



# Convergence of a hill-climbing genetic algorithm for graph matching

Andrew D.J. Cross, Richard Myers, Edwin R. Hancock\*

*Department of Computer Science, University of York, York YO1 5DD, UK*

Received 10 December 1998; received in revised form 12 July 1999; accepted 12 July 1999

---

## Abstract

This paper presents a convergence analysis for the problem of consistent labelling using genetic search. The work builds on a recent empirical study of graph matching where we showed that a Bayesian consistency measure could be efficiently optimised using a hybrid genetic search procedure which incorporated a hill-climbing step. In the present study we return to the algorithm and provide some theoretical justification for its observed convergence behaviour. The novelty of the analysis is to demonstrate analytically that the hill-climbing step significantly accelerates convergence, and that the convergence rate is polynomial in the size of the node-set of the graphs being matched. © 2000 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

*Keywords:* Genetic algorithms; Graph matching; Convergence analysis; Consistent labelling; Hybrid genetic algorithm; Bayesian consistency measure

---

## 1. Introduction

Configurational optimisation problems permeate all fields of machine intelligence. Broadly speaking, they are concerned with assigning symbolic or discretely defined variables to sites organised on a regular or irregular network in such a way as to satisfy certain hard constraints governing the structure of the final solution. The problem has been studied for over three decades. Concrete examples include the travelling salesman [1] and N-queens problems [2] together with a variety of network labelling [3,4] and graph matching [5] or graph colouring problems. The search for consistent solutions has been addressed using a number of computational techniques. Early examples from the artificial intelligence literature include Mackworth's constraint networks [3,4], Waltz's use of discrete relaxation to locate consistent interpretations of line drawings [6], Shapiro and Haralick's use of forward-checking and backtracking to

solve the consistent labelling problem [7], together with a host of applications involving the A\* algorithm [8,9]. More recently, the quest for effective search strategies has widened to include algorithms which offer improved global convergence properties. Examples include the use of simulated annealing [1,10,11], mean-field annealing [12], tabu-search [13–16] and most recently genetic search [17].

Despite stimulating a large number of application studies in the machine intelligence literature, the convergence properties of these modern global optimisation methods are generally less well understood than their classical counterparts. For instance, in the case of genetic search, although there has been considerable effort directed at understanding the convergence for infinite populations of linear chromosomes [18,19], little attention has been directed towards understanding the performance of the algorithm for discrete entities organised on a network structure. However, in a recent study, we have taken some first steps in the analysis of genetic algorithms for consistent labelling. There are two contributions of our earlier work which merit further discussion. First, we have provided a factor analysis of the significance of the different parameters required for

---

\* Corresponding author. Tel.: + 44-1904-43-2767; fax: + 44-1904-43-2767.

E-mail address: [erh@minster.cs.york.ac.uk](mailto:erh@minster.cs.york.ac.uk) (E.R. Hancock).

convergence [20]. In a second study, we have performed an empirical investigation of genetic search for the graph-matching problem [21].

Our motivation for embarking on this study of genetic algorithms for consistent labelling was previous work in which we had developed a Bayesian framework for gauging relational consistency [22]. We have not only used the new theory to show how the underlying consistency model accounts for a number of associative memory architectures [23,24], but have also developed some highly effective algorithms for inexact graph matching [25–27] and correcting structural errors in networks [27,28]. Our initial aim was to consider how the Bayesian consistency measure [22] could be used to assess fitness and how the process of correcting graph errors via edit operations [27] could be mapped onto the architecture of genetic search [21]. The main conclusions of our study were threefold. First, the consistent labelling of graphs was only amenable to genetic search if a hill-climbing operator was incorporated. Second, the quality of the final solution was greatly improved if cross-over (or genetic recombination) was conducted by exchanging connected subgraphs. Finally, we found the optimisation process to be relatively insensitive to the choice of mutation rate.

Unfortunately, our analysis of the empirical results has hitherto been extremely limited and has been couched only in terms of a rather qualitative model of the pattern space in which configurational optimisation is performed [21]. This has meant that we have been unable either to predict the convergence behaviour or to account for the three interesting empirical properties listed above. The aim in this paper is to remedy this shortcoming by presenting a detailed analysis of algorithm behaviour. It is important to stress that although there have been several analyses of genetic search, these differ from the study described here in three important ways. First, we are concerned specifically with the graph-matching problem. This means that we present an analysis that is more pertinent to the consistent labelling problem where there is network organisation rather than a linear chromosome. Second, we pose our analysis in terms of discrete assignment variables rather than continuous ones. Finally, we deal with the infinite population size assumption in a critical manner.

## 2. Paper outline

Stated succinctly, the aim of this paper is to provide an analysis of the convergence of the genetic optimisation procedure when applied to the problem of graph matching [5,29–31]. We commence by investigating each of the genetic operators in turn. Once we have understood the behaviour of the operators in isolation, we turn our attention to predicting their collective behaviour. In or-

der to demonstrate the validity of our theoretical predictions, we compare them with a Monte-Carlo study.

It is important at the outset to point out that the analysis presented in this paper commences from the same basic standpoint as the work of Qi and Palmieri [18,19]. However, there are two principal differences between their analysis and the one presented here. First, their model of the genetic algorithm operates in a continuous space. While this renders the analysis algebraically tractable, it represents a simplistic model of discretely defined configurational optimisation which is unrealistic. Second, their analysis makes non-specific assumptions concerning the form of the fitness function. In consequence, it is of limited use in understanding the graph-matching or consistent labelling problem.

By contrast, we provide an analysis which is more specific to the Bayesian framework developed by Wilson and Hancock [26] for consistent labelling problems. It is the Bayesian consistency measure developed in the work that has been explored in an empirical manner in the evolutionary search procedure developed by Cross and Hancock [21]. The new analysis presented here not only allows us to develop quantitative predictions of overall population behaviour, it also allows us to attempt a realistic analysis of the algorithm time complexity. One of the main conclusions of our empirical study of genetic search [21] was that the addition of a hill-climbing operator can yield significant improvements in both the convergence rate and solution quality. For this reason, we will supplement our analysis of the traditional operators with a study of the hybrid hill-climbing algorithm. This is one of the novel contributions of the paper.

While the results obtained for the mutation and cross-over operators are relatively generic, the analysis of the selection and hill-climbing operators require a more detailed model of the problem at hand. It is here that our analysis becomes problem-specific to graph matching.

The paper outline is as follows. In Section 3, we briefly review the fitness measure that underpins our graph-matching algorithm. Section 4 details the main stages of the genetic search procedure. These two sections are effectively a synopsis of our recent paper [21] which reports the details of the evolutionary graph-matching technique. In Section 5, we commence our modelling of the distribution of the Bayesian consistency measure which fulfils the role of fitness in our graph-matching technique. Section 6 exploits this distribution-model to predict the evolution of fitness under each of the genetic operators, i.e. mutation, selection, cross-over and hill-climbing. In Section 7, we use the individual operator characteristics to understand the iterative behaviour of the combined hill-climbing genetic operator. This section also comments on the validity of our analysis. Section 8 provides some illustrative examples of the graph-matching method for real-world images. Finally, Section 9 offers some conclusions.

### 3. Relational graphs

Central to this paper is the aim of matching relational graphs represented in terms of configurations of symbolic labels. We represent such a graph by  $G = (V, E)$ , where  $V$  is the symbolic label-set assigned to the set of nodes and  $E$  is the set of edges between the nodes. Formally, we represent the matching of the nodes in the data graph  $G_1 = (V_1, E_1)$  against those in the model graph  $G_2 = (V_2, E_2)$  by the function  $f: V_1 \rightarrow V_2$ . In other words, the current state of match is denoted by the set of Cartesian pairs constituting the function  $f$ .

In order to describe local interactions between the nodes at a manageable level, we will represent the graphs in terms of their clique structure. The clique associated with the node indexed  $j$  consists of those nodes that are connected by an edge of the graph, i.e.  $C_j = j \cup \{i \in V_1 | (i, j) \in E_1\}$ . The labelling or mapping of this clique onto the nodes of the graph  $G_2$  is denoted by  $\Gamma_j = \{f(i) \in V_2, \forall i \in C_j\}$ . Suppose that we have access to a set of patterns that represent feasible relational mappings between the cliques of graph  $G_1$  and those of graph  $G_2$ . Typically, these relational mappings would be configurations of consistent clique labellings which we want to recover from an initial inconsistent state of the matched graph  $G_1$ . Assume that there are  $Z_j$  relational mappings for the clique  $C_j$  which we denote by  $\Lambda^\mu = \{\lambda_i^\mu \in V_2, \forall i \in C_j\}$  where  $\mu \in \{1, 2, \dots, Z_j\}$  is a pattern index. According to this notation  $\lambda_i^\mu \in V_2$  is the match onto graph  $G_2$  assigned to the node  $i \in V_1$  of graph  $G_1$  by the  $\mu$ th relational mapping. The complete set of legal relational mappings for the clique  $C_j$  are stored in a *dictionary* which we denote by  $\Theta_j = \{\Lambda^\mu | \mu = 1, 2, \dots, Z_j\}$ .

The discrete relaxation procedure is based on maximising the joint probability of the matched label configuration, i.e.  $P(\Gamma_j)$ . It is therefore necessary to find a way of enumerating  $P(\Gamma_j)$  when the label configuration is highly inconsistent. The approach is to adopt a Bayesian viewpoint in which it is assumed that only consistent labellings in the dictionary are legal and have uniform non-zero a priori probabilities of occurrence, i.e.  $P(\Lambda^\mu) = Z_j^{-1}$ . Other configurations do not occur a priori but are the corrupted realisations of the dictionary items. This idea is realised by applying the axiomatic property of joint probability to expand  $P(\Gamma_j)$  over the space of consistent configurations

$$P(\Gamma_j) = \sum_{\mu=1}^{Z_j} P(\Gamma_j | \Lambda^\mu) P(\Lambda^\mu). \quad (1)$$

Further development of a useful objective function for discrete relaxation requires a model of the label corruption process, that is of the conditional probabilities of the potentially inconsistent configurations given each of the  $Z_j$  feasible relational mappings  $P(\Gamma_j | \Lambda^\mu)$ . We adopt a very simple viewpoint; matching errors are assumed to be memoryless and to occur with uniform probability.

The first consequence of the assumed absence of memory is that the errors are independent. As a result we can factorize the conditional probabilities over the individual nodes in the graph, i.e.

$$P(\Gamma_j | \Lambda^\mu) = \prod_{i \in C_j} P(f(i) | \lambda_i^\mu) \quad (2)$$

Our next step is to propose a model for the label corruption mechanism at each node in the graph. Again, taking recourse to the memoryless assumption, we take the probability of label errors on individual objects to be independent of the class of label. If  $P_e$  is the label error probability, then the distribution function for the label confusion probabilities is

$$P(f(i) | \lambda_i^\mu) = \begin{cases} 1 - P_e & \text{if } f(i) = \lambda_i^\mu, \\ P_e & \text{otherwise.} \end{cases} \quad (3)$$

As a result of this distribution rule, the conditional matching probabilities depend on the Hamming distance  $H(\Gamma_j, \Lambda^\mu)$  between the matched configuration  $\Gamma_j$  and the individual dictionary items  $\Lambda^\mu$ , i.e.

$$P(\Gamma_j | \Lambda^\mu) = (1 - P_e)^{|C_j| - H(\Gamma_j, \Lambda^\mu)} P_e^{H(\Gamma_j, \Lambda^\mu)}, \quad (4)$$

where the Hamming distance  $H(\Gamma_j, \Lambda^\mu)$  is defined using the Kronecker delta function to be  $H(\Gamma_j, \Lambda^\mu) = \sum_{i \in C_j} (1 - \delta_{f(i), \lambda_i^\mu})$ . The model components given in Eqs. (2)–(4) naturally lead to the following expression for  $P(\Gamma_j)$  in terms of the set of Hamming distances to the consistent labellings residing in the dictionary

$$P(\Gamma_j) = \frac{b}{Z_j} \sum_{\mu=1}^{Z_j} \exp[-k_e H(\Gamma_j, \Lambda^\mu)] \quad (5)$$

where  $b = (1 - P_e)^{|C_j|}$  and  $k_e = \ln(1 - P_e)/P_e$ . According to our simple model of label errors Hamming distance is the basic measure of consistency. Systematic softening of the constraints residing in the dictionary is controlled by the parameter  $P_e$ .

The configurational probability  $P(\Gamma_j)$  is the basic ingredient of our genetic search procedure. It represents the probability of a particular matching configuration evaluated over the state space of feasible possibilities (i.e. the dictionary). We use as our global measure of consistency the average of the clique configurational probabilities, i.e.

$$P_G = \frac{1}{|V_1|} \sum_{j \in V_1} P(\Gamma_j). \quad (6)$$

In the next section of this paper we will describe how this average consistency measure can be utilised as a fitness measure in the genetic search for relational matches.

### 4. Genetic search

In a recent paper [21], we showed that a hill-climbing genetic search procedure provides a very natural way of

locating the global optimum of the global consistency measure described in the previous section. In essence, the approach relies on generating a population of random initial matching configurations. These undergo cross-over, mutation and selection to locate the match that optimises the Bayesian consistency measure defined in the previous section of this paper. The main stages of the algorithm are outlined below and more detailed discussion can be found in Ref. [21].

#### 4.1. Initial population generation

The idea underpinning genetic search is to maintain a population of alternative solution vectors and to refine this population using various evolutionary operators. To distinguish the different solutions we use a population index  $\alpha$ . We let  $f^{(\alpha)}(i)$  denote the match assigned to the node  $i$  in the data-graph by the  $\alpha$ th solution in the current population. The fitness associated with the solution indexed  $\alpha$  is denoted by  $P_G^\alpha$ .

In order to initialise the algorithm we randomly assign matches. In other words, our initial solution vectors are random configurations of labels drawn from the model graph.

#### 4.2. Cross-over

Cross-over is the process which mixes the pool of solutions to produce new ones. If effectively controlled, the process can be used to combine pairs of suboptimal or partially consistent matches to produce one of improved global consistency. Typically, deterministic updating of the match will propagate constraints only over the distance of one neighbourhood with each iteration. Cross-over can accelerate this process by combining disconnected yet internally consistent subgraphs from the individual solutions in the pool.

The standard cross-over procedure involves selecting at random pairs of global matching configurations from the current population. Matches at the corresponding sites at randomly chosen locations in the two graphs are then interchanged with uniform probability  $\frac{1}{2}$ . However, this uniform cross-over mechanism will not necessarily facilitate the merging of locally consistent subgraphs. Moreover, the process also ignores the underlying structure of the graphs. A better strategy is to combine the solutions by physically dividing the graphs into two disjoint subgraphs. In this way internally consistent portions of the individual solutions may be exchanged at the structural level.

#### 4.3. Mutation

A further randomisation stage is applied to the individual matches to introduce new information into the

population of global matches through a process of mutation. This is effected by randomly re-assigning the matches on individual sites. The probability of re-assignment is uniform across the sites. In other words, we randomly re-assign a fixed fraction, the matches in  $f^\alpha(i)$ , with random labels selected from the set  $V_2$ .

#### 4.4. Hill-climbing

The aim in performing hill-climbing operations is to restore consistency to graphs modified by the cross-over and mutation operations. Although this can be effected by stochastic means, this is time consuming. The hill-climbing stage involves iteratively reconfiguring the graphs modified by cross-over or mutation to maximise the value of  $P_G^\alpha$ . Formally, this corresponds to a parallel iterative application of the following decision rule:

$$f^{(\alpha)}(i) = \arg \max_{V_2} P_G^\alpha. \quad (7)$$

This application of the rule has the effect of locating the nearest local optima of the global consistency measure. It therefore redistributes the population of solutions to reside at the modes of this fitness measure. Suboptimal modes become increasingly unlikely as they are removed from the population by the stochastic selection operations. This process not only accelerates convergence, but also diminishes the requirement for a large population of graphs.

#### 4.5. Selection

The final stochastic element of genetic search is the selection process. The aim here is to randomly admit the configurations refined by the hill-climbing process to the population on the basis of their fitness measure. The probability distribution defined in Eq. (5) lends itself naturally to the definition of a population membership probability. Suppose that  $P_G^{(\alpha)}$  denotes the global configurational probability for the  $\alpha$ th member of the pool (population) of graphs. By normalising the sum of clique configuration probabilities over the population of matches, the probability for randomly admitting the  $\alpha$ th solution to the pool of graphs  $\mathcal{P}$  is

$$P_s^\alpha = \frac{P_G^{(\alpha)}}{\sum_{\alpha' \in \mathcal{P}} P_G^{(\alpha')}}. \quad (8)$$

#### 4.6. Empirical findings

Based on an empirical study of the resulting graph matching algorithm, we reached the following conclusions concerning its convergence behaviour:

- The method was relatively insensitive to mutation rate. In fact, provided that the mutation probability did not

exceed 0.600, then the number of iterations required for convergence was approximately constant.

- The addition of the hill-climbing step considerably reduced the number of iterations required for convergence.
- Once the population size exceeded a critical value, then the convergence rate was essentially independent of population size.
- The number of iterations required for convergence was approximately polynomial in the number of graph nodes.

The aim in the remainder of this paper is to provide an analysis which supports these empirical findings.

### 5. Distribution analysis

We formulate our investigation of graph matching as a discrete-time process with the states defined over the state space of all possible correspondences between a pair of graphs. Our analysis of the population is a statistical one, in which we assume that the population is sufficiently large that we can invoke the central limit theorem. For this reason we direct our attention to the modelling of the probability density function for the distribution of solution vectors.

#### 5.1. Formal ingredients of the model

For the problem of graph matching, each solution vector  $f^{(\alpha)} : V_1 \rightarrow V_2$  represents the labelling of the nodes of a data graph  $V_1$  with the nodes of a model graph  $V_2$ . In order to simplify the analysis, we will assume that the pair of graphs have an identical number of nodes, i.e.  $|V| = |V_1| = |V_2|$ .

To commence our modelling of the distribution of solution vectors, we focus our attention on the fraction of mappings that are in agreement with known ground truth. If the configuration of ground-truth correspondence matches is denoted by  $\tilde{f}$ , then the fraction of correctly assigned matches for the solution vector indexed  $\alpha$  is equal to

$$F_\alpha^{(n)} = \frac{1}{|V|} \sum_{i \in V} \delta_{f^{(\alpha)}(i), \tilde{f}(i)}, \tag{9}$$

where  $\alpha$  is a population index of the solution vector,  $n$  is the iteration number and  $\delta$  is the Kronecker delta function. A solution vector  $f^{(\alpha)}$  in which each of the matches is correct would have  $F_\alpha^{(n)} = 1$ . By contrast, a solution vector in which none of the correspondence matches are correct would have  $F_\alpha^{(n)} = 0$ . In order to analyse how the genetic graph-matching process performs, we wish to evaluate the distribution of  $F_\alpha^{(n)}$  over the entire population of candidate solution vectors. At

iteration  $n$ , we denote the distribution of fractional matching error by  $P_D^{(n)}(F = \gamma)$ .

The overall goal of our analysis is to model how the distribution of the fraction of correct matches evolves with iteration number. For reasons of tractability, we largely confine our attention to understanding how the mean fraction of correct matches with iteration number. The quantity of interest is

$$\langle F_\alpha^{(n)} \rangle = \frac{1}{|\mathcal{P}|} \sum_{\alpha \in \mathcal{P}} F_\alpha^{(n)} = \int_\gamma \gamma P_D^{(n)}(F = \gamma) d\gamma. \tag{10}$$

Since the mutation and cross-over operators do not draw upon the fitness of the individuals in the population, it follows that we can model their effect on the distribution of correct matches without reference to the specific nature of our fitness measure. In contrast, the selection operator draws upon our measure of relational consistency in order to determine the probability that each individual belonging to the population survives into the next generation. For graph matching, it is clear that the fitness measure is not related in a monotonic manner to the fraction of correct matches. In other words, there is no one-to-one relationship between  $P_G$  and  $F_\alpha^{(n)}$ . It is for this reason that we must turn to genetic search rather than hill-climbing gradient ascent as a means of optimisation. However, since the fitness measure draws on Hamming distance between the super-cliques as a means of gauging relational consistency via Eqs. (5) and (6), we would expect that  $P_G$  becomes small when  $F_\alpha^{(n)}$  approaches zero. Conversely,  $P_G$  will be large when  $F_\alpha^{(n)}$  approaches unity. In other words, extreme survival probabilities should correspond to the extremes of the distribution of correct matches.

Most attempts to analyse the convergence properties of genetic algorithms [18,19] have relied on the assumption that the population size is infinite. Although this assumption is made for reasons of computational tractability, it is rarely possible to realise it in practice. In fact, such an assumption corresponds to taking the case in which the entire search space is populated by candidate solutions. This is clearly at odds with the spirit of genetic search, where it is the role of the evolutionary operators to advantageously position a small number of distinct individuals. Moreover, if the whole sample space were filled with solutions, then global optimisation could be trivially performed by exhaustive selection of the fittest individual. However, while we criticise the infinite population size assumption on the grounds of realism, we do not wish to detract from its importance in understanding the convergence properties of genetic search. One of the important features of our analysis is that even at relatively small population sizes, our results predict the convergence behaviour accurately. It is interesting to note that in their analysis Qi and Palmieri [18,19] show that the finite population density function approaches the infinite

population limit with a fractional error that is proportional to  $1/\sqrt{|V|}$ .

We commence our analysis by assuming that the number of correct matches in the initial population follows a binomial distribution. If the initial population is chosen in a random fashion, then at the outset the expected fraction of correct matches is given by

$$\langle F^{(0)} \rangle = \frac{1}{|V|} \tag{11}$$

Under the binomial assumption, the number of correct matches has mean  $F^{(0)}|V|$  and standard deviation  $\sqrt{|V|F^{(0)}(1 - F^{(0)})}$ . As a result the initial number of correct matches  $\tau$  is distributed in the following manner:

$$P(\tau) = \frac{(|V|)!}{\tau!(|V| - \tau)!} (\langle F^{(0)} \rangle)^\tau (1 - \langle F^{(0)} \rangle)^{|V| - \tau}. \tag{12}$$

Since we are interested in the fraction of correct matches, we turn our attention to the distribution of the random variable  $\gamma = \tau/|V|$ . By appealing to the central limit theorem under the assumption that the graphs have large numbers of nodes, we can replace the binomial distribution of the number of correct matches by a Gaussian distribution of the fraction of correct matches. In practice, we deal with graphs whose size exceeds 40 nodes, and so this approximation will be a faithful one. The probability distribution for the fraction of correct solutions in the population is therefore

$$P_D^{(0)}(F^{(n)} = \gamma) \approx \frac{1}{\sqrt{2\pi|V|\langle F^{(0)} \rangle(1 - \langle F^{(0)} \rangle)}} \times \exp\left(-\frac{(|V|\gamma - |V|\langle F^{(0)} \rangle)^2}{2|V|\langle F^{(0)} \rangle(1 - \langle F^{(0)} \rangle)}\right). \tag{13}$$

Because the distribution is Gaussian, the mode is located at the position  $\gamma = |V|\langle F^{(0)} \rangle$ .

### 6. Genetic operators

In this section we will investigate the role that the three traditional genetic operators, i.e. mutation, cross-over and selection, play in the evolution of the average fraction of correct solutions in the genetic population. We will supplement this analysis with a discussion of the hill-climbing process. At this stage, our interest lies not with the prediction of the collective behaviour of the operators, but with the effect that each one has in isolation upon the population. Collective behaviour is the topic of Section 7.

#### 6.1. The mutation operator

The goal of the mutation operator is to increase population diversity by performing a stochastic state swap-

ping process. This is effected by randomly re-assigning on the individual node matches. The process is applied to each of the different solutions that constitute the current population. This process proceeds independently for both the individual nodes and the individual solutions. This is in contrast with the cross-over which serves to exchange information between pairs of solutions in order to form new individuals that are admitted to the population on the basis of their fitness. The uniform probability of any match undergoing a random state swapping process is  $P_m$ . For each individual solution vector, there are three possible transitions that can occur in the state of match. First, an individual mutation could increase the number of correct matches by one unit; in this case the increase in the fraction of correct matches is  $P_m(1 - F_x^{(n)})(1/|V|)$ . The second possible outcome is a reduction in the number of correctly assigned matches by one unit; in this case the decrease in the fraction of correct matches is  $P_m F_x^{(n)}(|V| - 1)/|V|$ . Finally, the mutation could leave the number of correct correspondences unchanged; in this case the fraction of correct matches remains unchanged at the value  $F_x^{(n)}$ . For moderate mutation rates, the most likely change to the fraction of matches is due to the second transition. This corresponds to a disruption of the set of correctly assigned matches.

We are interested in the effect that the mutation operator has upon a solution vector in which the fraction of correct matches is  $F_x^{(n)}$  at iteration  $n$ . In particular, we would like to compute the average value of the fraction of correct matches at iteration  $n + 1$ . Based on the three assignment transitions outlined above, the new average fraction of correct matches is

$$\langle F_x^{(n+1)} \rangle = F_x^{(n)} + P_m(1 - F_x^{(n)})\frac{1}{|V|} - P_m F_x^{(n)}\frac{|V| - 1}{|V|}. \tag{14}$$

After some straightforward algebra, we can re-write this recursion formula in terms of the fraction of matches correct at the outset, i.e.  $F_x^{(0)}$ . As a result, the average fraction of correct matches at iteration  $n$  is

$$\langle F_x^{(n)} \rangle = (1 - P_m)^n \left( F_x^{(0)} - \frac{1}{|V|} \right) + \frac{1}{|V|}. \tag{15}$$

This result depends on only the iteration number and the initial fraction of correct matches. We can make the exponential character of the formula clearer by rewriting it in terms of the natural exponential function. As a result

$$\langle F_x^{(n)} \rangle = \frac{1}{|V|} + \left( F_x^{(0)} - \frac{1}{|V|} \right) \exp(-k_m n), \tag{16}$$

where  $k_m = \ln(1/(1 - P_m))$ . There are a number of interesting features of this formula that deserve further comment. First, the equation represents an exponential decay that tends towards a minimum value of  $1/|V|$ , i.e. the probability of randomly assigning a correct match. The

rate of decay is determined by the logarithm of the probability that a mutation operation does not take place, i.e.  $1 - P_m$ . In qualitative terms, the mutation process represents an exponential drift towards highly unfit solutions. The rate of drift is controlled by two factors. The first of these is the mutation rate  $P_m$ . As the mutation probability increases, the disruptive effect of the operator becomes more pronounced. The second factor is the initial fraction of correct matches that were present prior to mutation. As this initial fraction of correct matches increases, then so does the disruptive effect of the mutation operator. The effect of this second drift process is to impose a higher rate of disruption on solutions in the population that are approaching a consistent state. Poor or highly inconsistent solutions, on the other hand, are not significantly affected. This latter drift effect can be viewed as a natural mechanism for escaping from local optima that may be encountered when the global solution is approached in a complex fitness landscape.

After a very large number of mutation processes, the population will approach a completely random state. In this state the number of correct matches will be governed by a binomial distribution in which the expected value of the fraction of correct matches is  $1/|V|$ . In other words, we expect any initial distribution of solutions to drift towards a binomial one in an exponential fashion.

To verify the validity of the result, we have conducted a Monte-Carlo simulation. We set the mutation probability to a value of  $\frac{1}{40}$  and plot the distribution  $P_D$  as a function of iteration number. The resulting plot is given in Fig. 1. We observe the predicted exponential decay. Specifically, the peak of the population distribution drifts towards the origin in the fashion predicted. Moreover, the width of the distribution becomes narrower as it approaches a binomial distribution at the origin. In order to quantify this process we plot the most probable fraction of correct matches in the population against

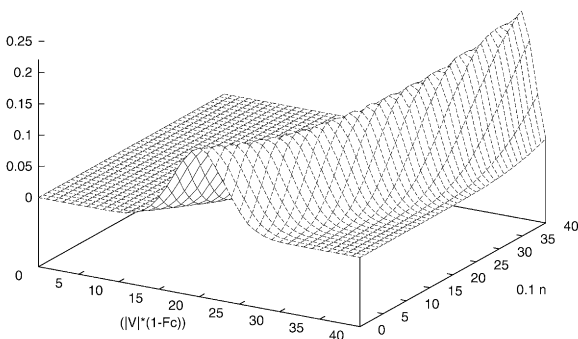


Fig. 1. A numerical simulation of the distribution of correct solutions in the population using only the mutation operator.

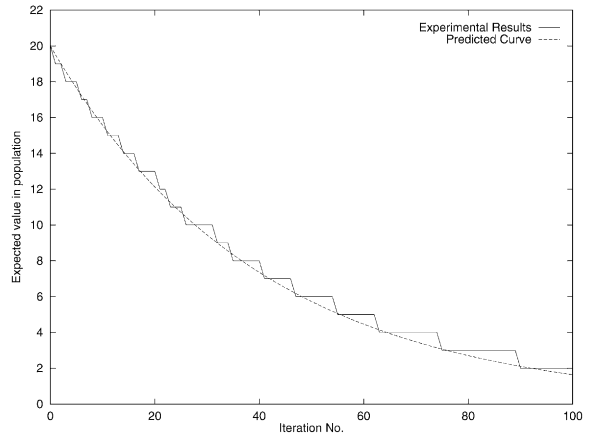


Fig. 2. The comparison of the analytic predictions of the mutation operator with the simulation run results.

iteration number. This plot is shown in Fig. 2. We note that there is a good agreement between our prediction of exponential decay and what is observed experimentally.

### 6.2. An analysis of the selection operator

In contrast with the mutation operator which is a uniform random process, selection draws upon the fitness function to determine the probability that the different solution vectors survive into the next population generation. Because of this task-specific nature of the fitness function, it is not possible to undertake a general analysis of the selection process. Moreover, in the case of graph matching the compound exponential structure of our fitness measure further complicates the analysis. To overcome this problem, we present an approximation to our Bayesian consistency measure which allows us to relate the survival probability to the fraction of correct matches. This approximate expression for the survival probability turns out to be polynomial in the fraction of correct matches.

We commence by writing the fitness using the expression for the super-clique matching probability given in Eq. (6). To make the role of error probability more explicit, we re-write the matching probability in terms of Kronecker delta functions that express the compatibility between the current matching assignments and the consistent matches demanded by the configuration residing in the dictionary. As a result

$$P(\Gamma_j) = \frac{1}{|\Theta_j|} \sum_{\Lambda^e \in \Theta_j} \prod_{i \in C_j} P_e^{(1 - \delta_{f(i), \Lambda^e})} (1 - P_e)^{\delta_{f(i), \Lambda^e}}. \quad (17)$$

Our aim is to compute the average value of the global consistency measure,  $P_G^z$ . Because the consistency function averages the matching probability  $P(\Gamma_j)$ , the

expected value of the global probability is equal to

$$P_G^z = E \left[ \frac{1}{|\Theta_j|} \sum_{\Lambda^e \in \Theta_j} \prod_{i \in C_j} P_e^{(1 - \delta_{f(i), \lambda_i^e})} (1 - P_e)^{\delta_{f(i), \lambda_i^e}} \right]. \quad (18)$$

We now note that the expected value of the exponential function under the product can be re-expressed in terms of the assignment probabilities in the following manner:

$$\begin{aligned} E[P_e^{(1 - \delta_{f(i), \lambda_i^e})} (1 - P_e)^{\delta_{f(i), \lambda_i^e}}] \\ = P_e P(f(i) \neq \lambda_i^e) + (1 - P_e) P(f(i) = \lambda_i^e). \end{aligned} \quad (19)$$

As a result, the expected value of the global matching probability, i.e. the probability of survival, is equal to

$$\begin{aligned} P_G^z = \frac{1}{|\Theta_j|} \sum_{\Lambda^e \in \Theta_j} \prod_{i \in C_j} \{ P_e P(f(i) \neq \lambda_i^e) \\ + (1 - P_e) P(f(i) = \lambda_i^e) \}. \end{aligned} \quad (20)$$

Unfortunately, this expression still contains reference to the dictionary of structure-preserving mappings. In order to further simplify matters, we observe that when the configuration of assigned matches becomes consistent then, provided the error probability  $P_e$  is small, we would expect the sum of exponentials appearing in Eq. (5) to be dominated by the single dictionary item that is fully congruent with the ground-truth match. The remaining dictionary items make a negligible contribution. Suppose that  $\hat{\mu}$  is the index of the correctly matching dictionary item, then we can write

$$\exp[-k_e H(\Gamma_j, \Lambda^{\hat{\mu}})] \gg \sum_{\Lambda^e \in \Theta_j - \Lambda^{\hat{\mu}}} \exp[-k_e H(\Gamma_j, \Lambda^e)]. \quad (21)$$

We can now approximate the super-clique matching probability by considering only the dominant dictionary item. This allows us to remove the summation over dictionary items. Finally, we note that the average value of the probability of correspondence match, i.e.  $P(f(i) = \lambda_i^e)$ , is simply equal to the fraction of correct matches  $F_x^{(n)}$ . By assuming that all super-cliques are of approximately the same average cardinality, denoted by  $|\hat{C}|$ , we can approximate the global probability of match in the following manner:

$$P_G^z = \frac{1}{|\Theta_j|} [P_e (1 - F_x^{(n)}) + (1 - P_e) F_x^{(n)}]^{|\hat{C}|}. \quad (22)$$

In other words, our measure of relational consistency is polynomial in the fraction of correct matches. Moreover, the order of the polynomial is equal to the average node connectivity  $|\hat{C}|$ . As the average neighbourhood size or node connectivity in the graphs increases, the discriminating power of the cost function becomes more pronounced.

The model developed in this section relates the Bayesian consistency measure to the fraction of correct

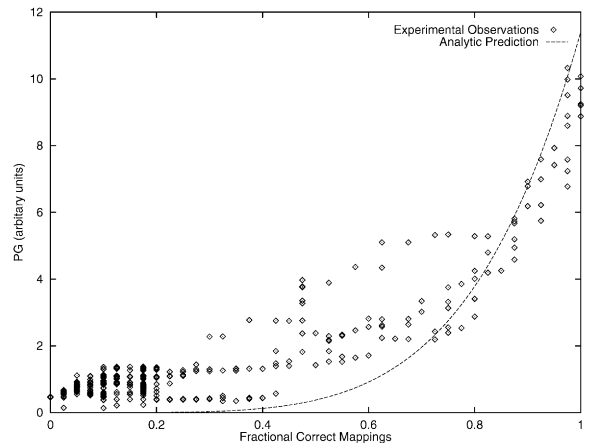


Fig. 3. A plot of the empirical values of our fitness measure as a function of the fraction of correct mappings. The dashed line represents our theoretical prediction, while the points are the collected data.

matches in the population. In order to verify the model, we have conducted the following experiment. We have generated a large number of random graphs. For each graph in turn, we have generated a set of random self-correspondences with varying degrees of error. For each set of correspondences, we have computed the Bayesian consistency measure. We have then compared the measured and predicted values of  $P_G^z$  for each value of the known fraction of correct matches. This plot is shown in Fig. 3 for randomly generated graphs containing 40 nodes. The graphs used in this experiment are Delaunay triangulations generated from random point sets. For the Delaunay triangulation, the average clique size or node connectivity is  $|\hat{C}| = 5.2$ . The predicted value agrees well with the true match probability  $P_G$  as the fractional error increases. At lower values, the disparity between our prediction and the true functional becomes more pronounced, although the general form of the curve is still reflected in the data.

Having derived an approximate relationship between the Bayesian measure of relational consistency and the fraction of correct matches, we recall that the survival probability for the solution vector indexed  $\alpha$  between generations  $n$  and  $n + 1$  under roulette wheel selection strategy [32] is given by

$$P_{Selection}^z = \frac{P_G^z}{\sum_{\beta \in \mathcal{P}} P_G^\beta}. \quad (23)$$

In order to model the selection process, we recast the roulette wheel probabilities in terms of the distribution of the fraction of correct matches. This involves taking the approximation for the matching probability and weighting according to the probability distribution for

the fraction of correct matches. The expression for the selection probability becomes

$$P_D^{(n+1)}(F^{(n)} = \gamma) = \frac{P_D^{(n)}(\gamma)P_G(\gamma)}{\int_0^1 P_D^{(n)}(\gamma')P_G(\gamma') d\gamma'} \quad (24)$$

The normalisation integral appearing in the denominator is not of prime interest to us since it depends only on iteration number and not upon the overall fraction of correct matches. Here we are concerned with the dependence on the fraction of correct matches. In order to proceed we observe that the normalisation factor serves only to guarantee that the total survival probability sums to unity. Stated alternatively, this means that the populations at generations  $n$  and  $n + 1$  are of the same size. Furthermore, if we confine our attention to the convergence properties of the mode of the distribution of the fraction of correct matches, we can neglect the normalization term and investigate the quantity

$$P_D^{(n+1)}(F^{(n+1)} = \gamma) = \Upsilon^{(n)}P_D^{(n)}(\gamma)P_G(\gamma), \quad (25)$$

where  $\Upsilon^{(n)}$  is an iteration-dependent normalization constant that is not a function of the fraction of matches correct. The recursive application of this equation yields the distribution of the fraction of correct matches at any generation in terms of the initial fraction. The relationship is a power law of the form

$$P_D^{(n)}(F^{(n+1)} = \gamma) = \alpha^{(n)}P_D^{(0)}(\gamma)P_G(\gamma)^n. \quad (26)$$

Substituting for the approximate initial distribution given in Eq. (13) together with our approximation to the cost function from Eq. (22), we find

$$P_D^{(n)}(F^{(n)} = \gamma) \approx \frac{1}{\sqrt{2\pi|V|F^{(0)}(1 - F^{(0)})}} \times \exp\left(-\frac{(|V|\gamma - |V|F^{(0)})^2}{2|V|F^{(0)}(1 - F^{(0)})}\right) \times (P_e(1 - \gamma) + (1 - P_e)\gamma)^{|\hat{C}|n}. \quad (27)$$

The required distribution is simply a Gaussian distribution that is modulated by a polynomial of order  $|\hat{C}|n$ . This demonstrates that the average fraction of correct matches in the population will tend to increase as the value of  $n$  increases. In other words, the iteration process improves the fraction of correct matches. By confining our attention to the solutions that occur most frequently in the population, we can track the iteration dependence of the mode or peak,  $F_{max}^{(n)}$ , of the distribution of correct matches. To locate the most frequently occurring solution in the population, we proceed as follows. First, we evaluate the derivative of the distribution function in Eq. (27) with respect to the fraction of correct mappings, i.e.  $\gamma$ . Next, we set the derivative equal to zero. By solving the resulting saddle-point equation for  $F^{(n)}$  and after rejecting the non-physical values of  $\gamma$  that fall outside the interval

$[0, 1]$ , we find that the maximum value  $P_D$  is located at the position

$$F_{max}^{(n)} = \frac{1}{2} \left( F^{(0)} + \frac{P_e}{\kappa_2} + \frac{\sqrt{\kappa^2(P_e^2 + 2P_eF^{(0)}\kappa_2 + F^{(0)}\kappa_2^2) + 2\kappa_1\kappa_2^2|\hat{C}|n}}{\kappa_1\kappa_2} \right), \quad (28)$$

where

$$\kappa_1 = \frac{(|V|)}{2F^{(0)}(1 - F^{(0)})} \quad (29)$$

and

$$\kappa_2 = 1 - 2P_e. \quad (30)$$

In Fig. 4 we plot the iteration dependence of the modal fraction of correct matches. The example shown in this plot has the label error probability set to  $P_e = 0.01$ , the graph size  $|V|$  is 40, and the average super-clique size  $|\hat{C}|$  is 5.5.

With this model of the iteration dependence of  $F_{max}^{(n)}$  under the selection operator to hand, we are in a position to compute the number of iterations required for algorithm convergence. Our convergence condition is that the modal fraction of correct matches is unity. We identify the value of the iteration index  $n$  that satisfies this condition by setting  $F_{max}^{(n)} = 1$  in Eq. (28). Furthermore, we assume that the initial population is randomly chosen and as a result  $F^{(0)} = 1/|V|$ . Solving for  $n$ , we find the number of iterations required for convergence to be equal to

$$n_{converge} = \frac{(|V|)^2(1 - P_e)}{|\hat{C}|(1 - 2P_e)}. \quad (31)$$

In other words, commencing from a simple model of the selection process that uses a number of domain-specific

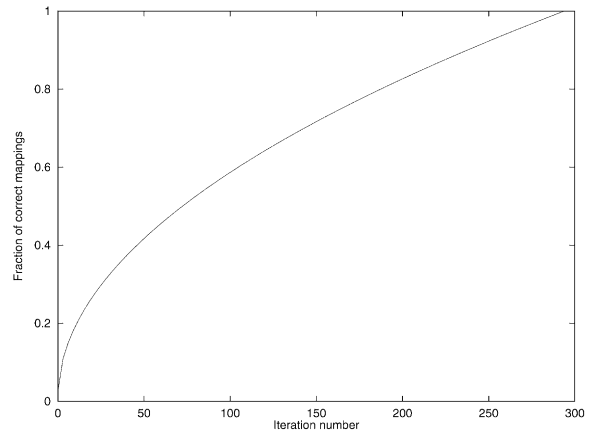


Fig. 4. The predicted fraction of correct mappings as a function of the iteration number when using only selection.

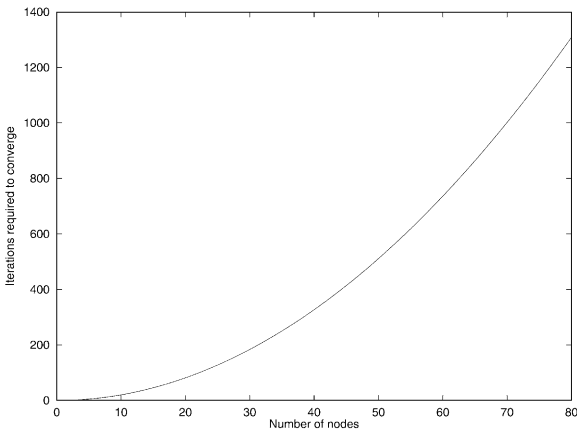


Fig. 5. The predicted number of iterations needed to ensure convergence as a function of the graph size.

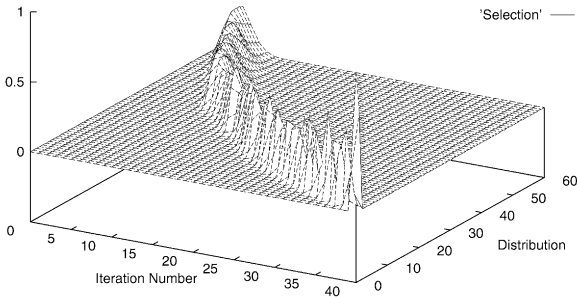


Fig. 6. A numerical simulation of the distribution of correct solutions in a genetic population using only the selection operator.

assumptions concerning our Bayesian consistency measure, we have shown that we would expect the number of iterations required for convergence to be polynomial with respect to the number of nodes in the graphs under match. A plot of the number of iterations required for convergence is shown in Fig. 5.

In order to provide some justification for our modeling of the population mode, we have investigated how the selection operator modifies the distribution of correct correspondences in the genetic population. To embark on this study, we use our simple simulation process to generate the distribution  $P_G^{(n)}$  under the selection operator as a function of the iteration number. The results are shown in Fig. 6. The main point to note is that the width of the distribution remains narrow as the iterations proceed. This is because only the selection operator is used. It is important to stress that there is no diversification process at play.

### 6.3. An analysis of the cross-over operator

The cross-over or recombination operator is responsible for exchanging the assigned matches at corresponding sites in pairs of solution vectors. This contrasts with the mutation and hill-climbing stages of the genetic search procedure, where the matches are re-assigned. In other words, cross-over mixes the matching configurations through an exchange process, while mutation and hill-climbing are responsible for reconfiguration.

To proceed with our analysis of the cross-over operator, let us assume that we are exchanging the assigned matches between the candidate solutions indexed  $\alpha$  and  $\beta$ . These two solutions are chosen at random from the population at iteration  $n$ . The fraction of matches correctly assigned in the solution indexed  $\alpha$  is  $F_\alpha^{(n)}$ , while for the solution indexed  $\beta$  the fraction is  $F_\beta^{(n)}$ . Suppose that the fraction of nodes exchanged from the solution indexed  $\alpha$  to the solution indexed  $\beta$  is denoted by  $\chi$ . Under these circumstances, we form two new solutions, indexed  $\alpha'$  and  $\beta'$  which have the following fractions of correct matches

$$F_{\alpha'}^{(n)} = \chi F_\alpha^{(n)} + (1 - \chi) F_\beta^{(n)}, \quad (32)$$

$$F_{\beta'}^{(n)} = (1 - \chi) F_\alpha^{(n)} + \chi F_\beta^{(n)}. \quad (33)$$

Since the recombination process involves only swaps and does not reconfigure the locations of the matches, it has no overall effect on the average number of correctly assigned node matches per solution in the population. It will however blur the distributions of both fitness and the fraction of correct solutions.

Let us consider the effect of a large number of cross-over operations. According to the central-limit theorem, if we consider the distribution of a random variable associated with the solutions in the population, then as the number of trials increases then so the distribution will approach a Gaussian. To provide an illustration of this effect, we construct an initial population in which each solution has half the matches correctly assigned, i.e.  $F^{(n)} = 0.5$ . After five cross-over iterations, the resulting population distributions are plotted in Fig. 7.

This blurring process, while not directly effecting the algorithm convergence, plays an important role when combined with the selection operator. The blurring occurs symmetrically about the mode of the distribution. If this blurred distribution were subjected to a selection operation, then the high fitness tail would be selected in preference to the low-fitness tail. The resulting distribution would therefore become skewed towards the higher fractions of correct matches.

### 6.4. An analysis of the hill-climbing operator

Since the hill-climbing operator is only used to make local changes that increase  $P_G$ , it is clear that it can only

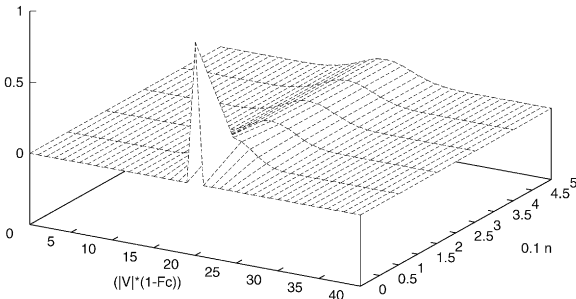


Fig. 7. A demonstration of the blurring process effected by the genetic cross-over operator.

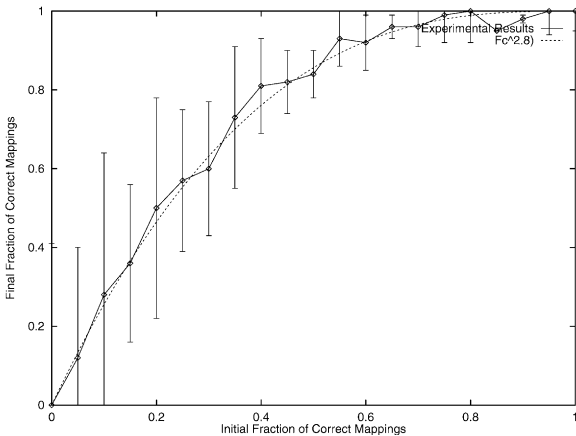


Fig. 8. Empirical results demonstrating how we expect gradient ascent to perform. The dotted curve represents the best fit that was found.

improve the quality of the match. In this section of our analysis, we aim to determine to what extent the hill-climbing operator effects the overall convergence rate of our algorithm.

Modelling the behaviour of the gradient ascent step of the algorithm is clearly a difficult problem since it is highly dependent on the local structure of the global landscape of the fitness measure. One way of simplifying the analysis is to adopt a semi-empirical approach. Here we aim to Monte-Carlo the gradient ascent process and extract a parameterisation of the iteration dependence of the required distribution parameters. Our starting point is to generate 1000 random graphs. Commencing from a controlled fraction of initially correct matches, we perform gradient ascent until the configuration of matches stabilises and no more updates can be made. We plot the final fraction of correct matches against the fraction initially correct in Fig. 8. The best fit to the data gives the following iteration dependence:

$$F_x^{(n+1)} = 1 - (1 - F_x^{(n)})^{2.8}. \tag{34}$$

This result relates the fraction of correct matches at iterations  $n$  and  $n + 1$  resulting from the application of the hill-climbing operator.

By expanding the recursion in iteration number, we can obtain the dependence on the initial fraction of correct matches. At iteration  $n$ , the fraction of correct matches is given by

$$F_x^{(n)} = 1 - (1 - F_x^{(0)})^{2.8n}. \tag{35}$$

We can use the empirical iteration dependence of the expected fraction of correct solutions to make a number of predictions about the convergence rate of population-based hill-climbing. We commence by assuming that the initial set of matches is selected in a random manner. As before, this corresponds to the case  $F^{(0)} = 1/|V|$ . Our condition for convergence is that less than one of the matches per solution is in error, i.e.  $F^{(n)} > (|V| - 1)/|V|$ . By substituting this condition into Eq. (35) and solving for  $n$ , we find

$$n = 0.36 \frac{\ln(|V|)}{\ln[|V|/(|V| - 1)]}. \tag{36}$$

So when the graphs are large  $n \simeq 0.36 |V| \ln |V|$ . The number of iterations required for convergence is plotted as a function of graph size in Fig. 9. The convergence rate increases slowly with graph size. For modest numbers of nodes the increase is approximately linear.

It is interesting to contrast the dependence on graph size with that for the selection operator (see Fig. 9). Whereas hill-climbing has a slow dependence on graph size, in the case of selection there is a more rapid polynomial dependence. As a figure of merit, for a graph of size 50 nodes, the number of iterations required by selection is a factor of 10 larger than that required by hill-climbing.

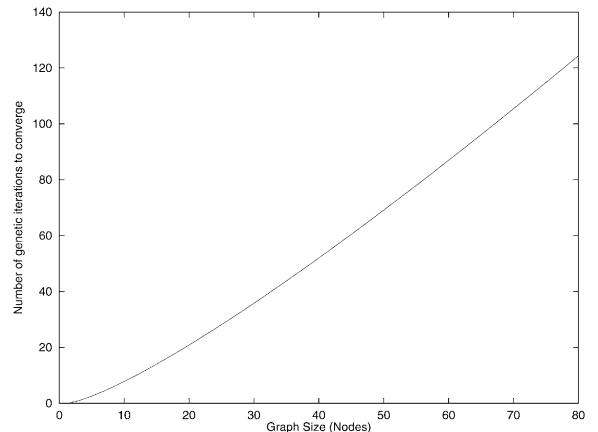


Fig. 9. A prediction of the number of iterations required to ensure convergence when using only the hill-climbing operator.

## 7. An analysis of the combined operators

As we pointed out earlier, the overall goal in this paper is an analysis of the combined effect of the mutation, cross-over, selection and hill-climbing operators described in Section 3. This is not a straightforward task. Previous attempts have included the work of Qi and Palmieri [18,19]. The distinguishing feature of this work was to use a general framework to derive a sufficient condition for monotonic increase of the average fitness under the processes of selection and mutation. Unfortunately, the framework assumes that the optimisation process operates in a continuous space rather than a discrete one. It is the discrete search space of the consistent labelling problem posed by graph matching which is the focus of attention in this paper.

### 7.1. Standard genetic search

Our aim is to extend the analysis of the individual operators presented in Section 6 by deriving a sufficient condition that ensures a monotonic increase of the expected fraction of correct solutions when composite operators are applied. To embark on this study, we must first consider the order in which the different genetic operators are applied. As the population of candidate solutions enters the new iteration  $n + 1$  from the preceding iteration  $n$ , we first perform the cross-over operation. As we have discussed earlier, this process results in a post-cross-over population that is distributed according to a Gaussian distribution. As a result the mode of the distribution is located where the fraction of correct solutions is equal to  $F^{(n)}$ , while we let the standard deviation of the distribution be equal to  $\sigma^{(n)}$ . The mutation operator is applied after the cross-over process. The main effect is to shift the mode of the Gaussian distribution to a lower fitness value. To model the combined effect of the cross-over and mutation operators, we use Eq. (15) in order to compute the change in the fraction of correct solutions due to a mutation operation. The change is equal to

$$\Delta F^{mutation} = -\frac{P_m(F^{(n)}|V| - 1)}{|V|}. \quad (37)$$

As expected, there is a decrease in the expected fraction of correct solutions. Immediately following the application of the mutation operator, we do not know the exact distribution of the fraction of correct matches. However, as demonstrated earlier, we know that for a large number of mutation operations, the distribution is binomial, which in turn can be approximated well by a Gaussian for large  $|V|$ . As a result, the required distribution can be approximated in a Gaussian manner. The mode of the distribution is located at the position

$$F^{mutation} = F^{(n)} + \Delta F^{mutation}. \quad (38)$$

If the mutation probability  $P_m$  is relatively small, as is usually the case, after a single mutation operation, then we can assume that the standard deviation of the Gaussian, i.e.  $\sigma^{(n)}$ , remains unchanged.

In order to determine how the peak or mode of this distribution is shifted by the selection operator we recall Eq. (28). Our interest is now with the change fraction of correct matches that the peak of the distribution undergoes under the combined selection and mutation operators. This quantity is equal to the peak value offset by the shift due to mutation, i.e.

$$\Delta F^{selection} = F_{max}^{selection} - \Delta F^{mutation}. \quad (39)$$

It is important to note that the distribution used as input to the selection operator is the result of the sequential application of the cross-over and mutation processes. Computing the distribution shift after selection is straightforward, but algebraically tedious. For this reason we will not reproduce the details here. Given that we now have a prediction of how we expect the peak of the population distribution to evolve under the processes of cross-over, mutation and subsequent selection, we are in a position to construct a condition for monotonic convergence. Clearly, for the population to converge, the downward shift (i.e. fitness reduction) due to the mutation operator must be smaller than the upward shift (i.e. fitness increase) resulting from selection. In order to investigate this balance of operators, we consider the break-even point between mutation and selection which occurs where

$$\Delta F^{selection} = \Delta F^{mutation}. \quad (40)$$

Substituting from Eqs. (37) and (39), and solving for  $P_m$ , the break-even condition is satisfied when

$$P_m \leq \frac{|V| \cdot |\hat{C}| (1 - 2P_e)}{(P_e/F^{(n)} + 1)(1 - 2P_e) - F^{(n)}(1 - 2P_e) - |V|P_e} \frac{\sigma}{F^{(n)}}. \quad (41)$$

It is important to note that this condition on the mutation probability is very similar to that derived by Qi and Palmieri [18,19]. In fact, the maximum mutation rate is proportional to the ratio of the variance of the fraction of correct mappings in the population to the current expected fraction of matches correct. Moreover, the limiting value of the mutation is proportional to the total number of edges in the graphs, i.e.  $|V| \cdot |\hat{C}|$ . Finally, we note that as the fraction of correct matches increases, the mutation rate must be reduced in order to ensure convergence. It is important to emphasize that we have confined our attention to deriving the condition for monotonic convergence of the expected fitness value. This condition does not guarantee that the search procedure will converge to the global optimum. Neither does it make any attempt to capture the possibility of premature convergence to a local optimum.

### 7.2. Hybrid hill-climbing

Having derived the monotonic convergence condition for the combined effect of the three standard genetic operators, we will now turn our attention to the hybrid hill-climbing algorithm used in our empirical study of graph matching [21]. As before, we compute the change in the fraction of correct matches that we would expect to result from the additional application of the hill-climbing operator. Since this step immediately follows mutation, the population shift is given by

$$\Delta F^{hillclimb} = 1 - (1 - F^{mutation})^{2.8} - F^{mutation}. \quad (42)$$

For completeness, our analysis should next focus on the effect of the selection operator. However, as we demonstrated in Section 6.4, the rate of convergence for the selection operator is significantly slower than that of the hill-climbing operator. This observation suggests that we can neglect the effects of selection when investigating the hybrid hill-climbing algorithm.

In order to identify the monotonic convergence criterion for the hybrid hill-climbing algorithm, we focus on the interplay between opposing population shifts caused by mutation and hill-climbing. This analysis is entirely analogous to the case presented in the previous subsection where we consider the interplay between mutation and selection for the standard genetic algorithm. In the case of the hybrid hill-climbing algorithm, the break-even occurs when

$$\Delta F^{hillclimb} \geq -\Delta F^{mutation}. \quad (43)$$

Proceeding as before, the inequality yields the following relationship:

$$1 - \left(1 + \frac{F^{(n)}|V|(P_m - 1) - P_m}{|V|}\right)^{2.8} \geq \frac{F^{(n)}|V|(P_m - 1) - P_m}{|V|}. \quad (44)$$

By collecting terms and solving for the mutation probability, we arrive at the following convergence condition:

$$P_m \leq \frac{|V|((1 - F^{(n)})^{1/2.8} + F^{(n)} - 1)}{F^{(n)}|V| - 1}. \quad (45)$$

To simplify the convergence condition further, we make the reasonable assumption that the size of the graphs is large, i.e.  $|V| \gg 1$ . Under this assumption, the dependence on  $|V|$  cancels with the result

$$P_m \leq \frac{((1 - F^{(n)})^{1/2.8} + F^{(n)} - 1)}{F^{(n)}}. \quad (46)$$

This limiting mutation rate is plotted in Fig. 10 as a function of  $F^{(n)}$ . In practice, we must select the operating value of  $P_m$  to fall within the envelope defined by the curve in

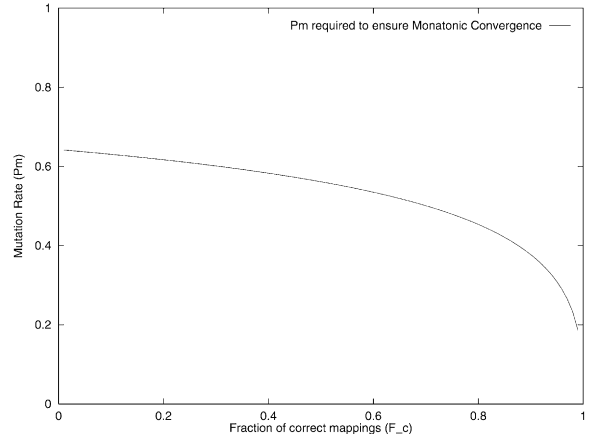


Fig. 10. The maximum mutation rate that may be used to ensure monotonic convergence of a hybrid genetic hill-climbing optimisation scheme.

Fig. 10. As the fraction of correct matches approaches unity (i.e. the algorithm is close to convergence), then the mutation rate should be annealed towards zero. More interestingly, we can use the convergence condition to determine the largest value of the mutation rate for which convergence can be obtained. By taking the limit as the fraction of correct matches approaches zero, we find that  $\lim_{F^{(n)} \rightarrow 0} P_m = 0.6430$ . This agrees well with the empirical findings reported in our previous work [21].

### 7.3. Result validity

Before providing some illustration of the matching method, we return to the question of population size. The aim here is to use a Monte-Carlo study to assess how the theoretical predictions, made under the central limit assumption for large population size, degrade as the population size becomes relatively small.

To embark on this study, we use the simulation method outlined in Section 5.2. The problem studied involves a 20 node graph. In the first set of runs, we use only the three traditional genetic operators, i.e. mutation, cross-over and selection. In Fig. 11, we plot  $F^{(n)}$  as a function of iteration number for increasing population sizes. It is clear that beyond a population size of about 50 solutions, the convergence curves for the different population sizes become increasingly similar.

We repeat this investigation, but supplement the standard operators with the hill-climbing operator. The results are shown in Fig. 12. As expected, we find that agreement between curves, even for small population sizes, is extremely high. We would consequently expect our convergence predictions for the hybrid hill-climbing genetic algorithm to be valid for all population sizes that would be used in practice.

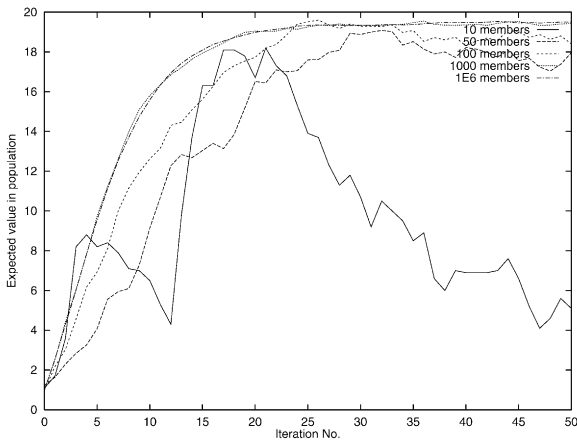


Fig. 11. The expected value for  $F^{(n)}$  under the traditional genetic operators for various population sizes.

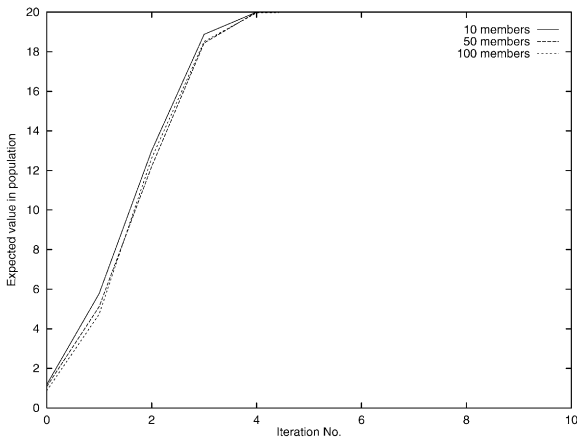


Fig. 12. The expected value for  $F^{(n)}$  under the hybrid hill-climbing genetic algorithm for different population sizes.

## 8. Real images

This section furnishes two examples of the performance of the hybrid genetic algorithm on matching feature sets. The first task was matching corners extracted from an office scene, as shown in panels (a) and (b) of Fig. 13. These are low-quality images taken with an IndyCam. There is no calibration information: ground truth data were obtained by hand. The corners sets were Delaunay triangulated, as shown in panels (c) and (d). By considering the number of mappings in the ground truth data which were inconsistent with the triangulations, the amount of relational corruption was estimated at 15%. Each graph contains about 70 nodes. The graphs were matched using a hybrid genetic algorithm with a

population size of 10. The cross-over and mutation rates were 1.0 and 0.4, respectively. The matching results are shown in panels (e) and (f). The initial guess is random and contains no correct mappings. After only five iterations, the hybrid genetic algorithm has converged on a final match containing 99% correct mappings.

The second example is more difficult. Panels (a) and (b) of Fig. 14 show the left and right images. This time, about 70 regions were segmented from each image using a simple thresholding procedure. As can be seen from the graphs in panels (c) and (d), there is considerable relational corruption. The ground truth suggested that there was as much as 50% relational corruption. The graphs were matched using the same algorithm as before. Again, the initial guess was poor. After 10 iterations, the algorithm had converged to a solution with about 67% correct mappings.

## 9. Conclusions

In this paper our aim has been to understand the role that the different genetic operators play in the convergence of a genetic algorithm. In addition, our investigation has allowed us to make predictions about the number of iterations required for convergence under the processes of selection and hill-climbing. In particular, we have found that the selection process converges in a polynomial number of iterations as a function of graph size. The typical number of iterations required for convergence were in their thousands. Hill-climbing was found to converge in very nearly linear time complexity with logarithmic behaviour as the graph sizes became very small. The typical number of iterations required for convergence was less than 10.

The mutation operator was found to produce an exponential drift of the population distribution towards incorrect mappings. The drift rate was found to depend on both the mutation rate and the current fraction of correct correspondences. In other words, there is greater disruption when the population is dominated by a single consistent solution. In the case when the population contains a large number of dissimilar yet poor solutions, there is less disruptive drift. In contrast with the other operators, the role of the cross-over operator is to exchange information via recombination. The net effect is to blur the distribution of the fraction of correct solutions in a Gaussian manner. In other words, the mean fraction of correct solutions remains stationary, while the corresponding variance increases.

Based on this operator-by-operator analysis, we have analysed the conditions that are required in order to guarantee monotonic convergence of the most likely item in the population. In particular, we have obtained two interesting convergence conditions. First, we derived the convergence condition for the standard genetic

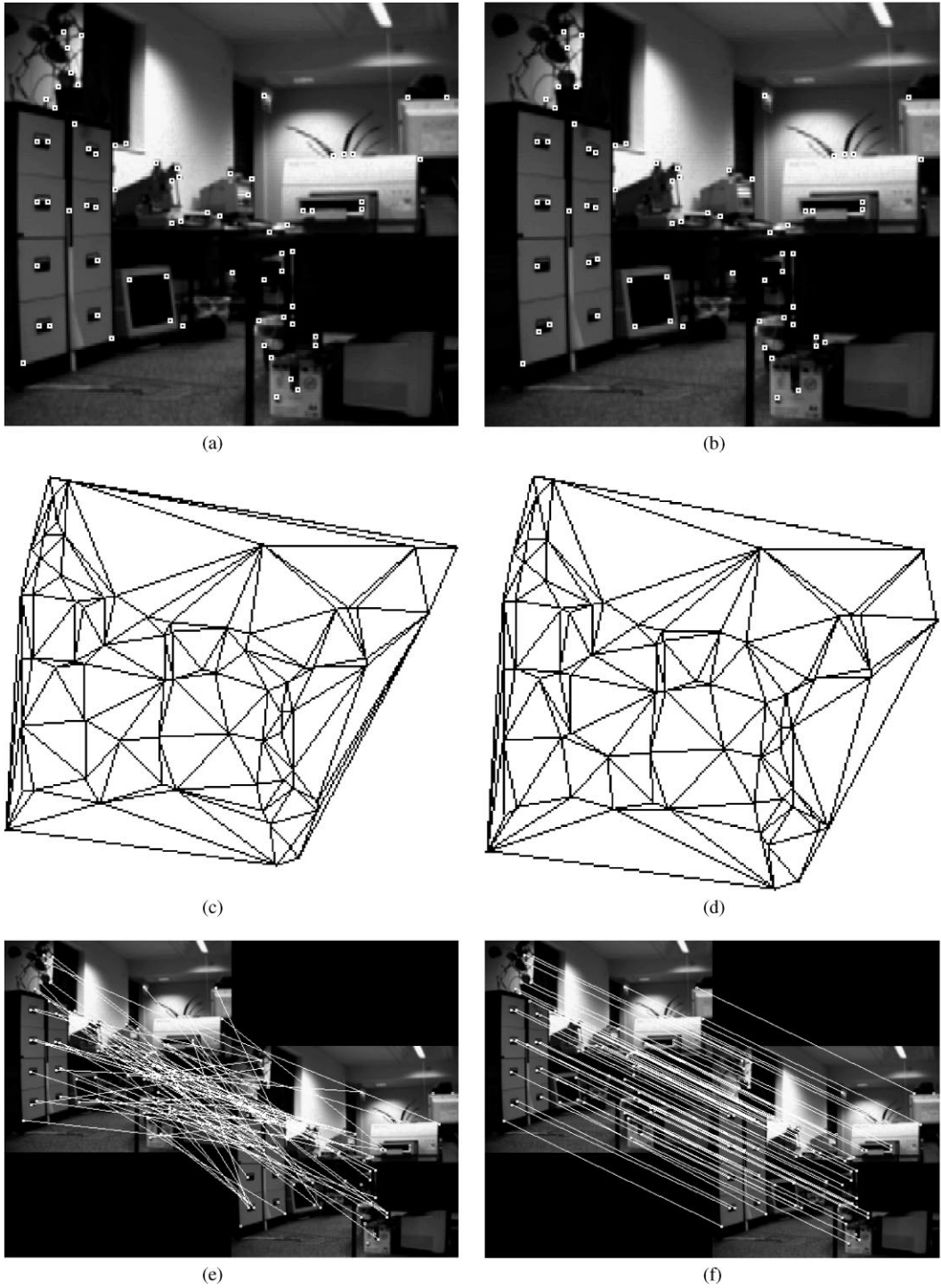


Fig. 13. Uncalibrated Stereogram 1. The camera positions are not known. (a) Left image; (b) right image; (c) left feature graph; (d) right feature graph; (e) initial guess (0%); (f) final match (99%).

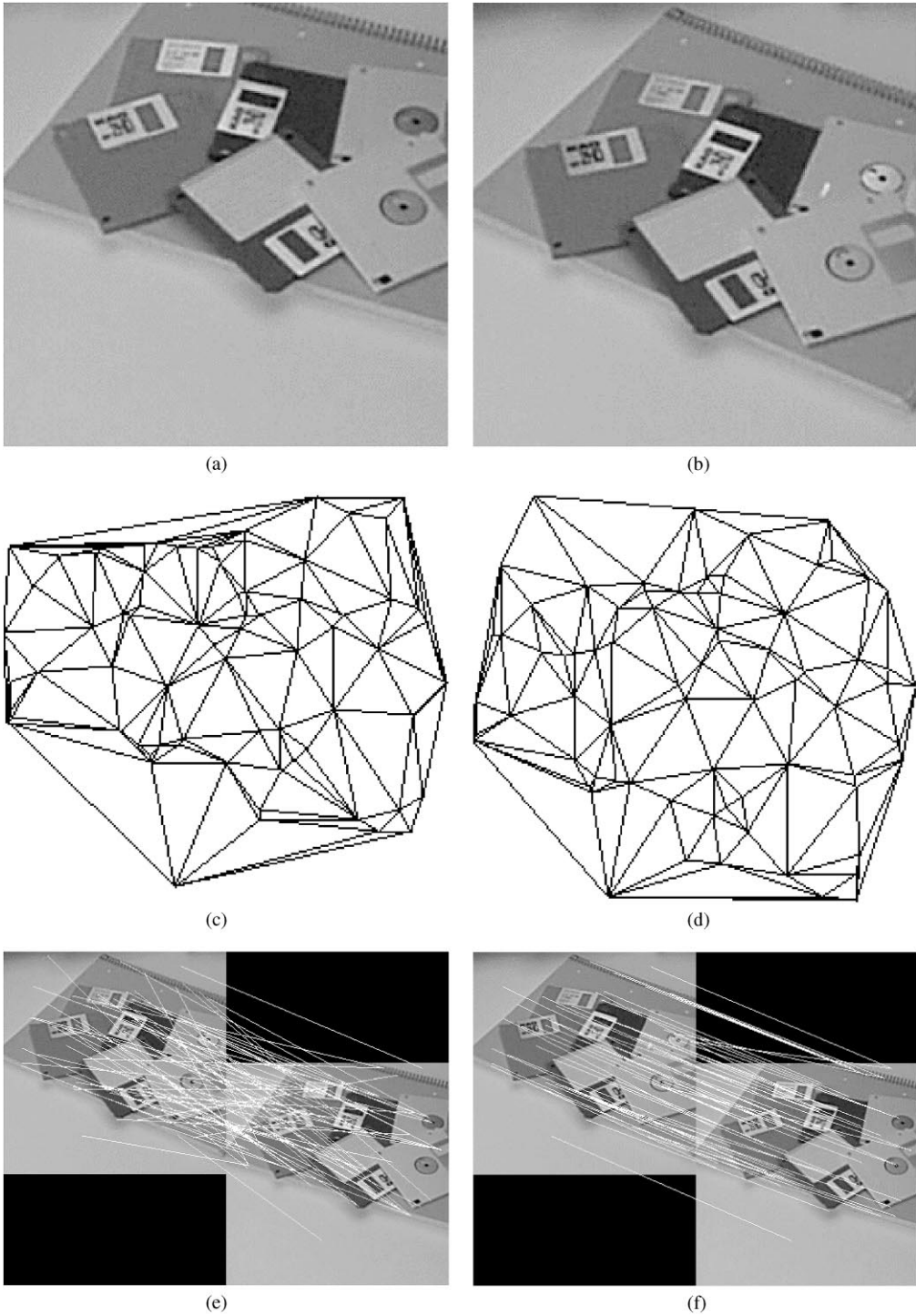


Fig. 14. Uncalibrated Stereogram 2. The camera positions are not known. There is considerable relational corruption: the two Delaunay triangulations are very different. (a) Left image; (b) right image; (c) Left feature graph; (d) right feature graph; (e) initial guess (0.05%); (f) final match (67%).

algorithm, composed of cross-over, mutation and selection. This condition was found to have a similar structure to that reported by Qi and Palmieri [18,19]. However, the condition for convergence makes the role of graph structure explicit in that the limiting value of the mutation probability is proportional to the total number of edges in the graphs being matched. The second result applies to the convergence of a hill-climbing genetic algorithm. The striking feature of the hybrid algorithm is that the limiting value of mutation probability is independent of the structure of the graphs being matched. This result accords well with our previous empirical findings.

In a final section we have investigated the validity of our results when generalized to problems with finite population sizes. In particular, we have discussed the errors introduced by the simplifying approximations that have been made in order to derive our convergence conditions. One approximation that we have made throughout is that the population distribution can be assumed to be Gaussian. By the central-limit theorem, as the size of the graphs increase, then so the approximation becomes increasingly accurate. While deriving the convergence results for the selection operator we were required to make an approximation of our functional which simplifies the dictionary model of consistency. Specifically, we assume that dominant contribution is from a single noise corrupted dictionary item. Of all the assumptions that have been made, this is the least satisfactory. When the fraction of correct matches is low, this approximation tends to underestimate the true value of the fitness functional. However, at higher values it yields accurate results. In other words, the approximation exaggerates the true discriminating power of our consistency measure. Consequently, we would expect to find that our theoretical convergence rates are underestimates. The final assumption has been to approximate the behaviour of the hill-climbing step using an empirical performance curve. In practice, the error resulting from the fitted polynomial approximation was never found to exceed 5%.

## References

- [1] S. Kirkpatrick, C.D. Gelatt, M.P. Vecchi, Optimisation by simulated annealing, *Science* 220 (1983) 671–680.
- [2] V. Kumar, Algorithms for constraint-satisfaction problems: a survey, *AI Magazine* 13 (1992) 32–44.
- [3] A.K. Mackworth, E.C. Freuder, The complexity of some polynomial network consistency algorithms for constraint satisfaction problems, *Artif. Intell.* 25 (1985) 65–74.
- [4] A.K. Mackworth, Consistency in a network of relations, *Aritif. Intell.* 8 (1977) 99–118.
- [5] A. Sanfeliu, K.S. Fu, A distance measure between attributed relational graphs for pattern recognition, *IEEE Trans. Systems Man. Cybernet.* 13 (1983) 353–362.
- [6] D. Waltz, Understanding line drawings of scenes with shadows, in: P.H. Winton (Ed.), *The Psychology of Computer Vision*, McGraw-Hill, New York, 1975.
- [7] L.G. Shapiro, R.M. Haralick, A metric for comparing relational descriptions, *IEEE Trans. Pattern Machine Intell.* 7 (1985) 90–94.
- [8] J. Pearl, *Heuristics: Intelligent Search Strategies for Computer Problem Solving*, Addison-Wesley, Reading, MA, 1984.
- [9] A.M. Yuille, J. Coughlan, Twenty questions, focus of attention and A\*: a theoretical comparison of optimisation strategies, in: M. Pelillo, E.R. Hancock (Eds.), *Energy Minimisation Methods in Computer Vision and Pattern Recognition*, Lecture Notes in Computer Science, vol. 1223, Springer, Berlin, 1997, pp. 197–212.
- [10] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions and Bayesian restoration of images, *IEEE Trans. Pattern Machine Intell. PAMI-6* (1984) 721–741.
- [11] B. Gidas, A re-normalisation-group approach to image processing problems, *IEEE Trans. Pattern Machine Intell.* 11 (1989) 164–180.
- [12] A. Yuille, Generalised deformable models, statistical physics and matching problems, *Neural Comput.* 2 (1990) 1–24.
- [13] F. Glover, Ejection chains, reference structures and alternating path methods for traveling salesman problems, *Discrete Appl. Math.* 65 (1996) 223–253.
- [14] E. Rolland, H. Pirkul, F. Glover, Tabu search for graph partitioning, *Ann. Oper. Res.* 63 (1996) 232–290.
- [15] F. Glover, Genetic algorithms and tabu search — hybrids for optimisation, *Discrete Appl. Math.* 49 (1995) 111–134.
- [16] F. Glover, Tabu search for nonlinear and parametric optimisation (with links to genetic algorithms), *Discrete Appl. Math.* 49 (1995) 231–255.
- [17] D.B. Fogel, An introduction to simulated evolutionary optimisation, *IEEE Trans. Neural Networks* 5 (1994) 3–14.
- [18] X.F. Qi, F. Palmieri, Theoretical analysis of evolutionary algorithms with an infinite population in continuous space: basic properties of selection and mutation, *IEEE Trans. Neural Networks* 5 (1994) 102–119.
- [19] X.F. Qi, F. Palmieri, Theoretical analysis of evolutionary algorithms with an infinite population in continuous space: analysis of the diversification role of cross-over, *IEEE Trans. Neural Networks* 5 (1994) 120–129.
- [20] R. Myers, E.R. Hancock, Genetic algorithm parameter sets for line labelling, *Pattern Recognition Lett.* 18 (1998) 1283–1292.
- [21] A.D.J. Cross, R.C. Wilson, E.R. Hancock, Inexact graph matching using genetic search, *Pattern Recognition* 30 (1997) 953–970.
- [22] E.R. Hancock, J. Kittler, A Bayesian interpretation for the Hopfield network, *IEEE International Conference on Neural Networks*, 1993, pp. 341–346.
- [23] E.R. Hancock, J. Kittler, Discrete relaxation, *Pattern Recognition* 23 (1990) 711–733.
- [24] E.R. Hancock, M. Pelillo, A Bayesian interpretation for the exponential correlation associative memory, *Pattern Recognition Lett.* 19 (1998) 139–149.
- [25] R.C. Wilson, E.R. Hancock, Gauging consistency and controlling structural errors, *IEEE Computer Society Computer Vision and Pattern Recognition Conference*, 1996.

- [26] R.C. Wilson, E.R. Hancock, Structural matching by discrete relaxation, *IEEE Trans. Pattern Machine Intell.* 19 (1997) 634–648.
- [27] R.C. Wilson, E.R. Hancock, Structural matching with active triangulations, *Computer Vision Image Understanding* 72 (1998) 21–38.
- [28] R.C. Wilson, E.R. Hancock, Relational matching with dynamic graph structures, *Proceedings of the Fifth International Conference on Computer Vision*, 1995, pp. 450–456.
- [29] H.G. Barrow, R.J. Popplestone, Relational descriptions in picture processing, *Machine Intell.* 6 (1971).
- [30] B.T. Messmer, H. Bunke, Efficient error-tolerant subgraph isomorphism detection, In: D. Dori, A. Bruckstein (Eds.), *Shape, Structure and Pattern Recognition*, 1995, pp. 231–240.
- [31] A.K.C. Wong, M. You, Entropy and distance of random graphs with application to structural pattern recognition, *IEEE Trans. Pattern Machine Intell.* 7 (1985) 599–609.
- [32] D. Goldberg, *Genetic Algorithms in search, Optimisation and Learning*, Addison-Wesley, Reading, MA, 1989.

**About the Author**—ANDREW CROSS gained his B.Sc. in Computational Physics with first-class honours from the University of Manchester Institute of Science and Technology in 1994. Between 1994 and 1998 he undertook research in the area of optimisation methods for computer vision at the University of York. He was awarded the D.Phil. degree for this work in July 1998. Following a period of postdoctoral research at York, Dr. Cross took up an appointment with NewTek in San Antonio, Texas. His interests are in computer vision, graphics and image processing.

**About the Author**—RICHARD MYERS took his B.A. in Natural Sciences from the University of Cambridge in 1989. In 1995 he gained a M.Sc. with distinction in Information Processing at the University of York. He has recently completed a D.Phil. in the Computer Vision Group at the Department of Computer Science at the University of York. The main topic of his research is the use of genetic algorithms to solve consistent labelling problems arising in the machine vision domain. In 1997 he spent two months working at NEC Corporation in Kawasaki, Japan sponsored a REES/JISTEC fellowship. His interests include evolutionary computation, perceptual organisation and labelling problems.

**About the Author**—EDWIN HANCOCK gained his B.Sc. in physics in 1977 and Ph.D. in high energy nuclear physics in 1981, both from the University of Durham, UK. After a period of postdoctoral research working on charm-photo-production experiments at the Stanford Linear Accelerator Centre, he moved into the fields of computer vision and pattern recognition in 1985. Between 1981 and 1991, he held posts at the Rutherford-Appleton Laboratory, the Open University and the University of Surrey. He is currently Professor of Computer Vision in the Department of Computer Science at the University of York where he leads a group of some 15 researchers in the areas of computer vision and pattern recognition. Professor Hancock has published about 180 refereed papers in the fields of high energy nuclear physics, computer vision, image processing and pattern recognition. He was awarded the 1990 Pattern Recognition Society Medal and received an outstanding paper award in 1997. Professor Hancock serves as an Associate Editor of the journal *Pattern Recognition* and has been a guest editor for the *Image and Vision Computing Journal*. He is currently guest-editing a special edition of the *Pattern Recognition* journal devoted to energy minimisation methods in computer vision and pattern recognition. He chaired the 1994 British Machine Vision Conference and has been a programme committee member for several national and international conferences.