入力 3D モデルの近似形状を持つ 3D 歯車集合体モデルの生成

下中進(学生会員) **宮田一乘**(正会員)

北陸先端科学技術大学院大学・知識科学研究科

Generation of a 3D gear aggregate model that approximates the shape of a specified 3D model

Susumu Shimonaka (Student Member) Kazunori Miyata (Member)

SCHOOL OF KNOWLEDGE SCIENCE, Japan Advanced Institute of Science and Technology

{ s1050023, miyata } @ jaist.ac.jp

アブストラクト

この論文では,指定した3Dモデルの近似形状を持つ3D歯車集合体モデルを生成する手法を提案する. 従来手法で生成された集合体とは違い,本研究で生成する集合体は互いにかみ合う,かつ大きさの異な る歯車で構成される.加えて,構成する歯車1つを回転させると構成する歯車すべてが回転する状態で ある.

本論文では、インボリュート標準平歯車を用いる.本手法の概要を以下に示す.はじめに、指定した モデルを複数の平面状の断面に分割する.次に、各断面に歯車を、本論文が定義するスコアを用いて、 1つずつ配置していく.そのスコアは、歯車のかみ合いや歯車のサイズを考慮して決定される.

Abstract

This paper proposes a method for generating a 3D gear aggregate model that approximates the shape of a specified 3D model. Unlike a 3D gear aggregate model generated by traditional methods, the 3D gear aggregate model generated by the proposed method consists of many engaged 3D gears of different size. Moreover, it is possible to rotate all engaged gears if one of them is rotated.

In this paper, involute spur gears are used. The outline of the method is as follows. First, a specified 3D model is divided into multiple planar cross-sections. Then, each cross-section is filled with gears, by placing the gears one by one according to a defined score. The score of a gear is calculated according to its engagement and size. キーワード: 歯車, 3D モデル生成, コンピュータグラフィクス, ノンフォトモデリング

Keywords: Non-Realistic Expressive Modeling,, Gear, Computer Graphics, 3D model Generation

1. 背景

歯車は機械要素の一種であるが、五円硬貨やイタリアの国章 などのように「工業」「労働」「労働者」のシンボルとして利用 されたり、映画"モダン・タイムス"では映画のセットとして使 用されたりしている.アーティスト Mike Libby[2]は昆虫の標本 に歯車やばねといった機械要素を加えて、insect-like robot や robot-like insect を表現している.これらの例のように、歯車は 機械要素を表す有用なシンボルとして広く用いられている.

本研究では、入力 3D モデルを歯車の集合体に変換することで、歯車が持つ機械的な表現を目指す.また、ユーザが歯車に

対する知識を有していなくても、労力をかけずに複雑な歯車集 合体を自動生成できるようにする.

2. 関連研究

指定した3Dモデルを別な3Dモデルへと変換する研究として, Non-Photorealistic Renderingの研究分野のひとつである形状モデ リング手法の他,3Dモデルを物理的に制作できる形に生成する ことを目指す fabrication 分野,対象となる3Dモデル通りに構成 要素を堆積させることを目指す photorealistic modeling 分野など を含めると多数提案されている.例えば、アルチンボルトの肖 像画のように複数の3Dモデルの構成体に変換する手法[3]、実 際に配置可能な缶の積み上げ集合体に変換する手法[4],複数の 折り紙から構成されるユニット折り紙に変換する手法[5],入力 モデルの形状を保持した Polyomino Puzzle を生成する手法[6], 複数の任意の3Dモデルを指定した3Dモデルの形状に合うよう に堆積させる手法[7],などが先行研究として挙げられる.

以上で示した手法では、歯車のかみ合わせのような構成要素 間の特殊な連結を考慮していない.本手法では、歯車のかみ合 わせを考慮しながら1つずつ歯車を配置する.

3. 目的

本研究では、指定した 3D モデルの近似形状を持つ歯車の集 合体を自動で生成する手法[1]を提案する.加えて、生成した集 合体は、歯車の特徴を視覚的に表現するために、以下の状態に する.

1. 構成する歯車間にかみ合いがある.

2.構成する歯車を1つ回転させるとすべての歯車が回転する.
3.集合体を大きさの異なる歯車で構成する.

4. 歯車集合体の生成について

4.1 歯車の種類・規格

本手法では、歯車集合体を構成する歯車として、日本工業規 格[8]で規格化された工業で一般的に使用されている「インボリ ュート標準平歯車」を用いる.平歯車は、歯車軸に平行な他の 歯車にかみ合い動力を伝達する歯車である.図1にインボリュ ート標準平歯車を示す.

本手法では、使用に耐えうる歯車の強度や工業的に使用され る寸法など, 製造に必要な物理値は考慮せず, 歯数や歯の大き さ、かみ合わせの位置の規格のみを考慮する. したがって、歯 車のピッチ円直径 d, モジュール m, 歯数 T の 3 つのパラメ ータで、歯車の形状を定義する. ピッチ円直径は歯車をかみ合 あわせるための条件に用いられるピッチ円の直径を、モジュー ルは歯の大きさを、歯数は歯車の歯の数をそれぞれ示しており、 これらの関係は d=mT で表すことができる. ここで, 歯数は 自然数でなければならない. また規格より, 図 2に示す歯車の 歯先円直径(最外径)CはC=d+2mで定義され、本研究では歯 車の干渉判定に使用する.同じく規格より、歯の根元を通る歯 底円の直径 d_f は d_f =d-2.5mで定義される.ここで、ピッチ円、 歯先円, 歯底円の中心位置は同じである. 2 つの歯車がかみ合 うには互いのピッチ円が一点で接触しており、かつ互いにモジ ュールが同じであるという条件を満たす必要がある.本手法で は、モジュールの値はユーザが初めに指定した1つの値(正の 数)をすべての歯車で使用する.これにより、歯車がかみ合う ために必要な条件は「互いのピッチ円が接触する」だけになる.

歯数が小さい歯車を加工する際に,加工工具によって歯の根 元部分が削られ、歯の形が変形する切り下げ[9]と呼ばれる不都 合が発生し、かみ合うことのできる歯面が減りかみ合いが悪く なる、強度が減るなど、歯車にとって好ましくないことが生じ る.そのため、切り下げを防ぐ目的で転位と呼ばれる歯形の調 整が行われている.転位を考慮すると解くべきパラメータが増 えるため、問題の単純化のために本研究では転位は扱わず、歯 車の最少歯数を切り下げが発生しない値と同じ 17 枚[9]に設定 する.また,歯幅(歯車の厚さ)はすべての歯車で同じ厚みとし, ユーザが0以外の正の値を設定する.





4.2 歯車集合体の生成手法

歯車の集合体の生成には、以下の2つの手順を踏む.図3に 手順の概略図を示す.

 図 3 (a)に示す入力 3D モデルに対して、複数の平行した YZ 平面レイヤを図 3 (b)に示すような等間隔に位置に設置することで、図 3 (c)のような各レイヤ位置における断面を得る.
得られた各断面内に、図 3 (a)-(f)のように複数の歯車を歯車

の平面部と断面が平行になる姿勢で、断面内に歯車が配置できなくなるまで配置していく.その結果、図3(g)に示すような歯 車集合体が生成される.

ここで、目的に挙げた「1つの歯車を回転させるとすべての 歯車が回転する」という条件がある. このため, 配置する歯車 は、同レイヤ内での他の歯車への回転の伝達だけでなく、他レ イヤへも回転を伝達する必要がある.この条件を満たすため, 本手法では歯車単体を1個ずつではなく、図4に示すような歯 車セットを1セットずつ配置する. 歯車セットは、1つ以上の 歯車が同じ歯車軸に固定しており、構成するいずれかの歯車1 つが回転すると歯車軸に固定された他の歯車も回転する. した がって、歯車セットを構成する歯車を異なるレイヤの断面に置 くことにより、他レイヤの歯車セットへ回転を伝達することが できる. 歯車セットが他の歯車セットに回転を伝達できるよう に、歯車セットを構成するいずれかの歯車1つを、図5で示す ようにすでに配置された歯車1つとかみ合わせながら、1つず つ配置していく.この配置方法により,配置した歯車セットは 必ずかみ合いによって回転するようになり、「1つの歯車の回 転ですべての歯車が回転する」状態で歯車を配置することがで きる.

芸術科学会論文誌 Vol. 12, No. 2, pp. 92-103



4.2.1 断面を得る処理

入力した 3D モデルを平行なレイヤに分割し、断面を得る方 法について説明する.まず前処理として、入力モデルを XYZ 空間に配置し、入力モデルのサイズの正規化、平行移動を以下 の手順で行う.ここで、入力モデルは 3D サーフェスモデルと する.

1) 入力モデルの Z 軸方向および, Y 軸方向の全長のうち, 長 い方が正規化サイズ F になるように入力モデルをスケーリン グする. F は 0 以外の任意の値を設定するものとし,本研究で は F=5.0 としている.

2)入力モデルを原点方向へ平行移動させるために, XYZ 空間 内に入力モデルを内包する境界ボックスを生成する.境界ボッ クスは以下の定義を持つ.

・YZ 平面とZX 平面とからなる直方体である.

・入力モデルを内包する.

・各面は内包する入力モデルの一部と接する.

生成した境界ボックスの最少の X,Y,Z 座標値をもつ頂点に
XYZ 空間の原点を平行移動させる.

以上の手順により,図 6に示すような直方体内に入力モデル を内包できる.



図 6 入力モデルのサイズの正規化後,平行移動後の図

次に図 7(a)に示すように、入力モデルに N 枚の YZ 平面レイ ヤを設置する. 設置するレイヤ枚数 N は自然数とし、ユーザが 設定する. ここで、原点から i 番目の位置のレイヤは図 7(a)の ようにレイヤ番号 $X_r = (i-1)$ を持ち、各レイヤは X 座標が (X_r +1) * L_D の位置に設置される. L_D は、図 7(c)のように入力モ デルを等間隔に分割するために、入力モデルの X 軸方向の全長 を(N+1)で割った値とする.

そして,各レイヤの位置で入力モデルの断面を 2D 平面テク スチャと見立ててテクセル化を行うことで図 7 (b)に示すよう な断面の形状を得る.



図 7 XYZ 空間のレイヤ位置とテクセル化の概要図

テクセル化処理の手順について図 8を用いて以下に説明する. 1) 各レイヤの X 座標の YZ 平面上で、一辺が Y 軸に平行、対 角の YZ 座標値が (0,0) - (F,F)の正方形平面を XYZ 空間内に生 成する. ここで、正方形平面は図 8 (a)に示すように 1 辺が L₆ の正方形セルで構成される正方形グリッドを持つ. L₆はテクセ ル数 B で F を割った値とする. B の値は自然数であり、ユーザ が設定する.

2) 各セルの内部に入力モデルの一部を含むかの判定を行い,図 8(b)のように入力モデルの一部を内包する各セルにテクセル値 として true を,他のセルには false を持たせる.

以上の処理を行うと、2 値のテクセル値を持つセルの集合が 得られる.そして、本手法では、true のテクセル値をもつ各セ ル内の XYZ 空間位置を入力モデルの断面とする.また、true を持つセルと false を持つセルの境界上の XYZ 位置は断面外と する.

芸術科学会論文誌 Vol. 12, No. 2, pp. 92-103



図8 各レイヤ位置での入力モデルのテクセル化の処理

4.2.2 断面に歯車を配置する処理

次に、4.2.1 項の処理で得た断面に対して歯車を配置する処理 について説明する.

まず, XYZ 空間に, 4.2.1 項のテクセル化の処理時に作成した図 8に示す正方形グリッド構造を持つ正方形平面を, テクセル化の処理時と同じサイズで同じ XYZ 位置に生成する.

次に,図 9に示す以下の手順に従い,歯車セットを配置する. 1) 各セルの中心位置ごとに歯車セットスコア S₁を計算する. 歯車セットスコア S₁の詳細については 4.2.2.1 で後述するが,計 算位置に歯車セットが置けるときのみ, S₁は正の値になる.

2) すべてのセルの中心位置で S_1 が 0 だった場合, 歯車が1つ も置けないとして歯車配置を終了し, 歯車集合体は生成されな い. この場合, B の値を増やす, N の値を増やすなど, ユーザ が入力パラメータを見直すことで対応できる.

3) 全セル内で最大 S_1 を持つセルの中心位置に、その S_1 を算出 した時のパラメータを持つ歯車セットを 1 つ配置する. 図 10 にこの処理の概要を示す.

4) 各セルの中心位置ごとに歯車セットスコア S_2 を計算する. S_2 の計算では、 S_1 と違い以下の点を計算時に考慮している.詳細については 4.2.2.2 で後述する.

・配置済みのいずれかの歯車とかみ合う歯車を持つ歯車セット が計算位置に置けるとき、かつ、配置済みの歯車および、歯 車軸と物理的干渉をしない歯車セットが計算位置に置けると きのみ、S₂は正の数になる.

5) すべてのセルの中心位置で S_2 が 0 だった場合, 歯車はこれ 以上置けないものとして歯車配置を終了する.

6) 全セル内で最大 S₂を持つセルの中心位置に、その S₂を算出した時のパラメータを持つ歯車セットを1つ配置する.
7)4)に戻る



図 10 1つ目の歯車セットを配置する手順の概要

なお、本手法で求める S_1 , S_2 は局所解であるため、歯車を配置 していない領域が発生する場合が生じる.この問題への対策は 4.2.3項で説明する.

4.2.2.1 1つ目の歯車セットのS1計算

1つ目に配置する歯車セットに対する S_1 の計算方法を説明する. 計算中のセルの中心位置のレイヤ番号を X_c とした場合,そのセルの位置を $P_c=(X_c, Y_c, Z_c)$ とする.

はじめに、直径 $d_g \ c P_c$ を通り X 軸に平行な歯車軸が断面内 に配置できるレイヤ範囲[X_u, X_d]を算出する. この処理の概略を 図 11 に示す. ここで、歯車軸の直径 d_g の大きさはユーザが 設定する. まず、 P_c で軸が断面内に配置できるか、図 11 (a)に 示すように P_c を中心とする直径 d_g の YZ 平面上の円板の一部 を内包する各セルのテクセル値で判断する. 接する各セルのテ クセル値がすべて mue の場合、軸は断面内に配置できる. 配置 できない場合、 P_c には歯車セットを配置できないため、 P_c の S₁ を 0 として S₁の計算を終える. 軸を配置できる場合、次に図 11 (b)(c)に示すように X_w, X_dを算出する. 歯車軸は X 軸と平行に配 置するので P_c と同じ Y.Z 座標を通るとする.



図 11 レイヤ範囲[Xu, Xd]を算出する処理の概略

 X_u を算出するために、 X_c から0までの各レイヤで軸が断面内 に配置できるか順に判定する. X_c から連続して軸が断面内に配 置できる最後のレイヤを X_u とする. X_d を算出するためには、 X_c からN-1までの各レイヤで軸が断面内に配置できるか順に判 定する. X_c から連続して軸が断面内に配置できる最後のレイヤ を X_d とする.

次に、歯車セットを構成する歯車の歯車スコア F_i を定義する. 添え字iはレイヤ番号を表す.そして、以下の式(1)のように S_1 は $[X_u, X_d]$ 範囲内の F_i の総和とする:

$$S_1 = \sum_{i=X_u}^{X_d} F_i$$

...(1)

式(1)に示す通り、 X_u から X_d まで順に F_i を計算する. 計算中 のレイヤ番号を X_s , $P_{xx}=(X_s, Y_c, Z_c)$ とし、 X_s での F_i を F_{Xs} とす る. P_{xs} に配置する歯車の歯数を T_{Xs} , ピッチ円直径を d_{Xs} , 歯 先円直径を C_{Xs} とする. これらを用いて以下の式を定義し、計 算方法を図 12を用いて説明する:

| $d_{Xs}=mT_{Xs}$ | (2) |
|-------------------------|-----|
| $C_{Xs} = d_{Xs} + 2 m$ | (3) |
| $F_{Xs} = max(C_{Xs})$ | (4 |

図 12 (a)のように式(2)における歯数 T_{Xs}の初期値を最少の歯

数の17とし、図12(a)から(b)のように1ずつ増加させていきな がら、式(3)の C_{xs} を持つ歯車が断面内に存在できるか直径 C_{xs} の円板の一部を内包する各セルのテクセル値を参照し判定する. そして式(4)のように断面内に存在できる最大の C_{xs} を F_{xs} とする. T_{xs} の初期値で式(3)の C_{xs} を持つ歯車が断面内に存在できない場合, P_{xx} には歯車が配置できないとして $F_{xs}=0$ とする.

以上の計算で、セルの中心位置 P_c における S_1 を求めることができる.



図 12 F_{Xs}の計算方法

4.2.2.2 2つ目以降の歯車セットのスコア S2計算

2 つ目以降の歯車セットに対する S₂の計算は基本的には 4.2.2.1 と同じ処理を行うが、配置済の歯車との関係から以下に 示す2つの処理を加える.

1) 計算中のレイヤ番号 X_c のセルの中心位置 $P_c \varepsilon(X_c, Y_c, Z_c)$ と する. P_c に配置済の歯車1つとかみ合える歯数 T_k の歯車が配置 できる場合のみ S_2 を計算する.ここで,配置済みのある歯車の ピッチ円と1点で接触できるピッチ円をもつ歯車が断面内に配 置できる場合,その歯車がかみ合える歯車となる.そして S_2 の計算では, F_{xc} は以下の3つの式を用いる:

| $d_{Xc}=mT_k$ | (5) |
|-------------------------|-----|
| $C_{Xc} = d_{Xc} + 2 m$ | (6) |
| $F_{Xc} = C_{Xc}$ | (7) |

 C_{Xc} を持つ歯車が断面内に配置できない場合, X_c でかみ合える 歯車を配置できないため, S_2 は0とする.

2)4.2.2.1における S_1 計算時のセルのテクセル値を参照して歯車や歯車軸を断面内に配置できるかの判定に、 S_2 の計算では配置済みの歯車の歯先円や歯車軸と接触するかの判定を追加する.

そして,接触する場合,断面内に配置できないと判定する.例 外として,1)の処理の接触判定では,かみ合う歯車同士の歯先 円は常に接触しているので,X。のレイヤでかみ合う配置済みの 歯車の歯先円との接触は判定しない.

以上の処理で、セルの中心位置 P_c における S_2 を求めることができる.

4.2.2.3 歯車セットスコアに関して

式(4), (7)に示すように、歯車の最外径である歯先円の直径が 大きいほど Fi が大きくなる.したがって、その総和で求める S_1 , S_2 は歯車セットが大きい歯車で構成されているほど値が大 きくなる.本手法では、 S_1 , S_2 の大きい歯車セットから順に配置 するため、大きい歯車で構成されている歯車セットが優先的に 配置される.したがって、断面が大きい部分には大きい歯車が 配置され、断面が小さい部分,または配置された歯車や歯車軸 の隙間に小さい歯車が配置される.これにより,生成される歯 車集合体を異なるサイズの歯車で構成する目的を満たせる.

4.2.3 歯車が配置されていない領域の発生の抑制

図 13に示す例を用いて、歯車の未配置領域が発生する場合に ついて説明する.

図 13 (a)に示すように入力モデルに対して X_r=0, X_r=1 のレイ ヤを設置した場合,図 13 (b)のような断面を得る.得られた断 面に逐次歯車セットを配置する場合,図 13 (c)のような最大 S₁ を持つ歯車セットを配置し、新たな歯車セットを既配置の歯車 セットとかみ合うように配置していく.そして最終的に、図 13 (d)のような歯車集合体を生成するが、生成した歯車集合体は X_r=1 の断面内の領域には歯車が配置されない.すなわち、歯車 の未配置領域が発生しているため、入力モデルの近似形状を十 分には得られていないことになる.

この原因について,図 14を用いて説明する.図 13(b)の断面 には、図 14(b)のような異なるレイヤ間の両領域を連結する歯 車セットが配置できる.その領域を構成するセルを本研究では 「跨ぎセル」と呼ぶ.図 13(b)の断面内の「跨ぎセル」を図 14 (a)に示す.そして、歯車が未配置の領域へ連結できる跨ぎセル を「未跨ぎセル」と呼ぶ.図 13(b)の断面内に、歯車セットを 配置した後の、歯車が未配置の領域を図 14(c)に、「未跨ぎセ ル」を図 14(d)にそれぞれ示す.

図 13の例では、この「未跨ぎセル」を歯車が塞ぐことで、 X_r=1 へ連結できる歯車セットが配置できなくなったため、歯車の未配置領域が発生した.



図 13 歯車集合体生成後の歯車が置かれない領域の発生の例



図 14 「歯車集合体生成後の歯車の配置されていない領域」 が発生する原因

歯車の未配置領域が発生する問題への対策として、4.2.2.1 お よび、4.2.2.2 で述べた S_1,S_2 の計算内に「未跨ぎセル」を塞ぐよ うな歯車を配置しない処理を加える.

上記の処理の準備として、テクセル化後に、「断面を構成する 領域」と「跨ぎセル」を抽出する.そして、S₁,S₂の計算内で、 「断面を構成する領域」と「跨ぎセル」の情報をもとに、計算 している歯車セットが塞ぐセルが「未跨ぎセル」かどうか判定 する.それぞれの処理について、4.2.3.1と4.2.3.2で説明する.

4.2.3.1 テクセル化後に追加する処理

入力モデル断面のテクセル化後に以下の2つの処理を行う.

「断面を構成する領域」を抽出する処理

断面に対して以下の処理を行う.

 テクセル値が true であるセルを各領域に分ける.領域の定義として、YZ 平面上で互いに4連結しているセルの集合を1 つの領域と定義する.

2) 各領域に ID を振り, 領域内の各セルにその ID を持たせる. この ID を本論文では「断面 ID」と呼ぶ

「跨ぎセル」を抽出する処理

異なるレイヤ間の領域を連結する歯車セットを構成する歯車 を配置できる領域は以下の条件を満たす.

・構成する各歯車は同じ歯車軸の方向上に中心を持ち,歯車を 配置するために、その位置を中心とした最少歯数の歯車の歯先 円の大きさ以上の領域が確保されている.

そして、その領域を構成するセルが跨ぎセルである.

跨ぎセルの抽出は以下の処理を行う.手順の説明に図 15を用 いる.以下の処理は画像処理のオープニング処理[10]を応用し ている.

1) X 方向に断面が連続するセルを,以下の式(8)を用いて抽出 する. そして抽出した各セルに「フラグ A」を持たせる.

X 方向に断面が連続するセルの判定について、レイヤ番号が

U, Y 座標が V, Z 座標が W の XYZ 空間座標 (U,V, W)におい て,式(8)が TRUE であれば,(U,V,W)を内包するセルは X 方向 に断面が連続するセルである.

(G(U-1,V,W) ∨ G(U+1,V,W)) ∧ G(U,V,W) ...(8) ここで, G(I, J, K)関数は座標(I, J, K)を内包するセルのテクセル

値(TRUE or FALSE)を返す.

フラグA」を持つ各セルに対して、以下の手順を行う.
図 15(a)のように、セルの中心位置に、その位置を中心とした最少歯数の歯車の歯先円直径を持つ円板をYZ平面に置く.
図 15(b)のように、置いた円板に接触する各セルが「フラグA」を持つセルのみ場合、円板の中心位置のセルに「フラグB」を持たせる.

3) 「フラグB」を持つ各セルに対して、以下の手順を行う.
① 図 15(c)のように、そのセルの中心位置に、その位置を中心とした最少歯数の歯車の歯先円直径を持つ円板を YZ 平面に置く.

 ② 図 15(d)のように、置いた円板に接触する各セルに「フラグ C」を持たせる.

4) 2) 3)の処理により、X方向に断面が連続するセルの集合 内から、セルの中心位置に最少歯数の歯車の歯先円の大きさの 領域を持たないセルが取り除かれる.そして、フラグCを持つ セルが跨ぎセルである.

4.2.3.2 S1,S2の計算内に追加する処理

前述した S₁,S₂の計算における歯車軸の範囲[X_u, X_d]の計算後に、以下の手順を持つ処理を追加する.

1) 配置済みの歯車が置かれている領域の断面 ID を「歯車配置 済み領域リスト」に登録する.そして、[X_u, X_d]の範囲内の X_r で、YZ 座標が(Y_c, Z_o)の位置を内包する各セルが持つ断面 ID を「軸リスト」に記録する.ここで、「軸リスト」内の断面 ID を持つ領域は、計算しているSの歯車セットで置かれる領域を 表す.

2) [X_u, X_d]の範囲内の X_rで, YZ 座標が(Y_c, Z_o)の各位置で以下 の処理を行う. (X_r, Y_c, Z_o)の位置に最少歯車の歯先円直径の円板 を中心に置いた場合,円板に接触する各セルが、後述する式(9) を用いて、未跨ぎセルかどうか判定する.未跨ぎセルだった場 合、(X_r, Y_c, Z_o)の位置を内包するセルが持つ断面 ID を「軸リス ト」から取り除く. 2)の処理を、断面 ID が「軸リスト」から取 り除かれなくなるまで繰り返す.

式(9)について,式(9)が TRUE なら(U,V,W)を内包するセルは 「未跨ぎセル」である.

 $\begin{array}{l} (M(U-1,V\!,\!W) \land H(U\!-\!1,V\!,\!W)) \lor (M(U\!+\!1,V\!,\!W) \land H(U\!+\!1,V\!,\!W)) \land \\ M(U\!,V\!,\!W) \ldots (9) \end{array}$

M(I, J, K)関数は, 座標(I, J, K)を内包するセルが跨ぎセルなら TRUE を返す. H(I, J, K)関数は, 座標(I, J, K)を内包するセルが 持つ断面 ID が「歯車配置済み領域リスト」, または「軸リスト」 内にないなら TRUE を返す.

そして, F_i計算時の「歯車が断面内に置けるか」の判定に以下の判定を加える.

・直径 Cxsの円板に接触するセルが未跨ぎセルの場合、Cxsを持

つ歯車は断面内に配置できないと判定する.

以上の処理により、歯車の未配置領域の発生を抑制できる.



4.2.4 描画のためのメッシュの生成

算出した歯車セットをレンダリングするために,歯車セット を構成する歯車と歯車軸の形状メッシュを以下のように生成す る.

歯車の形状メッシュは,算出したパラメータ m, d, T, をも つ歯車の外形をなす各部位の寸法を歯車の規格[9][11]に当ては めて決定する.決定する部位を図 16に示す.



図 16 歯車の外形をなす各部位

歯車の外形と歯幅をもとに図 17のようなメッシュを生成する.

上記の処理により,任意のパラメータ m, d, T, 歯幅を持つ 歯車の形状メッシュを生成できる.

歯車軸の形状メッシュにはユーザが指定した歯車軸径 $d_g e$ 最外径として持つ N_γ 角柱メッシュを用いる.本研究では、 N_γ =16~20の値を使用している. N_γ 角柱の長さは生成対象の歯車 セットの全長とする.ここで、本研究では、歯車軸を目立たせ るため、 N_γ 角柱の長さを微小量伸ばしている.

以上の処理で、図 18 のような任意のパタメータ d_g、歯車セットの全長を持つ歯車軸の形状メッシュを生成することができる.



図 17 歯車の形状メッシュ 図 18 歯車軸の形状メッシュ

最後に、歯車および歯車軸の中心の XYZ 空間位置は S₁, S₂ の計算時のセルの中心位置とする.また、各歯車の角度に関しては、かみ合っている歯車同士の歯が干渉しない角度を用いる.

4.3 全数処理と簡易処理の比較実験

本手法の実行環境は CPU : Intel Core i7 CPU 930, メモリ: 6GB である.

ここで本手法では、最大の S_1 を計算する処理に対し、各セルの中心位置で S_1 を計算し、全セル内で一番大きい S_1 を用いる 全数処理と、以下に記す簡易処理の2つの方法を適用し、計算 コストを比較した。全数処理は、各セルの中心位置で S_1 を計算 するため、セルの総数分計算する必要があり、全数処理の総計 算コストは莫大になると予想できる。その計算コストを抑えつ つ、歯車の配置に必要な S_1 を計算するために以下の簡易処理を 行う。

1) ランダムに選んだ1つのセルの中心位置で S_1 を計算する 2) $S_1>0$ を持つ位置を内包するセルが N_{S1} 個になるまで1)を繰り 返す. N_{S1} はユーザが設定する.本手法では $N_{S1}=10$ を用いてい る.

3) N_{S1}個内で最大の S₁を用いる.

全数処理と簡易処理を用いた手法で同じパラメータを用いて 歯車集合体を生成した結果を図 19, 図 20に, その計算時間に ついて表 1 に示す. 生成されたモデルのレンダリングには Mitsuba Renderer[12]を用いている.表1に示した計算時間をみ ると,この2つの例では1つ目の歯車セットを計算するのに全 数処理では膨大な時間を要している.これに対して,簡易処理 では1つ目の歯車セットの計算時間は,全数処理の計算時間の 数万分の1である.

本手法は、 S_1 で配置するのは歯車集合体を構成する無数の歯 車セットの1つだけで、その他の歯車セットは S_2 に従って配置 されるため、モデルを構成する歯車の傾向は主に S_2 に従うと考 えられる. それは図 19, 図 20のどちらの手法の歯車集合体で も、入力モデルの大きな部分に当たる部分には大きな歯車を配 置できていることからもわかる. このことから、計算時間を大 幅に削減できる簡易処理を本手法では採用することとした.



図 19 全数処理と簡易処理で生成した歯車集合体の例1

| (a)入力モデル | |
|----------|---------|
| | |
| (b)全数処理 | (c)簡易処理 |

図 20 全数処理と簡易処理で生成した歯車集合体の例2

| | 図 19 | | 図 20 | |
|------------------------|--------------------------------|------|---------|--------|
| 処理の種類 | 全数 | 簡易 | 全数 | 簡易 |
| テクセル数 B | 600 | | 900 | |
| レイヤ数 N | 10 | | 15 | |
| モジュール m | F/600 | | F/900 | |
| 歯車軸の太さ d _g | (最小歯数の歯車の d _f)*4/5 | | | |
| 全歯車セット の計算時間 | 50,035s | 965s | 26,286s | 2,527s |
| 歯車セット1 つ目の 配置に要した時間 | 49,144s | 1s | 23,668s | 1s |

5. 結果および考察

5.1 結果

4章で説明した本手法で生成した歯車集合体を図 21-図 24に 示し,表 2にそれぞれの計算時間と入力パラメータについて示 す.入力パラメータの歯幅については、レイヤ間の歯車が接触 しないような値を用いている.

図 21,図 23-図 24に示すように生成した歯車集合体は、入 カモデルの近似形状を保持しており、異なったサイズの歯車で 構成されていることが視認できる.また、図 25の拡大図に示す ように生成モデルは歯車同士のかみ合いを表現できている.

本手法は、各レイヤで得られる入力モデルの断面形状を近似 するように歯車集合体を生成するので、レイヤの枚数 N が多い ほど、生成される歯車集合体の形状は入力モデルの形状に近づ く.N 以外の入力パラメータが同じ場合の生成モデル例を図 23, 図 24に示す.この例からもNが多くなるほど、歯車集合体が入 力モデルの形状により近づくことがわかる.

本研究は、各セルの中心位置ごとに歯車セットが置けるか計 算を行うため、テクセル数Bは十分に大きい必要がある.本研 究では、計算時間との兼ね合いから経験的にB=600~1200を用 いている.

ユーザが入力するモジュール m は、歯車の歯の大きさであ り、値が大きいほど歯が大きくなる.しかし、本手法での最少 の歯数は一定なので、ピッチ円直径 d=m* 歯数 T の関係から m を大きく設定するほど、配置される最小歯数の歯車が大きく なる.配置する歯車は大きくなるほど、断面に敷き詰めること が困難になるので、m は大きいほど生成モデルが入力モデルの 近似形状を得るのが困難になる.

入力モデルの大きさは有限なので、断面内に配置する際に、 選択できる歯車のサイズには最大値が存在する.したがって、 選択できる歯車のdには最大値が存在する.dが最大値の時のT の値は、d=m*Tの関係から、mが小さいほど大きくなる.また、本手法では、選択できる歯車のTには最小値が存在しており、Tは自然数なので、選択できる歯車のTの値の個数はTの 最大値からTの最小値を引いたものになる.そして、そのTの 値の個数は、選択できる歯車のdの値の個数を示す.これらの ことにより、mが小さいほど、選択できる歯車のdの値の個数 が増える.

ここで、本手法はかみ合う歯車を置いていく手法であり、配 置済みの歯車のピッチ円と1点で接触するピッチ円を持つ歯車 を離散した位置に置いていく手法である.したがって、その各 位置で選択できる歯車のdの値の個数が多いほど、より多くの 位置で配置済みの歯車のピッチ円と接触できる歯車を配置でき る.これより、mが小さい方が、かみ合う歯車の配置に有利に なり、逆にmが大きいほどかみ合う歯車は配置しづらくなる.

一方,mが小さくなるほど歯車の歯が小さくなり,生成モデルを見たときに歯車の歯が視認しづらくなる.

入力モデルの形状と入力パラメータが同じでも入力モデルが 持つ座標軸の向きによって、レイヤの並びが変わるので生成さ れる歯車集合体の見えは変わる.この例を付録資料に示す.ま た,生成モデルに金属の反射を適応して描画した例,歯車セットを構成する歯車に同じ色を付けた例,それをアニメーション させた動画例も付属資料に示す.動画は,歯車の回転を視覚的 にわかりやすくするため,歯車に装飾を施している.

5.2考察

図 22のウサギ形の歯車集合体において、入力モデルの片耳に あたる部分に歯車が配置されていない状態であることがわかる. これは、その部分の断面のセル数が少ないため、もしくは断面 が小さすぎたためであると考えられる. この問題への対処法と して、テクセル数の増加、モジュールを小さくするなどである 程度改善はできるが、本処理は離散的であるセルの中心位置ご とに計算するため、断面が歯車を置けるのに十分な大きさを持 っていたとしても歯車が置かれないという状況を完全になくす ことは不可能である.

さらに本手法は、配置済みの歯車に歯車セットを必ずかみ合わ せて置いていく方法である.このことから、特定の断面に歯車 が置かれなかった場合、その特定の断面とのみ X 方向に連続す る断面にも歯車が置かれないといった問題がある.この問題が 発生していると思われるのが図 22で、その問題の箇所を図 26 に示す.図 26ではマークしているウサギの耳の付け根に当たる 断面に歯車が配置されなかったため、その断面としか X 方向に 連続しなかったと思われる耳の先方向の断面にも、歯車が置か れなかったと考えられる.

また、生成されたモデルの傾向として、しわや眼球・ロ周辺 の凹凸、小さい突起などの細かい形状的特徴が消失しているが、 胴体の形状など大きい形状的特徴を保持していることが確認で きる.このことから本手法は、入力モデルの大局的な特徴を保 持する歯車集合体を生成できる手法であるといえる.

6. まとめ

本論文では、入力モデルの大局的な特徴を保持し、歯車の特 性を表現した歯車集合体を自動生成する手法を提案した.提案 手法は、入力モデルから複数の平行な断面を計算し、断面ごと にその断面形状を近似するように歯車を敷き詰めることで、入 カモデルの大局な特徴に当たる近似形状を得ている.そして、 歯車の特性を表しながら歯車を敷き詰める手法として、歯車の 規格を用いながら歯車セットを1つずつ置いていく局所的な手 法を採用している.また、局所的な手法なために起こる問題を 防ぐべく、好ましくない歯車のサイズ・配置に制限をかける対 策を講じた.

今後の課題として以下の点が挙げられる.

・現状では、図 26に示すウサギの耳のような、入力モデルの近 似形状を得るのに必要な部分でも、一部の断面が小さければ、 歯車が配置されない問題がある.この問題を解決するため、入 力モデルのメッシュスムーズ化などの前処理や、断面が小さい 場合でも、必要に応じて、断面からはみ出ても歯車を置くなど の断面の大きさの制約にとらわれないで歯車を配置する仕組み が必要である. ・本手法はテクセル数が大きいほど、レイヤ数が多いほど、モジュールが小さいほど計算コストが高くなる. これらの計算コストを並列化や GPU 化などで低減を図る必要がある.

| 表の | 生成モデルの計管時間と入力パラメータ |
|-----|-----------------------------|
| 1 4 | 主成し / / / の可异时间と八// / / / / |

| | 図 21 | 図 22 | 図 23 | 図 24 |
|------------------------|-------------------------------|---------|---------|--------|
| テクセル数 B | 800 | 800 | 800 | 800 |
| レイヤ数 N | 30 | 30 | 30 | 5 |
| モジュール m | F/1200 | F/1200 | F/800 | F/1200 |
| 軸の太さ dg | (最小歯数の歯車のd _r)*4/5 | | | |
| 全歯車セットの 計算時間 | 11,327s | 7,210s | 13,816s | 476s |
| テクセル化と 領域分割の計算時間 | 2,631s | 2,569s | 2,403s | 490s |
| 跨ぎセルの オープニング処理 | 15s | 14ss | 10s | 1s |
| 全体の計算時間 (メッシュ作成を含む) | 14,300s | 10,041s | 16,690s | 1,052s |





| モデル | | モデル |
|------|----|--------|
| 図 22 | 生成 | モデルの例2 |



図 23 生成モデルの例3





図 25 生成モデルの一部拡大図



図 26 連続する断面で起きる問題

参考文献

- [1] 下中 進, 宮田 一乗, 入力モデルの近似形状を持つ歯車 集合体モデルの生成, pp.44-49, NICOGRAPH 2012, 2012.
- [2] Insect Lab Studio, http://insectlabstudio.com
- [3] Ran Gal, Olga Sorkine, Tiberiu Popa, Alla Sheffer, Daniel Cohen-Or, 3D collage: expressive non-realistic modeling, Proceedings of the 5th international symposium on Non-photorealistic animation and rendering, pp.7-14, 2007.
- [4] 高橋 和茂,高井 昌彰,高井 那美,近似形状の構築が可能な缶アート制作支援システム,情報処理学会研究報告. グラフィクスと CAD 研究会報告, Vol.2011-CG-145 No.26, 2011.
- [5] 田村 友和,高井 昌彰,高井 那美,ユニット折り紙を用いた3次元メッシュモデルの近似形状構築,情報処理学会研究報告.グラフィクスと CAD 研究会報告, Vol.2010-CG-141 No.13, 2010.
- [6] Kui-Yip Lo, Chi-Wing Fu, Hongwei Li, 3D polyomino puzzle,

SIGGRAPH Asia '09 ACM SIGGRAPH Asia 2009 papers, Article No. 157, 2009.

- [7] Chongyang Ma, Li-Yi Wei, Xin Tong, Discrete element textures, SIGGRAPH '11 ACM SIGGRAPH 2011 papers, Article No. 62, 2011.
- [8] 技能士の友編集部,技能ブックス(13) 歯車のハタラキ, pp.32-33,株式会社大河出版,2007, ISBN978-4-88661-413-1.
- [9] 根本 良三, 絵とき歯車の基礎のき そ,pp.20-21,pp.20-21,pp.30-31,pp.41-42, 日刊工業新聞社, 2007, ISBN 978-4-526-05814-1.
- [10] ディジタル画像処理編集委員会,ディジタル画像処理, pp.179-180, 財団法人 画像情報教育振興協会, 2004, ISBN978-4-903474-01-4.
- [11] 蓮見 義久、パソコンによる歯車の設計計算、 pp.35-41,pp.20-32 理工学社、1993、ISBN4-8445-2520-4.
- [12] Mitsuba physically based renderer, http://www.mitsuba-renderer.org/

著者略歴

下中 進



2009 年度 大阪府立工業高等専門学校 専攻科 修了.2012 年度 北陸先端科学技術大学院大学 知識科学研究科 博士前期課程 修了予定.大学院では主に 3DCG を対象に研究している.

宮田 一乘



1986年東京工業大学大学院・総合理工学研究科・物理情報工学 専攻修士課程修了.同年,日本アイビーエム(株)東京基礎研 究所入社.1998年東京工芸大学学術部助教授.2002年より, 北陸先端科学技術大学院大学知識科学教育研究センター教授. 2012年より,同大学知識科学研究科教授.博士(工学).コン ピューターグラフィックスおよびメディア表現に関する研究に 従事.情報処理学会,芸術科学会,映像情報メディア学会,ACM, IEEE 等会員.