

ABSTRAK

Konfigurasi parameter yang tepat merupakan awal dari penggunaan sebuah algoritma. Konfigurasi ini bertujuan untuk memaksimalkan kinerja dari suatu algoritma. Algoritma genetika dan neural networks merupakan kedua algoritma yang memerlukan pemilihan parameter yang tepat. Banyaknya penggunaan generasi dan populasi pada algoritma genetika, dan jumlah neuron pada layer neural networks dapat mempengaruhi performanya. Dalam permainan ular, performa di sini mencakup dari score yang didapat dan juga efisiensi runtime pada program. Berdasarkan hal itu, dilakukan pengujian parameter untuk menemukan konfigurasi yang tepat untuk memaksimalkan kinerja dari kedua algoritma. Permainan ular memiliki kinerja metrik yang jelas, seperti score dan beberapa rintangan yang ada sehingga digunakan sebagai model eksperimen dalam pengujiannya. Pengujian dilakukan sebanyak 60 kali eksperimen yang membandingkan jumlah generasi dan populasi, mutation chance, dan neuron pada hidden layer. Hasil penelitian menunjukkan konfigurasi parameter dengan generasi lebih besar dari populasi adalah yang paling optimal. Konfigurasi ini menghasilkan skor yang hampir setara dengan generasi dan populasi sama besar, namun dengan runtime yang jauh lebih efisien. Pengujian mutation chance menunjukkan bahwa nilai 0.1% adalah yang terbaik dan ukuran hidden layer 16 neuron terbukti lebih efisien dibandingkan 24 neuron, baik dari segi skor maupun runtime.

Kata kunci: Algoritma genetika, Neural networks, Permainan ular klasik, Mutation chance, Hidden layer, Neuron

ABSTRACT

Configuring the right parameters is crucial for the optimal use of an algorithm. This configuration aims to maximize the performance of an algorithm. Genetic algorithms and neural networks are two types of algorithms that require proper parameter selection. The number of generations and population size in genetic algorithms, as well as the number of neurons in the layers of neural networks, can affect their performance. In the game of Snake, performance encompasses both the score achieved and the runtime efficiency of the program. Therefore, parameter testing is conducted to find the right configuration to maximize the performance of both algorithms. The Snake game has clear performance metrics, such as score and obstacles, making it suitable as an experimental model for testing. Sixty experiments were conducted to compare the number of generations, population size, mutation chance, and neurons in the hidden layer. The research results show that a parameter configuration with a larger number of generations than population size is the most optimal. This configuration yields a score nearly equal to that of equal generations and population size, but with much more efficient runtime. Mutation chance testing indicates that a value of 0.1% is the best, and a hidden layer size of 16 neurons proves to be more efficient than 24 neurons, both in terms of score and runtime.

Keywords: *Genetic Algorithm, Neural networks, Classic snake game, Mutation chance, Hidden layer, Neurons*