

2007年度 卒業論文

ファジィ理論を用いた
ゲームバランス調整作業の効率化

指導教員：渡辺 大地講師

メディア学部 ゲームサイエンスプロジェクト
学籍番号 M0104092
大西 克哉

2007年度 卒業論文概要

論文題目

ファジィ理論を用いた
ゲームバランス調整作業の効率化

メディア学部

学籍番号：M0104092

氏名

大西 克哉

指導
教員

渡辺 大地講師

キーワード

ゲームデザイン、ゲームバランス
ファジィ理論、確率分布

近年、ゲーム産業の総出荷規模は2兆9,364億円と無視できない市場を持っている。ゲームはハードウェア技術の進歩により表現の幅は広がったが、ソフトウェア技術におけるゲーム制作の体系化といった研究は余り進んでいないのが現状である。そこで本研究ではソフトウェア技術の中でもゲームバランス調整作業に着目する。

ゲームバランス調整作業とは、ゲームの難易度をゲームデザイナーの理想とするものに近付けるためにゲーム内に設定している各々の数値を調整する作業のことである。本研究ではゲームバランス調整作業の中でも確率分布を扱うものに着目した。確率分布とは、ある事柄に対して確率によって算出する結果があるとき、それぞれの結果がどの程度の確率で起きるかをまとめたものである。確率分布をゲームに用いる場合は、一様乱数では出すことのできない戦略の奥深さを出すために用いる。従来手法を用いて確率分布を調整する場合は確率分布から実際にどの程度ばらつくかを想像できる必要があり、ゲームデザイナーの経験に頼っているのが現状である

そこで本研究では、ファジィ理論を用いて確率を直接操作せずに直感的に確率分布を生成する手法を提案する。ファジィ理論とは真と偽の間のグレーゾーンを認める理論で、曖昧な入力からシステム制御を行うことを得意とする理論である。本提案手法により、統計の素養がないゲームデザイナーでも直感的に確率分布を調整することが可能となった。提案手法の有用性を検証するため、提案手法をを実装したツールを用いて検証実験を行った。検証実験の結果、理想に近い確率分布を直感的に作成できたことがわかった。

目次

第1章	はじめに	1
1.1	研究背景と目的	1
1.2	論文構成	2
第2章	ゲームバランス調整手法とその問題点	3
2.1	ゲームにおける不確実性の表現	3
2.2	確率分布の種類と実例	5
2.3	従来手法とその問題点	7
第3章	ファジィ理論を用いたゲームバランス調整手法の提案	8
3.1	ファジィ理論の概要	8
3.2	本提案手法へのファジィ理論の適用	9
3.3	本研究の提案手法	11
3.4	ファジィ化とファジィルール	11
第4章	検証と考察	17
4.1	実験概要	17
4.2	検証実験	17
4.3	実験結果	18
4.4	考察	18
第5章	まとめ	20
	謝辞	21
	参考文献	22

目 次

2.1	標準正規分布のグラフ	6
3.1	ブール理論におけるメンバーシップ関数	10
3.2	ファジィ理論におけるメンバーシップ関数	10
3.3	攻撃ダメージ 96 のメンバーシップ関数	12
3.4	評価 A におけるメンバーシップ関数合成前	13
3.5	評価 A におけるメンバーシップ関数合成後	13
3.6	補正後の評価 B におけるメンバーシップ関数	14
3.7	補正後の評価 C におけるメンバーシップ関数	14
3.8	補正後の評価 D におけるメンバーシップ関数	15
3.9	本提案手法によって生成した確率分布	16

第 1 章

はじめに

1.1 研究背景と目的

ゲーム業界の 2007 年出荷実績は総出荷規模 2 兆 9,364 億円 [1] と無視できない大きさの市場を持っており、国際競争力が高い分野である。ここでいうゲームとは、クロフォードのゲームデザイン論 [2] におけるコンピュータゲームのことを指す。ゲームにおける技術はハードウェアに関する技術とソフトウェアに関する技術の 2 種類に分けることができる。ハードウェア技術とは 3DCG を用いたリアルな描写や DVD-ROM による大容量化といった技術のことを指し、ソフトウェア技術とは開発プロセスやゲームルールをデザインする技術のことを指す。近年ハードウェア技術の進歩がめざましいが、ソフトウェア技術の体系化といった研究は余り進んでいないのが現状である [3]。そこで本研究ではソフトウェア技術の中でもゲームバランス調整作業に着目する。

ゲームバランス調整作業とは、ゲームの難易度をゲームデザイナーの理想とするものに近付けるためにゲーム内に設定している各々の数値を調整する作業のことを指す。本研究ではゲームバランス調整作業の中でも確率分布を操作するものに着目する。確率分布とは、ある事柄に対して確率によって算出する結果があるとき、それぞれの結果がどの程度の確率で起きるかをまとめたものである。ゲームにおいて確率分布を用いる場合は、一様乱数では出すことのできない奥深さを出すために用いる。従来手法で確率分布を調整する場合、確率分布から実際にど

の程度ばらつくかを想像できる必要があり、ゲームデザイナーの経験に頼っているのが現状である。

そこで本研究では、理想とする確率分布を求める際にファジィ理論を用いることで直感的に確率分布を求める手法を提案する。ファジィ理論とは、真と偽しかなかった従来のブール理論に対して、真と偽の間であるグレーゾーンを認める理論である。ファジィ理論は曖昧な入力から値を導き出すことを得意としており、ゲームプログラミングにおいては AI などにファジィ理論を用いている。本研究では、ゲームの確率分布を調整する作業に対してデータを主観的に扱えるファジィ理論を用いる手法を提案する。これにより、統計の素養がないゲームデザイナーでも直感的に確率分布を調整することが可能となる。提案手法の検証には提案手法を実装したツールを用いて検証実験を行った。検証実験は RPG の戦闘シーンにおける攻撃力の値を例として取り上げた。テストプレイヤーは理想とする確率分布を想定しながらツールを用いて直感的に確率分布を生成し、生成した確率分布によって算出されたダメージのリストを第三者に評価してもらった。検証実験の結果から、理想に近い確率分布を直感的に作成できたことがわかった。

1.2 論文構成

本論文は全 5 章で構成する。まず第 2 章で用語を定義し、ゲームデザインの現状と問題点を整理する。その後、第 3 章でファジィ理論を用いた本研究の提案手法を述べ、第 4 章で検証実験の方法とその結果を考察する。最後に、第 5 章で本研究のまとめを述べる。

第 2 章

ゲームバランス調整手法とその問題点

本章ではゲームバランス調整とはどのようなものを述べ、従来手法とその問題点について述べる。

2.1 ゲームにおける不確実性の表現

本研究はゲームデザイナーがゲームバランスを調整する作業の中でも確率分布を用いるものを対象とする。ゲームデザイナーとはゲームデザインを行う人のことを指し、ゲームデザインとは文字通りゲームをデザインすることを指す。前田の著書 [4] によるとゲームデザイナーの仕事には企画、プレゼン、システムデザイン、バランス調整とあるが、桜井の著書 [5] にもある通り会社によってその役割や呼び名は混在しており、企画担当はプランナーと呼び明確に分ける例もある。そこで本研究におけるゲームデザインとはゲームのルールを決定するシステムデザイン、ゲームを面白くするためのバランス調整作業と定義する。本研究ではゲームデザイナーの仕事の中でもゲームバランス調整作業に着目する。ゲームバランス調整作業とは、ゲームデザイナーが理想とするゲームに近づけるためにゲーム内に設定している値を調整する作業のことを指す。そして、ゲーム内に設定する値の中でも確率分布の調整作業に着目する。確率分布とは、ある事柄に対しての結果があるとき、それぞれの結果がどの程度の確率で起きるかをまとめたものである。つまり

確率分布とは乱数の偏りを表し、ゲームにおいては不確実性の表現に用いている。そこで、確率分布の説明をするためにまずは乱数の説明を行う。乱数をよく用いる例としてRPG（ロールプレイングゲーム）の戦闘シーンのダメージ計算式を取り上げる。RPGの概略に関しては井上の研究 [6] に詳しい。戦闘シーンとはRPGを構成する要素の1つであり、味方キャラクターと敵キャラクターの戦闘を行うシーンを指す。ダメージ計算式とはそれぞれのキャラクターが持つ固有の値を用いる計算式で、戦闘の勝敗を決するために用いる計算式のことである。RPGでは、このダメージ計算式においてダメージを計算する際に乱数を用いることが多い。乱数を用いた具体的なダメージ計算式の例として『ドラゴンクエスト [7]』のダメージ計算式 [8] を取り上げる。味方キャラクターが敵キャラクターに与えるダメージ値を D 、味方キャラクターの攻撃基本値を d 、味方キャラクターの攻撃力を A 、敵キャラクターの防御力を B 、乱数 r の幅を $(0 \sim 255)$ としたとき、味方キャラクターが敵キャラクターに攻撃した場合のダメージ計算式が以下の (2.1)(2.2)(2.3) である。

$$d = A - \frac{B}{2} \quad (2.1)$$

$$D = 1 \quad D = 2(d < 2) \quad (2.2)$$

$$D = \frac{d + (d + 1) \left(\frac{r}{256} \right)}{2} (d \geq 2) \quad (2.3)$$

ダメージ計算式はこの『ドラゴンクエスト』のようにシンプルなものから『テイルズオブシンフォニア』のように21種類もの要素が関わるもの [9] まで多種多様である。このようにダメージ計算式に乱数を用いる理由としては、乱数を用いる

ことで与えるダメージに揺らぎが発生するからである。これにより戦闘シーンにある程度の不確実性が生まれ、ダメージが固定値である場合に比べて深い戦略をプレイヤーに要求することが可能である。しかし、このダメージ計算式で用いている乱数 r は一様乱数と呼ばれる乱数で、全ての値の出る確率が等しい乱数である。ここからさらに奥深い戦略を要求する手段として、本研究が対象とする確率分布がある。確率分布とは乱数によって出る値について、それぞれどの程度の確率で発生するかを表すものである。確率分布を用いることで生まれる戦略の例としては武器ごとのダメージがある。2つの武器があり、どちらも与えるダメージの上限を120、下限を80とする。1つ目の武器は100付近の値が出やすい武器とし、2つ目の武器は上限値である120付近と下限である80付近の値が出やすい武器とする。プレイヤーが少しでも早く敵を倒したいと考えた場合、敵の残り体力が90である場合は100付近の値が出やすい前者を、敵の残り体力が110である場合は120付近の値が出やすい後者を選ぶという戦略を取ることが可能である。このように、同じ上限と下限を持つ武器でもその確率分布に差を持たせることで武器毎の特徴を表現し、プレイヤーに選択肢を提供することからゲームにおいて確率分布は重要であると言える。

2.2 確率分布の種類と実例

本研究の研究対象である確率分布にはどのようなものなのか具体的に説明する。確率分布には様々な種類があるが、その中でも有名な分布として正規分布がある。正規分布はガウス分布とも呼ぶ分布で、身長や体重、成績など様々な事象をモデル化することが可能な分布である。その根拠は中心極限定理を中心としたものであり、中心極限定理に関しては福田の著書 [10] に詳しい。正規分布とは式 (2.4) を確率密度関数として持つものである。

$$fX(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, -\infty < x < +\infty, \sigma > 0 \quad (2.4)$$

正規分布の中でも期待値 0、分散 1 の正規分布を標準正規分布と呼ぶ。標準正規分布のグラフを図 2.1 に示す。

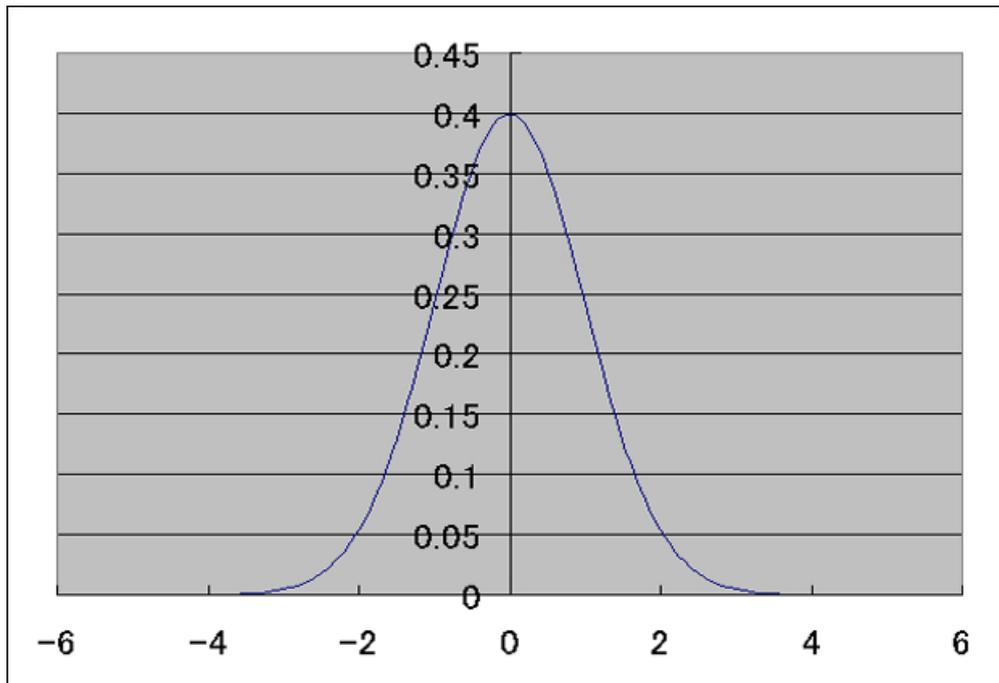


図 2.1: 標準正規分布のグラフ

馬場らの著書 [11] によると理論面、実用面どちらにおいても非常に重要な役割を演じている。ゲームデザインにおいても扱いやすい標準正規分布を用いることがある。しかし、標準正規分布では表現できない確率分布をゲームに用いたいケースが存在する。それは第 2 章第 2 節で述べた武器毎の与えるダメージの表現などである。月本らの著書 [12] によると、確率分布は標準正規分布以外にもコーシー分布やレーレー分布など数多くの確率分布があるが、それらがゲームデザイナーの理想とする確率分布であるケースは稀である。そこで、ゲームデザイナーは理想とする確率分布を得るために確率そのものを調整することになる。本研究は、このゲームデザイナーが確率分布を直接調整しなければならないという状況に着目したものである。次節で確率分布の確率を直接操作する従来手法とその問

題点について述べる。

2.3 従来手法とその問題点

現在、RPGの戦闘シーンにおけるダメージ値の確率分布を調整する際は次のような手順で行う。

1. ここでは例として剣による攻撃のダメージ確率分布を調整する。剣の基本となるダメージ値を100、ダメージの揺れ幅20%としたとき、剣によって敵に与えるダメージの上限は120、下限は80となる。その後、それぞれの値が出る確率を仮に決定する。
2. 1. で設定した確率分布を用いてテストプレイを行う。テストプレイの結果、剣による攻撃ダメージは基本値付近でもう少し安定してもよいとゲームデザイナーが感じたとする。そこで、基本値である100付近の確率を上げ、相対的に80、120の確率が下がるように確率を直接操作して調整する。このとき、理想とするダメージのばらつきにするためにどの程度確率を増減させるかはゲームデザイナーの経験と勘によって定まる。
3. 2. で新たに定めた確率分布を用いてテストプレイを重ね、再度確率を調整するという繰り返りで理想とする確率分布に近づけていく。

このように、ゲームデザイナーがゲームバランス調整のために確率分布を調整する場合、試行錯誤を繰り返すことで理想とするゲームバランスに近づけている。しかし、従来手法の場合確率を直接操作する際に確率分布のグラフから実際の戦闘でどの程度ダメージがばらつくかをある程度想像できる必要がある。そこで本研究ではファジィ理論を用いることで確率分布そのものを直接操作することなく、理想とする確率分布を作成できる手法を提案する。提案手法の詳細を次章で述べる。

第 3 章

ファジィ理論を用いたゲームバランス調整手法の提案

本章ではファジィ理論の概要を説明し、提案手法においてどのようにファジィ理論を用いて確率分布を調整するか述べる。

3.1 ファジィ理論の概要

ファジィ理論とは 1965 年にカリフォルニア州立大学バークレー校の Lotfi Asker Zadeh 教授が発表したもので、曖昧なものを扱うことができる理論である。この曖昧なものの例としては人間が普段使う言葉がある。例えば、暑い・寒い・高い・低い・速い・遅いなどの言葉はそれらの物理量である温度・高度・速度などのスケールに基づいて表現しているものの、これらの言葉は主観的でありその意味する境界がはっきりしない。こういった曖昧なものをファジィと呼び、それに対して数値で表現できるものをクリスプと呼ぶ。従来までの理論であるブール理論では 1 (真) か 0 (偽) で判断するしかなかったのに対して、ファジィ理論はその 1 と 0 の間を認めるという特徴がある。これを用いることでファジィ理論はファジィなデータとクリスプなデータを変換することを得意としており、現在では列車の運転や伝統工芸品の制作に用いている [13]。また、主観的な物事を取り扱えることが

らゲーム内に登場する敵の行動 AI やプレイヤー以外のキャラクター (Non Player Character : NPC) AI にもファジィ理論を用いている [14][15]。

3.2 本提案手法へのファジィ理論の適用

David の著書 [16] によると、一般的にファジィ理論のプロセスは大きく分けて 3 つの手順で構成する。第 1 に、クリスプなデータをファジィなデータにするためにファジィ化を行う。このとき、メンバーシップ関数という特性関数を用いて表現する。メンバーシップ関数については後で詳しく述べる。第 2 に、独自のファジィルールに基づいてメンバーシップ関数を組み合わせる。第 3 に、メンバーシップ関数を元にして非ファジィ化を行い、該当する数値のクリスプ出力を生成する。この手順を踏まえたうえで、本提案手法へのファジィ理論の適用について詳しく述べる。

本提案手法におけるファジィ化とは数値を感覚的な評価にマッピングする作業である。RPG の戦闘ダメージを例に挙げると、96 ダメージ以上は『強い』というようなマッピングをメンバーシップ関数を用いて行う。メンバーシップ関数とは値ごとの度合い (0~1) を定義する関数で、このメンバーシップ関数によって定義される度合いをメンバーシップ度と呼ぶ。また、本研究独自の定義としてメンバーシップ度が 0 から 1 へ漸進的に変化する X 座標の幅を漸進幅と呼ぶ。漸進幅 $g = 10$ のとき、先ほどの戦闘ダメージの例で用いた 96 ダメージ以上は『強い』というマッピングをブール理論におけるメンバーシップ関数で表現したものを図 3.1、ファジィ理論におけるメンバーシップ関数で表現したものを図 3.2 に示す。

図 3.1、図 3.2 からわかるように、ブール理論を用いた場合は例え 96 ダメージに近い 95 ダメージであっても『強い』に属さない。しかしファジィ理論を用いた場合は 0 と 1 の間を認めているため、96 ダメージ以上の『強い』度合いを 1 としたときに 95 ダメージの『強い』度合いを 0.9 として扱うことが可能である。このように、1 つの値に対する評価に対して 1 つのメンバーシップ関数を生成していくことでファジィ化を行う。

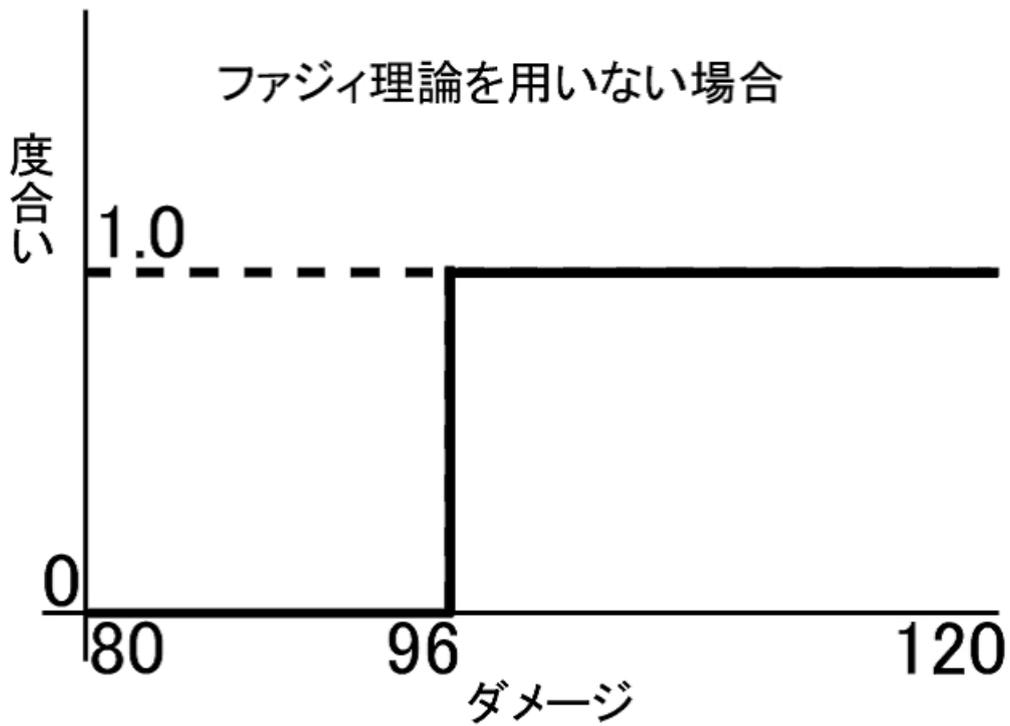


図 3.1: ブール理論におけるメンバーシップ関数

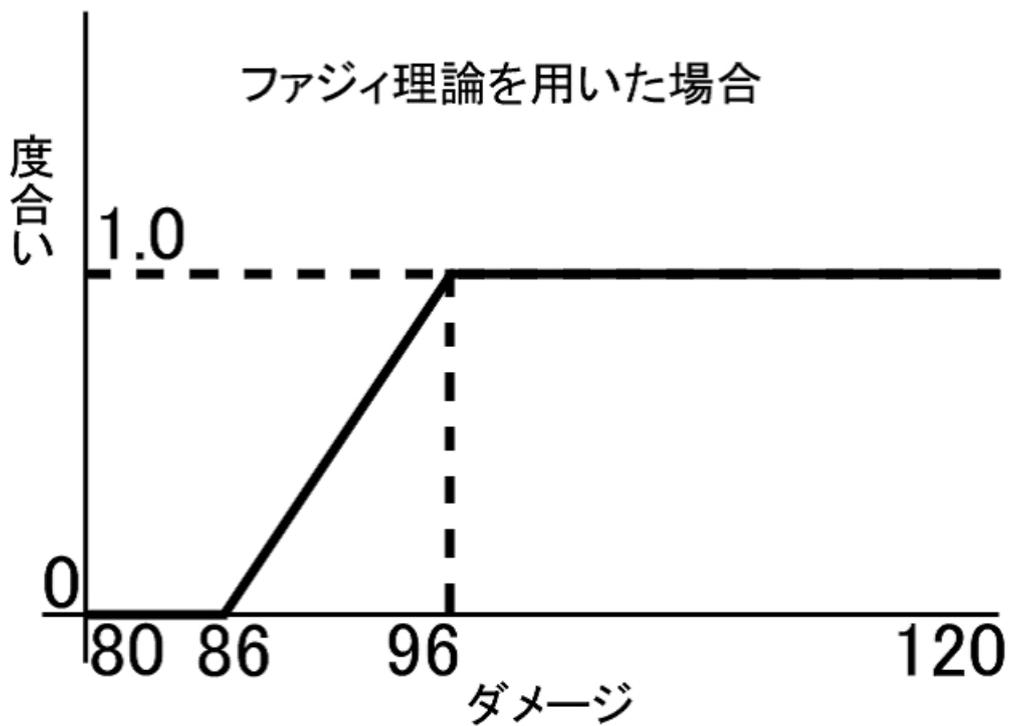


図 3.2: ファジィ理論におけるメンバーシップ関数

このように主観的な評価に基づいた値ごとの度合いをメンバーシップ関数としてアウトプットできることから、そのメンバーシップ関数の形状を組み合わせることで確率分布の形状に適用できないかという考えが本研究の提案手法である。次節よりその詳細について述べる。

3.3 本研究の提案手法

従来手法を用いてゲームデザイナーが確率分布の調整を行う場合、確率分布を直接調整する必要がある。そのため、理想とする確率分布に近づけるためにある程度の経験や知識が必要であった。そこで本研究ではファジィ理論を用いて確率を直接操作せずに確率分布を求める手法を提案する。これにより、確率分布に関する知識が無くとも確率分布の調整が可能ではないかと考えた。以下に提案手法を述べる。

3.4 ファジィ化とファジィルール

提案手法を説明するための具体的な例として、RPGの戦闘シーンにおけるゲームバランス調整を想定する。調整する値は敵に与える攻撃ダメージの確率分布とする。まず、テストプレイヤーは漸進幅 g を決定する。次に、与える攻撃ダメージの値域から任意の値を取り出し、その値が理想とする確率分布に近いどうかの評価を行う。評価は『最も出て欲しい (A)』、『やや出て欲しい (B)』、『出てほしい (C)』、『出て欲しくない (D)』の4段階で評価する。ここでは例として攻撃ダメージ96に対する評価を A 評価としたとする。X軸を攻撃ダメージ、Y軸をメンバーシップ度、攻撃ダメージ値域80~120、漸進幅 $g = 10$ としたときのメンバーシップ関数を図3.3に示す。

このように1つの値に対する評価に対して1つのメンバーシップ関数を生成する。2つ目以降に選択した値についても同様に評価を行なっていき、評価した数だけメンバーシップ関数を生成する。ダメージに対する評価を終えた後は同じ評

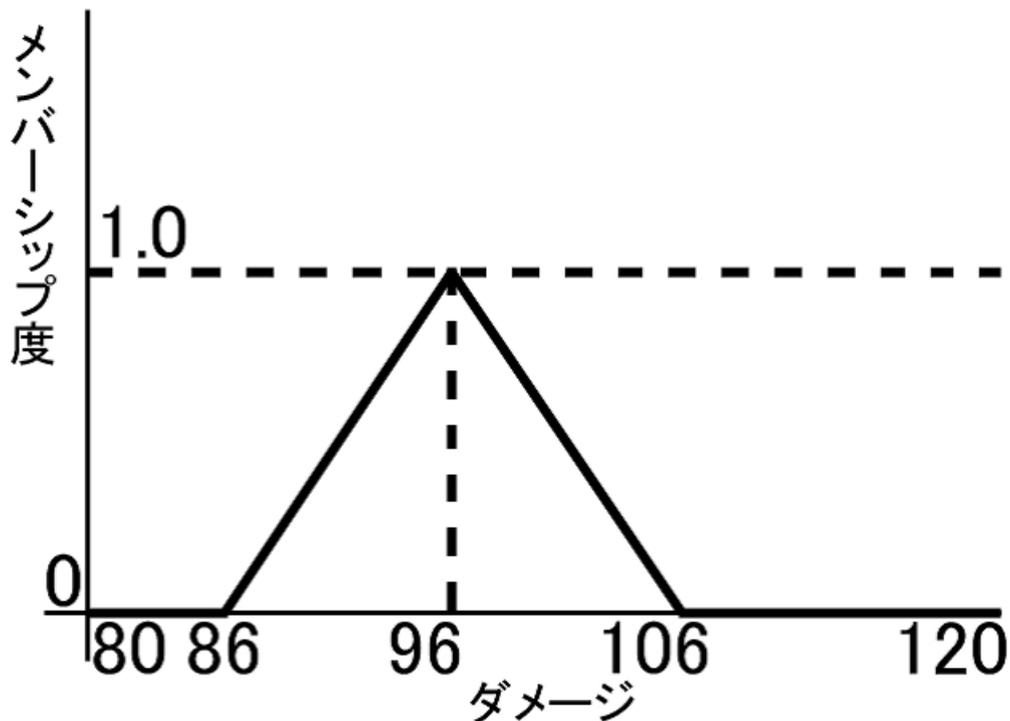


図 3.3: 攻撃ダメージ 96 のメンバーシップ関数

値に属するメンバーシップ関数の合成を行う。合成はメンバーシップ関数同士を単純に加算して行うものとする。漸進幅 $g = 10$ 、攻撃ダメージ 96 と 109 に対しての評価を A としたとき、評価 A におけるメンバーシップ関数合成前を図 3.4、合成後を図 3.5 に示す。

各評価毎にメンバーシップ関数の合成を行うことで、計 4 つのメンバーシップ関数を生成する。これら 4 つのメンバーシップ関数を用いて確率分布を求める。まずは評価内容を確率分布の形状に反映するため、各メンバーシップ関数のメンバーシップ度に補正をかける。評価 A のメンバーシップ度を基準として、評価 B のメンバーシップ度を 0.7 倍、評価 C のメンバーシップ度を 0.4 倍する。また、評価 D に関しては出て欲しくない値であるためメンバーシップ度を -1.0 倍する。攻撃ダメージ 110 に対しての評価を B 、攻撃ダメージ 90 と 103 に対しての評価を C 、攻撃ダメージ 80 に対しての評価を D としたときの補正後のメンバーシップ関数を図 3.6、図 3.7、図 3.8 に示す。

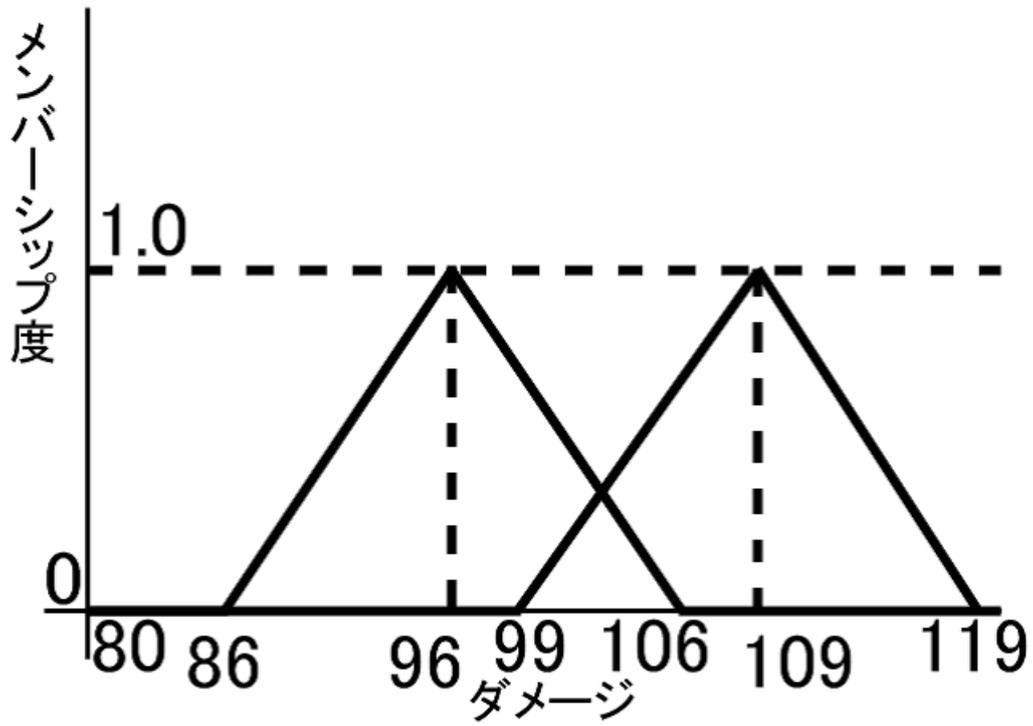


図 3.4: 評価 A におけるメンバーシップ関数合成前

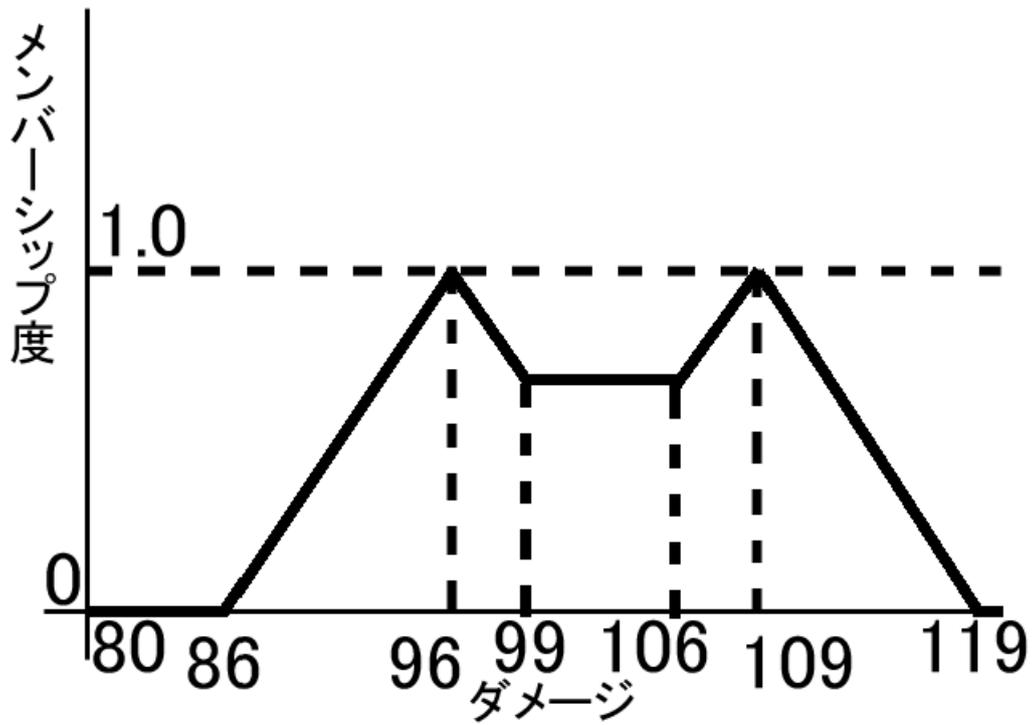


図 3.5: 評価 A におけるメンバーシップ関数合成後

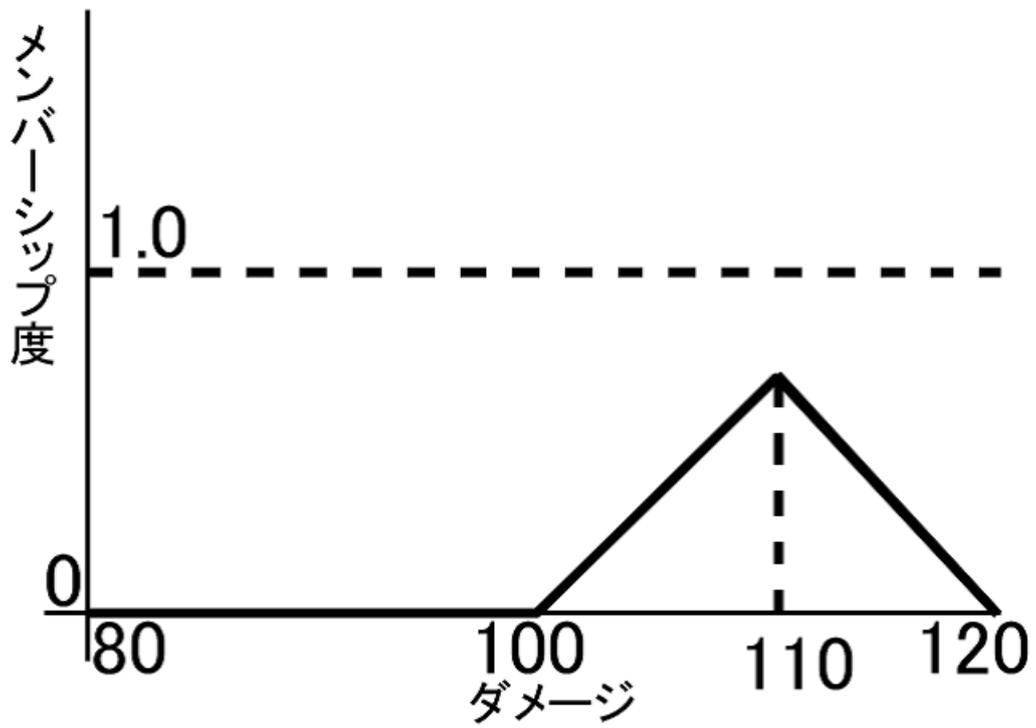


図 3.6: 補正後の評価 B におけるメンバーシップ関数

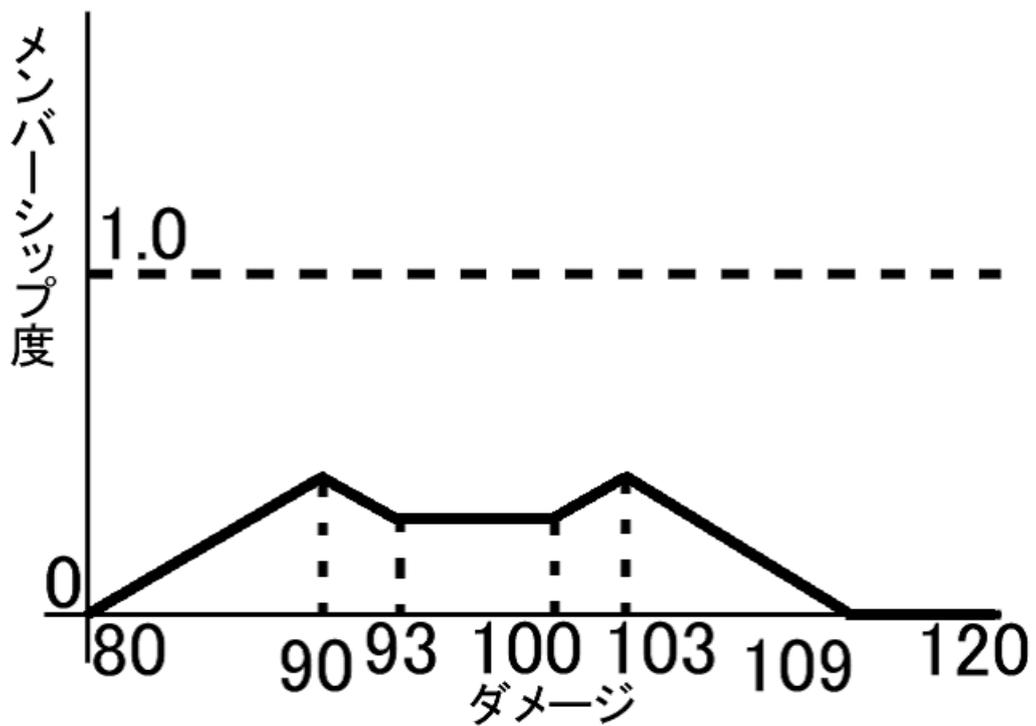


図 3.7: 補正後の評価 C におけるメンバーシップ関数

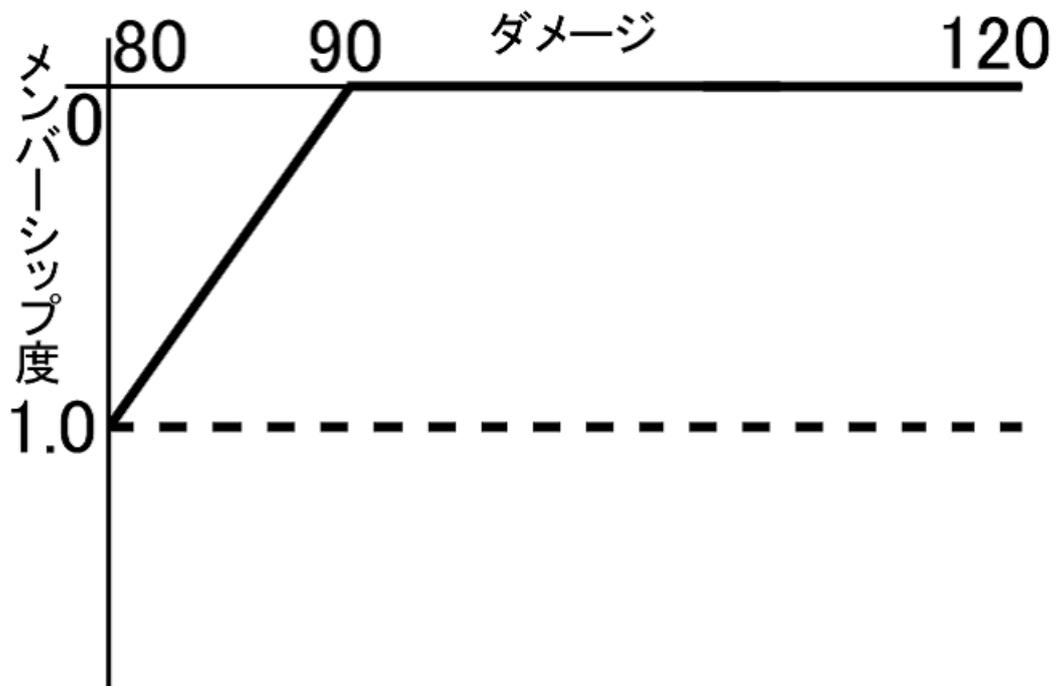


図 3.8: 補正後の評価 D におけるメンバーシップ関数

最後に、これら 4 つのメンバーシップ関数を合成することで確率分布を求める。このとき、メンバーシップ度が負の値を取った場合は 0 として扱う。図 3.5、図 3.6、図 3.7、図 3.8 を合成し、生成した確率分布を図 3.9 に示す。

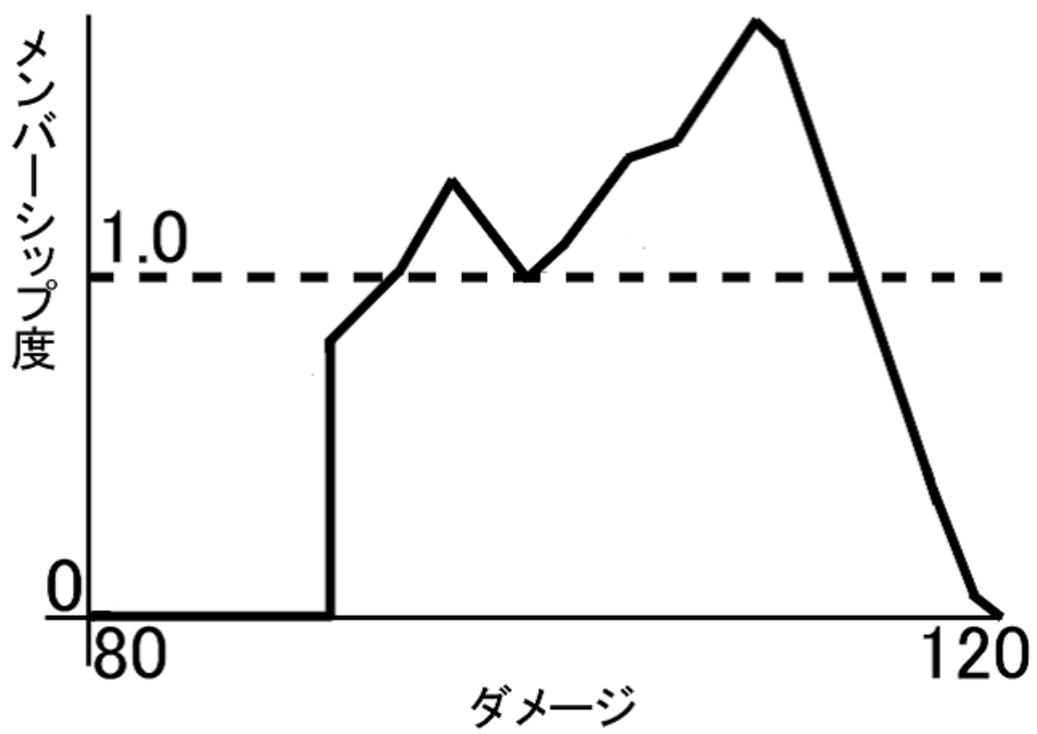


図 3.9: 本提案手法によって生成した確率分布

第 4 章

検証と考察

本章では提案手法の有用性を確かめるため検証実験を行い、その結果に対する考察を述べる。

4.1 実験概要

実験には提案手法を実装した「Microsoft Excel」による検証ツールを用いた。検証実験は RPG の戦闘シーンにおける攻撃ダメージの確率分布を想定し、攻撃ダメージ幅は 80~120、漸進幅 $g = 10$ 、確率補正は A 評価を基準として B 評価を 0.7 倍、C 評価を 0.4 倍、D 評価を -1.0 倍とした。詳しい手順については次節で述べる。

4.2 検証実験

本検証実験の目的は、本研究の提案手法を用いてテストプレイヤーが直感的に理想とする確率分布を生成できるかどうかを検証することを目的とする。そこで、以下のような検証実験を行った。

- 調査対象

有効回答数 7 名、男性 7 名、女性 0 名

- 調査時期

2008年8月

- 実験手順

テストプレイヤーは理想とする確率分布を思い浮かべ、簡単な文章にまとめる。次に、先ほど理想とした確率分布を念頭に置きつつツールを用いて確率分布を生成する。その後、生成した確率分布を用いて50個の攻撃ダメージを算出し、その結果がテストプレイヤーが想定した確率分布に近いかどうかを別のテストプレイヤーが評価した。評価は『理想通りの分布である』『理想に近い分布である』『理想に近い傾向の分布である』『理想から外れた分布である』の4段階評価による回答とした。これを観察者同士で行い、その結果を集計した。

4.3 実験結果

- 『理想通りの分布である』 …… 3名
- 『理想に近い分布である』 …… 3名
- 『理想に近い傾向の分布である』 …… 1名
- 『理想から外れた分布である』 …… 0名

4.4 考察

各テストプレイヤーが理想とした分布は、『特定の値付近が出やすい確率分布』だけでなく、『80付近と120付近が出やすい確率分布』や『まんべんなく出る確率分布』など多岐に渡っていたが、実験結果を見ると概ねテストプレイヤーの理想通りの確率分布が生成できたといえる。ただ1人『理想に近い傾向の分布である』という評価が出てしまったが、これについて評価の理由を聞いたところ「特定の

値付近が出やすい確率分布だったにも関わらずその範囲外の数値が何度か出たため」と述べている。実際の確率を確認したところミスはなかったため、これは試行回数が50回と少なかったことによる乱数の偏りであると判断した。

また、本実験の問題点として4段階評価の設定がある。これは、『理想に近い分布である』という評価と『理想に近い傾向の分布である』という評価の間に明確な差がなく、日本語として曖昧であったというものである。しかしこれに関しては実験の前に『理想通りの分布である』、『理想に近い分布である』、『理想に近い傾向の分布である』、『理想から外れた分布である』の順に高い評価であり、『理想から外れた分布である』以外は肯定的な評価であることを説明していたため、実験結果への影響は無かったと言える。

第 5 章

まとめ

本研究では、ゲームバランス調整作業の中でも確率分布を調整するものに着目し、ファジィ理論を用いた手法で直感的に確率分布を生成する方法を提示した。検証実験の結果、直接確率分布を調整する従来手法とは違い、主観による回答を繰り返すだけで理想とする確率分布を得ることが可能であることがわかった。今後の展望としては、検証実験では固定していた漸進幅や確率補正を自動的に変化することで、より少ない評価数で理想とする確率分布を得ることができると考える。

謝辞

本論文を締めくくるにあたり、終始適切にご指導頂きました渡辺大地講師をはじめ、日頃から研究のサポートをして頂いたゲームサイエンスプロジェクトの皆様に心より御礼申し上げます。

参考文献

- [1] “ 2008CESA ゲーム白書 2008CESA Games White Paper ”, 社団法人コンピュータエンターテインメント協会, 2008.
- [2] Chris Crawford, Shino OJ 訳, “ クロフォードのゲームデザイン論 - コンピュータゲームは芸術たりうるか - ”,
<<http://www2.airnet.ne.jp/ojima/acgd/Coverpagej.html>>.
- [3] 大野満秀, “ 日本におけるゲームソフト開発プロセスと海外共同開発・製作の状況 ”, 同志社大学 ワールドワイドビジネス研究センター, 2002.
- [4] 前田圭士, “ ゲームデザイナーの仕事 ”, ソフトバンククリエイティブ株式会社, 2008.
- [5] 桜井政博, “ 桜井政博のゲームについて思うこと Think about the Video Games ”, 株式会社エンターブレイン, 2005.
- [6] 井上明人, “ テレビゲームプレイヤーの誕生 ”, 慶應義塾大学大学院 政策・メディア研究科修士論文, 2005.
- [7] ドラゴンクエスト公式サイト 天空の城下町,
<<http://www.square-enix.co.jp/dragonquest/>>.
- [8] カ2ロマリア, “ ドラゴンクエスト1 情報編 解析データ ”,
<<http://www.geocities.jp/hoppygeo/DQ1/DQ1infomation.html>>.

- [9] マジスティックファンタジアン, “ テイルズオブシンフォニア解析 ”,
<<http://tales.albion9.com/tos/analyze.html>>.
- [10] 福田明, “ 理工系のための応用確率論 [基礎編] ”, 森北出版株式会社, 2003.
- [11] 馬場敬之, 久池井茂, “ スバラシク実力がつくと評判の確率統計キャンパス・ゼミ ”, マセマ出版社, 2003.
- [12] 月本洋, 松本一教, “ やさしい確率・情報・データマイニング ”, 森北出版株式会社, 2004.
- [13] 廣田薫, “ だからファジィが面白い ”, 裳華房, 1993.
- [14] Mark DeLoura, 川西 裕幸, 狩野 智英, “ GAME PROGRAMMING GEMS ”,
株式会社ボーンデジタル, 2001.
- [15] Mark DeLoura, 川西 裕幸, 狩野 智英, “ GAME PROGRAMMING GEMS2 ”,
株式会社ボーンデジタル, 2002.
- [16] David M.Bourg, Glenn Seemann, “ ゲーム開発者のための AI 入門 ”, 株式会社オライリージャパン, 2005