

# Confidence value predication of called genetic bases using a fuzzy predication system.

Brian D. French, Cristian Domnisoru, Habtom Resson, Mohamad T. Musavi

*University of Maine  
Department of Electrical & Computer Engineering  
Intelligent Systems Laboratory  
201 Barrows Hall, Orono, ME 04469*

## Abstract

*We propose a novel solution for determining the confidence of a particular base call within a sequence being called correctly. The confidence value measure has proven to be an important tool when it comes to assembling a final DNA sequence, which is apparent from their explicit use in sequence-assembling algorithms. Though the current model for determining confidence value has been employed successfully and has gained wide acceptance, utilizing just one static model for all sequences, in light of new sequencing techniques and process variability, leads to an inflexible model. Our model employs fuzzy logic, which forwards flexibility, adaptability and intuition through the use of linguistic variables and fuzzy membership.*

## Keywords

Confidence Value, Base Calling, Fuzzy Logic, Consensus Algorithms.

## 1. Introduction

### 1.1 Motivation

The unspoken goal of research into base-calling algorithms is to attain 100% accuracy, which would completely obviate the need for any intervention to determine the correct sequence. But given the current state of the art, more pragmatic goals for the next few years are for error rates around <1%. Although this is very low, it is certainly not zero. Which means intervention, including consensus algorithms [1]

and human operators, cannot be eliminated anytime soon. Paradoxically, as base-calling algorithms' error rates drop, the smaller base-call errors can become obfuscated and difficult to locate. That is why assembling algorithms and human operators use the confidence value measure to determine how well the base-calling algorithm has performed at particular base calls, which clearly makes it easier to uncover potential errors and to correct them, thus increasing throughput of genetic sequencing. It is unmistakable that confidence value prediction has emerged as an essential tool in contemporary genome mapping projects.

While developing a novel adaptive base calling strategy we wanted to add a confidence value measure feature. One method [2] that has been utilized generates a value that is predictive of the true error of a base-call sequence by means of a lookup table that contains trace features as the index and the confidence value as the result. Though this method has gained wide acceptance, employing just one lookup table for all sequences leads to an inflexible model. As sequencing machines, sequencing chemistry, and base-calling algorithms improve; models must adapt in order to reflect the technological progress. Even worse there can be variations between sequencing machines, which can compromise the model rendering it not truly predictive of the error.

### 1.2 Previous work

By far the main body of work accomplished in the area of confidence value was done primarily

in support of the development of the *phred* base calling system [2]. The goals of their work were to produce a predictive confidence value measure that would directly correlate to true trace error rates and to have this value be useful in discriminating where possible errors are located. By employing a greedy algorithm on a large data set they were able to create their model (a lookup table). The input space consists of four trace data features. The output space is the resulting confidence value, which should relate to the true error probability of a base call by the following equation.

$$q = -10 \cdot \log_{10}(e)$$

Where  $q$  is the confidence value and  $e$  is the error probability. The error value was log transformed because the error probabilities they were working with were so small.

One contributing measure that their system introduced was the discrimination power of the confidence value. That is, how well does the system perform at locating the regions with errors and regions that are error free? This is best illustrated by an example. Suppose we have a base call sequence that contains 5 errors within a 100 base trace. A perfectly correct confidence value for each base call could be the value 13. This number comes from the fact that each base call is given an error probability of 5/100. So the confidence value is calculated by  $(-10) \cdot \log_{10}(5/100) \approx 13$ . Even though the confidence value is correlated to the error rate it doesn't give us any idea where the errors are located. A better example would be splitting the 100 bases in half into two groups of 50 bases each. Suppose also that we find the first half has 4 errors and the second half has 1 error. This would mean that the bases in the first half could all be assigned the error probabilities 4/50, while the other half of the bases could be assigned 1/50, thus corresponding to confidence values of 11 and 17 respectively. We see that this example does a better job at discriminating the poor region (the first half) from the region that performed well (the last half). This leads to their definition of discriminating power.

$$P_r = |B_r| / |B|$$

Where  $P_r$  is discriminating power factor for error rate  $r$ .  $|B|$  is the number of bases in set  $B$  and  $|B_r|$  is the number of bases in  $B_r$ .

*$B_r \subseteq B$  such that the error rate in  $B_r \leq r$  and whenever  $B_r$  includes a base call  $b$ ,  $B_r$  includes all other base calls where their error probabilities are less or equal to that of  $b$ 's.*

The last condition in the above statement insures that we have the maximum number of  $|B_r|$  that we can have. So we can see that we would want the largest set of  $B_r$  we can have for small error rates  $r$ . In effect spreading out the error probabilities, thus increasing its discrimination properties.

Finally let's examine the trace features used in *phred* base calling system:

1. *Peak Spacing: This feature is the ratio of the largest peak-to-peak ratio to the smallest peak-to-peak ratio within a window of seven peaks.*
2. *Uncalled/called ratio: This feature is the ratio of the amplitudes of the largest uncalled peak to the smallest called peak.*
3. *Uncalled/called ratio2: The only difference between the second and third features is the window size used, which was seven and three peaks respectively.*
4. *Peak resolution: This final feature is the number of bases between the current base and the next unresolved base (in the *phred* system an unresolved base call is labeled with an N).*

We can see that by utilizing a table we introduce discontinuities between table entries, which could cause resolution problems for input features near the edges of the entries. Also the nature of the greedy algorithm that they used may, depending on the training data, produce large sections that lump in large ranges parameters together. This means if data comes along later with true variations in error rates within the large sections, the confidence value could be flattened. Also this system does not allow for the model to adapt to newer base

calling techniques, variations in sequencing machines, and deviations in other quality control measures all leading to a very inflexible model. Finally the model doesn't forward any intuition with the trace features as they relate to the confidence value.

### 1.3 Fuzzy Logic

The incentive for using a fuzzy model is so that we can take advantage of the linguistic variables feature inherent in fuzzy logic. Since linguistic variables are based from natural language, we are able to fashion models from the insight of experts even if they have minimal mathematical backgrounds. The result is an intuitive model instead of a mystical black box. At the same time, fuzzy logic is powerful enough to model complex non-linear systems that are ubiquitous in contemporary predictive and control domains. Finally, after a rough model is produced with the help of experts, the system can be improved further by tuning the model through the use of various mathematical techniques such as genetic algorithms and clustering.

Fuzzy Logic is a natural extension of traditional Boolean Logic. To illustrate this point we should first entertain what is meant by traditional set membership with respect to Boolean Logic. In this case a value either has membership or does not membership within a defined set. The step function in figure 1 is a great illustration of two-valued logic where an object's membership is either alive or not alive. For example a rock is not alive and under no circumstances is it alive. To contrast this we can sample a certain population of people on whether or not is it warm outside over a varying degrees of temperatures and plot the number of people who think it is warm versus temperature. The result would most likely be a Gaussian membership of degrees of 'warm' as seen in Figure 2, thus reflecting the naturally ambiguous term 'warm'. Instead of a value having a membership of 0 or 1, the degree of membership in Fuzzy Logic lies between 0 and 1 inclusively, which allows for a value of .5 'warm' to be a possible value. We can see that the membership function in Figure 2 captures the essences of

what the linguistic variable 'warm' means much better than two-valued logic ever could.

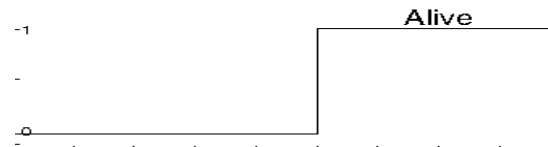


Figure 1: Boolean set of being Alive. Notice the membership is either 0 or 1.

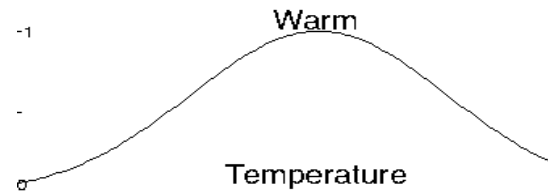


Figure 2: Fuzzy set of being Warm. Notice the membership degree is along the interval [0,1].

Now that we have examined a Fuzzy variable it seems natural to discuss Fuzzy operators.

$$\begin{aligned}
 A \text{ and } B &\Leftrightarrow \mu_{A \cap B}(x, y) \Leftrightarrow \wedge(\mu_A(x), \mu_B(x)) \\
 A \text{ or } B &\Leftrightarrow \mu_{A \cup B}(x, y) \Leftrightarrow \vee(\mu_A(x), \mu_B(x)) \\
 \text{Not } A &\Leftrightarrow \mu_A(x)^c \Leftrightarrow 1 - \mu_A(x).
 \end{aligned}$$

Here we define both the 'and', 'or', and 'not' operators. We see that the crisp variables  $x, y$  can be over different universes of discourse. For example  $x$  could be the absolute temperature measured in degrees (a universe of discourse) and  $y$  could be the humidity measure in percent (another universe of discourse). The  $\mu_A(\bullet)$  function return the degree of membership for a crisp value in fuzzy set A, which would be a value between 0 and 1 inclusively. The union and intersection of the degrees of memberships result in 'or' and 'and' operators respectively. The compliment results in the 'not' operator. The unions and intersection of fuzzy sets can be defined in many ways, but they are most commonly defined as the *maximum* function and *minimum* functions respectively. The not operator is commonly defined as  $1 - \mu_A(x)$ .

The Fuzzy model we used employed implications in the form of if-then rules [3].  $A$  implies  $B$  can be determined by finding the *minimum* of fuzzy sets A and B.

$$\text{If } x \text{ is } A, \text{ then } y \text{ is } B \Leftrightarrow A \cap B$$

Lastly we need to find the crisp output by finding the center of gravity of the aggregate of all the results of the implication. We apply the center-of-gravity method because the aggregate implication results in a new fuzzy output set, while in fact we need a single crisp output. Applying the *maximum* function to all the resulting implications performs the aggregation.

The entire Mamdani inference is outlined below.

1. Map the crisp inputs to their fuzzy memberships.
2. Aggregate the antecedents of the if-then rules by applying fuzzy operators.
3. Perform the implication.
4. Aggregate all the results of the if-then rules.
5. Map the aggregate set to a crisp output by apply the center-of-gravity method.

## 2. Fuzzy Model Implementation

The initial motivation for developing the model was so our base-calling algorithm could have a confidence value to complete the system. We wish to collect trace features and use them as inputs to the fuzzy model. In this fuzzy model we are able to collect three trace features from our base-calling algorithm. The first feature is the *height*, which is simply the height of peak. The second is the *peakness*, which is a measure related to the concavity at the top of a peak. The final feature is the base spacing, which are just the location differences from one peak to another. In addition to determining if a base is being called correctly, we added a second fuzzy model to produce a second confidence value related to base insertion/deletion errors. Another thing to keep in mind here is that our base-calling algorithm not only identifies the most likely base call candidate within a local position, but it is also capable of identifying a second most likely base call candidate. This gives us a starting point from which we can define input variables to our fuzzy system. Later on we will define rough membership functions along with some intuitive if-then rules. These can always

be tuned later using clustering, genetic algorithm techniques, and neural-fuzzy techniques.

### 2.1 Confidence Fuzzy Model

Below is the list of inputs/outputs of the fuzzy system along with a short description.

- $P_{called}$ : Peakness of the base called.
- $P_{2nd}$ : Peakness of the 2<sup>nd</sup> candidate.
- $H_{called}$ : Height of the base called.
- $H_{2nd}$ : Height of the 2<sup>nd</sup> candidate.
- $\Delta S_{previous}$ : The difference between the actual and predicted distance to the previous base.
- $\Delta S_{next}$ : The difference between the actual and predicted distance to the next base.
- $C_p$ : Confidence value of the base called relative to the peakness variable.
- $C_H$ : Confidence value of the base called relative to the height variable,
- $C_{\Delta S}$ : Confidence value of the base called relative to  $\Delta S$  variable.
- $C$ : Overall confidence value of the base called.

In figure 3 we can see that the confidence model consists of three fuzzy sub-systems along with one main fuzzy system. The 6 inputs for height ( $H_{called}$ ,  $H_{2nd}$ ), peakness ( $P_{called}$ ,  $P_{2nd}$ ), and delta-spacing ( $\Delta S_{previous}$ ,  $\Delta S_{next}$ ) are direct inputs to the fuzzy sub-systems which result in 3 intermediate confidence inputs ( $C_p$ ,  $C_H$ ,  $C_{\Delta S}$ ) that feed into the main confidence system. The result of the main system is the final confidence value ( $C$ ) for the given trace features.

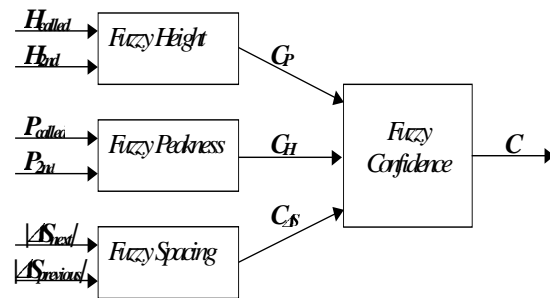


Figure 3: Model Overview

The next thing to consider was the membership functions. What function should the fuzzy sets take, and how many regions should each universe of discourse be divided up into? There is truly no perfect solution, but since we are setting up a general system to begin with, a good shape to begin with for member functions are Gaussians. Each fuzzy variable was arbitrarily divided up into 3 or 4 fuzzy sets, which was based on intuition. For example the height was separated into 3 fuzzy sets *low*, *medium*, and *high*, as seen in figure 4 below. We see the centers of the Gaussian functions were spread evenly over the universe of discourse and that all input variables were normalized. The widths of the Gaussian functions with three and four member functions were set to have  $\sigma$  values of .2123, and .1416 respectively. This gives us a solid starting point for the confidence value model.

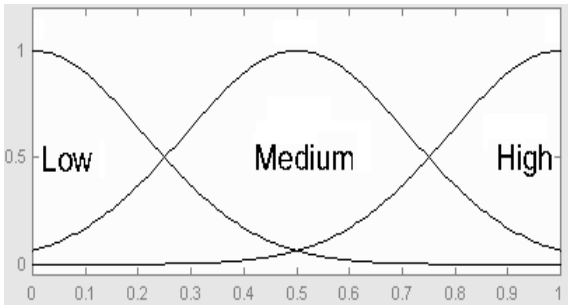


Figure 4: Height fuzzy sets

### 2.3 If-Then Rules

The final thing to sketch out is a starting point for the if-then fuzzy rules that were gleaned from intuition and experience. All the implications, fuzzy operators within the antecedents, and implication aggregation follow those guidelines drawn out in the Mamdani method described in the Fuzzy Logic section 1.3. In these if-then tables we use the fuzzy sets *Very Low (VL)*, *Low (L)*, *Medium (M)*, *High (H)*, *Very High (VH)*. We can see from Table 1 that we can extract the if-then rules. An example if-then rule is: *if  $H_{called}$  Low and  $H_{2nd}$  is high then the output is Very Low.*

Table 1: Height fuzzy if-then rules for fuzzy sub-system

		$H_{2nd}$		
		Low	Medium	High
	Low	<b>M</b>	<b>L</b>	<b>VL</b>
	Medium	<b>H</b>	<b>L</b>	<b>VL</b>
	High	<b>VH</b>	<b>H</b>	<b>L</b>

Table 2: Peakness fuzzy if-then rules for fuzzy sub-system.

		$P_{2nd}$			
		Concave	Flat	Medium	Sharp
	Flat	<b>M</b>	<b>L</b>	<b>VL</b>	<b>VL</b>
	Medium	<b>H</b>	<b>M</b>	<b>L</b>	<b>VL</b>
	Sharp	<b>VH</b>	<b>H</b>	<b>L</b>	<b>L</b>

Table 3: Spacing fuzzy if-then rules for fuzzy sub-system.

		$ \Delta S_{previous} $		
		Small	Medium	Large
$ \Delta S_{next} $	Small	<b>VH</b>	<b>H</b>	<b>M</b>
	Medium	<b>H</b>	<b>M</b>	<b>L</b>
	Large	<b>M</b>	<b>L</b>	<b>VL</b>

Table 4: A Sample of the Main system fuzzy if-then rules

$C_P$	$C_H$	$C_{\Delta S}$	$C$
<b>H</b>	<b>L</b>	<b>H</b>	<b>M</b>
<b>H</b>	<b>H</b>	<b>L</b>	<b>H</b>
<b>VH</b>	<b>M</b>	<b>H</b>	<b>VH</b>
...	...	...	...

Table 4 only shows a partial table that contains the 125 ( $5^3$ ) if-then rules. An example if-then rule here might be: *if  $C_P$  high and  $C_H$  is low and  $C_{\Delta S}$  is high then  $C$  is medium.*

### 2.4 Insertion/deletion confidence value

In addition to the traditional confidence value our model includes a supplementary confidence value that predicts the insertion or deletion error in the neighborhood of the base called. We denote this confidence value as  $C_{indel}$ . In this case, a large negative confidence value would mean that the actual spacing is significantly

smaller than the expected one and therefore has a high probability of an insertion error. Similarly, a very large positive confidence value will indicate a high probability for a deletion error near by.

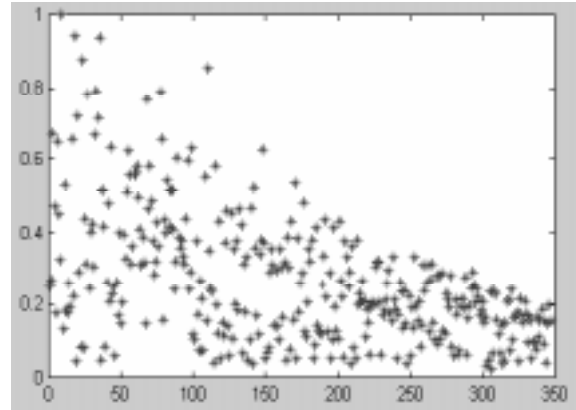
This fuzzy system uses two inputs variables  $\Delta S_{previous}$  and  $\Delta S_{next}$  to calculate  $C_{indel}$ . The linguistic terms for  $\Delta S_{previous}$ ,  $\Delta S_{next}$ , and  $C_{indel}$  will having varying degree of membership in the following fuzzy sets: *large negative (LN)*, *small negative (SN)*, *zero (Z)*, *small positive (SP)*, and *large positive (LP)*. The fuzzy rules for this model are shown in Table 5.

**Table 5:** Insertion/Deletion fuzzy if-then rules

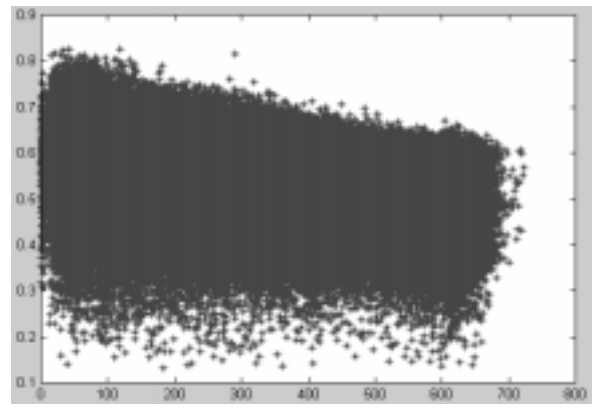
		$\Delta S_{previous}$				
		LN	SN	Z	SP	LP
$\Delta S_{next}$	LN	LN	SN	SN	SN	Z
	SN	SN	SN	Z	Z	SP
	Z	SN	Z	Z	Z	SP
	SP	SN	Z	Z	SP	SP
	LP	Z	SP	SP	SP	LP

## 2.5 Improved Fuzzy Model and Model Tuning

After looking at the *height* and *peakness* data, it is apparent there is a strong dependency on position. This is apparent from Figure 5 when we plot the height versus base-position. This stems from the fact that our base-calling system doesn't scale (in the same way) the data in an effort to preserve features as most other base-calling systems do. It is apparent that if we use the data as-is the confidence value will have a strong dependency on base position, which may inevitably be a correct final result, but does not follow the local spirit in regards *height* and *peakness* in our fuzzy model. This dependency of the confidence values on the base position is depicted in Figure 6 created based on the processing of 800 files. One solution is to send a more even-scaled version of this data. We propose to change the fuzzy model described in the above paragraphs (2.1-2.3) such that the new inputs to the system are ratios of the base call heights and of the peaknesses as opposed to using the absolute values of those variables. In this way, the discriminating power of the Fuzzy Logic system is increased.



**Figure 5:** Normalized Height vs. Base Position in file of typical un-scaled ABI 3700 file.



**Figure 6:** Confidence vs. Base Position from 800 ABI 3700 file using initial fuzzy model. Note here that the confidence ranges from 0 (Not Confident) to 1 (Very Confident) linearly.

Although we don't anticipate changing the type of membership functions, it is possible for the number of fuzzy sets to increase and the centers and widths of these Gaussian functions to change as well. This would take place in a tuning phase where we would identify numbers and centers of regions through clustering techniques such as fuzzy c-means clustering that can be performed on the input and output space. The results would directly relate to new member function locations and widths. These membership functions could be tuned further using genetic algorithms. In addition, if-then rules that we have established may be added or removed using neural-fuzzy techniques in an effort to further improve the model.

### 3. Results & Conclusion

In this paper we have proposed a novel Fuzzy Logic approach for estimating confidence values for DNA base calling. In addition to the traditional base call confidence value, a new indicator is being proposed to assess the probability that a base call is an insertion error or that in the corresponding position there is an indication of a deletion error. We have more than 1100 ABI 3700 type files to work with. We will use 2/3 of the files for tuning our fuzzy model and the rest for testing. Measures that will be important in determining performance will be how close the error probabilities for base calls match the true error probabilities. The second major factor will be comparing error rates and number of base calls that fall within those error rates. This will provide information needed to calculate the discriminating power factor outlined in section 1.2.

### Acknowledgements

The work presented in this paper was in part supported by the grant "An Accurate DNA Base Caller," NSF award # DBI:0090738

### References

- [1] Bonfield, J.K. and Staden, R. (1995). The application of numerical estimates of base calling accuracy to DNA sequencing projects. *Nucleic Acids Research*, **23**, 1406-1410.
- [2] Ewing, B. and P. Green. (1998). Base-calling of automated sequencer traces using *phred*: II. Error probabilities. *Genome Research*, **8**, 186-194.
- [3] Mamdani, E.H. (1977). Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Transactions on Computers*, **26**, 1182-1191.