE (XBOX, Lionhead Studio, published by Microsoft)
<http://www.xbox.com/ja-JP/games/f/fable/>
- (2) Façade <http://www.interactivestory.net/>
- (3) Oz Project
<http://www.cs.cmu.edu/afs/cs.cmu.edu/project/oz/web/papers.html>

プログラマーとゲーム

ハードコーディング型実装

ゲームを作ってリリース。ゲーム性は固定。



プログラミング
実装



プログラマーとゲーム

動的に変化するゲーム

リアルタイムにユーザーの特性やゲーム状態を解析して
自分自身(ゲーム性)を変化させるゲーム



プログラミング
実装

GAME



メタAIの出現

万人向けに
完成したバランスのゲーム

ユーザーに応じた
ゲームがその場で生成

デジタルゲームの世界は、
「誰にでも楽しめるバランスを取れるゲーム」に対して、
「ユーザー特性やスキルに応じたゲームをリアルタイム制御で生成」
するメタAI技術が台頭しつつある。

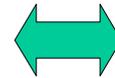
ゲーム性をリアルタイムにコントロールするメタAI (Meta AI)

本授業の主旨

4つの言葉を覚えて帰しましょう

- (1) エージェント (Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

ゲーム体験 (AI vs プレイヤー)

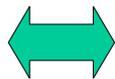
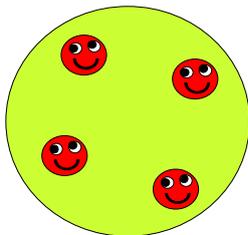


それなりに賢くなる(通常のゲーム)
一体(巨大)に特化して賢くなる
ボス戦などで特別なアルゴリズム

例) ワンダと巨像

イラスト: アンの小箱 <http://www.ame-box.com/>

ゲーム体験 (AIたち vs プレイヤー)

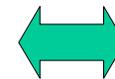
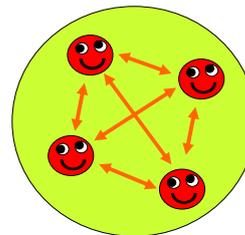


複数の敵が出るが連携しない

例) JRPGの敵たち

イラスト: アンの小箱 <http://www.ame-box.com/>

ゲーム体験 (マルチエージェント vs プレイヤー)

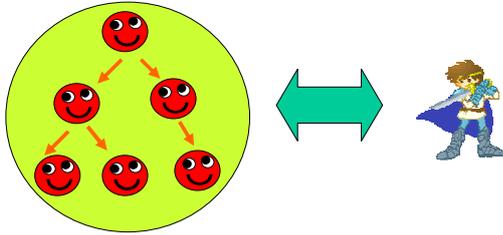


相互に連携する(マルチエージェント)

例) クロムハウズ(第3回資料、CEDEC2007資料参照)

イラスト: アンの小箱 <http://www.ame-box.com/>

ゲーム体験 (AIチーム vs プレイヤー)

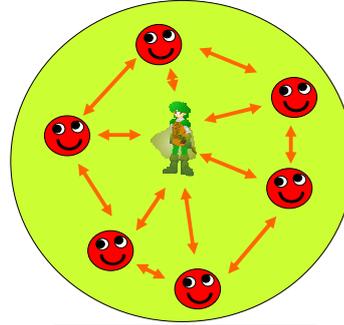


組織的構造を入れてチームとして行動する

例) Killzone2 (GDC10 報告会資料参考) など FPS

イラスト: アンの小説 <http://www.amn-box.com/>

ゲーム体験 (プレイヤー in AI社会)



AIたちの作る社会の中で社会的な活動する

例) The Sims (GDC10 報告会資料参考) など

イラスト: アンの小説 <http://www.amn-box.com/>

FPS(戦闘) ⇔ 社会系のAIの比較

	FPS	社会系
内面	シンプル	複雑
思考	複雑	シンプル
周囲の環境の見え方	視線・射線に対する障壁	多様な行為の アフォーダンス
産み出したいもの	戦場	社会シミュレーション
組織・人間関係	階層的・厳密	平坦的・緩い繋がり
スコープ時間	短期～長期	短期～長期
重要なもの	敵味方の生死	自分の幸福度
インタラクション	暴力・治癒	多様
学術との関係	アルゴリズム	哲学、認知科学、 心理学など

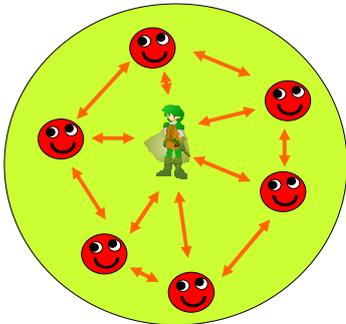
キャラクターAIの技術はこの両極の間にある

FPS(戦闘) ⇔ 日常系のAIの比較



ゲームAIは広大にして多様
FPS⇔日常系はそれを捉える一つの軸

ゲーム体験 (プレイヤー in AI社会)

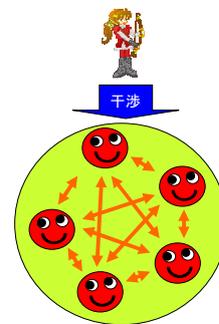


AIたちの作る社会の中で社会的な活動する

例) The Sims (GDC10 報告会資料参考) など

イラスト: アンの小説 <http://www.amn-box.com/>

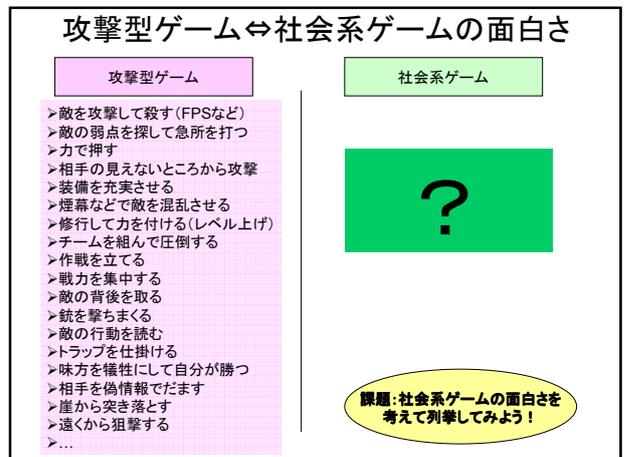
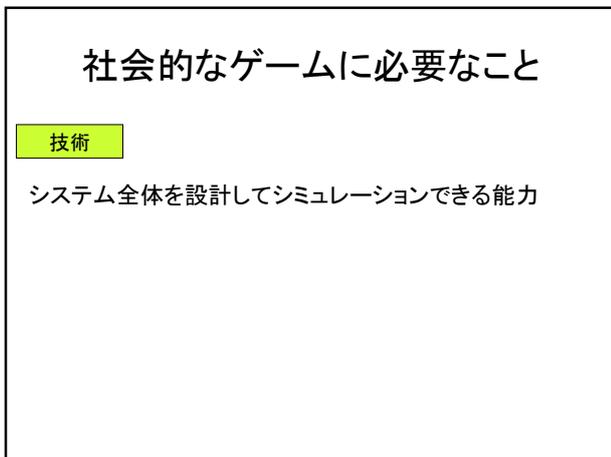
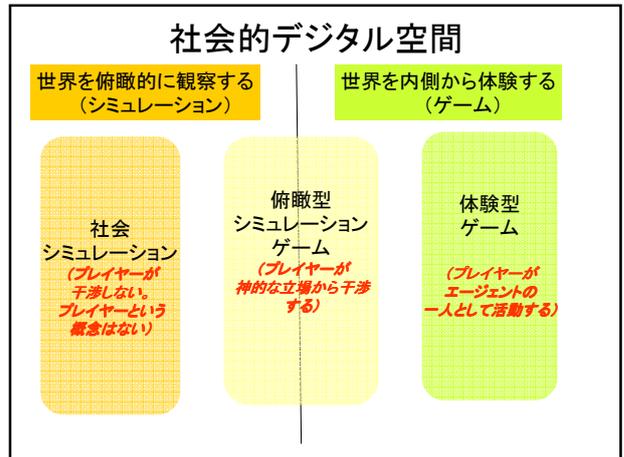
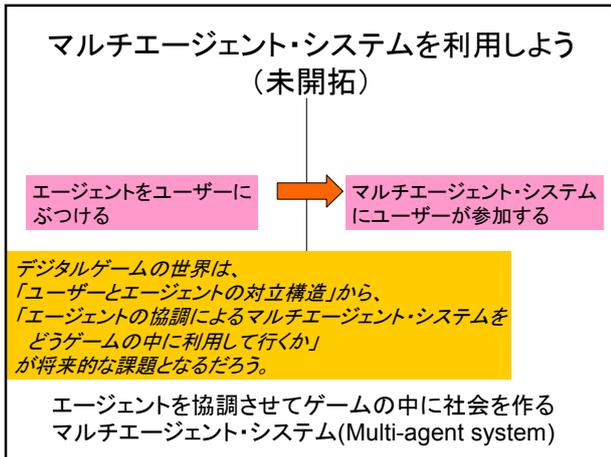
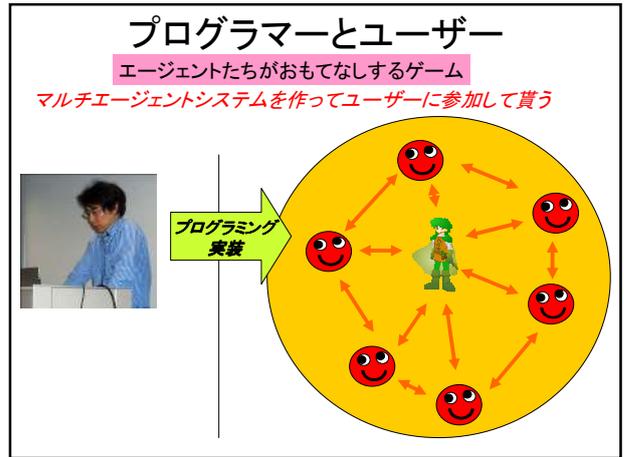
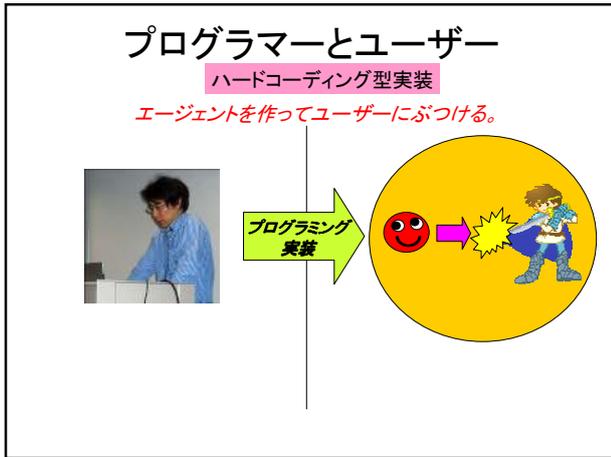
ゲーム体験 (プレイヤー to AI社会)



AIたちの作る社会に対して干渉する

例) くりきん、牧場系ゲーム、The Sims (所謂GODゲーム)

イラスト: アンの小説 <http://www.amn-box.com/>



攻撃型ゲーム⇔社会系ゲームの面白さ

攻撃型ゲーム	社会系ゲーム
<ul style="list-style-type: none"> 敵を攻撃して殺す(FPSなど) 敵の弱点を探して急所を打つ 力で押す 相手の見えないところから攻撃 装備を充実させる 煙幕などで敵を混乱させる 修行して力をつける(レベル上げ) チームを組んで圧倒する 作戦を立てる 戦力を集中する 敵の背後を取る 銃を撃ちまくる 敵の行動を読む トラップを仕掛ける 味方を犠牲にして自分が勝つ 相手を偽情報でだます 崖から突き落とす 遠くから狙撃する ... 	<ul style="list-style-type: none"> 喋る 経済活動をする 社会的活動をする 日常生活を行う 衣食住を充実させる 物を交換する 知識や技術を見につける コミュニティを作る 計画を立てる 一つのことに取り組む こっそり人の役に経つ 演説する 社会の情勢、経済の動向を読む 街を整備する 人のために何かする 正確な情報を伝える/偽情報を流す 崖から落ちそうな人を助ける 近くでお年寄りをサポートする ...

攻撃型ゲーム⇔社会系ゲームの面白さ

攻撃型ゲーム	社会系ゲーム
<ul style="list-style-type: none"> 敵を攻撃して殺す(FPSなど) 敵の弱点を探して急所を打つ 力で押す 相手の見えないところから攻撃 装備を充実させる 煙幕などで敵を混乱させる 修行して力をつける(レベル上げ) チームを組んで圧倒する 作戦を立てる 戦力を集中する 敵の背後を取る 銃を撃ちまくる 敵の行動を読む トラップを仕掛ける 味方を犠牲にして自分が勝つ 相手を偽情報でだます 崖から突き落とす 遠くから狙撃する ... 	<ul style="list-style-type: none"> 喋る 経済活動をする 社会的活動をする 日常生活を行う 衣食住を充実させる 人の役に立つ 知識や技術を見につける コミュニティを作る しかし... こうした楽しみがデジタルゲームで可能なのだろうか?

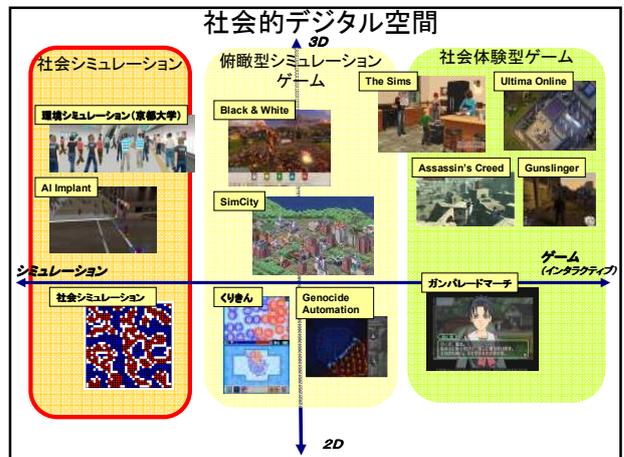
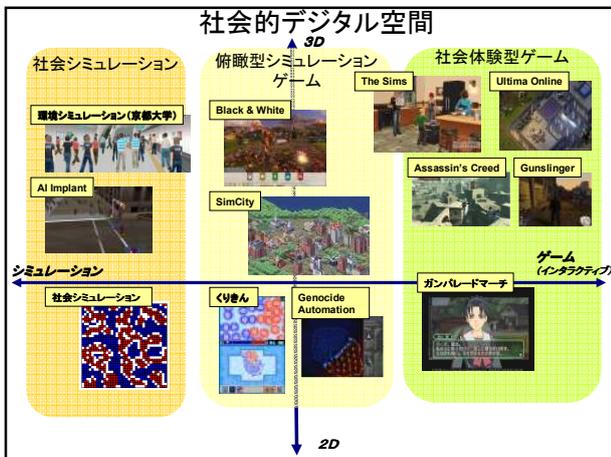
社会的なゲームに必要なこと

技術
システム全体を設計してシミュレーションできる能力

企画
社会的な喜びをゲーム内で再現する能力

社会的デジタル空間

世界を俯瞰的に観察する (シミュレーション)	世界を内側から体験する (ゲーム)	
<p>社会シミュレーション (プレイヤーが干渉しない。プレイヤーという概念はない)</p>	<p>俯瞰型シミュレーションゲーム (プレイヤーが神的な立場から干渉する)</p>	<p>体験型ゲーム (プレイヤーがエージェントの一人として活動する)</p>



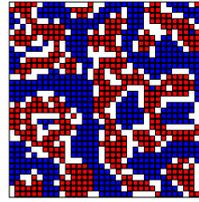
社会シミュレーション

完全なシミュレーション

地域社会の住み分け

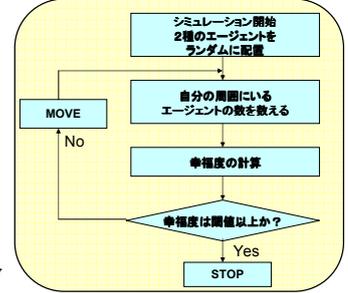
「シェリング「分居」モデル

2種類のエージェントが周囲に自分と同じ種類のエージェントが住まう環境を求めて行動する。



DEMO!

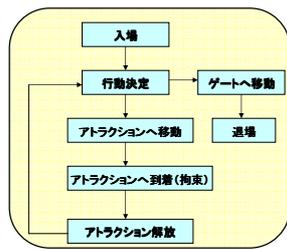
幸福度 = 近傍にいる同種エージェント数 / 近傍にいる総エージェント数



田村誠、板山真弓、山影道、第7章「地域社会の住み分け」、コンピューターの中の社会(共立出版)

テーマパーク・シミュレーション

DEMO!



アトラクション効用(アトラクションを選ぶ確率)
= アトラクションの嗜好度
- $\alpha \times$ アトラクションまでの直線距離
- $\beta \times$ アトラクションの混雑状況
(α, β はシミュレーション定数)

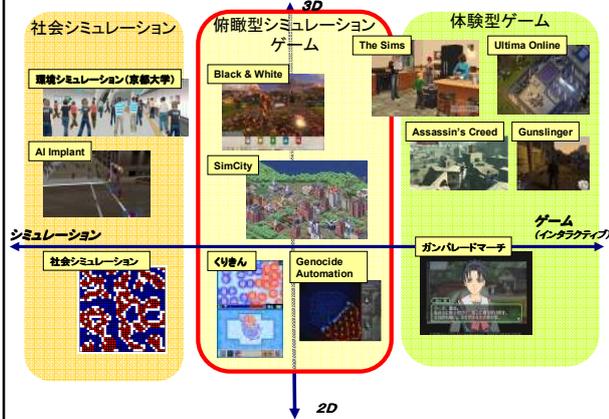
エージェントが持つパラメーター
(1) 各アトラクションへの嗜好度
(2) すでに訪問したアトラクション
(3) 満足度 ((1)(2)より計算)
(4) 帰宅満足度(満足度がこの値より大きければ帰宅)

辺見和晃、第8章「来場者に優しいテーマパーク」、コンピューターの中の社会(共立出版)

社会的デジタル空間



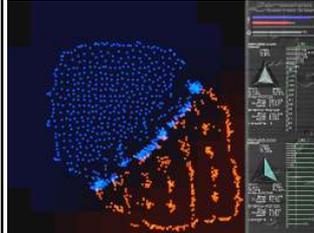
社会的デジタル空間



群知能乱戦シミュレータ「Genocide Automation」

Sense of Wonder Night 入賞作品、佐々木直哉、<http://www11.plala.or.jp/normal/>

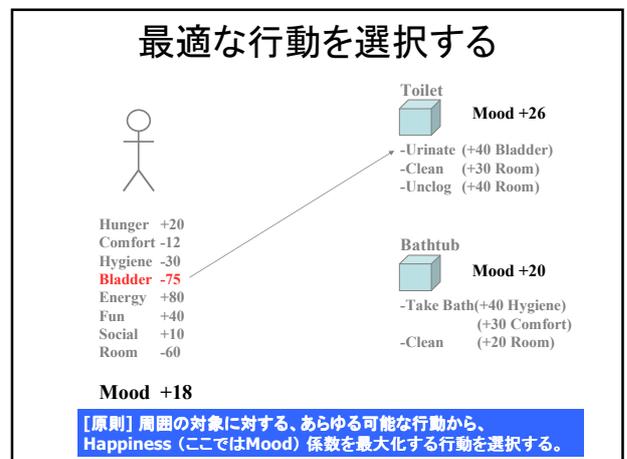
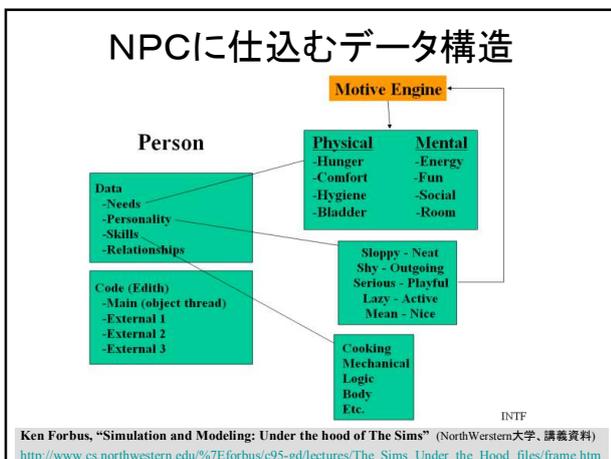
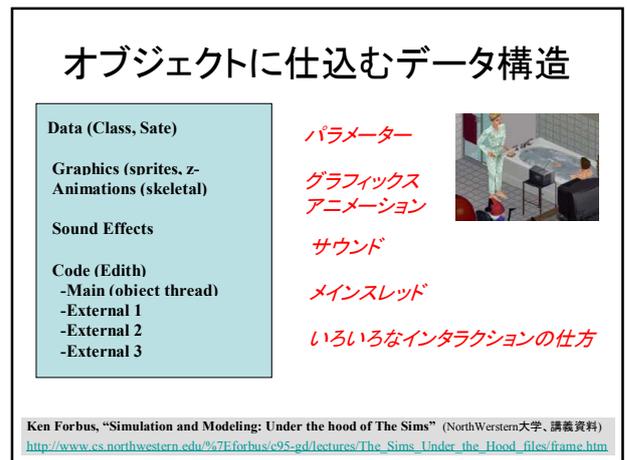
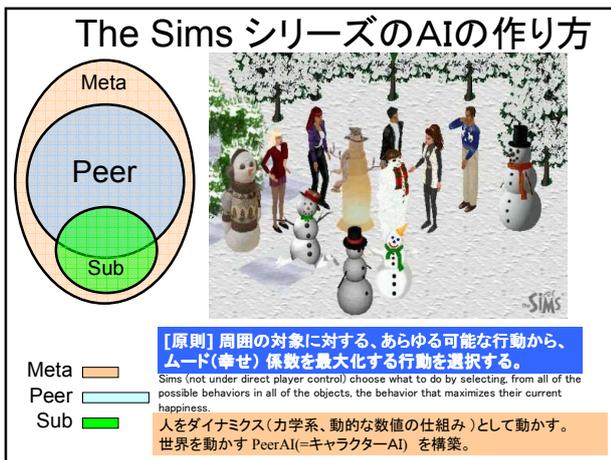
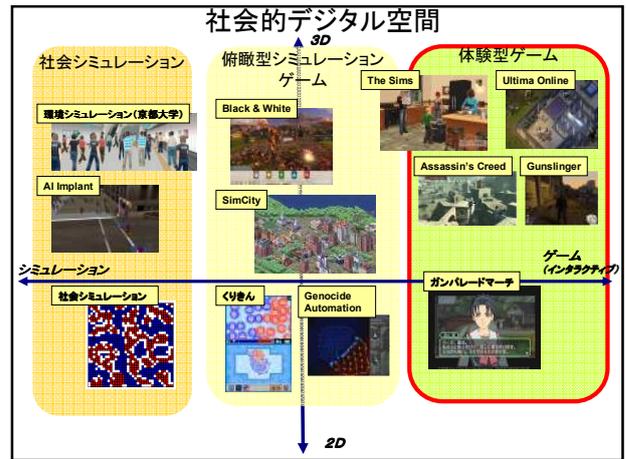
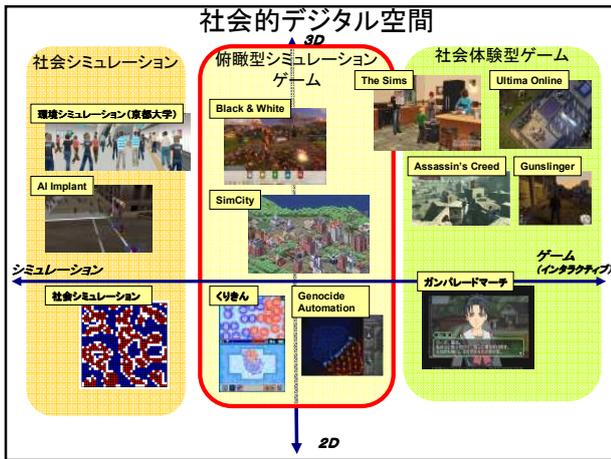
DEMO!

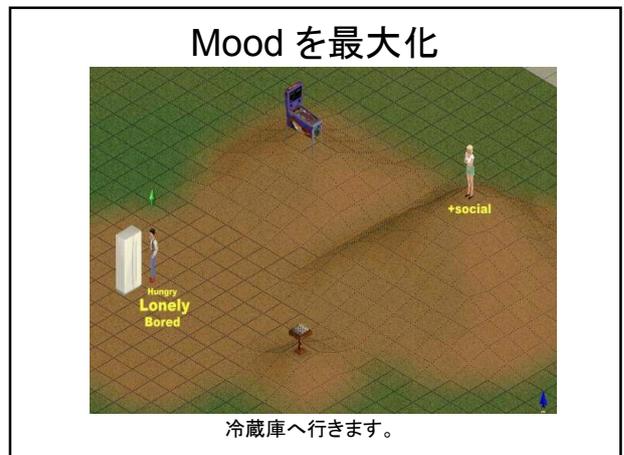
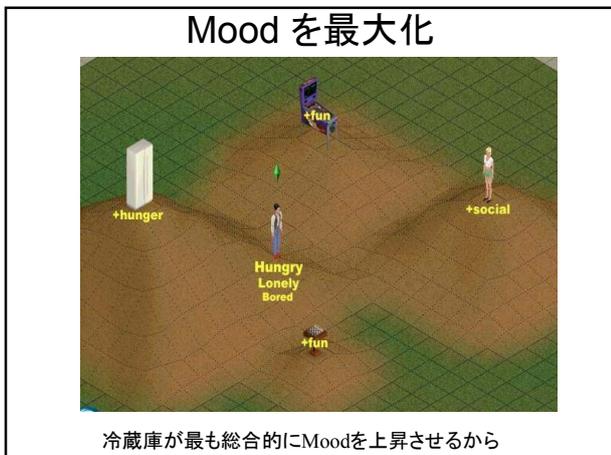
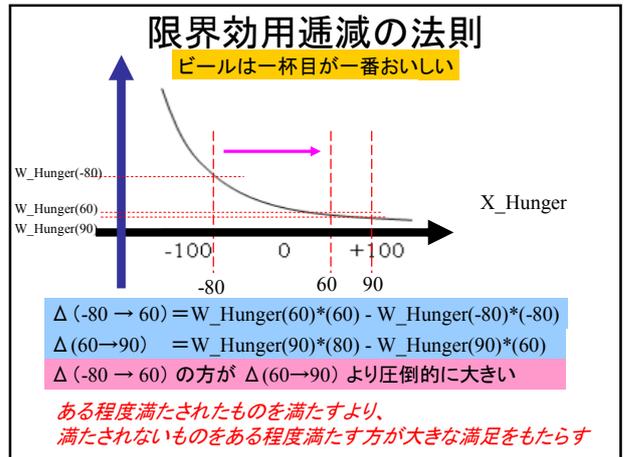
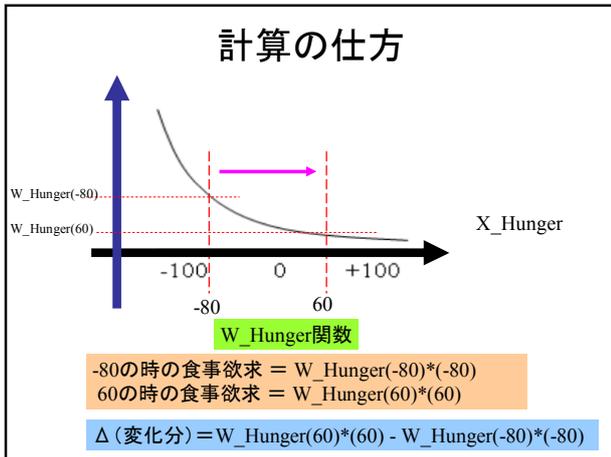
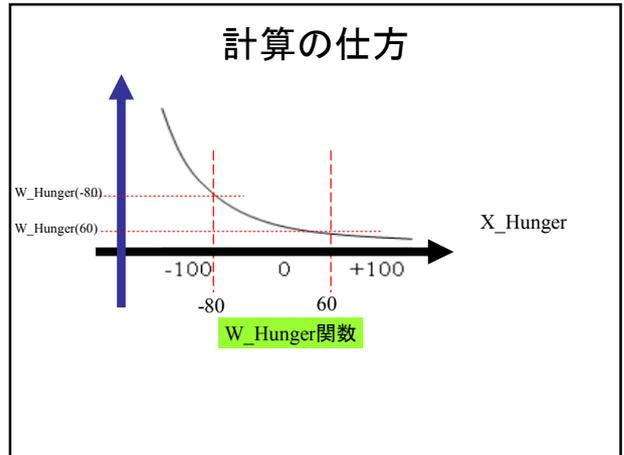
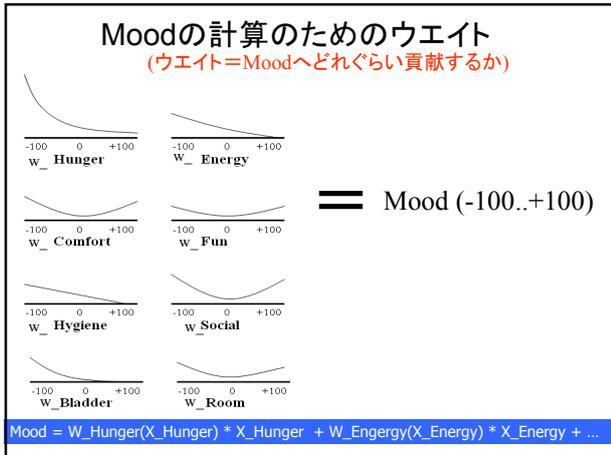


- (1) ユニットに分かれている
- (2) ユニットは「資源」を消費。資源が不足するとダメージ
- (3) エリアは敵、味方の数から占有率を計算。
- (4) ユニット毎に味方、敵戦力を計算。優勢と判断すれば戦う、でなければ逃げる。
- (5) ユニットは合流閾値によって合流する。

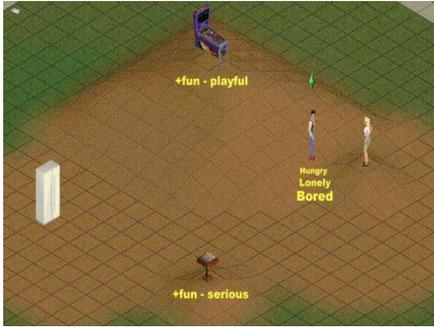
[左クリック]
・視界内に敵がいなければカーソル位置へ向かう。
・表示される矩形内において、敵が視界内にいる場合には、味方の戦力を5倍に見積もる(好戦的になる)

[右クリック]
・視界内に敵がいなければカーソル位置から遠ざかる。
・表示される矩形内において、敵が視界内にいる場合には、敵の戦力を5倍に見積もる(消極的になる)





Mood を最大化



お腹が膨れたので、ちょっと退屈だから、女の子と話します。

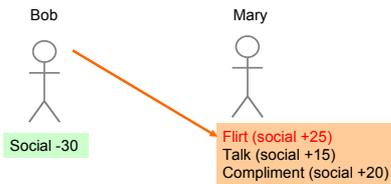
The Sims の最初の目標

エージェントが社会生活を営みながら、
インタラクションする社会を作りたい。

どうすればよいか？

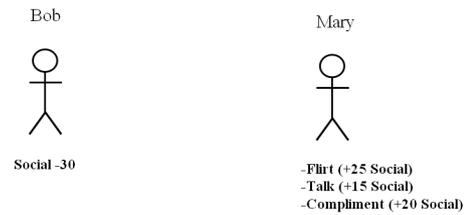
- (1) 自律的に生活を営む。
- (2) エージェント同士の社会的にインタラクション
- (3) Edith による各行動のつくり込み

(2) NPC-NPC インタラクション



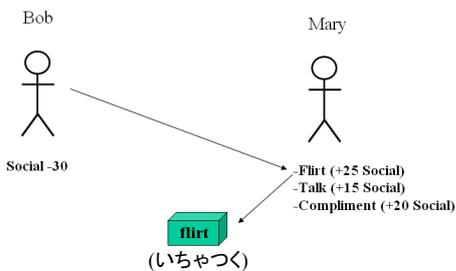
NPC 同士も、相手が自分のパラメーターをどう変化させるかを計算して決定する。上記では、Bob は、社交パラメーターが低下しており、Flirt (いちゃつく) ことで、それが回復するので、Flirt を選択する。

NPC-NPC インタラクション



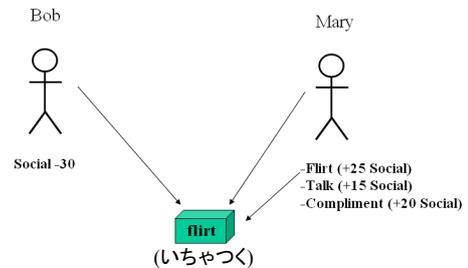
Kenneth D. Forbus "Some notes on programming objects in. The Sims"
http://www.qrg.northwestern.edu/papers/Files/Programming_Objects_in_The_Sims.pdf

NPC-NPC インタラクション

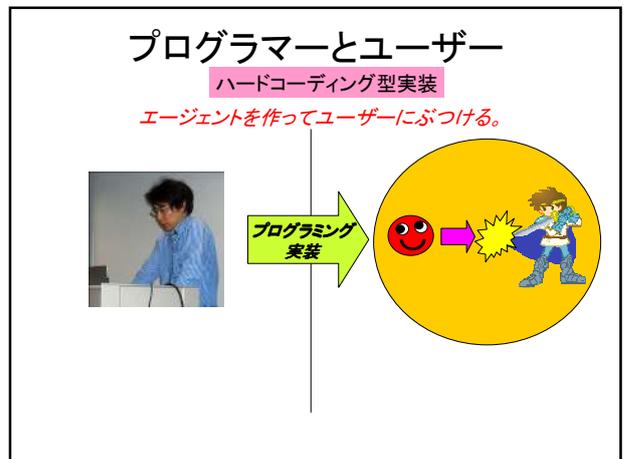
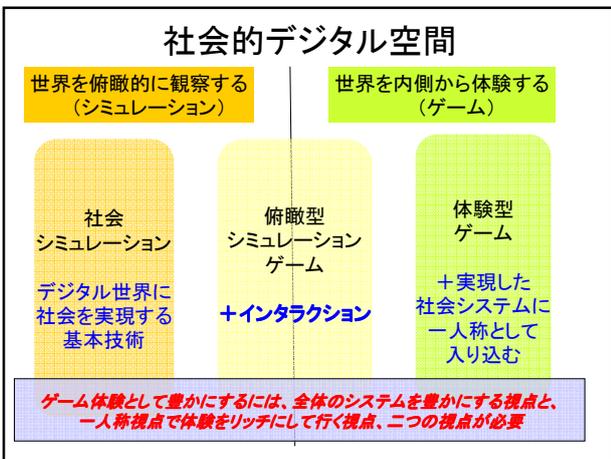
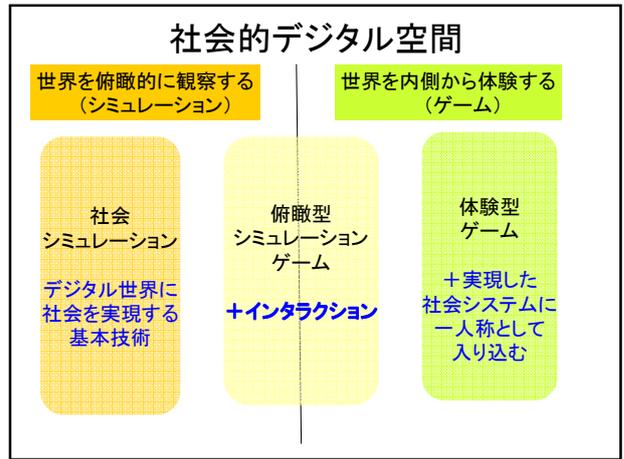
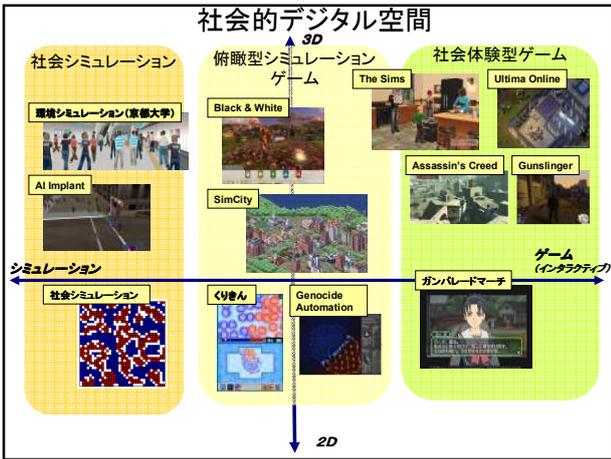
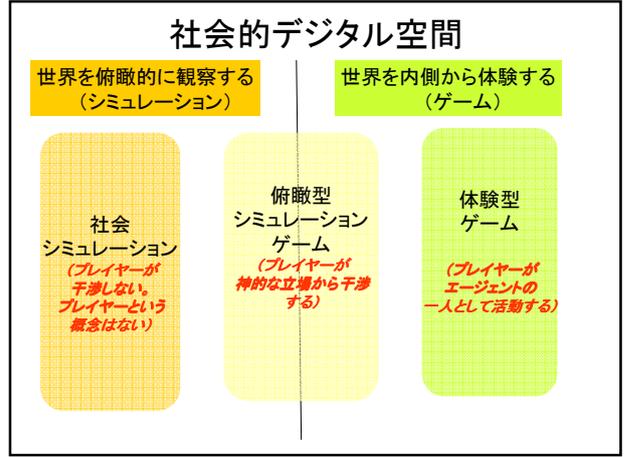


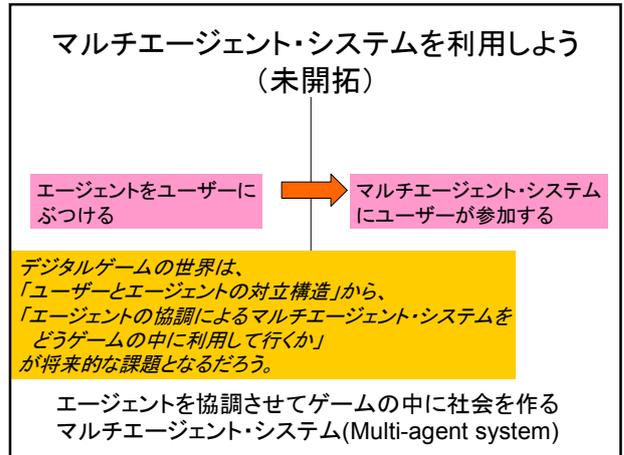
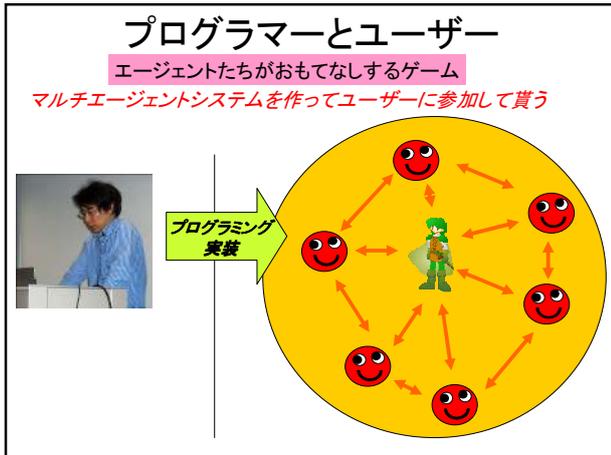
Kenneth D. Forbus "Some notes on programming objects in. The Sims"
http://www.qrg.northwestern.edu/papers/Files/Programming_Objects_in_The_Sims.pdf

NPC-NPC インタラクション



Kenneth D. Forbus "Some notes on programming objects in. The Sims"
http://www.qrg.northwestern.edu/papers/Files/Programming_Objects_in_The_Sims.pdf



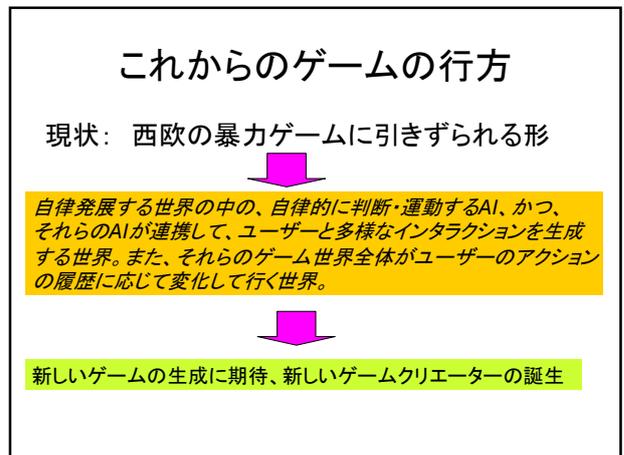
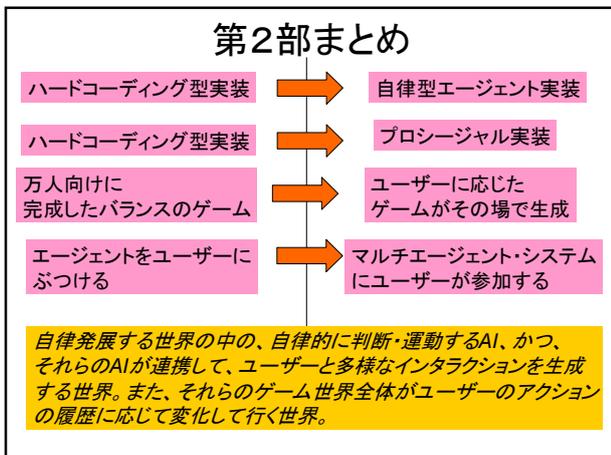


第2部 まとめ

第2部まとめ

デジタルゲームにおける人工知能技術

- 第一章 自律型エージェント (Autonomous Agent)
- 第二章 プロシージャル (Procedural)
- 第三章 メタAI (Meta-AI)
- 第四章 マルチエージェント(Multi-agent)



[付録] ゲームAIの学習・研究の仕方

- (I) 日本語の文献が殆どない、或いは、資料は英語でよいものが多いので英語は必須。
- (II) ゲームAIの教科書は英語で多く出版されている。
- (III) WEB上にゲームAIでよい英語論文が見つかる。
- (IV) 日本語の資料は少ないが質が高い。

(I) 参考文献(日本語)

- (1) 「FSM」「プランニング」「評価値法」など、ゲームAIの基礎技術については、



オライリー・ジャパン

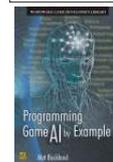
「実例で学ぶゲームAIプログラミング」
(Mat Buckland 著、松田晃一 訳)

の解説が優れています。

ソースコードはWEB

<http://www.wordware.com/files/ai/>

できれば原書で読みましょう！



(I)参考文献(日本語)

(2)

- ①「世界表現」「プランニング」については、IGDA日本のHPの「ダウンロード」から、三宅が書いた第1、2、5、6回セミナーの教科書、CEDEC2006の資料がDLできます。

<http://www.igda.jp/>

(センサーの実装の仕方、記憶の利用法などを知りたい方は必読)
上記サイト復旧中のため一時的に以下のフォルダから

<http://server02.joeswebhosting.net/~ig1347/modules/mydownloads/>

- ② デジタルコンテンツ協会

デジタルコンテンツ制作の先端技術応用に関する調査研究報告書(第3章)

http://www.dcaj.org/report/2007/ix1_07.html

(PDFファイルがダウンロード出来ます。)

- ③ 人工知能学会誌 Vol. 23 No. 1 (2008年1月)「ゲームAI特集」
「デジタルゲームにおける人工知能技術の応用」(三宅)

(II)参考文献(英語)

WEB

Mat Buckland

ai-junkie <http://www.ai-junkie.com/ai-junkie.html>

Craig Reynolds

RAYNOLDS <http://www.red3d.com/>

リンク集 <http://www.red3d.com/cwr/games/>

Steven Rabin

GameAI <http://www.gameai.com/>

CGF-AI

CGF-AI <http://www.cgf-ai.com/>

リンク集 <http://www.cgf-ai.com/links.html>

(II)参考文献(英語)

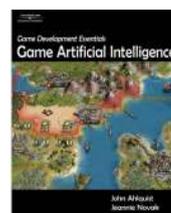
書籍

AI Game Programming Wisdom 1 - 4



ゲーム開発者、研究者による、
それぞれのタイトルの実装例、研究成果

書籍



欧米のゲームAIの歴史から最先端までがわかりやすく、解説されています。

文科系の方も理科系の方も読んで楽しめる本です。

John Ahlquist, Jeannie Novak

Game Development Essentials: Game Artificial Intelligence



References for Killzone

- [1] William van der Sterren (2001), "Terrain Reasoning for 3D Action Games", http://www.cgf-ai.com/docs/gdc2001_paper.pdf
- [2] William van der Sterren (2001), "Terrain Reasoning for 3D Action Games(GDC2001 PPT)", http://www.cgf-ai.com/docs/gdc2001_slides.pdf
- [3] Remco Straatman, Arjen Beij, William van der Sterren (2005), "Killzone's AI : Dynamic Procedural Combat Tactics", http://www.cgf-ai.com/docs/straatman_remco_killzone_ai.pdf
- [4] Arjen Beij, William van der Sterren (2005), "Killzone's AI : Dynamic Procedural Combat Tactics (GDC2005)", http://www.cgf-ai.com/docs/killzone_ai_gdc2005_slides.pdf
- [5] Damian Isla (2005), "Dude, where's my Warthog? From Pathfinding to General Spatial Competence", <http://www.aiide.org/aiide2005/talks/isla.ppt>



Reference for Halo & Halo2

- Damian Isla (2005), "Dude, where's my Warthog? From Pathfinding to General Spatial Competence", <http://www.aiide.org/aiide2005/talks/isla.ppt>
http://nikon.bungie.org/misc/aiide_2005_pathfinding/index.html
- Damian Isla (2005), Handling Complexity in the Halo 2 AI, Game Developer's Conference Proceedings., http://www.gamasutra.com/gdc2005/features/20050311/isla_01.shtml
- Jaime Griesemer(2002),The Illusion of Intelligence: The Integration of AI and Level Design in Halo, <http://halo.bungie.org/misc/gdc.2002.haloai/talk.htm>
- Robert Valdes(2004), "In the Mind of the Enemy The Artificial Intelligence of Halo2", <http://www.stuffo.com/halo2-ai.htm> (現在はclosed)

References for C4 Architecture

- (1) MIT Media Lab Synthetic Characters Group, <http://characters.media.mit.edu/>
- (2) R. Burke, D. Isla, M. Downie, Y. Ivanov, B. Blumberg, (GDC2001), "CreatureSmarts: The Art and Architecture of a Virtual Brain", <http://characters.media.mit.edu/Papers/gdc01.pdf>
- (3) D. Isla, R. Burke, M. Downie, B. Blumberg (2001), "A Layered Brain Architecture for Synthetic Creatures", <http://characters.media.mit.edu/Papers/ijcai01.pdf>
- (4) D. Isla, B. Blumberg (2002), "Object Persistence for Synthetic Characters", <http://characters.media.mit.edu/Papers/objectPersistence.pdf>
- (5) Movies of Duncan, <http://web.media.mit.edu/~bruce/whatsnew.html>
- (6) Object Persistence for Synthetic Characters. D. Isla, B. Blumberg. In the Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS2002., <http://characters.media.mit.edu/Papers/objectPersistence.pdf>



References for F.E.A.R.

- 論文 Orkin, J. (2006), **3 States & a Plan: The AI of F.E.A.R.**, Game Developer's Conference Proceedings.
- Jeff Orkin, Applying Goal-Oriented Action Planning to Games, AI Game Programming Wisdom 2, Charles River Media., 217-228, (2003)
- 参考文献(I) Mat Buckland, "Programming Game AI by Example", Chapter 9, WORDWARE publishing (第9章とそのサンプルコードはゴール指向型プランニングの優れた解説です。教科書をお探しの方は、ゲームAIについて最良の書の一つです。推薦します。) **オンラインジャーナルより「実例で学ぶゲームAIプログラミング」として翻訳が出版されています。**
- 参考文献(II) Jeff Orkin' HP <http://web.media.mit.edu/~jorkin/> (Jeff Orkin は、米におけるゲームAIにおけるゴール指向型プランニングの推進者の一人。著者のサイトに豊富な情報がありません。)
- 参考文献(III) 星野 瑠美子, "F.E.A.R.のAI - 3つの状態とゴール指向プランニングシステム" http://www.igda.jp/modules/xeblog/?action_xeblog_details=1&blog_id=62 (IGDA Japan サイト内、GDC2006講演の紹介)

ご清聴ありがとうございました。

質疑応答

これ以外に、意見や質問があれば、メールへ

y_miyake@fromsoftware.co.jp

ご清聴ありがとうございました。



これ以外に、意見や質問があれば、メールかアンケートへ

y_miyake@fromsoftware.co.jp

(IGDA Japan登録アドレス yoichi-m@pk9.so-net.ne.jp)

<http://www.igda.jp>