A Comparison Between FS-NEAT and FD-NEAT and an Investigation of Different Initial Topologies for a Classification Task with Irrelevant Features

Evgenia Papavasileiou and Bart Jansen
Department of Electronics and Informatics
Vrije Universiteit Brussel
Pleinlaan 2, 1050 Brussel, Belgium AND
iMinds
Technologiepark-Zwijnaarde 19, 9052 Gent, Belgium
Email: {epapavas,bjansen}@etrovub.be

***Abstract*—Feature Selective Neuroevolution of Augmenting Topologies (FS-NEAT) and Feature Deselective Neuroevolution of Augmenting Topologies (FD-NEAT) are two popular methods for optimizing the topology and the weights of Artificial Neural Networks (ANNs) simultaneously with performing feature selection. However, no study exists that systematically investigates their performance on the exclusive-or (XOR) problem with increasing complexity. Moreover, it is unknown whether the choice of a different initial topology of the ANNs would influence the performance of the two algorithms. For this reasons, this paper investigates the performance of FD-NEAT and FS-NEAT in terms of accuracy, number of generations required for their convergence to the optimal solution and their ability of selecting the relevant features in artificial datasets with irrelevant features. The comparisons are performed based on hypothesis tests (Wilcoxon rank sum test, $p<0.05$). The results show that the choice of the initial topology can affect the performance of the two algorithms, resulting in higher accuracy, faster convergence and better feature selection abilities.**

I. Introduction

NeuroEvolution of Augmenting Topologies (NEAT) [1] is a method that uses genetic algorithms (GAs) for learning both the connection weights and the topology of Artificial Neural Networks (ANNs). Its successor, called Feature Selective NEAT (FS-NEAT) [2], performs feature selection simultaneously with the optimization of the topology and the connectivity of the underlying nodes. Feature Deselective NEAT (FD-NEAT) [3] is an alternative feature selection method that has shown great promise for classification tasks [3]–[5], although NEAT and FS-NEAT stem from reinforcement learning.

Although a study exists that compares FD-NEAT and FS-NEAT on artificial datasets [3], to our knowledge this is the first time that FD-NEAT and FS-NEAT are systematically compared on the simple, non linear XOR problem with multiple dimensions (5, 10 and 20 inputs). In addition, there are no studies that systematically investigate the effect of modifying the initial topologies on these algorithms. The goals of this paper are to compare the performance of FS-NEAT and FD-NEAT in the XOR problem with irrelevant features as well as to investigate the effect of changing the initial topologies on the classification of the samples, their feature selection abilities and the time required for convergence.

This work is supported by the Fonds Wetenschappelijk Onderzoek-Vlaanderen (FWO).

This paper is organized as follows. Section II describes the state of the art and the methods that will be compared. Section III presents the experimental setup required for the comparison of the existing methods as well as the experiments carried out for investigating the importance of the initial topology. Section IV presents and discusses the acquired results and finally section V concludes this paper.

II. Methods

A. Neuroevolution

Neuroevolution (NE) is a learning method that uses GAs to optimize the parameters of ANNs. NE methods are distinguished based on the type of encoding the ANNs into the genotype (direct or indirect) and whether they optimize only the connection weights or both the connection weights and the topology of the ANNs. NE started as a method to evolve only the connection weights of fixed topology ANNs [6]–[10]. Later Topological and Weight Evolving Artificial Neural Networks (TWEANNs) [1], [11]–[13] optimized both the weights and the topology of the underlying nodes. TWEANNs offered significant advantages, as finding the optimal topology of an ANN requires time-consuming evaluations of potential architectures. Especially, in complex problems, the number of neurons and connections that are required scales with the complexity of the problem [14] and thus the manual definition of the optimal topology is even more difficult. Moreover, the topology defines the size of the search space; selecting a fixed topology smaller than the optimal means that the search is performed in a lower dimensionality and thus the optimal solution will not be found. On the other hand, selecting a bigger topology than the optimal one implies that the search is performed in an unnecessarily high dimensionality. However, in TWEANNs the problem of identifying the right ANN topology is tackled by their ability to automatically discover the optimal architecture.

B. NEAT

NEAT [1] belongs to the TWEANN methods enabling the learning of the structure of ANNs at the same time it optimizes their connectivity weights. It was found to outperform other fixed topology methods on benchmark reinforcement learning tasks [15]. The proficiency of the method is attributed to its three main innovations for which ablation studies in [1] showed that each of the introduced components is crucial to its performance. These three main features are described in the following subsections.

1) Historical markings: NEAT encodes the ANNs with a set of connection genes and a set of node genes. The ANNs' structure is evolved over the generations by mutation and crossover operations. The evolution starts minimally with simple networks whose structure becomes gradually more complicated over generations. A new gene is introduced in the genome by structural mutations that add a new connection or a new node in the network. The connection weights can also be changed by mutation operators. To perform crossover, the system must line up matching genes between the individuals of the population. NEAT facilitates this with historical markings, i.e by assigning an incrementally increasing innovation number when a new gene is added to the genome thus functioning as a chronological parameter. The innovation number is inherited by the offspring through crossover maintaining the historical origin throughout evolution.

2) Speciation: The introduction of a new structure initially reduces the fitness of a network. Without speciation the new individual would compete with the entire population and there is a high probability that it would be replaced before it is given time to be optimized. NEAT protects topological innovations through speciation by allowing individuals to compete within their own niche instead of the entire population. The population is divided into species based on their topological similarity. This is measured based on a distance function defined on the number of excess and disjoint genes and the average differences of weights in the matching genes. Excess, disjoint and matching genes are defined based on the genome alignment that is facilitated with the historical markings discussed in the previous subsection.

3) Search space minimization: NEAT starts the evolution with a population of minimal structures, i.e. with a population of ANNs with no hidden layers and fully connected input and output layers with random weights. It gradually evolves more complex networks by introducing new nodes and connections through structural mutations. These topological innovations are maintained only if they are found to increase the network's performance, i.e. NEAT tends to discover small networks without unnecessary structure.

C. FS-NEAT

FS-NEAT [2] is a NE method that extends NEAT in performing feature selection simultaneously with learning the ANNs topology and weights. The selection of the right set of features is performed without requiring human expertise and without relying on metalearning or labeled data. It follows the three basic principles of NEAT described in section II-B; historical markings, speciation and starting with minimal structure with the exception that it starts the evolution even more minimally than NEAT. The initial ANNs' topologies in NEAT consist of an input and an output layer with all the inputs directly connected to the output nodes. The evolvable topologies in FS-NEAT start also from networks with no hidden layers but only with one random input connected to one random output node. In the course of generations more inputs will be connected but only the connections that come from the useful (relevant) inputs will tend to survive, performing in this way implicit feature section.

D. FD-NEAT

FD-NEAT [3] is an extension to original NEAT, i.e. it follows the basic principles of NEAT of historical markings, speciation and minimal starting of the topology. It functions similarly to FS-NEAT in terms of performing implicit feature selection simultaneously with topology and weight learning. The main difference between FS-NEAT and FD-NEAT lies in the way that feature selection is performed. FD-NEAT starts with the same minimal topologies as NEAT and drops irrelevant inputs throughout the evolution. This is done by a new mutation operator that functions only in the input layer. Only the inputs that result in increasing the performance of the individual tend to survive and in this way FD-NEAT performs implicit feature selection.

III. Experimental Set-up

A. Dataset

The artificial datasets that we construct are based on the XOR dataset. The XOR problem is the simplest representative of a non-linearly separable hypothesis space and constitutes one of the first datasets a researcher would consider to verify the success of their approach [1], [3], [16]. The original XOR dataset consists of two inputs (known also as attributes or features), one output and 4 samples with values 00, 01, 10 and 11 and outputs 0, 1, 1, 0 respectively, i.e. the output is true when an odd number of attributes is true and false otherwise. In order to be able to investigate the feature selection and classification abilities of FS-NEAT and FD-NEAT we need to construct an artificial 2 out of k dataset (referring to as $2/k$), where $k \in \{5, 10, 20\}$. The 2 inputs will be the relevant features and the remaining $k - 2$ inputs will be assigned to irrelevant booleans. In order to include these irrelevant features and investigate the classification and feature selection abilities of these machine learning methods, we need to increase the number of samples to avoid imposing any bias, as the larger the dataset the less probable it is that there is an underlying correlation between the randomly generated data and the output. The resulting dataset is a challenging task for a feature selection algorithm, because it is constructed in such a way in

TABLE I
PARAMETERS FOR FS-NEAT AND FD-NEAT

Parameter	Value	Meaning
Population Size	150	The number of individuals in the population
P_r(add node)	0.03	The probability that a new node will be added
P_r(add link)	0.05	The probability that a new connection will be added
P_r(mutate weight)	0.9	The probability that the weight of a connection will change
P_r(remove input connection)	0.1	The probability that connections from the input layer will be removed (in FD-NEAT)
Coefficient 1	1.0	Compatibility coefficient of excess genes
Coefficient 2	1.0	Compatibility coefficient of disjoint genes
Coefficient 3	0.4	Compatibility coefficient of average weight difference
Compatibility threshold	3.0	Compatibility threshold for speciation
Max generations	250 450	The max number of allowed generations in: the 2/5 2/10 XOR the 2/20 XOR
Crossover percentage	0.8	The portion of individuals in the population to participate in crossover
P_r(crossover interspecies)	0.001	The probability that the two parents of crossover will belong to different species
P_r(multipoint crossover)	0.6	The 1-P_r that the weights of the offspring are the average weights of the parents

order each of its individual attributes to be equally informative for predicting the output.

B. Experiments

The purpose of the experiments described in this section is to compare the performance of FD-NEAT and FS-NEAT in different settings of initial topologies. Each experiment is run as ten fold cross validation, it is repeated for ten times and the results are averaged over the independent runs.

1) Genetic Algorithms Settings: The fitness function of FD-NEAT and FS-NEAT is the same as the one used in NEAT at the XOR problem [1]. Its definition is based on the error between the output of the ANN and the correct output of the training set and is shown in equation 1.

$$Fitness = (N - \sum_{i=1}^{N} |O_i - T_i|)^2 \tag{1}$$

where N is the size of the training dataset, O_i the output of the ANN on the i^{th} pattern of the training set and T_i the real output that corresponds to the i^{th} pattern of the training set. The sum of the error between the network's output and the real output is subtracted from the number of training samples N so that higher fitness would indicate better network structure. Finally, this difference is squared in order for the fitness to get proportionally a higher value as the network approaches the solution. The evolution in FD-NEAT and FS-NEAT stops when the fitness gets its highest value, i.e. when all the input patterns of the training set are classified correctly or when the maximum allowed number of generations is reached.

The parameters used for the setting of FD-NEAT and FS-NEAT are presented in Table I.

2) Measures for evaluation of the performance : The performance of FD-NEAT and FS-NEAT is evaluated based on the following criteria:

- Accuracy on the test set: it measures how successful the method is in finding the right relationship between input and output.
- Number of generations: it measures how fast the algorithm converges to the solution.

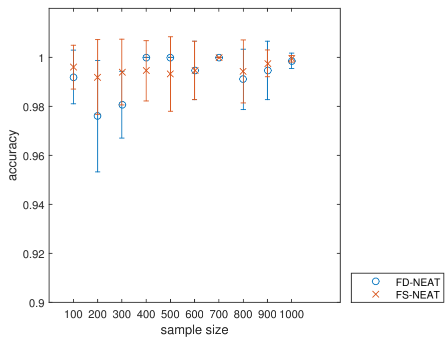
For the evaluation of the feature selection abilities of each method we employ the measures of frequency and sum of absolute weights that were also employed in [3] and the thresholded frequency described below.

- Frequency: the fraction of individuals in the population that include the correct set of features. The relevant features are expected to be selected with higher frequency than the irrelevant ones. For the calculation of this measure, every input that has a connection initiating from it with a weight of absolute value greater than 0 is considered selected.
- Thresholded frequency: the fraction of individuals in the population that include the correct set of features. For this measure an input is selected only if the absolute weight that initiates from it is greater than a predefined threshold. Here we choose a threshold equal to 0.5. The value of this threshold is found experimentally by examining the weights of the connections of the ANNs that are learned by the two algorithms. The difference between the frequency of the relevant inputs and the frequency of irrelevant ones is expected to be higher, as the successful algorithms are expected to assign higher weights to the relevant inputs compared to the irrelevant ones.
- Sum of absolute weights: the summation of the absolute weights of the connections that initiate from each input node. This measure is calculated on the best network of the final population and not on the whole population like the (un)thresholded frequency. The connections that initiate from the relevant inputs are expected to have higher weights compared to the connections of the irrelevant inputs, thus their absolute sum is expected to be higher.

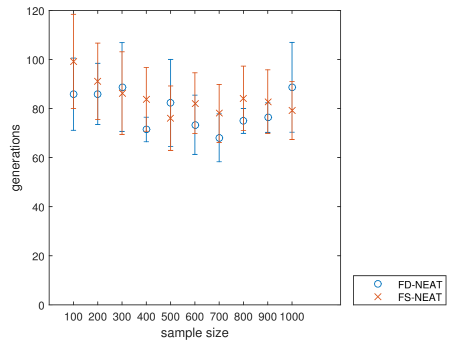
The algorithms can be considered as having a feature selection ability when there is a statistical difference between the values of frequency or of sum of absolute weights between relevant and irrelevant inputs.

3) Comparison of conventional FD-NEAT and FS-NEAT: The purpose of the initial experiments is to investigate the behaviour of conventional FD-NEAT and FS-NEAT in the 2/5 XOR problem. We repeat the experiments for an XOR dataset of increasing number of samples from 4 to 100 with a step of 4 and then from 100 to 1000 with a step of 100. From these experiments we want to identify the ideal number of samples for which the accuracy on the test set is higher, the number of generations is lower and the frequency of the relevant features is higher than the frequency of the irrelevant ones.

4) Investigation of the importance of the initial topology: Using this selected number of samples we perform the second



(a) Accuracy on the test set



(b) Number of generations

Fig. 1. Results of the comparison of conventional FD-NEAT and FS-NEAT for increasing number of samples in the 2/5 XOR problem

set of experiments in order to investigate the importance of the initial topology. The choice of the starting topology is very important and according to our knowledge it has not been studied in a systematic manner. As it was already mentioned, both FS-NEAT and FD-NEAT start minimally. However, all non-trivial, real-life problems are characterized by a non-linear decision boundary and thus require at least one additional layer. This means that the network that needs to be evolved will have at least one additional hidden layer. For this purpose, we include one hidden layer in the initial topology and we vary the number of hidden nodes from 1 to 10, following an approach used with NEAT in [17].

The goals of these experiments are: 1) We want to examine the influence of the choice of the initial topology on the performance of FD-NEAT and FS-NEAT. Towards this purpose, we perform hypothesis tests (Wilcoxon rank sum test, $p<0.05$) to investigate if there is a statistical difference for the values of accuracy and number of generations required for convergence among the topologies of different number of hidden nodes, i.e. we want to examine if there is a good topology for starting the evolution that results in better performing networks compared to starting the evolution with a different topological setting. 2) We want to examine the feature selection ability of each of the two algorithms for each different chosen initial topology. This means that we perform hypothesis tests (Wilcoxon rank sum test, $p<0.05$) to investigate if there is a statistical difference between the relevant and irrelevant features for the values of frequency (unthresholded and thresholded) and of the sum of absolute weights. 3) We want to compare the performance of FD-NEAT and FS-NEAT for problems of increasing complexity, so we perform hypothesis tests (Wilcoxon rank sum test, $p<0.05$) to examine if there is a statistical difference in the performance between the two algorithms. This comparison concerns the values of all the measures (accuracy, number of generations, (un)thresholded frequency and sum of absolute weights).

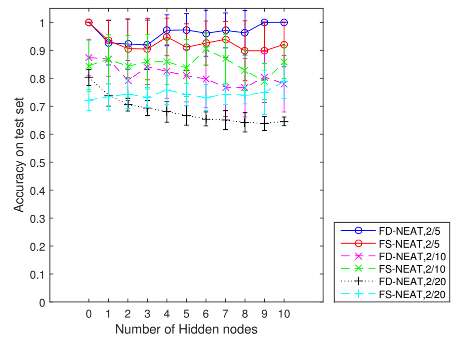


Fig. 2. Results of accuracy for FD-NEAT and FS-NEAT for the 2/5, 2/10 and 2/20 XOR problems

IV. Results and Discussion

The results of the experiments shown in Figures 1-5 show the mean values and the standard deviations calculated over the independent runs.

A. Results on comparing conventional FD-NEAT, FS-NEAT

From the results shown in Figure 1, it is observed that the best accuracy on the test set (mean=1, std=0) is accomplished for FD-NEAT for samples=400, 500 and 700 and for FS-NEAT for samples=700. By taking into account the number of generations and the frequency of selected features in these sample values, we choose the sample size=700 to perform the further experiments. For this sample size the number of generations required for convergence is 68.08±9.79 for FD-NEAT and 78.02±11.76 for FS-NEAT.

B. Results on the investigation of the initial topology

1) Accuracy: In Figure 2 we present the results of accuracy for both FD-NEAT and FS-NEAT for the 2 out 5, 10 and 20 XOR problem. For the 2/5 XOR problem FD-NEAT and FS-NEAT are able to solve the problem with accuracy (1±0) for initial topologies of 0, 9 and 10 hidden nodes and 0 hidden nodes respectively. Comparing FD-NEAT and FS-NEAT, FD-NEAT seems to perform better but a statistical difference only exists for 8-10 initial hidden nodes. For the 2/10 XOR problem, FD-NEAT performs better for initial topologies of 0 and 1 hidden nodes (0.87±0.06). For FS-NEAT the worst topology is the one of 9 initial hidden nodes, whereas the topology of 6 hidden nodes seems to bring better results.

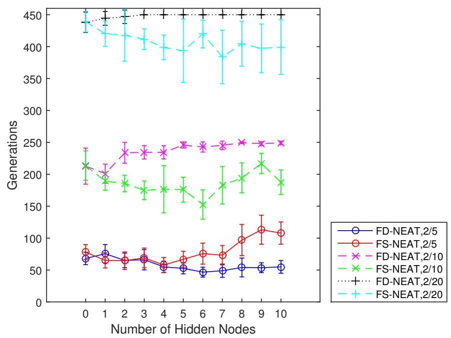


Fig. 3. Results of number of generations for FD-NEAT and FS-NEAT for the 2/5, 2/10 and 2/20 XOR problems

Comparing FD-NEAT and FS-NEAT, we can observe that FS-NEAT tends to solve the problem with higher accuracy than FD-NEAT, but the two algorithms are statistically different only for the initial topology of 6 hidden nodes. Finally, at the 2/20 XOR problem, FD-NEAT performs better for an initial topology of 0 hidden nodes (0.80$\pm$0.03), while this topology is the worst for FS-NEAT (0.72$\pm$0.03). For FS-NEAT, the topologies of 10 hidden nodes are better (0.79$\pm$0.06) than most of the other topologies. Comparing the performance of the two algorithms, FS-NEAT is better than FD-NEAT for topologies of 2 to 10 initial hidden nodes.

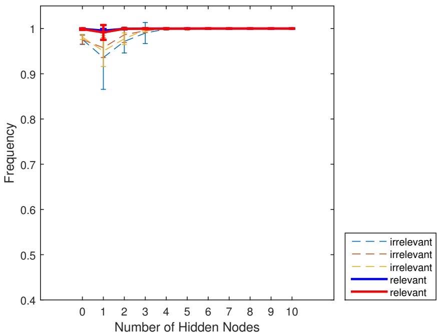
From the above results we observe that the ability of the algorithms to solve the problem decreases as the complexity of the problem increases, e.g. the accuracies in the 2/20 problem are much lower than the accuracies in the 2/10 which are lower than the ones of the 2/5. Overall, FS-NEAT seems to perform better than FD-NEAT for most of the chosen initial topologies. FD-NEAT shows its best performance for an initial topology of 0 hidden nodes. On the other hand, a conclusion on FS-NEAT's optimal size of the hidden layer cannot be drawn as it depends on the problem at hand, e.g. in the 2/10 problem a large hidden layer yields the worst performance whereas in the 2/20 problem it results in a good one.

2) Number of generations: Figure 3 shows the number of generations required by FD-NEAT and FS-NEAT to converge to the solution, or the number of generations that they were allowed to evolve. At the 2/5 XOR problem, FD-NEAT converges in statistically more generations for topologies of few hidden nodes (0-3), whereas FS-NEAT requires more generations for bigger initial topologies (8-10 hidden nodes). Comparing the two algorithms, FD-NEAT requires fewer generations than FS-NEAT. As the number of hidden nodes in the initial topology increases, the number of generations required for the convergence of FS-NEAT increases whereas the number of generations required for FD-NEAT remains almost the same, or slightly decreases. Indeed, a hypothesis test confirms that FS-NEAT converges in more generations than FD-NEAT for an initial topology of 5-10 hidden nodes. At the 2/10 XOR problem, as the initial number of hidden nodes increases, FD-NEAT is not able to converge as it reaches the limit of 250 generations. In fact, the topologies of 0 and 1 initial hidden nodes are statistically better. On the other hand, FS-NEAT is able to converge independently of the choice of the initial topology. In general, it converges in statistically fewer generations than FD-NEAT for almost all the chosen initial topologies (2-10 hidden nodes). Finally, at the 2/20 inputs problem, the topology of 0 hidden nodes is the best for FD-NEAT and the worst for FS-NEAT. In general, FD-NEAT cannot converge within the predefined limit of 450 generations, whereas FS-NEAT converges for most topologies.

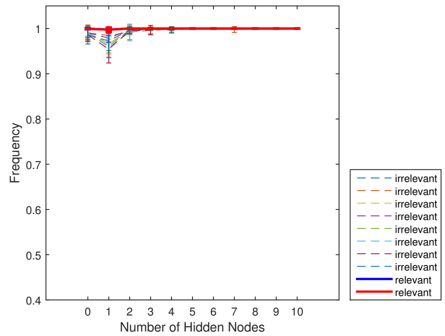
From the above analysis we can observe that the harder the problem, the more generations are required for convergence to the optimal solution. For most of the topologies in the 2/10 and 2/20 inputs problems, FD-NEAT was not able to converge within the predefined maximum number of generations. This can also explain why the accuracy of FD-NEAT in these cases is lower than the accuracy of FS-NEAT (Figure 2).

3) Frequency: In Figure 4 we present the results of thresholded and unthresholded frequency. At the 2/5 XOR problem (Figures 4a, 4d, 4g, 4j), FD-NEAT's ability in distinguishing between relevant and irrelevant inputs is better for topologies of 1-2 hidden nodes, while the topologies of 0 and 9 hidden nodes are the worst for FS-NEAT. In the 2/10 inputs problem (Figures 4b, 4e, 4h, 4k) the initial topology which results in higher difference between relevant and irrelevant features is the one of 1 hidden nodes for FD-NEAT, whereas the topologies of few hidden nodes (0-2 nodes) are the worst for FS-NEAT and the topology of 6 hidden nodes seems to provide better results. At the 2/20 inputs problem (Figures 4c, 4f, 4i, 4l), FD-NEAT performs better for topologies of 0 and 1 hidden nodes (Figures 4c, 4i). For FS-NEAT different results are obtained between frequency and thresholded frequency; on the one hand by taking into account the results of the unthresholded frequency (Figure 4f), the topology of 10 hidden nodes yields better results than most other topologies and on the other hand topologies of few hidden nodes (0-2) yield better results (Figure 4l).

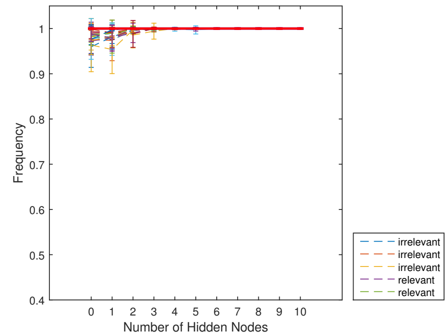
From the results in Figures 4a, 4b, 4c, 4g, 4h, 4i we observe that as the number of hidden nodes increases, FD-NEAT is not able any more to distinguish between the relevant and the irrelevant inputs, as the frequency of all the inputs converges to 1 (i.e. all the inputs are selected by all the networks in the final population). This is not the case for FS-NEAT that has a better ability in distinguishing between relevant and irrelevant inputs, as the distance between the frequency of relevant and irrelevant inputs indicates (Figures 4d, 4e 4f, 4j, 4k, 4l). Also, it is observed that the best topology for FD-NEAT for initiating its networks' evolution would be the one of few hidden nodes (e.g. 0 and 1 hidden nodes), while it is found again that these are the worst topologies for FS-NEAT. Again, a clear conclusion cannot be drawn for FS-NEAT's initial topology as it depends on the complexity of the problem. Comparing FD-NEAT and FS-NEAT, hypothesis tests show that FS-NEAT has a better ability than FD-NEAT in distinguishing between relevant and irrelevant inputs independently of the choice of the initial topology for all the three problems. Although the



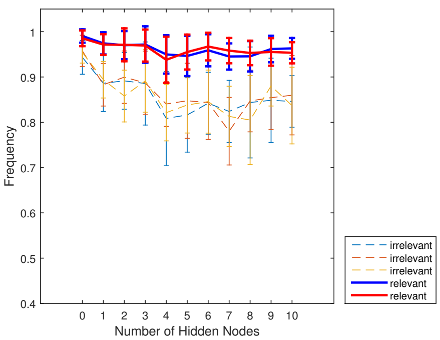
(a) Frequency FD-NEAT, 2/5 XOR



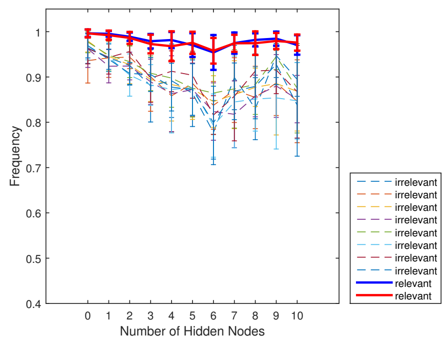
(b) Frequency FD-NEAT, 2/10 XOR



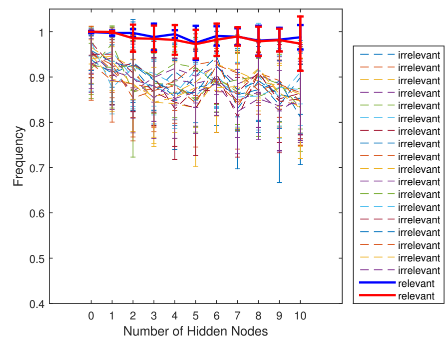
(c) Frequency FD-NEAT, 2/20 XOR



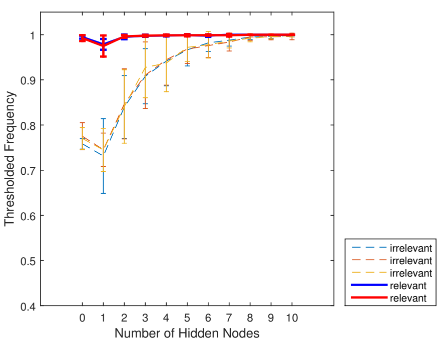
(d) Frequency FS-NEAT, 2/5 XOR



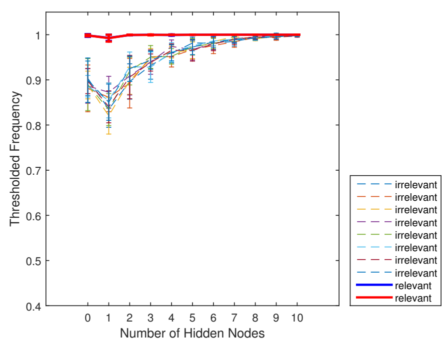
(e) Frequency FS-NEAT, 2/10 XOR



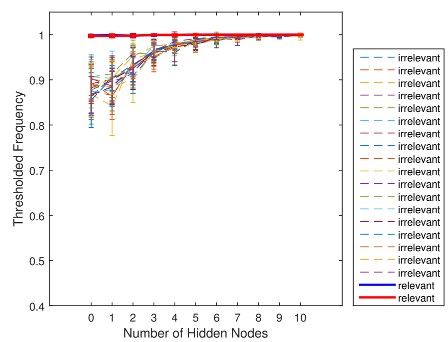
(f) Frequency FS-NEAT, 2/20 XOR



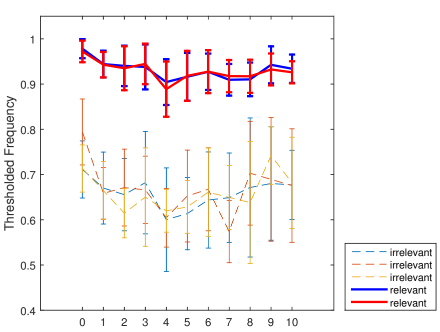
(g) Thresholded frequency FD-NEAT, 2/5 XOR



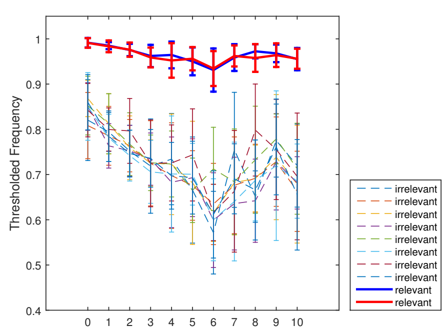
(h) Thresholded frequency FD-NEAT, 2/10 XOR



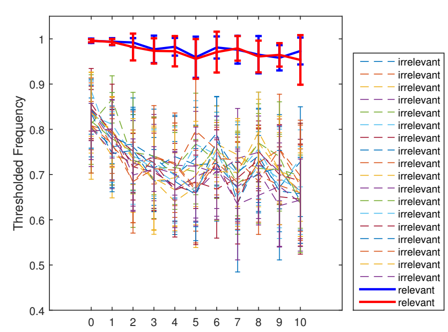
(i) Frequency FD-NEAT, 2/20 XOR



(j) Thresholded frequency FS-NEAT, 2/5 XOR

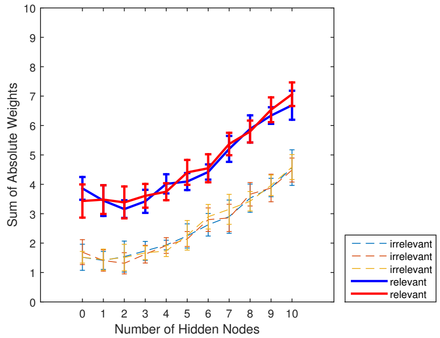


(k) Thresholded frequency FS-NEAT, 2/10 XOR

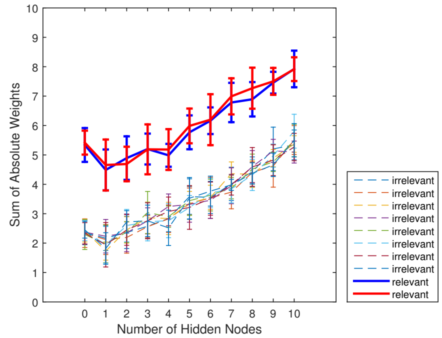


(l) Frequency FS-NEAT, 2/20 XOR

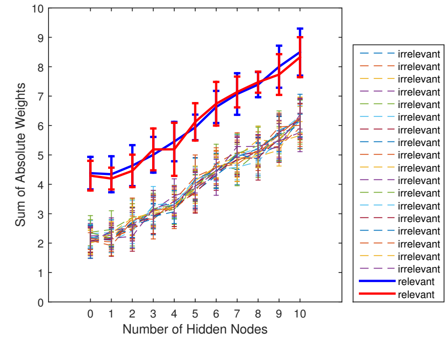
Fig. 4. Results of the frequency of FD-NEAT and FS-NEAT with initial topology of one hidden layer and increasing number of hidden nodes for the 2/5, 2/10 and 2/20 XOR problems.



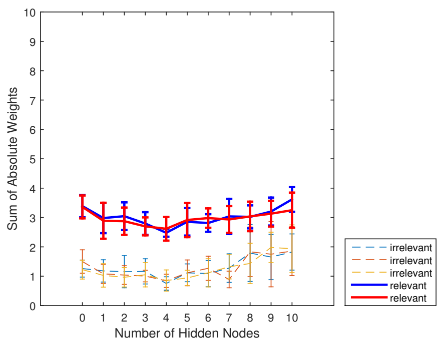
(a) Sum of absolute weights FD-NEAT, 2/5 XOR



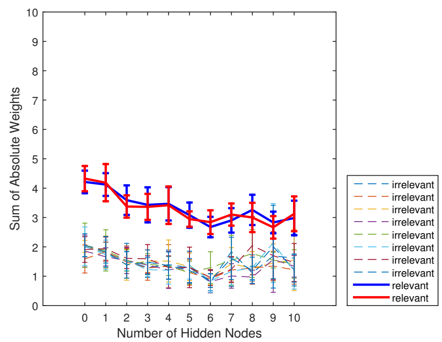
(b) Sum of absolute weights FD-NEAT, 2/10 XOR



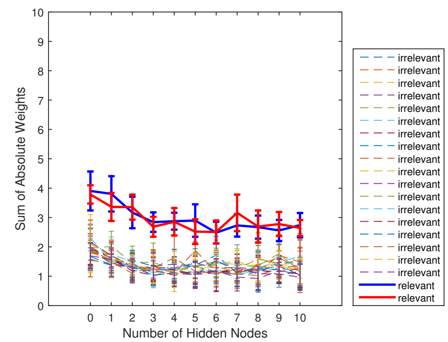
(c) Sum of absolute weights FD-NEAT, 2/20 XOR



(d) Sum of absolute weights FS-NEAT, 2/5 XOR



(e) Sum of absolute weights FS-NEAT, 2/10 XOR



(f) Sum of absolute weights FS-NEAT, 2/20 XOR

Fig. 5. Results of the sum of absolute weights of FD-NEAT and FS-NEAT with initial topology of one hidden layer and increasing number of hidden nodes for the 2/5, 2/10 and 2/20 XOR problems

details on the graphs may not be easily observed, we can notice the difference between relevant and irrelevant inputs. Also, the standard deviation of the frequency of selecting irrelevant inputs is higher than the standard deviation of the frequency of selecting the relevant ones, which shows the consistency of the two algorithms in finding the relevant features.

4) Sum of absolute weights: In Figure 5 we present the results of the sum of absolute weights. At the 2/5 problem, the difference between the relevant and irrelevant inputs is the worst for an initial topology of 0 hidden nodes for FD-NEAT and 8-10 hidden nodes for FS-NEAT. Comparing FD-NEAT and FS-NEAT, FD-NEAT has a statistically higher difference between relevant and irrelevant inputs than FS-NEAT for 7-10 hidden nodes. At the 2/10 XOR problem (Figures 5b, 5e), the topologies of 0 hidden nodes for FD-NEAT and of 0 and 1 nodes for FS-NEAT are statistically better than most of the other topologies. For FS-NEAT the worst initial topology is the one of 9 nodes. In comparison, FD-NEAT is more able than FS-NEAT to distinguish between relevant and irrelevant features for most of the topologies. Finally, at the 2/20 inputs problem (Figures 5c, 5f), FD-NEAT's ability in distinguishing between relevant and irrelevant inputs is better for initial topologies of 8-10 hidden nodes while FS-NEAT's for 0 hidden nodes. In comparison, FD-NEAT performs better feature selection than FS-NEAT independently of the choice of the initial topology.

From the graphs of the sum of absolute weights (Figure 5) we can observe that as the number of hidden nodes increases FD-NEAT has the tendency to assign higher values to all the inputs (relevant and irrelevant ones) (Figures 5a, 5b, 5c), whereas FS-NEAT assigns higher weights to the relevant inputs compared to the irrelevant ones (Figures 5d, 5e, 5f). FD-NEAT seems to have a better feature selection ability for an initial big hidden layer and FS-NEAT for an initial small hidden layer, but taking into account that for these topologies the accuracy is smaller and the number of generations is higher, we tend not to accept this conclusion. Also, by comparing the two algorithms, FD-NEAT is found to outperform FS-NEAT in feature selection which is a result contradictory to the conclusion drawn from the analysis of the frequencies. These two remarks imply that one should not solely rely on one of these measures in order to identify the feature selection ability of an algorithm. In order to choose the more appropriate measure, we should take into account that the frequency shows the fraction of the population that has selected the relevant inputs. It is assumed that in the course of the generations, all the networks of the population should converge to the correct solution and thus it indicates how homogeneous the population is in selecting the relevant inputs. On the other hand, the sum of absolute weights concerns the weights that are assigned to the inputs by the best network of the population. If we take into account that the goal of FS-NEAT and FD-NEAT is to provide an ANN with the right topology, then this measure is more appropriate for investigating the feature selection ability of the algorithm as the best network is expected to assign higher weights to the connections that initiate from the relevant inputs compared to the ones initiating from irrelevant. Also, the use of this measure does not require the a priori definition of any threshold above which the inputs should be considered selected. In this paper, the selection of the threshold was performed based on analysis of the range of weights of the relevant inputs and the irrelevant ones. This requires the a priori knowledge of the correct set of features, which is something that does not happen in real datasets. Therefore, in real applications, the definition of such a threshold is very important, as its value is the border between selecting the correct set of relevant inputs and leaving them out.

V. CONCLUSION

The experiments of this paper, although limited to one artificial dataset with restricted complexity, indicate that the the choice of an initial topology plays an important role for the performance of FD-NEAT and FS-NEAT. Although a general conclusion cannot be drawn on the exact number of hidden nodes that yield the best performance, we observe that overall, FD-NEAT seems to have a better performance for initial topologies of small number of hidden nodes (0-3), whereas FS-NEAT performs worse for an initial topology of zero hidden nodes. This means that FS-NEAT performs better when the networks of the initial population have one hidden layer with at least one hidden node. Finally, FS-NEAT performed better than FD-NEAT in most cases in terms of accuracy, number of generations and ability of selecting features. This can be explained by the different mechanisms for feature selection in the algorithms. FS-NEAT starts with one selected input and it gradually adds inputs, whereas FD-NEAT starts with all the inputs selected and it gradually removes inputs. In a dataset where the portion of relevant features is less than the irrelevant ones it was expected that FS-NEAT would outperform FD-NEAT because it would be easier to select these relevant ones compared to FD-NEAT that would have to deselect the irrelevant ones. However, it should be noted that none of these algorithms knows a priori the number of relevant and irrelevant features, which is something that will happen in an application to real world datasets, where the algorithms will be called to select the relevant inputs.

In future work we are going to extend the set of experiments on problems of increased complexity by employing well-known benchmark datasets, such as the spiral and surface plots, the Monk's problems [18] and others. Also, we are going to investigate the behaviour of these algorithms on datasets with more features and also by increasing the ratio between relevant and irrelevant features to approach more realistic problems.

REFERENCES

[1] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary Computation*, vol. 10, no. 2, pp. 99–127, 2002.

[2] S. Whiteson, P. Stone, K. O. Stanley, R. Miikkulainen, and N. Kohl, "Automatic feature selection in neuroevolution," in *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*. ACM, 2005, pp. 1225–1232.

[3] M. Tan, M. Hartley, M. Bister, and R. Deklerck, "Automated feature selection in neuroevolution," *Evolutionary Intelligence*, vol. 1, no. 4, pp. 271–292, 2009.

[4] M. Tan, R. Deklerck, B. Jansen, and J. Cornelis, "Analysis of a feature-deselective neuroevolution classifier FD-NEAT in a computer-aided lung nodule detection system for CT images," in *Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation*. ACM, 2012, pp. 539–546.

[5] B. Jansen, M. Tan, I. Bautmans, B. Van Keymolen, T. Mets, and R. Deklerck, "Accelerometer based gait analysis," in *Proceedings of the International Conference on Bio-inspired Systems and Signal Processing*, 2011, pp. 138–143.

[6] D. Whitley, T. Starkweather, and C. Bogart, "Genetic algorithms and neural networks: optimizing connections and connectivity," *Parallel Computing*, vol. 14, no. 3, pp. 347–361, 1990.

[7] D. J. Montana and L. Davis, "Training feedforward neural networks using genetic algorithms." in *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, vol. 89, 1989, pp. 762–767.

[8] F. Gomez and R. Miikkulainen, "Incremental evolution of complex general behavior," *Adaptive Behavior*, vol. 5, no. 3-4, pp. 317–342, 1997.

[9] D. E. Moriarty and R. Miikkulainen, "Efficient reinforcement learning through symbiotic evolution," *Machine Learning*, vol. 22, no. 1-3, pp. 11–32, 1996.

[10] F. Gomez and R. Miikkulainen, "2-d pole balancing with recurrent evolutionary networks," in *Proceedings of the 8th International Conference on Artificial Neural Networks*. Springer, 1998, pp. 425–430.

[11] P. J. Angeline, G. M. Saunders, and J. B. Pollack, "An evolutionary algorithm that constructs recurrent neural networks," *IEEE Transactions on Neural Networks*, vol. 5, no. 1, pp. 54–65, 1994.

[12] B.-T. Zhang and H. Muhlenbein, "Evolving optimal neural networks using genetic algorithms with Occam's razor," *Complex systems*, vol. 7, no. 3, pp. 199–220, 1993.

[13] D. T. Cli, I. Harvey, and P. Husbands, "Incremental evolution of neural network architectures for adaptive behaviour," in *Proceedings of the European Symposium on Artificial Neural Networks (ESANN'93)*, 1992, pp. 39–44.

[14] T. Manning and P. Walsh, "Automatic task decomposition for the neuroevolution of augmenting topologies NEAT algorithm," in *Proceedings of the European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*. Springer, 2012, pp. 1–12.

[15] A. P. Wieland, "Evolving neural network controllers for unstable systems," in *Proceedings of the International Joint Conference on Neural Networks*, vol. 2. IEEE, 1991, pp. 667–673.

[16] D. James and P. Tucker, "A comparative analysis of simplification and complexification in the evolution of neural network topologies," in *Proceedings of Genetic and Evolutionary Computation Conference*, 2004.

[17] S. Xu, H. Moriguch, and S. Honiden, "Sample efficiency analysis of neuroevolution algorithms on a quadruped robot," in *In Proceedings of the IEEE Congress on Evolutionary Computation*. IEEE, 2013, pp. 2170–2177.

[18] S. B. Thrun, J. Bala, E. Bloedorn, I. Bratko, B. Cestnik, J. Cheng, K. De Jong, S. Dzeroski, S. E. Fahlman, D. Fisher *et al.*, "The monk's problems: A performance comparison of different learning algorithms," Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-CS-91-197, December 1991.