見え方の時系列変化を考慮した注視行動計画[†]

——遺伝的アルゴリズムを用いた最適化——

坂 根 茂 幸*・車 利 治**・小 俣 透*・佐 藤 知 正***

Planning Focus of Attention with Consideration of Time Varying Aspects

—— Search of the Best Plan by Using a Genetic Algorithm ——

Shigeyuki SAKANE*, Toshiji KURUMA**,

Toru OMATA* and Tomomasa SATO***

Visual monitoring of an object manipulated by a multi-fingered hand has a problem of occlusions caused by the fingers. In this paper, we present a new planning method, based on the 3-dimensional model of the environment, to focus attention in such a monitoring task. We formulate the planning of focus of attention as a scheduling problem for the whole task period rather than for a specific time. We introduce a Genetic Algorithm to cope with a combination explosion of the image features considered at all time steps. The Genetic Algorithm is a population based search method which finds solutions efficiently in a large search space by simulating some features of biological evolution. The algorithm consists of selection, crossover, and mutation procedures. To solve the planning problem, we have implemented the algorithm using a coding scheme for a gene to produce consistent offspring after the crossover. Experimental results demonstrate the usefulness of the proposed method by finding the pseudo optimal plan within a short computation time.

Key Words: robot vision, focus of attention, multi-fingered hand, sensor planning, scheduling, genetic algorithm

1. はじめに

ロボットシステムが作業環境の不確実性に対処するためには、センシング機能をもつことが必要になる。しかし、従来のセンサの固定的利用法では柔軟性に欠ける場合が少なくない。たとえば、カメラや照明が環境に対して固定的に設定されていると、対象物と環境物体の関係の変動に伴い、オクルージョンや影の問題が生じるために環境情報の入力が適切に行えなくなり、ロボット作業

の実行に重大な支障をきたす。これに対処するためには、センシングシステムの設定に自由度を与え、状況に応じて適応的に変更可能にすること、そして、その設定をシステムが自律的に計画する機能をもたせることが必要になる。

対象物の柔軟な操作を実現するために、通常の単純なグリップに代わり多指ハンドを用いる場合には、このような能動的センシングが一層重要な課題となる。つまり多指ハンドでは物体と指リンク間の相対位置関係が絶えず変化するので、作業を確実に実行させるためには視覚フィードバック制御が欠かせない。この操作の過程においては、多数の指によりオクルージョンが頻繁に生じ、物体の見え方が時々刻々と変化する。したがって、ある一時刻における固定的な注視計画では不十分であり、作業時間全体をとおしてみた能動的な注視行動の計画が必要になる。さらに、この計画問題では、視覚特徴、作業時間の増加に伴い、注視特徴のシーケンスの組合わせ数は指数関数的に増加するため、効率的な探索手法が求め

[†] 第 14 回ロボット工学部会研究会で発表 (1991・6)

* 電子技術総合研究所 つくば市梅園 1-1-4

** (有)トリムテクニカ 大阪市北区末広町 2-37

*** 東京大学先端科学技術研究センター

東京都目黒区駒場 4-6-1

* Electrotechnical Laboratory, Tsukuba

** Trimtechnica Co. Ltd, Osaka

*** Research Center for Advanced Science and Technol-

ogy, The University of Tokyo, Meguro-ku, Tokyo

(Received October 14, 1991)

(Revised March 16, 1992)

られる。

視覚フィードバック制御における注視行動の計画問題は、これまでもいくつか扱われてきた。坂根ら¹⁾は把握物体のエッジ特徴の選択問題として、また、Feddemara²⁾は、対象物にある穴の特徴選択問題として視覚フィードバック制御の問題の中で取り扱っているが、いずれも、静的な状態、すなわち、ある特定の時点における状態に基づく注視行動計画であった。さらに、坂根ら³⁾は、把握物体が時間経過と共に移動する際の照明設定計画の中で注視すべきエッジ特徴の選択も含めた計画を行ったが、照明軌道の計画が主であり、注視行動の計画を陽にスケジューリング問題として定式化するには至っていない。一方、多指ハンドに視覚センサを組み合わせたシステムの研究では、視覚で得られた対象物体の形状に応じて把握モードを生成する研究^{4),5)}がある。しかし、把握物体を視覚フィードバックに基づいて操作する際の注視行動計画は、まだ扱われていない。本研究で扱うような多指ハンド作業の注視行動計画の研究が従来なかった理由には、注視特徴の組合せ、および、それらを時間軸方向に組み合わせた積として、組合せ爆発問題に対処しなければならないという困難性が挙げられる。

本研究の目的は、多指ハンドを用いた物体操作を確実にするため、3次元幾何モデルに基づいて物体の見え方を予測し、作業時間全体をとおした注視行動計画を行うことである。ここでは指と対象物の間の相対的關係を作業時間をとおした幾何モデルに基づく予測として与えるが、実際には誤差を生じるのでこの誤差を修正するためにセンシングが必要である。もし、その誤差が一定の範囲内にあると仮定すると、幾何モデルから見え方を予測でき、正確なセンシングが可能である。逆に、幾何モデルに基づく予測が可能ならば、誤差をさらに小さくすることが可能になる。

本論文では、この注視行動計画を注視特徴のスケジューリング問題として定式化する。さらに、その探索手法として遺伝的アルゴリズム^{6),7)}(以下、GAと略す。)を利用することにより組合せ爆発に対処し、比較的短時間に良好な解が得られる方法を提案する。多指ハンドの幾何モデルを用いた実験結果は、本方法の有効性を示している。

2. 注視行動計画の定式化

注視特徴として物体のエッジを用いることとし、多指ハンドによる物体操作例として、物体を水平面上で回転させたときの様子を Fig. 1 に示す。多指ハンドと物体の相対位置関係の変化により、注視特徴であるエッジの見え方が時々刻々と変化することがわかる。ここでは、物体のトラッキングを確実にするための能動的センシング

Fig. 1 Manipulation of an object by a multi-fingered hand

計画を、時間と共に変化する視覚特徴のうちどれを注視すべきかという、注視特徴のシーケンスのスケジューリング問題として定式化する。この定式化における課題には、(1)評価関数の設定、(2)スケジューリングの解法がある。本研究では、(1)については以下に述べる二つの評価基準を導入した。また、(2)については、その組合せ爆発問題に対処するために、GAによる探索方法を導入した。まず評価基準については、つぎの二つの観点から評価を行う。

- 1) 注視エッジの組合せに起因するトラッキングの信頼性の評価を考える。ここでは数値解析手法の一つである特異値分解を用いて、物体の3次元の位置パラメータからカメラの透視平面上のエッジ位置パラメータへの関数のヤコビ行列の条件数を各時刻ごとに計算、正規化し、その総和をこの評価関数とする(詳細は文献1))。
- 2) 注視エッジの連続性の評価を考える。すなわち、注視エッジそのものの評価がそれほど変わらなければ、注視領域の設定にかかる負荷を減らす意味からも、なるべく同じエッジを注視し続けたほうがよい。そこで、注視エッジの切り換え数を各時刻ごとに計算し、その総和をこの評価関数とする。

上述の各評価をスケジューリング問題として定式化すると、つぎのようになる(以下、問題(1)と略記する)。

$$\text{最小化 } z = c_s \sum_{i=0}^{p-2} f_s(x^i, x^{i+1}) + c_c \sum_{i=0}^{p-1} f_c(x^i) \quad (1)$$

$$\text{条件 } x^{iT} x^i = e^{iT} x^i = m \quad (i=0, \dots, p-1)$$

(条件式の右肩の T はベクトルの転置を表わす)

ただし、

n : 物体の総エッジ数, m : 注視するエッジ数

c_s, c_c : 重み係数

(c_s, c_c は z が $0 \leq z \leq 1$ で正規化されるように決められる。)

$$f_s(x^i, x^{i+1}) = (x^i - x^{i+1})^T (x^i - x^{i+1}) / 2$$

: 時刻 i と $i+1$ 間の注視エッジの切り換え数

$f_c(x^i)$: 時刻 i の注視エッジの組合せから計算される行列の条件数

また、作業時間全体は p 個の時刻に離散化されている。物体の各エッジにはあらかじめ (0 から始まる) 固有の番号がつけられていて、可視エッジベクトル e^i 、注視エッジベクトル x^i はつぎのように定義される。

$$e_j^i = \begin{cases} 1: \text{時刻 } i \text{ に } j \text{ 番目のエッジが見える。} \\ 0: \text{時刻 } i \text{ に } j \text{ 番目のエッジが見えない。} \end{cases}$$

$$x_j^i = \begin{cases} 1: \text{時刻 } i \text{ に } j \text{ 番目のエッジを注視する。} \\ 0: \text{時刻 } i \text{ に } j \text{ 番目のエッジを注視しない。} \end{cases}$$

ただし、 e_j^i, x_j^i は $e^i, x^i (\in \mathbf{R}^n)$ の第 j 要素である。たとえば物体の総エッジ数 $n=10$ とし、作業開始時に 3, 5, 6, 7 番目のエッジが見えていたとすると、 e^0 はつぎのように表わされる。

$$e^0 = (0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 1 \ 1 \ 0 \ 0)^T$$

なお可視エッジベクトル e^i は、エッジがほかの物体により隠されないかどうか (可視性)、画像からエッジを検出するためにエッジを p 画素領域のあいだの明暗値の差 (コントラスト) が十分あるか、エッジの長さが十分であるか否か (エッジ長) の各評価から決定される。

3. 遺伝的アルゴリズムを用いた最適化

前章で定式化されたスケジューリング問題は、可視エッジ数や作業時間の増加に伴い、組合せ的爆発を生じる。たとえば、各時刻の可視エッジ数がすべて w 個であったとすると、解 $x_i (i=0, \dots, p-1)$ のとりうるすべての場合の数は $(wC_m)^p$ になる。ここで、 $p=10, w=6, m=3$ とおくと、その数は約 10^{13} という膨大な数になる。そこで、本研究では、この組合せ爆発に対処するために遺伝的アルゴリズム (GA: Genetic Algorithm)^{6), 7)} を解の探索に導入する。

GA は、生物の進化の過程を最適化問題に応用した探索アルゴリズムである。その特長として、多数の遺伝子集団 (population) を基に探索するので探索領域が広くとれること、また、確率的要素を用いて局所解からの脱出手段を有することなどがあげられる。

Table 1 Synopsis

T	number of generation
K	number of genes (even number)
$Y(t)$	a population at time $t (t=0, 1, 2, \dots, T)$
$y^k(t)$	a gene ($k=0, 1, 2, \dots, K-1$)
$v(y^k(t))$	fitness function
$P_s^k(t)$	probability of selection
P_c	probability of crossover
P_m	probability of mutation
G_s	selection operator
G_c	crossover operator
G_m	mutation operator

GA に用いられる記号を Table 1 に示す。なお、遺伝子 $y^k(t)$ は p 次元の行ベクトルでその第 i 要素を $y_i^k(t)$ と表わす。GA では、遺伝子 $y^k(t)$ を評価する尺度として適応度 $v(y^k(t))$ が用いられる。その適応度 v が大きい遺伝子ほどつぎの世代に残る確率が高くなる。適応度 v には、問題 (1) の z に適当な変換を施して用いられる。

本研究で用いた GA は、文献 6) の SGA (Simple Genetic Algorithm) を基にして構成したもので、遺伝子数 K や世代交代数 T は一定という単純な構造をもつ。以下、図の説明を簡単に行う。まず、探索開始時に初期遺伝子集団 $Y(0)$ を生成する。つぎに、この遺伝子集団を親集団とし、選択確率 $P_s^k(t)$ (後で定義される) に基づいて二つの遺伝子を選択 (G_s) する。さらに交差 (G_c)、突然変異 (G_m) を行い、遺伝子の内容を変化させて子集団 $Y(1)$ の要素とする (G_s, G_c, G_m については後で説明する)。この世代交代を繰り返すことにより、適応度の大きい遺伝子が残る、最適解が求められる。なお、選択確率 $P_s^k(t)$ はつぎのようにして計算される。

$$P_s^k(t) = \frac{v(y^k(t))}{\sum_{k=0}^{K-1} v(y^k(t))}$$

親集団から子集団を生成させるときに使われる三つの操作について説明する。

1) 選択 (selection): G_s

選択確率 $P_s^k(t)$ に従って親集団 $Y(t)$ から遺伝子 $y^k(t)$ を二つ取り出す。この選択確率 $P_s^k(t)$ が大きい遺伝子ほど子集団に残る可能性が高くなる。

2) 交差 (crossover): G_c

選択 P_s^k で選択された二つの遺伝子に対し、等確率で 2 か所の切れ目を選び、それらの切れ目に p まれた要素を交換する。ビット列で構成された遺伝子 y^{k1}, y^{k2} に対して交差 G_c を行った例を下に示す ($0 \leq k1, k2 \leq K-1$)。

$$y^{k1}(t) = (0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0)$$

$$y^{k2}(t) = (1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1)$$

↓

$$y_c^{k1}(t) = (0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0)$$

$$y_c^{k2}(t) = (1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1)$$

ただし、 $y_c^{k1}(t), y_c^{k2}(t)$ は遺伝子集団 $Y(t)$ から $Y(t+1)$ が生成されるまでの一時的な遺伝子である。

3) 突然変異 (mutation): G_m

遺伝子の各要素に対し、突然変異確率 P_m で値を変化させる。下の例では、 $\sim$ の部分に対して突然変異 G_m が適用されている。

$$y_c^{k1}(t) = (0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 0)$$

$$y_c^{k2}(t) = (1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1)$$

↓

$$y_m^1(t) = (1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1 \ 0 \ 0)$$

$$y_m^2(t) = (1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 1)$$

$y_c^1(t)$, $y_c^2(t)$ と同様, $y_m^1(t)$, $y_m^2(t)$ も一時的な遺伝子である。

遺伝子をどのように構成するかは, GA の探索性能を左右する重要な問題である。問題(1)の解 x^i ($i=0, \dots, p-1$) をそのまま時系列に並べると np 個のビット列で構成される遺伝子がコード化できる。しかし, この遺伝子に交差 G_c や突然変異 G_m を適用すると, 問題(1)の条件を満たさない遺伝子(致死遺伝子)が多数でてくる。たとえば, $n=4$, $m=2$, $p=2$ とし, 可視エッジベクトル e^0, e^1 を以下のようにおく。

$$e^0 = (1 \ 1 \ 0 \ 1)^T, \quad e^1 = (1 \ 1 \ 1 \ 0)^T$$

さらに, 選択 G_s により $y^{k1}(t)$, $y^{k2}(t)$ がつぎのように選ばれ,

$$\begin{array}{c|c} x^{0T} & x^{1T} \\ \hline y^{k1}(t) = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0) \\ y^{k2}(t) = (0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0) \end{array}$$

交差 G_c がつぎのように行われたとする。

$$\begin{array}{c|c} y^{k1}(t) = (1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0) \\ y^{k2}(t) = (0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0) \end{array}$$

↓

$$\begin{array}{c} y_c^1(t) = (1 \ 1 \ 0 \ 1 \ 0 \ 1 \ 1 \ 0) \\ y_c^2(t) = (0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0) \end{array}$$

$y_c^1(t)$ では, $e^{0T}x^0=3$, $y_c^2(t)$ では $e^{0T}x^0=1$ となり, いずれも $m=2$ の条件を満たさない。さらに $y_c^1(t)$, $y_c^2(t)$ の後半 4 ビットは, いずれのビットに突然変異 G_m が適用されても, $x^{1T}x^1=1$ または $x^{1T}x^1=3$ となり $m=2$ の条件を満たさなくなる。

この例のように, 注視エッジベクトルを時系列に並べた遺伝子では問題(1)の条件($x^{iT}x^i=e^{iT}x^i=m$)を満たさない遺伝子(致死遺伝子)が多数できてしまい, 世代交代が行えなくなってしまう。

致死遺伝子をなくすためには, 適応度 v にペナルティ関数をつけ加えて致死遺伝子の適応度を極端に下げたり, または, 遺伝子操作によって致死遺伝子が出ないようなコード化を行うことが考えられる。ここでは, 後者の方法を取り, コード化を工夫することにする。まず, 注視エッジベクトル x^i のとりうるすべてのベクトルをあらかじめ生成しておいて, 適当に番号をつける。この番号を用いて遺伝子をコード化する。先の例では, x^0, x^1 の候補として

$$x^0 \rightarrow 0: (1 \ 1 \ 0 \ 0)^T, \quad 1: (1 \ 0 \ 0 \ 1)^T, \quad 2: (0 \ 1 \ 0 \ 1)^T$$

$$x^1 \rightarrow 0: (1 \ 1 \ 0 \ 0)^T, \quad 1: (1 \ 0 \ 1 \ 0)^T, \quad 2: (0 \ 1 \ 1 \ 0)^T$$

のように番号をつけると, $y^{k1}(t)$, $y^{k2}(t)$ はつぎのようにコード化できる。

$$y^{k1}(t) = (0 \ 1)$$

$$y^{k2}(t) = (2 \ 2)$$

このコード化により, 交差 G_c による致死遺伝子は生じることがわかる。なお, 突然変異 G_m は, 適用される要素を, それ以外の残りの要素から等確率で選ばれた要素に変える操作とする。

さて, ここで用いた GA では, 探索性能を上げるために以下の三つの項目について改良を行った。

① 評価関数 $z(y^k(t))$ から適応度 $v(y^k(t))$ への変換について

問題(1)は最小化問題であり, $0 \leq z \leq 1$ であるから, 簡単に

$$v(y^k(t)) = 1 - z(y^k(t)) \quad (2)$$

とおくことにより適応度 v を構成することができる。ここでは, (2)式をさらに 4 乗し,

$$v(y^k(t)) = \{1 - z(y^k(t))\}^4 \quad (3)$$

とすることにより, 適応度の大きい遺伝子の選択確率を大きくする。

② 初期遺伝子集団 $Y(0)$ の生成法について

GA では, 初期遺伝子集団 $Y(0)$ はランダムに生成されることが多いが, ここではその生成過程で, 各遺伝子の適応度により確率的に遺伝子をふるいにかけて, $Y(0)$ 全体の評価を高めることにする。この確率的なフィルタリングでは, 具体的に以下の処理をしている。初期遺伝子集団をつくる際に, 各時点における可能なエッジの組合せがヤコビアン条件数と共に求められており, それらは条件数の小さい順序でソートされている。そこで, これが今 n 個(時刻によりこの個数は異なる)あるとすると, まず, 0 から $n-1$ の間の数をランダムに選ぶ。これを r とすれば, つぎに 0 から r の間の数をランダムに選び, これをコードの一部にする。条件数の小さい順でソートされているので, より条件数の小さいほう(したがって評価の良いもの)が選ばれる確率が高まる。

③ 最も適応度の大きい遺伝子の保存について

遺伝子集団 $Y(t)$ で適応度の最も高い遺伝子(二つ)の子集団 $Y(t+1)$ にそのまま遺伝させる。これは, GA の確率的な操作により, 適応度の最も大きい遺伝子が破壊されるのを防ぐためである。なお, 遺伝子の各世代における総数は一定値としているので, 適応度が低い遺伝子は世代交代の進行に伴い徐々に消滅していく。

以上の三つの改良項目は, 優秀な遺伝子の生成, 評価, 保護の度合いを強めたものであり, 次章の実験でその効果の有無を調べる。

4. 注視行動計画の実験例

ここでは, Fig. 1 に示すように, 多指ハンドで水平面上

の物体を一定の角速度で回転させる作業に対し、注視行動計画を行った。実験は、ワークステーション (SPARC-Station 2) 上でオブジェクト指向言語 EusLisp[®] を用いて記述し、多指ハンドのモデルには文献 9) のものを用いた。本実験で用いた各パラメータ値を Table 2 に示す。なお、重み係数 c_s, c_c の値は、作業環境、作業内容などに基づいて決められるべきであるが、本稿では、単に $c_s \sum f_s, c_c \sum f_c$ の最大値が 0.5 となるように c_s, c_c をとった。

まず、交差確率 P_c 、突然変異確率 P_m を決めるために、作業時間数 p を短くして ($p=11$) 予備実験を行った。なお、GA によって求められた最良値 (最終遺伝子集団 $Y(T)$ で最も適応度の大きい遺伝子の評価値 z) は探索を行うごとに異なるので、同じ条件で 10 回探索を行い、求められた最良値の平均値をとって評価する。その実験結果を Table 3 に示す。この結果より、 $P_c=0.8, P_m=0.01$ で最良値が極小になっていることがわかる ($P_c=0.8, P_m=0.01$ に決める)。このときの最良値の各統計量と最適値との比較を Table 4 に示す (最適値は逐次探索により求めた)。最良値にはある程度のばらつきがあるものの、平均して最適値に近い値が得られている。最も最適

値に近かった最良解と最適解の比較を Fig. 2 に示す。同図の縦軸は、幾何モデルから予測された画像のエッジをつけた ID 番号である。横軸は、作業の進行に伴う時間を示している。○印をつけたエッジは可視エッジを、●印をつけたエッジは最適解を示している。そして、◇印をつけたエッジは、本方法により計画された注視エッジの組合せを示している。これより、その最良解は最適解と比較して時刻 $t=7$ の部分が違っているだけで、ほぼ最適解に近い解が得られていることがわかる。

さらにこの予備実験での条件で、前章で説明した三つの改良点について実験を行って比較した (Table 5)。各

Fig. 2 Comparison of edge combinations between the optimal ($z=0.699$) and the best-of-generation ($z=0.0673$) plans

Table 2 Parameter values used in the experiments

total number of edges of the object	n	36
number of time steps in the task	p	21
number of edges to pay attention to	m	3
number of generations	T	100
number of genes	K	50
probability of crossover	P_c	0.8
probability of mutation	P_m	0.01

Table 3 Comparison of the best-of-generation (average of 10 runs) in the preliminary experiments with various conditions of P_c and P_m

P_c	P_m	0.001	0.01	0.1	0.5
0.05		0.210	0.205	0.192	0.258
0.3		0.181	0.158	0.136	0.228
0.6		0.149	0.138	0.126	0.209
0.8		0.141	0.096	0.098	0.221
1.0		0.134	0.107	0.108	0.236

Table 4 Comparison between the optimal and the best-of-generation plans ($P_c=0.8, P_m=0.01$)

optimal value of Z	0.0669
average of Z values	0.096
the maximum of Z values	0.134
the minimum of Z values	0.0673
standard deviation	0.024

Table 5 Performance of the best-of-generation with various conditions

conditions			Z value of the best-of-generation (average of 10 times)
A	B	C	
A 0	B 0	C 0	0.283
A 1	B 0	C 0	0.177
A 0	B 1	C 0	0.247
A 0	B 0	C 1	0.171
A 1	B 1	C 1	0.096

A 1 : Use fitness function $v=(1-z)^4$

A 0 : Use fitness function $v=(1-z)$

B 1 : Use filtering for initial population

B 0 : Use random selection for initial population

C 1 : Preserve a gene of the best performance for the next generation

C 0 : Do not preserve a gene of the best performance for the next generation

Table 6 Results of the experiments 1

average of Z values	0.144
the maximum of Z values	0.205
the minimum of Z values	0.094
standard deviation	0.017

Fig. 3 Performance of the algorithm in the experiments 1

Fig. 4 Edge combinations of the best-of-generation plan ($z=0.094$) in the experiments 1

Table 7 Results of the experiments 2

average of Z values	0.163
the maximum of Z values	0.206
the minimum of Z values	0.123
standard deviation	0.021

Fig. 5 Performance of the algorithm in the experiments 2

項目とも最良値が減少しており、さらにこれら三つの項目を組み合わせることにより最良値が大きく減少し、探索性能の向上に効果があることがわかる。

つぎに、作業時間数 $p=21$ にして Table 2 の条件で 50 回行った実験結果 (実験 1) を Table 6 に示す。なお、最適値は逐次探索で 10 万秒演算を行っても求められなかった。Fig. 3 に遺伝子集団の世代推移を示す。集団の平均評価値は、局所的には増えたり減ったりしながら漸近的に減少し、それに伴い集団の最小評価値も減少してい

Fig. 6 Edge combinations of the best-of-generation plan ($z=0.123$) in the experiments 2

Table 8 Comparison of computation times between the sequential search and the Genetic Algorithm

algorithm		time	6	11	16	21
Exp.1	SS		0.3	11.5	1872.7	...
	GA		21.5	36.6	52.0	67.3
Exp.2	SS		338.8	10018.0	...	...
	GA		22.3	37.0	54.0	71.6

SS : sequential search, GA : genetic algorithm

Table 9 Comparison of Z values between the sequential search and the Genetic Algorithm

algorithm		time	6	11	16	21
Exp.1	SS		0.061	0.067	0.075	...
	GA		0.090	0.125	0.140	0.132
Exp.2	SS		0.075	0.069	...	...
	GA		0.171	0.128	0.147	0.161

SS : sequential search, GA : genetic algorithm

る。Fig. 4 に最も値の良かった最良解を示す。注視エッジは、ひんぱんに変化せずある程度まとまって選ばれており、ほぼ最適解が得られていると考えられる。

さらに、カメラや照明の位置を変えて同じように行った実験結果 (実験 2) を Table 7, Fig. 5, Fig. 6 に示す。実験 1 とほぼ同様な結果が得られており、作業環境が変わっても良好な解が得られることがわかった。

最後に、作業時間数をいろいろ変えて、逐次探索と GA による探索の、計算時間と求められた最適値、最良値 (10 回探索を行ったときの平均値) の比較結果を Table 8, Table 9 に示す。逐次探索に要する計算時間が指数関数的に増加しているのに対し、GA による探索に要する計算時間は線形的な増加で抑えられ、かつ、最適値に近い値が得られた。これより GA による探索は可視エッジ数、作業時間数の増加に十分対応できることが確認され

た。

5. おわりに

ロボットのセンシングシステムの環境への適応能力を高度化することは、多指ハンドを用いるロボット作業の場合に一層重要な課題となる。たとえば、多指ハンドにより物体操作を行う際の視覚フィードバックにおいては、指のオクルージョンが頻繁に生じ、見え方が時々刻々と変化するために、それらを考慮した注視行動計画を行う必要がある。しかし、従来、このような注視行動計画はほとんど扱われてこなかった。

本論文では、この注視行動計画を注視特徴のスケジューリング問題として定式化すると共に、注視特徴の組合せ爆発問題に対処する探索手法として遺伝的アルゴリズムを用いる計画方法を提案した。そして、多指ハンドの幾何モデルを用いた実験により、本方法の有効性を確認した。

今後の課題としては、多指ハンドの姿勢も考慮した注視行動計画、および、可動にしたカメラ視点／照明位置の設定計画と結合した注視行動計画などがある。

最後に、本研究を行うにあたり御支援頂きました高瀬國克知能システム部長、弓場敏嗣前知能システム部長（現情報アーキテクチャ部長）、FusLispを開発された松井俊浩主任研究官、ならびに、協力頂いた知能システム部ロボットグループの方々に感謝します。

参考文献

- 1) 坂根, 佐藤, 柿倉: 視覚フィードバック制御の注視行動プランニング, 計測自動制御学会論文集, 24-6, 608/615(1988)
- 2) J. T. Feddema, C. S. G. Lee and O. R. Mitchell: Weighted Selection of Image Features for Resolved Rate Visual Feedback Control, IEEE Trans. on Robotics and Automation, 7-1, 31/47 (1991)
- 3) 坂根, R. Niepold, 佐藤, 白井: 環境モデルに基づくハンドアイシステムの照明設定プランニング, 日本ロボット学会誌, 7-3, 130/141 (1989)
- 4) K. Rao, G. Medioni, H. Liu, G. A. Bekey: Robot Hand-Eye Coordination: Shape Description and Grasping, Proc. IEEE Int. Conf. Robotics and Automation, 407/411 (1988)
- 5) S. A. Stansfield: Knowledge-Based Robotics Grasping, Proc. IEEE Int. Conf. on Robotics and Automation, 1270/1275 (1990)
- 6) D. E. Goldberg: Genetic Algorithms in Search, Optim-

ization, and Machine Learning, Addison Wesley (1989)

- 7) 須貝, 平田: 組合せ最適化アルゴリズムとその応用, 計測自動制御学会論文集, 29-12, 1084/1091 (1990)
- 8) T. Matsui and M. Inaba: EusLisp: An Object-Based Implementation of Lisp, Journal of Information Processing, 13-3, 327/338(1990)
- 9) 小俣, 松井, 車: 4指14自由度ロボットハンドの制御システム構成, 日本機械学会ロボティクス・メカトロニクス講演会講演論文集(1991)

[著者紹介]

坂根 茂 幸 (正会員)

1972年, 東京工業大学工学部制御工学科卒業, 74年同大学大学院修士課程修了. 同年電子技術総合研究所入所. 現在, 知能システム部自律システム研究室室長(工学博士). 日本ロボット学会の会員.

車 利 治

1990年, 名古屋大学大学院工学研究科情報工学修了. 同年電子技術総合研究所入所. 91年, (有)トリムテクニカへ入社, 現在に至る. 画像処理関係を主としたソフトウェア業務に従事.

小 俣 透 (正会員)

1981年, 東京工業大学制御工学科卒業. 83年同修士課程修了. 86年同博士課程修了(制御工学専攻). 同年工業技術院電子技術総合研究所入所. ロボット工学の研究に従事. 日本ロボット学会の会員.

佐 藤 知 正 (正会員)

1971年, 東京大学産業機械工学科卒業. 76年同博士課程修了, 電子技術総合研究所入所. 83年から84年まで米国ウッツホール海洋学研究所に客員研究員として滞在. 91年東京大学先端科学技術研究センターに移籍. 超音波撮像, 知能ロボットの研究に従事.