Multiple Fault Diagnosis Via The GA

W.D. Potter
Department of Computer Science
and
Artificial Intelligence Programs
University of Georgia

ABSTRACT

Diagnosis is the process of determining the correct problem from a collection of problems given a set of symptoms that indicate a problem exists. Common experiences with this process include visits to the physician in order to determine our illness (disease) and visits to our local mechanic to determine the cause (fault) of a poorly operating car. In either case, we report the symptoms of the problem to the diagnostician (physician or mechanic) who determines the most likely cause that best explains these symptoms (an example of abductive reasoning). In terms of the complexity of finding the correct problem, the diagnostician must find a diagnosis from a set of possible diagnoses. That is, if a total of 10 problems are being considered where only one of these is the correct one then at most 10 diagnoses will need to be evaluated.

However, in the more typical case where several problems (diseases/faults) may occur simultaneously, the complexity of finding a proper diagnosis increases exponentially with the number of problems. For example, using the 10 problems considered above, the situation changes to where any of the 1024 possible combinations* of problems may turn out to be the correct diagnosis. In this paper, we discuss an automated method for diagnosing multiple simultaneous problems. In particular, we focus on the Genetic Algorithm heuristic (testing only a small percentage of the total combinations, yet finding a satisfactory diagnosis).

INTRODUCTION

In medicine as well as electronics and other domains, multiple problem diagnosis, henceforth called *multiple fault diagnosis,* is the identification of a set of problems (disorders, diseases or faulty components) that best corresponds to or explains some observed abnormal behavior that is indicated by a set of symptoms (manifestations) [Peng87a, Reit87]. This type of problem solving is commonly referred to as abductive inference, and automating this approach has been the focus of extensive research efforts [Davi84, deKl87, Gene84, Jose87, Peng87a, Peng87b, Regg83, Reit87]. Common among these research efforts is the nature of their systems. Namely, these approaches to diagnosis follow the "reasoning from first principles" paradigm where a description of some physical system's structure and behavior is maintained and compared to abnormal behavior. This is in sharp contrast to the "experiential" paradigm which is driven by the problem solving rules of thumb or heuristics acquired from a human expert

* Actually, there are only 1023 combinations considered as possible diagnoses since the combination indicating a normally functioning system need not be evaluated.

diagnostician [Reit87]. The MYCIN expert system is based on the experiential approach.

A closer look at multiple fault diagnosis reveals three major stumbling blocks between a diagnostic problem and a "reasonable" automated solution:

1) the large number of possible diagnoses,
2) measuring the relative "goodness" of a particular diagnosis,
3) the search strategy used to find highly reliable diagnoses.

In this discussion, the "most reasonable" solution corresponds to the diagnosis or diagnoses that best explain the observed symptoms. This best explanation is determined and wholly dependent on the calculation of the goodness of a diagnosis. We call a diagnosis reliable or optimal if according to the goodness measure, no better diagnosis exists.

Consider for example the small hypothetical situation where we have 20 components in a telephone communications system, each component may exhibit faulty behavior via a set of, say, 10 alarms or symptoms, and that we have some mechanism for ranking each of the 2^{20} (that's 1,048,576) possible diagnoses. An intuitively appealing representation for a diagnosis is a 20-bit binary string where each of the 20 components is associated with one of the bit positions; component 1 with position 1, component 2 with position 2, and so on. In a diagnosis, a 0 in a particular bit position means that the corresponding component is not considered to be at fault while a 1 means that this component helps explain some or all of the symptoms.

One possible approach for finding the best diagnosis is to simply generate each of the 2^{20} diagnoses, calculate the goodness of each, and report the best one. However, for systems with more than about 25 components (that's over 33.5 million possible diagnoses), this approach becomes infeasible, especially if the system is in an aircraft or spacestation where quick diagnosis followed by correction would be crucial. In practice, it is not unusual for a medium-sized system to have upwards of 50 to 75 components, similar to medical domains that have large numbers of diseases or causes to consider [Regg83]. [Specialized heuristic search strategies have been proposed as alternatives to the exhaustive search strategy [deKl87, Jose87, Peng87b, Pott90, Reit87] when combined with a mechanism for distinguishing the goodness of a diagnosis.]

Exhaustive search may be speeded up by using a limited exhaustive search. In this approach, only the disorders (diseases/components) associated with the manifestations (symptoms) observed are considered. In the worst case where all disorders are indicated, this approach reverts to the regular exhaustive search. However, for those times when only a subset of the disorders are indicated, this method can be relatively fast and, of course, reliable.

In terms of reliability, another approach is the branch and bound method. It too is an approach that is guaranteed to find the optimal solution. Essentially, what we do in this approach is determine the possible next moves toward a solution and eliminate those moves that are more costly than the cheapest accumulated thus far in the search. For example, in a minimum route finding problem such as the traveling salesman problem where the best solution is the one with the shortest distance to travel, we keep track of a solution and terminate exploring an alternate solution whenever its distance exceeds the current tracked solution. Shorter solutions replace the current solution and we continue the search until there are no possible solutions that may displace our current best solution.

A good heuristic method used to attack the problem of multiple fault diagnosis is based on the genetic algorithm [Gold89, Holl75]. This strategy incorporates the determination of a goodness measure or "likelihood" that a particular diagnosis explains the observable symptoms. The genetic algorithm follows the notion of *natural selection* in nature. That is, a small population of solutions is randomly generated. The individual solutions that are the most promising (most likely to explain the observed symptoms) are used to determine or create another population (the next generation). This *evolutionary* process continues until no improvement in the likelihood of some best solution is observed.

Each of these methods has advantages and disadvantages for diagnosing problems with systems where more than one problem may occur at the same time. Before continuing further, we briefly describe the Probabilistic Causal Model from Parsimonious Covering Theory developed by Peng and Reggia [Peng87a, Peng87b, Regg83]. The reason for this digression is because we use a "goodness" measure for a diagnosis, regardless of the diagnostic strategy, that is based on their "relative likelihood" measure. Our goodness measure is called the *modified relative likelihood* [Pott90].

THE PROBABILISTIC CAUSAL MODEL

One of the leading theories of diagnosis is based on the notion of parsimoniously covering a set of observable symptoms [Regg83], that is, finding a minimal set covering* (i.e., a set of diseases in their medical domain) that explains a given set of symptoms. The fact that a cover is necessary in order to explain the symptoms is intuitively clear but minimality is another matter and, in some typical cases, is inappropriate. In order to overcome this major shortcoming, Peng and Reggia introduced the probabilistic causal model (PCM) [Peng87a, Peng87b]. The PCM integrates "symbolic cause-effect inference with numeric probabilistic inference" to solve multiple fault diagnosis problems.

In their approach, a multiple fault diagnosis problem is characterized as a 4-tuple:

$$<D, M, C, M^+>$$

where

D is a finite nonempty set of disorders (i.e., diseases or faulty components).

M is a finite nonempty set of manifestations (i.e., symptoms).

C is a relation, called the tendency matrix, which is a subset of $D \times M$. This relation pairs diseases with associated symptoms such that $(d, m) \in C$ means that disease d **may** cause symptom m.

M^+ is a subset of M which identifies the observed manifestations. Note that manifestations not identified in M^+ are assumed to be absent.

A diagnosis DI (a subset of D) identifies the disorders that are possibly responsible for the symptoms in M^+. Diagnosis DI covers M^+ if each of the individual manifestations in M^+ is associated with at least one of the disorders in DI as determined using C. As with M^+, disorders not identified in DI are assumed to be absent.

Associated with each disorder d_j in D is a prior probability p_j where $0 < p_j < 1$. Values are assumed to exist and disorders in D are assumed to be independent. Associated with each "causal association" in C is a causal strength c_{ij} such that $0 \le c_{ij} \le 1$ and represents how

* See [Edmo62] and [Gare79] for a description of the set covering problem and the membership of minimal set covering in the class of NP-complete problems, respectively.

frequently a disorder d_j causes manifestation m_i. One of their major contributions centers on the fact that c_{ij} is not equivalent to the conditional probability $P(m_i \mid d_j)$ used in earlier Bayesian approaches. The causal strength does represent the conditional probability $P(d_j \textit{ causes } m_i \mid d_j)$, which has the advantage of being unaffected by coincident disorders, that is, we may expect the frequency with which d_j causes m_i given d_j to remain stable. An additional assumption stipulates that no manifestation may exist in M^+ unless it is actually caused by some disorder in D.

Now, we have $|D|$ prior probabilities and $|D| \times |M|$ causal strengths. Using these values, Peng and Reggia derive a formula for calculating the "relative likelihood," denoted $L(DI, M^+)$, of a diagnosis DI given observable manifestations M^+. The likelihood is the product of three terms:

$$L(DI, M^+) = L_1 L_2 L_3$$

where

$$L_1 = \prod_{m_i \in M^+} \left[1 - \prod_{d_j \in DI} (1 - c_{ij}) \right],$$

is the likelihood that disorders in DI cause the manifestations in M^+. For diagnoses that do not cover M^+, L_1 evaluates to 0 thus forcing L to 0. Unfortunately, this denies any analysis of non-cover diagnoses. This limitation is avoided in our modified relative likelihood calculation.

$$L_2 = \prod_{d_j \in DI} \prod_{m_l \in effects(d_j) - M^+} (1 - c_{lj}),$$

is the likelihood that disorders in DI do not cause manifestations outside of M^+ (e.g., in $M - M^+$). In their words, L_2 is "a weight based on manifestations expected with DI but which are actually absent." Ideally, a good diagnosis has an L_2 value that is close to 1. Unfortunately, this term denies any analysis of super-cover diagnoses. Again, our modified relative likelihood calculation avoids this limitation.

$$L_3 = \prod_{d_j \in DI} \frac{p_j}{(1 - p_j)},$$

is the likelihood that a highly probable (very common) disorder d_j contributes significantly in the overall likelihood of a diagnosis DI containing d_j.

To summarize, L_1 forces L to focus in on only diagnoses that cover or explain all manifestations in M^+, L_2 encourages L to focus on "irredundant" and "relevant" covers, and L_3 forces L to focus on more likely or common disorders rather than on rare or less likely disorders (a reasonable but sometimes risky strategy in medical diagnosis). Irredundant covers do not contain any excess disorders which could be removed and still be left with a cover. Relevant covers (which contain the set of irredundant covers) ensure that disorders associated with manifestations in M^+ are the most seriously considered.

The Genetic Algorithm approach is dominated by the goodness of a diagnosis, also called the objective function. For this reason, we use the *modified relative likelihood* [Pott90], a variant of the relative likelihood (RL) of Peng and Reggia because the RL: 1) has a solid theoretical foundation, 2) has an efficient implementation within the search algorithms, 3) uses a relatively small amount of data to operate (number of disorders times the sum of one plus the number of manifestations), 4) uses data that may originate as subjective expert certainty factors but then evolve into statistically justifiable probabilities, and 5) is easily modified without affecting its fundamental nature or incurring any computational expense.

Our modified relative likelihood (MRL) allows the search to converge on a global maximum using diagnoses that are not covers (e.g., do not completely explain the manifestations) as well as diagnoses that are super-covers (e.g., contain redundant disorders). The reason for this modification is to allow progress whenever the search space terrain resembles Monument Valley with a generally large flat surface except for occasional very thin high peaks or needle-like structures. A search strategy may become "lost" in the flat area and never "see" a nearby peak without some broad convergence mechanism. This corresponds to a restricted diagnostic problem where very few covers for the manifestation set exist. With an unmodified RL, all non-cover diagnoses would have a zero likelihood and would provide almost no search improvement information. This situation is avoided by ensuring that term L_1 is never zero (recall $L(DI, M^+) = L_1 \, L_2 \, L_3$). That is, term L_1 is forced to zero in the RL whenever the causal associations between some disorder in the diagnosis and symptoms in the manifestation set are equal to zero, but these associations are set to a value very close to zero in the MRL computation. Therefore, the differentiating factors become the disorder prior probability and the expected but absent manifestations associated with a diagnosis. Also, L_1 significantly increases as more of the manifestations are explained.

The other modification that aids convergence is associated with term L_2. In certain cases using the RL, L_2 is forced to zero in the event of a redundant or irrelevant cover diagnosis. This occurs primarily when some disorder in a diagnosis has a unit causal association with a manifestation that is not present in the observed manifestation set. The modified relative likelihood substitutes a value very close to one for these situations. This allows diagnoses to be evaluated and compared in order for the search strategies to converge to a global maximum because terms L_1 and L_2 cause MRL values to be less than optimal (but not zero) when evaluating non-covers and super-covers, respectively.

As an example, consider the situation where we have a tendency matrix with 15 disorders and 10 manifestations, see Figure 1. Given the observed manifestations

$$M^+ = \{m_1, m_2, m_4, m_5, m_7, m_8, m_9, m_{10}\}.$$

We find that the terms and modified (or adapted) relative likelihood, $L_{adapted}$, of the optimal diagnosis are:

$$L_1 = 1.99315527715718e{-}01$$

$$L_2 = 2.11844640547108e{-}01$$

$$L_3 = 1.82012538561086e{+}00$$

$$L_{adapted}(DI, M^+) = 7.68528401831913e{-}02$$

where

$$DI = 001100001001100.$$

This indicates that $\{d_3, d_4, d_9, d_{12}, d_{13}\}$ gives us the best explanation for the observed manifestations.

www.cs.uga.edu/~potter/CompIntell/
TendencyMatrix10x25.txt

we will use this tendency matrix

ExhaustiveResults10x25.txt

we will use this for comparison/reliability

	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}	d_{12}	d_{13}	d_{14}	d_{15}
p_i	0.12	0.14	0.39	0.64	0.01	0.21	0.26	0.19	0.59	0.29	0.06	0.47	0.56	0.41	0.06
m_1	0.58	0.00	0.00	0.00	0.00	0.25	0.00	0.96	0.00	0.00	0.85	0.00	0.74	0.00	0.38
m_2	0.00	0.00	0.00	0.00	0.00	0.15	0.81	0.00	0.32	0.00	0.36	0.77	0.30	0.61	0.00
m_3	0.00	0.00	0.44	0.00	0.00	0.00	0.00	0.11	0.00	0.97	0.64	0.00	0.63	0.85	0.00
m_4	0.43	0.67	0.79	0.00	0.26	0.72	0.07	0.00	0.00	0.00	0.84	0.00	0.00	0.42	0.64
m_5	0.46	0.10	0.00	0.58	0.00	0.00	0.46	0.00	0.57	0.40	0.00	0.51	0.00	0.97	0.00
m_6	0.91	0.00	0.00	0.00	0.00	1.00	0.28	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.12
m_7	0.90	0.94	0.07	0.28	0.00	0.00	0.00	0.00	0.97	0.00	0.00	0.91	0.48	0.23	0.72
m_8	0.00	0.00	0.14	0.17	0.24	0.00	0.30	0.00	0.26	0.00	0.00	0.05	0.00	0.00	0.00
m_9	0.00	0.00	0.13	0.12	0.17	0.04	0.00	0.00	0.97	0.00	0.00	0.00	0.43	0.08	0.00
m_{10}	0.00	0.63	0.07	0.75	0.12	0.00	0.00	0.45	0.00	0.88	0.21	0.00	0.45	0.86	0.19

Figure 1. Prior Probability & Tendency Matrix: 10x15 One-Half Dense.

THE GENETIC ALGORITHM

Genetic Algorithms [Holl75, Gold89] are heuristic search routines that are guided by a model of Darwin's theory of natural selection or the survival of the fittest. Here the fittest means the most highly ranked solution in a large solution space. The basic idea behind the genetic search strategy is to generate solutions that converge on the global maximum (i.e., the best solution in the search space) regardless of the "terrain" of the search space. A typical terrain might resemble the Great Smoky Mountains with many peaks and valleys, an area that is relatively flat, and a highest peak (Clingman's Dome). One characteristic of genetic algorithms is that they are relatively unaffected by hill-climbing or being misled by some local maximum such as ascending Mt. LeConte and assuming that you are on the highest peak in the Smokies since other nearby peaks appear lower, depending on visibility. Likewise, with genetic algorithms the key to finding the global maximum lies in the ability to evaluate and compare possible optimal solutions.

The basic operations involved in a genetic algorithm (GA) are: 1) mate selection, 2) crossover, and 3) mutation. Typically, the major data structure is a binary string representing the possible solutions. In GA terms, a bit string corresponds to an individual, and a set of individuals is called a population. The fitness or strength of an individual is computed using some objective or fitness function, and is used to compare an individual with other individuals in the same population. During mate selection, parent strings are stochastically selected according to their fitness from the current population and "mated" to produce offspring for the next generation. Fitter parents contribute more offspring to the next generation than weaker parents because they have a higher probability of being selected for mating. This is the step that models the process of natural selection in nature.

Crossover, the second operation, determines the characteristics of a "child" or next generation individual. In nature, children inherit good as well as bad features of their parents in varying degrees of dominance. Crossover performs this same function in a GA. One of the simplest crossover approaches is to split each parent string at the same randomly chosen location and swap their tail sections. This ensures a certain amount of inheritance and ideally, the good/strong features will dominate the children. The inheritance of features that produce stronger children throughout the generations is the source of the GA's ability to converge on the global maximum in a relatively short time.

The last basic operation is called mutation. Mutation is that extremely rare "glitch" in the inheritance mechanism that introduces or modifies some feature with unpredictable consequences. Mutation occurs in a GA immediately after the creation of a next generation individual yet before the next generation has become static. Once the new generation becomes static, we move forward in order for it to become the new current generation. Ideally, mutants would contain some useful features that may have been inadvertently lost in earlier generations.

The simple genetic algorithm described in [Gold89] follows these three basic steps. Additional operations and modifications are described as well. One major modification to the simple crossover approach, called two-point crossover, has been shown to be an easily implemented and effective alternate to simple crossover. With two-point crossover, an individual bit string is viewed as a ring and sections of parents are interchanged. This is like cutting equal sized sections from two donuts and swapping the sections to form a new (more appetizing) pair of snacks. Another effective crossover approach is the "greedy" approach described in [Liep90].

Regarding the internal operating parameters of the GA, Goldberg recommends a population size between 50 and 200. The majority of our experiments have been performed with population sizes of 50, 100, and 150. Other features and characteristics of the GAs include a crossover probability of 0.6, and a standard mutation probability of 0.0333. The crossover probability identifies the likelihood that parents will have offspring. The standard mutation probability identifies the likelihood of a bit in an individual string being changed. In addition, several improvements to the simple GA have been incorporated either to improve the efficiency of the GA or to improve the reliability of the GA results. Finding improvements that equal or surpass our results are left to the reader, good luck.

REFERENCES

[Davi84]
Davis, R., "Diagnostic Reasoning Based on Structure and Behavior," in *Artificial Intelligence*, Vol. 24, No. 1-3, pp. 347-410, December 1984.

[Edmo62]
Edmonds, J., "Covers and Packings in a Family of Sets," in *Bulletin of the American Mathematics Society*, Vol. 68, pp. 494-499, 1962.

[Gare79]
Garey, M.R., and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman Publishers, San Francisco, CA, 1979.

[Gene84]
Genesereth, M.R., "The Use of Design Descriptions in Automated Diagnosis," in *Artificial Intelligence*, Vol. 24, No. 1-3, pp. 411-436, December 1984.

[Gold89]
Goldberg, D.E., *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Co., 1989.

[Holl75]
Holland, J.H., *Adaptation in Natural and Artificial Systems*, Ann Arbor: The University of Michigan Press, 1975.

[Jose87]
Josephson, J., B. Chandrasekaran, J. Smith, and M. Tanner, "A Mechanism For Forming Composite Explanatory Hypotheses," in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-17, No. 3, pp. 445-454, May/June 1987.

[deKl87]
de Kleer, J., and B.C. Williams, "Diagnosing Multiple Faults," in *Artificial Intelligence*, Vol. 32, No. 1, pp. 97-130, April 1987.

[Liep90]
Liepins, G.E., M.R. Hilliard, J. Richardson, and M. Palmer, "Genetic Algorithm Applications to Set Covering and Traveling Salesman Problems," in Brown (ed.), *OR/AI: The Integration of Problem Solving Strategies*, 1990.

[Peng87a]
Peng, Y., and J.A. Reggia, "A Probabilistic Causal Model for Diagnostic Problem Solving, Part I: Integrating Symbolic Causal Inference With Numeric Probabilistic Inference," in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-17, No. 2, pp. 146-162, March/April 1987.

[Peng87b]
Peng, Y., and J.A. Reggia, "A Probabilistic Causal Model for Diagnostic Problem Solving, Part II: Diagnostic Strategy," in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-17, No. 3, pp. 395-406, May/June 1987.

[Peng89]
Peng, Y., and J.A. Reggia, "A Connectionist Model for Diagnostic Problem Solving," in *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-19, No. 2, pp. 285-298, March/April 1989. Not cited in the text but it has some interesting stuff and a example.

[Pott90]
Potter, W.D., B.E. Tonn, M.R. Hilliard, G.E. Liepins, R.T. Goeltz, and S.L. Purucker, "Diagnosis, Parsimony, and Genetic Algorithms," in *Third International Conference on Industrial & Engineering Applications of Artificial Intelligence and Expert Systems*, July, 1990.

[Regg83]
Reggia, J.A., D. Nau, and P. Wang, "Diagnostic Expert Systems Based on a Set Covering Model," in *International Journal of Man-Machine Studies*, Vol. 19, No. 5, pp. 437-460, November, 1983.

[Reit87]
Reiter, R., "A Theory of Diagnosis From First Principles," in *Artificial Intelligence*, Vol. 32, No. 1, pp. 57-95, April 1987.

$$L(DI, M^+) = L_1 L_2 L_3$$

where

$$L_1 = \prod_{m_i \in M^+} \left[1 - \prod_{d_j \in DI} (1 - c_{ij}) \right],$$

is the likelihood that disorders in DI cause the manifestations in M^+. For diagnoses that do not cover M^+, L_1 evaluates to 0 thus forcing L to 0. Unfortunately, this denies any analysis of non-cover diagnoses.

$$L_2 = \prod_{d_j \in DI} \prod_{m_l \in effects(d_j) - M^+} (1 - c_{lj}),$$

is the likelihood that disorders in DI do not cause manifestations outside of M^+ (e.g., in $M - M^+$). L_2 is "a weight based on manifestations expected with DI but which are actually absent." Ideally, we prefer L_2 values that are close to 1.

$$L_3 = \prod_{d_j \in DI} \frac{p_j}{(1 - p_j)},$$

is the likelihood that a highly probable (very common) disorder d_j contributes significantly in the overall likelihood of a diagnosis DI containing d_j.

Multiple Fault Diagnosis -- Experiment SetUp

Phase 1 (individual diagnosis):
Prepare your GA using simple parameter settings. Run individual diagnosis tests by entering M+, a bit string representing the symptoms our patient has, and outputting the diagnosis (bit string) and fitness value of the solution proposed by the GA. Repeat Phase 1 several times in order to convince yourself that your GA is working properly. Discuss your results in class at the soonest possible class meeting.

Phase 2 (reliability phase):
Here we will run a set of trials where each trial has different parameter settings. Of course, we need to run each trial at least 10 times and use the average result (be sure to track best, worst, and average). You may use additional parameter settings but be sure to include the following:

- Population sizes: {80, 120, 160}
- Crossover probabilities: {0.4, 0.6, 0.8}
- Mutation probabilities: {0.001, 0.006, 0.011}
- Elitism: {with, without}
- Roulette wheel selection

This amounts to 54 standard trials (one for each parameter setting combination), yet 540 complete trials (recall, we repeat each standard trial 10 times).

You will need to decide your own convergence criteria and stopping criteria. For example, you might decide to recognize convergence when the average population fitness fails to change by some amount over five generations. Another example would be to recognize convergence when there is no improvement in the best individual after five generations. A possible stopping criteria might be to simply terminate the GA after 30 generations. Just make sure that whatever you use, it's reasonable.

Each trial constitutes a reliability run. In a reliability run, we run the GA on each of the 1023 M+, symptom set combinations. Base line statistics we want to track include (you may include additional statistics):

- Optimal Reliability: number of times the GA found the optimal solution divided by 1023.
- Runner Up Reliability: times the GA found the runner up solution / 1023.
- 2^{nd} Runner Up Reliability: times the GA found the 2^{nd} runner up solution / 1023.

Examples of additional statistics you may want to track include:

- Generation when the best individual was found.
- Average population fitness when convergence occurred.

Results should be organized and presented in a "laboratory report" that resembles a draft research publication. You will want to include an abstract, introduction, background, experiment setup, justification, visual and narrative results, conclusions, implications, and possible future work.

Submit your first draft as soon as possible. It's typical for initial drafts to be returned for revisions in order to become acceptable.