The Evolutionary Learning Rule in System Identification

Oscar Montiel, Oscar Castillo, Patricia Melin, Roberto Sepulveda

Summary. In this chapter, we are proposing an approach for integrating evolutionary computation applied to the problem of system identification in the well-known statistical signal processing theory. Here, some mathematical expressions are developed in order to justify the learning rule in the adaptive process when a Breeder Genetic Algorithm is used as the optimization technique. In this work, we are including an analysis of errors, energy measures, and stability.

14.1 Introduction

The problem of determining a mathematical model for an unknown system by observing its input-output data pairs is known as system identification, and it is an important step when we wish to design a control law for a specific system. Real systems are non-linear and have time variations; hence the best control laws that we can obtain are those based using real-time data from continuous-time stochastic processes [1].

Traditionally, system identification has been performed in two ways:

1. Using analytic models, i.e., obtaining mathematically the transfer function.
2. Using experimental input-output data. In this way, the identification can be achieved in two forms: nonparametric and parametric.

In this chapter we are interested in parametric models. As we mentioned, there are several well-known techniques to perform the system identification process. Most of the parametric techniques are gradient-guided and are limited in highly multidimensional search spaces. The system identification process generally involves two top-down steps, and these are structure identification and parameter identification. In the first step, we need to apply a priori knowledge about the target system for determining a class of model within the search for the most suitable model is going to be conducted [2] [3].

Here, we are using an evolutionary algorithm known as Breeder Genetic Algorithm (BGA) that lays somehow in between Genetic Algorithms (GAs) and Evolutionary Strategies (ESs). Both methods usually start with a randomly generated population of individuals, which evolves over the time in a quest to get better solutions for a specific problem. GAs are coded in binary forming strings called chromosomes; they produce offsprings by sexual reproduction. Sexual reproduc-
tion is achieved when two strings (i.e., parents) are recombined (i.e., crossover). Generally, the parents are selected stochastically, the search process is mainly driven by the recombination operation, and the mutation is used as secondary search operator with low probability, to explore new regions of the search space. An ES is a random search, which models natural evolution by asexual reproduction [4]. It uses direct representation, that is, a gene is a decision variable and its allele is the value of the variable [5] in ES the mutation is used as the search operator, and it uses the \((\mu, \lambda)\)-strategy as a selection method. Thus, the BGA can be seen as a combination of ESs and GAs, because it handles direct real variables, and truncation selection, which is very similar to \((\mu, \lambda)\)-strategy, and the search process is mainly driven by recombination making BGAs similar to GAs [6] [7] [8] [9] [10].

14.2 The Generic Identification Problem

Figure 14.1 shows the generic problem of system identification. Here, we have a digital signal input \(x(n)\) that is fed to the unknown system and to the adaptive filter at the same time. In this figure there is a "black box" enclosed by dashed lines; its output is called the desired response signal and it is represented by \(d(n)\). The adaptive system will compute a corresponding output signal sample \(y(n)\) at time \(n\). Both signals, \(d(n)\) and \(y(n)\), are compared subtracting the two samples at time \(n\), to obtain a desired response signal. This concept is expressed in equation form as

\[
e(n) = d(n) - y(n)
\] (14.1)

This block might have a pole-zero transfer function, an all-pole or auto-regressive transfer function fixed or time-varying, a nonlinear mapping, or some other complex system. In the dashed “black-box”, we have an additive noisy signal known as the observation noise signal because it corrupts the observation of the signal at the output of the unknown system [17]. Thus, the real desired signal \(\hat{d}(n)\) is contaminated with noise; hence the signal \(d(n)\) is given by Eq. (14.2):

\[
d(n) = \hat{d}(n) + \eta(n).
\] (14.2)

In the adaptive system block we could have any system with a finite number of parameters that affect how \(y(n)\) is computed from \(x(n)\). In this work, we are using an adaptive filter with a Finite Impulse Response (FIR filter), and it is represented by the equation

\[
y(n) = \sum_{i=0}^{L-1} w_i(n)x(n-i)
\] (14.3)

or in vectorial form as