社団法人　情報処理学会　研究報告
IPSJ SIG Technical Report

2004－ＩＣＳ－135　(23)
2004／3／16

決定木生成のための共生進化における部分解の検討

大谷　紀子†　　志村　正道†

† 武蔵工業大学環境情報学部
〒 224-0015 横浜市都筑区牛久保西 3-3-1
E-mail: †{otani,shimura}@yc.musashi-tech.ac.jp

あらまし　分類規則の表現技法の１つである決定木は，記憶容量やクラス判定処理の速度，分類規則の解釈の容易さの点から，未知事例分類に関して高正解率であると同時に，冗長性が少なく簡素であることが望ましい．本研究では，遺伝的アルゴリズムの１手法である共生進化を利用することで，予測正解率が高く簡素な決定木の生成を目指す．共生進化の特徴は部分解を個体として保持する点にある．新しい決定木学習システム SESAT2 を構築し，部分解の違いが及ぼす影響について検討する．部分解の遺伝子表現と集団構成が異なる４種類の SESAT2 を用意し，UCI リポジトリのデータにより評価した．その結果，システムの違いが訓練適応度と予測正解率に与える影響は少ないが，簡素さと学習時間を左右することが示された．

キーワード　決定木，遺伝的アルゴリズム，共生進化，部分解

Examination of Partial Solution in Symbiotic Evolution for Decision Tree Generation

Noriko OTANI† and Masamichi SHIMURA†

† Faculty of Environmental and Information Studies, Musashi Institute of Technology
Ushikubo-nishi 3-3-1, Tsuzuki, Yokohama, 224-0015, Japan
E-mail: †{otani,shimura}@yc.musashi-tech.ac.jp

Abstract　In representing classification rules by decision trees, simplicity of tree structure is as important as predictive accuracy especially in consideration of the memory capacity and the time required to classify. This paper addresses the issue of the generation of accurate and simple decision trees based on symbiotic evolution. It is distinctive of symbiotic evolution that individuals represent partial solutions. We construct a new system SESAT2, and examine the effect of some matters for partial solutions in it. Experiments were performed with four types of SESAT2 on several datasets in UCI repository. Our results show that the differences between the four systems produce no effect in training fitness and predictive accuracy, but some effect in simplicity and training time.

Key words　decision tree, genetic algorithm, symbiotic evolution, partial solution

1. はじめに

決定木は分類規則の木構造による表現技法である．決定木の非終端ノードには属性の種類，アークには属性値，終端ノードにはクラスが割り当てられており，事例の各属性値に従って根ノードから終端ノードまで決定木を辿ることで，事例の属するクラスが判定される．属するクラスが既知である訓練事例の分類結果を指標として，木の形状および値の割り当てを決定し，未知事例分類のための決定木を生成する．

これまでに提案された ID3 [1]，CART [2]，C4.5 [3] 等の情報量に基づく決定木生成システムでは，予測誤り率による枝刈りやブースティングアルゴリズム等の利用により，未知事例の分類において高い正解率を得ている．しかし，決定木の分類正解率と簡素さはトレードオフの関係にあり，分類正解率の高い決定木は複雑で大きいのが一般的である．記憶容量や分類処理の速度，分類規則の解釈の容易さの点から，簡素さは決定木の評価において重要な要因といえる．高正解率を保持しつつ木を簡素にすることは，決定木生成での一目標となっている．

最適化問題における解の探索手法の１つに遺伝的アルゴリズム（Genetic Algorithm; GA）[4] がある．与えられた問題に対する解候補を個体として表現し，個体の集団において評価，選択，生殖というサイクルを繰り返すことで，解として最適と思わ

れる個体を生成する．Moriarty らは，GA の一手法として共生進化（symbiotic evolution）を提案し，ニューラルネットワークの隠れ層を学習するシステム SANE を構築している [5]〜[7]．共生進化の特徴は，部分解と全体解をそれぞれ個体とする 2 つの集団を並行して進化させる点にある．全体解は部分解の組み合わせにより表現される．SANE では隠れ層のニューロンを部分解，ニューロンの組み合わせであるネットワーク構成子（network blueprint）を全体解として集団を形成し，両者を並行進化させる．このとき，全体解の評価に基づいて部分解を評価し，その評価値に従って進化した部分解を全体解に反映する．両者を相互に関係付けながら進化させることで集団内の個体の多様性が維持され，局所解への収束を回避した効率的な最適解探索が可能となっている．

共生進化を決定木生成に適用するにあたり，まず最初に部分解の表現が課題となる．決定木の最小構成単位であるアークとノードに着目し，アークとそれに連結するノードの組を部分解とすると [8], [9]，上下に位置するノードの種類と属性値を示すアークの関係を部分解集団で学習できる．しかし，1 つのノードから出るアーク同士の関係や，アークとノードの組の上下の結合関係の学習は，すべて全体解集団に委ねられるため，部分解集団と全体解集団の学習対象項目が非常に偏っている．

これに対して決定木生成システム SESAT [10] では，決定木を高さ 1 の部分木の組み合わせと見なし，高さ 1 の部分木を部分解，部分木の組み合わせで表現される決定木を全体解とした．部分解集団で 1 つの属性による分類規則の獲得を目指し，全体解集団では部分木の上下の結合関係の獲得のみを目指すことによって，両集団の学習対象項目の均衡を実現している．ところが，部分解の遺伝子表現に属性値を表す箇所を含むため，染色体が長くなり，処理に膨大な時間がかかったり，処理可能なデータが限られているなどの問題が見られた．

本稿では，未知事例を正確に分類できる簡素な決定木の生成を目的として，決定木生成のための共生進化に有効な部分解の遺伝子表現と集団構成について検討する．部分解を高さ 1 の部分木としたときにも，部分解の遺伝子表現や部分解集団の構成の仕方によって，各集団における学習対象項目が異なる．学習対象項目の変化が決定木に与える影響を実験により明らかにする．評価実験では，SESAT を改良した決定木生成システム SESAT2 を用いる．SESAT において，前述の問題を回避するために属性値を遺伝子座で表現し，適応度と進化戦略を改良したものが SESAT2 である．

以下，2. 章で SESAT2 における決定木生成手法と検討事項について説明する．3. 章では，部分解の遺伝子表現と集団構成の影響を比較するために，UCI リポジトリのデータを用いて行なった評価実験の結果を示す．評価実験の結果を踏まえて 4. 章で考察を行ない，5. 章で結論を述べる．

2. 決定木生成における共生進化

ニューラルネットワークでは，入力層からの情報が隠れ層にある複数のニューロンを介して出力層に伝達される．隠れ層の各ニューロンの動作は互いに独立であるため，隠れ層を構成するニューロンが決まると，ネットワーク全体の動作は一意に定まる．SANE では，全体解を隠れ層のニューロンの組み合わせで表現しているが，その組み合わせ方には制約がないため，交叉や突然変異などの遺伝操作により構造上不備のあるネットワークが生成されることはない．

一方，決定木は複数の部分木の組み合わせと見なすことができるが，決定木に含まれる部分木が指定されても，決定木の構造は一意に定まらない．各部分木の結合関係により決定木の構造は変化する．部分木の数に過不足が生じたり，冗長で無意味な決定木が生成される場合もある．従って，決定木生成に共生進化を適用する際には，部分解の単なる組み合わせで表現された全体解を一般的な遺伝操作で進化させることは望ましくない．部分解の結合関係を保持した形で全体解を表現し，解候補として適切な決定木のみを集団の個体とすることが必要となる．以下，SESAT2 における決定木生成方法の詳細を説明する．

2.1 決定木の制約

GA により木構造の解を探索するという点で，本手法は遺伝的プログラミング（Genetic Programming; GP）に類する．GP では，学習過程で木のノード数が急速に増大するブロート現象が発生しやすい．ブロート現象の原因は，実行されても正解率に関与しない冗長な部分木（意味論的イントロン），および実行されない無意味な部分木（構文的イントロン）の存在にある．イントロン（intron）とは DNA 中で遺伝情報を持たない部分を指す．SESAT2 においては，隣り合う同一の部分木(注1)が前者，どの事例も到達しない部分木が後者にあたる．イントロンは最適解探索に有益な場合もあるが，実行時間，解の複雑さ，過学習の点を踏まえると，木の複雑化は可能な限り回避すべきと思われる．SESAT2 では，複雑な木の生成を避けるため，解候補として生成される木に対して次の 4 つの制約を課す．

制約 1： 木の高さは H 以下である．

制約 2： 非終端ノードは 2〜M 個の子ノードを持つ．

制約 3： 隣り合う部分木は必ず異なる．

制約 4： 全ノードにいずれかの訓練事例が到達する．

制約 3 が意味論的イントロン，制約 4 が構文的イントロンへの対処である．H および M の値，さらに後出のパラメータ p_m, N_g, N_{sp}, N_{tp} の値は，データや目的に応じてあらかじめユーザが指定する．SESAT2 の学習過程で保持する決定木は，すべて上記 4 制約を満たすものとする．

2.2 sprig

SESAT2 の部分解 sprig を図 1 に示すような高さ 1 の部分木で表し，根ノードを属性ノード，葉ノードをクラスノードと呼ぶ．事例に出現する属性数を A，クラス数を C とすると，属性ノードには 1〜A の属性番号が入り，このノードにおいて当該属性により分岐することを表す．

sprig が決定木に組み込まれる際，クラスノードはクラスを表す終端ノード，あるいは別の sprig の属性ノードが接続される非終端ノードのいずれかとなる．終端ノードになる場合はク

(注1)：隣り合う部分木とは，共に同一の親を持ち，かつ隣り合っているノードを根ノードとする部分木を指す．

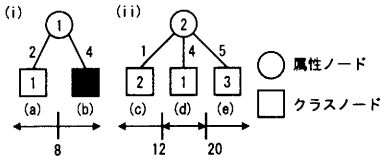

図 1　sprig の例

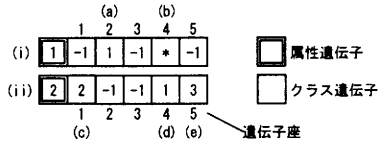

図 2　sprig の染色体の例 （$M = 5$ の場合）

ラスノードに 1〜C のクラス番号を入れ，非終端ノードになる場合にはそれを示す数値を入れる．図 1 では，非終端ノードになるクラスノードを網かけで示している．

各アークは 1〜M の属性値番号によりラベル付けされている．事例の属性値に従って属性ノードからクラスノードへと走査するときは，隣り合うアークのラベル平均が表す属性値を閾値として，辿るアークを選択する．図 1 では，M を 5 と指定し，訓練事例における属性 1 の属性値が 5〜9，属性 2 の属性値が 2〜18 であるとしたときの閾値を示している．

sprig を表す染色体は，1 個の属性遺伝子と M 個のクラス遺伝子からなる．図 1 の sprig を表す染色体の例を図 2 に示す．属性遺伝子が属性ノードに対応し，クラス遺伝子がクラスノードに対応する．図中の "*" は，決定木中で非終端ノードになるクラスノードの値が入ることを示す．クラス遺伝子が -1 のとき，sprig は対応するクラスノードを持たない．図 2 のクラス遺伝子 (a)〜(e) が，それぞれ図 1 のクラスノード (a)〜(e) に対応する．-1 以外のクラス遺伝子の位置を表す遺伝子座 1〜M がアークのラベルとなる．

sprig の染色体を新たに生成する場合は，制約 2 および制約 3 を満たす範囲で各遺伝子をランダムに設定する．N_{sp} 個の sprig を生成し，初期集団とする．

2.3　sprig に関する検討事項

sprig を部分解とすると，部分解集団（sprig 集団）と全体解集団の学習対象項目は以下のようにまとめられる．

(1)　1 つの非終端ノードに関する以下の項目 (a)〜(d)
- (a)　属性
- (b)　子ノード数
- (c)　各子ノードの種類
- (d)　各子ノードへ分岐するための属性値

(2)　(a)〜(d) が定まった非終端ノードの上下の結合関係

項目 (1) が部分解集団，項目 (2) が全体解集団の学習対象項目である．次のように sprig の遺伝子表現および集団構成を変化させると，両集団の学習対象項目も変化する．

(i)　下位属性を sprig のクラス遺伝子で指定するか否か

(ii)　1 つの sprig 集団に各属性の sprig を混在させるか，属性ごとに sprig 集団を用意するか

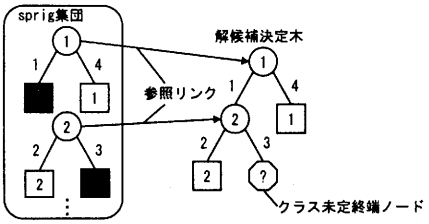

図 3　決定木構成子の例（P1）

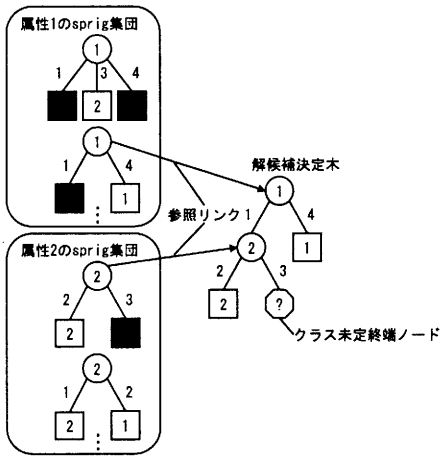

図 4　決定木構成子の例（PA）

(i) に関して，下位属性を指定する場合は，項目 (c) に関して「どの属性の sprig を接続するか」をも学習対象とするため，全体解集団では「指定された属性の sprig のうち，どれを接続するか」のみを学習する．一方，指定しない場合は属性に関する制約なしに項目 (2) を学習することになり，両集団の学習対象項目が異なる．前者を CY，後者を CN と呼び，クラスノードにはそれぞれ次の値を入れることにする．

CY：　-1〜$-A$（符号を反転した値を属性番号とする）

CN：　0

(ii) に関して，属性ごとに sprig 集団を形成すると，項目 (a) が学習対象ではなくなり，各 sprig 集団では当該属性に適した項目 (b)〜(d) を学習すればよいことになる．前者を P1，後者を PA とすると，sprig 集団の数は次のようになる．

P1：　sprig 集団数 $= 1$

PA：　sprig 集団数 $= A$

P1 と PA で sprig 集団の個体総数を等しくするために，PA の各 sprig 集団の個体数を N_{sp}/A 個とする．

後述する 3. 章では，CY と CN，P1 と PA の各組み合わせについて実験を行ない，各集団の学習対象項目の違いによる影響を比較する．

2.4　決定木構成子

SESAT2 における全体解を決定木構成子と呼ぶ．決定木構成子は，図 3，4 に示すような解候補決定木と参照リンクから構成されている．図 3 は P1，図 4 は PA の場合の例である．

Step.1	sprig 集団から無作為に選択した 1 つの sprig の値に基づいて部分木を生成し，解候補決定木とする．
Step.2	解候補決定木の非終端ノードのうち，下位の部分木が不定の非終端ノードに対して，無作為に選んだ sprig による部分木を付加する．レベルが H の非終端ノードはクラス未定終端ノードとする．
Step.3	すべての非終端ノードについて下位の部分木が定まるまで，Step.2 を繰り返す．
Step.4	クラス未定終端ノードのクラスを決定する．
Step.5	訓練事例の到達しない部分木を削除する．
Step.6	隣り合う同一部分木を統合する．
Step.7	子ノードを 1 つしか持たない非終端ノードを削除し，当該ノード以下の部分木を接続する．

図 5　決定木構成子生成手順

解候補決定木は，sprig 集団から選択した sprig のノードの値およびアークのラベルを参照して形成される．参照リンクは解候補決定木の生成時に参照した sprig からのリンクであり，解候補決定木の非終端ノードを終点とする．決定木構成子の生成手順を図 5 に示す．

Step.2 では，CY と CN，P1 と PA の組み合わせにより，次のように sprig を選択する．

CN-P1：　任意の sprig を選択する．

CY-P1：　指定された属性の sprig を選択する．

CN-PA：　任意の sprig 集団から，任意の sprig を選択する．

CY-PA：　指定された属性の sprig 集団から，任意の sprig を選択する．

クラス未定終端ノードとは，制約 1 を満たすために暫定的に作られた終端ノードであり，Step.4 において訓練正解率が最も高くなるようにクラスが割り当てられる．これにより，同一の sprig を参照する非終端ノードでも子ノードの種類が異なる可能性が生じる．Step.5 と Step.6 では，それぞれ制約 4 および制約 3 が満たされるように決定木構成子を変形する．子ノードが 1 つとなって制約 2 が満たされなくなった場合は Step.7 で修正する．

決定木構成子集団の個体数として指定された N_{tp} 個の決定木構成子を生成し，初期集団とする．複数の決定木構成子から参照される sprig や，どの決定木構成子からも参照されない sprig も存在し得る．

2.5 適 応 度

C4.5 では，分類正解率を高める指標として情報利得比を，過学習を回避するための指標として分類誤り率を採用している．情報利得比に基づいて決定木を生成した後，分類誤り率による枝刈りを行なう．一方，決定木構成子は解候補決定木の生成後に評価するため，決定木構成子の適応度には，生成後の決定木の分類正解率と過学習の可能性を同時に計ることが必要となる．

多くの訓練事例を正しく分類するよう，訓練事例に特化した決定木を生成した場合，各訓練事例は異なる終端ノードで正解と判断され，終端ノードごとの正解事例数の散らばりは大きくなる．このとき，訓練事例における正解率は高くなるが，過学習が起こっているために高い予測正解率は望めない．終端ノードごとの正解数の散らばりが小さいほど過学習の可能性が低いと考えられる．以上の考察より，正解数の散らばりを表す正解局在率を定義し，正解率が高く正解局在率が低いときに高くなる値を決定木構成子の適応度とする．

決定木構成子 T の正解局在率 $bias(T)$ は次式に従って算出する．ここで，全正解事例数を c，n 個の終端ノードにおける正解事例数をそれぞれ c_1〜c_n とする．

$$bias(T) = \begin{cases} \dfrac{-\sum_{i=1}^{n} \frac{c_i}{c} log_2 \frac{c_i}{c}}{-log_2 \frac{1}{c}} & (c \neq 0) \\ 1.0 & (c = 0) \end{cases} \qquad (1)$$

正解局在率は，すべての正解事例が 1 つの終端ノードに到達したときに 0，互いに異なる終端ノードに到達したときに 1 となり，終端ノードごとの正解数の散らばりが大きいほど大きな値を取る．

訓練事例の分類正解率 $acc(T)$ と正解局在率 $bias(T)$ から，決定木構成子を評価するための適応度 $tfit(T)$ を算出する．

$$tfit(T) = acc(T) \cdot (1 - 0.2 \cdot bias(T)) \cdot 100 \qquad (2)$$

sprig は参照リンク先の部分木により評価する．sprig S を参照する部分木のうち，最も適応度の高い決定木構成子に属する部分木を S の最良部分木と呼ぶ．最良部分木の属する決定木構成子の適応度を S の適応度とする．制約を満たすための変形により，sprig が表す木と参照リンク先の部分木が異なる場合があるが，sprig の適応度算出の際に，sprig が最良部分木を表現するよう sprig の遺伝子に変更を加える．これにより，適応度の高い決定木構成子に参照される sprig が高い評価を受け，その性質が集団内に広まる可能性が高くなる．

2.6 世 代 交 代

sprig 集団の世代交代では，[6] と同様にして，N_{tp} 個の個体のうち上位半数をそのまま次世代に残す．下位半数の個体は，上位四半数から選んだ 2 つの個体を親として交叉を行ない，生成された 2 つの子のいずれかと，2 つの親のいずれかで置き換える．交叉により終端ノード数が 2 個未満になった場合は，ランダムに位置番号と遺伝子の値を設定し，制約 2 を満たすようにする．子が制約 3 を満たさない場合は，問題箇所の遺伝子を他の遺伝子に置き換える．すべての個体の遺伝子に対して確率 p_m で突然変異を発生させ，次世代の個体とする．

決定木構成子集団の世代交代モデルとしては，[11] で提案されている MGG（Minimal Generation Gap）モデルを採用する．MGG モデルは，局所解収束の回避と進化的停滞の抑制を意図して考案されたモデルである．集団からランダムに非復元抽出された 2 個体を親として子を生成し，親と子の個体のうち，最良個体およびルーレット選択で選ばれた 1 個体の計 2 個体を次世代に残す．

子として生成されるのは，以下の 4 種類の木 C_1〜C_4 を解候補決定木とする決定木構成子である．ここで，親個体の解候補決定木を P_1，P_2 とし，P_1 と P_2 からランダムに選択したノードを n_1，n_2 とする．

Step.1　sprig の進化
Step.2　決定木構成子集団から親を選択
Step.3　子を生成
Step.4　子の解候補決定木を評価
Step.5　sprig の評価
Step.6　次世代に残す個体を選択

図 6　一世代の処理手順

表 1　各検討事項に関するシステム

		sprig 集団数	
		1	属性数 (A)
接続する属性ノードの属性指定	なし	CN-P1	CN-PA
	あり	CY-P1	CY-PA

表 2　パラメータ

パラメータ	値
突然変異確率 p_m	0.01
sprig 集団の個体数 N_{sp}	400
決定木構成子集団の個体数 N_{tp}	1000
世代交代回数 N_g	50000
木の高さの上限値 H	10
sprig の子ノード数の上限値 M	5

C_1：　P_1 の n_1 以下を P_2 の n_2 以下の部分木で置換した木

C_2：　P_2 の n_2 以下を P_1 の n_1 以下の部分木で置換した木

C_3：　P_1 の各ノードに sprig の遺伝子を反映した木

C_4：　P_2 の各ノードに sprig の遺伝子を反映した木

C_1，C_2 は交叉により生成された木であり，C_3，C_4 は解候補決定木の各部分木と参照リンク元の sprig の表す木が同じになるよう変更を加えた木である．変更処理は根ノードから順に行なう．子ノードの減少，あるいは非終端ノードから終端ノードへの変更が発生した場合には不要な部分木を削除する．子ノードの増加，あるいは終端ノードから非終端ノードへの変更が発生した場合には新たに部分木を追加する．

子の生成後，C_1〜C_4 の全非終端ノードに対して確率 p_m で突然変異を発生させる．突然変異では，当該ノードの参照リンク元の sprig を変更し，以下の部分木を作り変える．P_1，P_2，C_1〜C_4 のうち，最良個体およびルーレットで選択された 2 個体を次世代に残す．

一世代の処理の流れを図 6 に示す．初期集団を生成した後，図 6 の処理を N_g 回繰り返し，最良個体の解候補決定木を SESAT2 の出力解とする．

3. 評価実験

2.3 節に記した検討事項の影響を調査するため，CY と CN，P1 と PA の各組み合わせについて表 1 のような 4 つの SESAT2 を構築し，評価実験を行なった．実験で設定したパラメータを表 2 に示す．

実験には UCI 機械学習リポジトリのデータ [12] を用いた．使用した 12 種類のデータの事例数，属性数，クラス数を表 3 に示す．属性値には，実数値，整数値，数値以外の値があるが，数値以外の属性値には 1 から順に整数の番号を割り当てた．low，middle，high など順序付けが可能な場合は，番号の大小と属性値の順序が一致するようにした．

表 3　実験用データ

データ名	事例数	属性数	クラス数
aust	690	14	2
balance	624	4	3
breast	683	9	2
bupa	345	6	2
glass	214	9	6
heart-c	297	13	2
iris	150	4	3
monks1	124	6	2
monks2	170	6	2
monks3	123	6	2
pima	768	8	2
post	87	8	3

表 4　訓練事例における平均適応度（括弧内は標準偏差）

データ	CN-P1	CY-P1	CN-PA	CY-PA
aust	85.0 (0.5)	85.1 (0.6)	84.8 (0.5)	84.9 (0.6)
balance	77.4 (0.5)	77.4 (0.5)	77.6 (0.6)	77.9 (0.6)
breast	94.9 (0.2)	94.9 (0.2)	95.0 (0.2)	94.9 (0.2)
bupa	66.5 (1.0)	66.6 (1.1)	66.9 (1.2)	67.0 (1.2)
glass	66.5 (1.6)	66.9 (1.6)	67.8 (1.7)	68.0 (2.0)
heart-c	81.8 (0.6)	82.2 (0.6)	82.4 (0.6)	83.1 (0.9)
iris	92.8 (0.4)	92.8 (0.4)	92.8 (0.4)	92.7 (0.4)
monks1	91.8 (0.8)	92.0 (0.1)	91.7 (1.2)	91.2 (2.0)
monks2	72.4 (2.2)	72.4 (3.0)	72.1 (2.6)	72.6 (2.4)
monks3	89.7 (0.7)	89.7 (0.6)	89.8 (0.6)	89.8 (0.6)
pima	75.1 (0.7)	75.2 (0.7)	75.3 (0.7)	75.3 (0.7)
post	75.1 (1.8)	75.1 (1.8)	75.1 (1.8)	75.1 (1.8)
全平均	80.7	80.9	80.9	81.1

4 つのシステムにおいて，表 3 に示した事例数のうち，10 分の 9 を訓練事例，10 分の 1 をテスト事例とする 10-fold クロスバリデーションを 10 回繰り返した．このとき，決定木学習の達成度と所要時間，予測正解率，および生成された木の簡素さを比較する．

各データの訓練事例で学習を行なった際，最良個体の適応度の平均は表 4 のようになった．どのデータにおいても各システムの平均適応度に大差が見られないことから，式 2 で定義された適応度が最も高くなる決定木の探索には，学習対象項目の変化は影響しないといえる．

表 5 はテスト事例における平均正解率と生成された決定木の平均ノード数である．平均正解率を全データの平均で比較すると，最大で 0.7%の差が見られた．一方，平均ノード数の全平均は CN-P1，CY-P1，CN-PA，CY-PA の順で小さくなり，CN-P1 のノード数は CY-PA の 81%程度となった．

学習に要した平均時間の一部と全平均を表 6 に示す．各システムは同一のマシン環境において実行されており，同数の sprig と決定木構成子を保持している．この結果，CY-P1 と CY-PA で学習時間が長く，CN-P1 は CY-P1 の 70%程度の時間で学習が行なえることが示された．

表 5　テスト事例における平均正解率と平均ノード数（括弧内は標準偏差）

データ	CN-P1		CY-P1		CN-PA		CY-PA	
	正解率 [%]	ノード数	正解率 [%]	ノード数	正解率 [%]	ノード数	正解率 [%]	ノード数
aust	84.9 (4.1)	16.4 (3.3)	84.7 (4.2)	18.2 (3.7)	84.9 (4.0)	16.1 (3.6)	84.8 (3.6)	18.8 (6.6)
balance	78.7 (4.9)	26.5 (5.1)	78.4 (5.1)	27.0 (4.6)	78.8 (5.1)	28.5 (5.8)	78.6 (4.9)	30.9 (5.8)
breast	95.9 (2.3)	16.1 (2.9)	95.8 (2.2)	17.4 (3.1)	95.9 (2.0)	18.2 (3.3)	95.7 (2.1)	18.2 (4.2)
bupa	63.1 (8.0)	18.4 (3.8)	63.0 (9.2)	19.2 (4.3)	62.9 (9.4)	20.9 (4.0)	62.1 (8.7)	23.5 (4.5)
glass	63.1 (11.6)	20.0 (3.0)	63.3 (10.5)	21.3 (2.9)	64.1 (11.2)	22.2 (3.5)	62.7 (13.9)	26.4 (4.0)
heart-c	77.4 (7.6)	21.6 (3.5)	76.3 (8.0)	24.8 (4.3)	77.2 (7.6)	25.8 (4.3)	77.3 (7.9)	34.3 (6.3)
iris	95.6 (5.1)	6.9 (1.7)	94.4 (7.3)	7.2 (1.8)	94.6 (7.1)	7.4 (2.0)	94.1 (7.7)	7.0 (1.8)
monks1	99.6 (2.5)	14.3 (0.8)	100.0 (0.0)	14.4 (0.6)	99.5 (2.3)	14.2 (1.0)	98.6 (4.6)	14.3 (1.5)
monks2	67.6 (12.8)	21.5 (4.8)	67.1 (12.1)	21.9 (6.2)	65.7 (13.1)	22.4 (5.2)	66.3 (12.7)	25.1 (5.8)
monks3	92.5 (5.2)	7.0 (2.9)	92.0 (5.3)	7.4 (3.6)	90.9 (5.8)	8.0 (4.1)	91.6 (5.2)	7.5 (3.9)
pima	73.3 (6.4)	20.3 (4.1)	73.1 (6.6)	21.9 (3.7)	72.6 (6.7)	23.0 (4.1)	73.1 (6.6)	28.8 (5.8)
post	72.3 (17.7)	9.8 (2.3)	72.1 (17.4)	9.9 (2.5)	70.9 (17.3)	10.0 (2.4)	70.6 (16.8)	10.9 (3.7)
全平均	80.3	16.6	80.0	17.5	79.8	18.1	79.6	20.5

表 6　平均学習時間 [秒]

データ	CN-P1	CY-P1	CN-PA	CY-PA
aust	15.9	29.7	26.5	29.2
breast	15.8	27.7	26.5	27.6
bupa	11.2	19.1	17.7	18.8
全平均	14.5	20.6	18.7	20.2

4. 考　　察

最初に CY と CN の違いに着目する．表 5 の全平均では，CN と CY の予測正解率の差は 0.25%であり，CN のノード数は CY の約 91%となっている．CY では，CN における決定木構成子集団の学習対象項目の一部を sprig 集団に移行している．sprig 集団で決定木の構造をより細部まで学習し，それが決定木構成子集団で学習される結合関係を制限しているため，CY は CN ほど簡素な決定木を生成できなかったものと思われる．

次に P1 と PA の結果を比較すると，予測正解率の差は 0.45%であり，P1 のノード数は PA の約 88%となっている．PA では，各 sprig 集団における学習対象項目は P1 よりも少なく，遺伝操作は同じ属性遺伝子を持つ個体同士に対して施されるので，特定の属性に関する項目 (b)〜(d) をより緻密に学習できる．しかし，属性遺伝子の異なる個体を共存させる P1 に比べて sprig 集団内の個体の多様性が少なく，その結果決定木構成子集団の多様性も減少するため，PA による決定木は P1 と比べてノード数が多いと考えられる．

学習時間に関しては，CY と CN の違いによる影響が大きい．CY ではクラスノードに接続可能な sprig が限られているため，解候補決定木の生成や遺伝操作に時間がかかったと推測される．CN と P1 を組み合わせることで学習時間がより短縮され，CY-P1 の 70%程度の時間で学習が行なえることから，CN-P1 は学習時間短縮に効果的といえる．

以上より，学習対象項目は訓練適応度と予測正解率に大きな影響を与えないが，簡素さと学習時間を左右する要因と考えられる．予測正解率と簡素さだけでなく学習時間をも考慮すると，SESAT2 では CN-P1 が最も有用である．

5. おわりに

本稿では，未知事例を正確に分類できる簡素な決定木の生成を目的として，決定木生成のための共生進化に有効な部分解の遺伝子表現と集団構成について検討した．UCI リポジトリのデータを用いて 4 種類の SESAT2 を評価したところ，学習対象項目が訓練適応度と予測正解率に与える影響は少ないが，簡素さと学習時間を左右するという結果が得られた．予測正解率と簡素さだけでなく学習時間をも考慮すると，sprig のクラス遺伝子で下位属性を指定せず，部分解集団として多様な sprig が混在する 1 集団を保持する方が，SESAT2 の遺伝子表現および集団構成として有効であることが示された．

文　　献

[1] J. Quinlan: "Induction of decision trees", Machine Learning, **1**, 1, pp. 139–159 (1986).

[2] L. Breiman, J. Friedman, R. Olshen and C. Stone: "Classification and Regression Trees", Wadsworth & Brooks (1984).

[3] J. Quinlan: "C4.5: Programs for Machine Learning", Morgan Kaufmann (1993).

[4] D. Goldberg: "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley (1989).

[5] D. Moriarty and R. Miikkulainen: "Efficient learning from delayed rewards through symbiotic evolution", Proc. of the 12th International Conference on Machine Learning, pp. 396–404 (1995).

[6] D. Moriarty and R. Miikkulainen: "Efficient reinforcement learning through symbiotic evolution", Machine Learning, **22**, pp. 11–32 (1996).

[7] D. Moriarty and R. Miikkulainen: "Hierarchical evolution of neural networks", Proc. of IEEE World Congress on Computational Intelligence, pp. 428–433 (1998).

[8] オスカル, 沼尾, 志村："共存進化における決定木の学習に関する研究", 第 11 回人工知能学会全国大会予稿集, pp. 171–174 (1997).

[9] 中山, 大谷, 志村："共生進化に基づく決定木の最適化に関する研究", 第 14 回人工知能学会全国大会予稿集, pp. 327–328 (2000).

[10] 大谷, 志村："コミュニティ指向共生進化による決定木の生成", 第 15 回人工知能学会全国大会予稿集, 3D2-01 (2001).

[11] 佐藤, 小野, 小林："遺伝的アルゴリズムにおける世代交代モデルの提案と評価", 人工知能学会誌, **12**, 5, pp. 734–744 (1997).

[12] http://www.ics.uci.edu/~mlearn/MLRepository.html.