Genetic Algorithms for PID Parameter Optimisation: Minimising Error Criteria

T. O'Mahony & C.J. Downing, Cork Institute of Technology, Cork, IRELAND.
tomahony@cit.ie
cdowning@cit.ie

Klaudiusz Fatla, Wroclaw University of Technology, Wroclaw, POLAND.
fatla@elektryk.ie.pwr.wroc.pl

ABSTRACT
There is sparse mention in the literature of the application of Genetic Algorithm (GA) optimisation techniques to the tuning of PID controllers. Where mention does exist the application tends to be confined to an illustration of the concept on a simple model, without even a comparison with conventional tuning techniques provided. In the opinion of the authors this does not exploit the full potential of the GA approach which is tailored towards locating high performance areas in complex domains.
A close perusal of the PID tuning literature reveals a confusing plethora of techniques for simple systems and, conversely, a dearth of techniques for more complex processes. Furthermore it is generally accepted that the techniques applicable for simple processes are unsuited to higher-order processes and in addition that they frequently fail even for simple processes when a significant time delay exists.
The GA overcomes these difficulties by offering the possibility of a general tuning technique which is applicable to a wide variety of processes and may be based on either performance/robustness criteria or model following.
This paper reports on the application of GAs to a range of processes and compares the results obtained with those obtained using conventional tuning techniques. Simulation results indicate that the GA overcomes many of the difficulties associated with existing tuning rules, and produces satisfactory results for systems that are normally considered difficult to tune.

INTRODUCTION
Despite the abundance of PI and PID tuning techniques available in the literature, interest in, and development of new methodologies continues unabated. This is doubtless due to the widespread application of this versatile controller in industry and also due to the recognition that the majority are poorly tuned [4]. However, most current tuning methods will only yield PI or PID parameters for a restricted class of process models. There is no general methodology for arbitrary process models other than approximating them with a first-order or second-order time-delayed model and applying an appropriate rule. Furthermore as process complexity increases e.g. first-order unstable time-delayed system, the number of applicable tuning rules decreases and in many cases vanishes altogether. The rules that are available for such processes tend to be ad hoc in nature and not based on performance or robustness criteria, consequently little can be said for them other than that 'they work'. Therefore it typically happens that if a designer wishes to tune a regulator to minimise, say, the Integral of Absolute Error (IAE) load criterion, then depending on the type of process in question s/he may be faced with a plethora of contesting rules, or none at all. Clearly this scenario is not ideal.
The objective of this paper is to highlight a tuning method that utilises a unified approach and yields consistent performance - subject to that which is achievable - over a wide range of process models. To achieve this objective a Genetic Algorithm (GA) will be utilised. The GA approach is an intuitive and mature search and optimisation technique based on the principles of natural evolution and population genetics. Typically the GA starts with little or no knowledge of the correct solution and depends entirely on responses from an interacting environment and its evolution operators to arrive at good solutions. By dealing with several independent points, the GA samples the search space in parallel and hence is less susceptible to converging to a suboptimal solution. In this way, GAs have been shown to be capable of locating high performance areas in complex domains without experiencing the difficulties associated with high dimensionality or false optima, as may occur with gradient descent techniques. Thus GAs have been recognised as a powerful tool in many control applications such as parameter identification and control structure design [1]. GAs have also found widespread use in
controller optimisation particularly in the fields of fuzzy logic [8] and neural networks [12]. The application of GA to PID parameter optimisation has however been largely overlooked, particularly as an alternative tuning technique for processes which are otherwise difficult to tune.

The contribution of this paper is to highlight a number of deficiencies that exist in the current tuning literature and explore the potential of the GA methodology to overcome these deficiencies. To that end this paper concentrates on the minimisation of a variety of error criteria for a wide variety of processes including unstable time-delayed systems. Where possible, suitable tuning rules are identified and applied with the results then compared with a similarly optimised controller whose parameters were obtained by applying a genetic algorithm. The results indicate that the modern control engineer would do well to include the GA in his/her repertoire of design methodologies.

CONTROL ALGORITHM DESIGN

Due to the variety of PID control law permutations it is necessary to specify a minimum set of attributes which in this paper are as follows:

1) Only single-degree of freedom controllers will be treated
2) The PID controller is assumed to be of textbook (non-interacting) form as defined below:

$$G_c(s) = K_p + K_i \frac{1}{s} + K_d s$$

Eq.1

By suitable transformation of the parameters this form can always be converted to interacting form, whereas the converse is not always true. The textbook controller structure was adjudged to be most suitable as the majority of tuning relations are developed based on this structure.

Minimising one of the following error criteria generates the controller parameters of equation 1:

$$ISE = \int_0^T [r(t) - y(t)]^2 dt$$
$$IAE = \int_0^T |r(t) - y(t)| dt$$
$$ITAE = \int_0^T t \cdot |r(t) - y(t)| dt$$

Eq.2

where: $r(t) =$ reference input,
$y(t) =$ measured variable,

In this paper these error criteria will be minimised by applying an existing tuning algorithm as illustrated in the next section or through the application of a genetic algorithm, as will presently be elucidated. For a comprehensive overview and introduction to GA see Davis [3]. The GA works on a coding of the parameters $(K_p, K_i, K_d)$ to be optimised rather than the parameters themselves. In this study Gray coding was used where each parameter was represented by 16 bits and a single individual or chromosome was generated by concatenating the coded parameter strings. In contrast to traditional stochastic search techniques the GA requires a population of initial approximations to the solution. Here 40 randomly selected individuals were used to initialise the algorithm. The GA then proceeds as follows:

**Determine fitness:** The first step of the GA procedure is to evaluate each of the chromosomes and subsequently grade them. Each individual was evaluated by decoding the string to obtain the PID parameters which were then applied in a Simulink representation of the closed-loop system. The Simulink environment was chosen as it is intuitive, allows for the simple extension of the process model and avoids having to approximate the time-delay, which would otherwise be necessary in the MATLAB environment. On completion of the simulation the manipulated variable was automatically returned to the GA and the chosen error criterion used to evaluate the performance and assign a fitness value to that individual.

**Selection:** The five fittest individuals were automatically selected while the remainder were selected probabilistically, according to their fitness. This is an elitist strategy that ensures that the next generation's best will never degenerate and hence guarantees the asymptotic convergence of the GA.

**Generation:** using the individuals selected above the next population is generated through a process of single-point cross-over and mutation. Mutation was applied with a very low probability - 0.001 per bit. Reproduction through the use of crossover and mutation ensures against total loss of any genes in the population by its ability to introduce any gene which may not have existed initially, or, may subsequently have been lost.

**Repeat:** This sequence was repeated until the algorithm was deemed to have converged (50 iterations). As was indicated previously the simulation and evaluation of the GA tuned PID controller was achieved using the MATLAB/Simulink environment. In addition the GA Toolbox [2] was utilised to aid the implementation of the GA.
TUNING THE PID ALGORITHM

As was noted in the introduction one of the objectives of this paper is to highlight the usefulness of GA for PID controller design, by comparison with a conventionally tuned, but similarly optimised controller. Since tuning rules minimising the error criteria of equation 2 abound for processes modelled on the premise of first-order lag plus dead-time, this structure was used to begin the evaluation. More specifically, the following process model was assumed

$$G_{p2}(s) = \frac{e^{-\tau_m s}}{s + 1} \quad \text{Eq.3}$$

with $\tau_m = 0.2$ sec. The design objective was to achieve the minimum $\text{ISE}$, $\text{IAE}$ or $\text{ITAE}$ servo response using (a) the GA approach and (b) existing tuning rules. These rules are derived by Zhuang and Atherton [15] for the $\text{ISE}$ criterion and by Rovira et al. [11] for the $\text{IAE}$ and $\text{ITAE}$ criteria. For both cases the controller coefficients are listed in table 1 below, along with the total error and peak overshoot. The latter metric is defined as

$$P_{OS} = \max \left| y(t) - y_{ss} \right| \quad \text{Eq.4}$$

where $y_{ss}$ is the steady state value of the output and $\theta_c$ is a vector of PID parameters, while the former is defined according to

$$E_{\text{Total}}(\theta_c) = \int_0^T J_{DO} \, dt \quad \text{Eq.5}$$

In equation 5, the design objective, $J_{DO}$, is either $\text{ISE}$, $\text{IAE}$ or $\text{ITAE}$ as defined by equation 2. As might be expected both the GA and the conventional tuning rules perform very similarly.

Table 1 - Servo optimised FOLPD

<table>
<thead>
<tr>
<th>Tuning Rule</th>
<th>Error Criterion</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
<th>$E_{\text{TOTAL}}$</th>
<th>$P_{OS}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAE</td>
<td>4.40</td>
<td>3.14</td>
<td>0.35</td>
<td>0.35</td>
<td>0.05</td>
<td>0.05</td>
</tr>
<tr>
<td>ISE</td>
<td>4.44</td>
<td>4.98</td>
<td>0.52</td>
<td>0.30</td>
<td>0.11</td>
<td>0.11</td>
</tr>
<tr>
<td>ITAE</td>
<td>3.79</td>
<td>2.91</td>
<td>0.26</td>
<td>0.07</td>
<td>3.8e-3</td>
<td>3.8e-3</td>
</tr>
</tbody>
</table>

| GA          | IAE            | 4.27 | 3.04 | 0.34 | 0.35              | 0.03   |
|            | ISE            | 5.59 | 3.62 | 0.50 | 0.28              | 0.18   |
|            | ITAE           | 3.77 | 2.87 | 0.27 | 0.07              | 3.7e-4 |

Likewise to optimise the regulator response (assuming a step disturbance applied at the process input) a glut of contesting rules exist - see O'Dwyer [10] for a useful summary. In this application the controller tuning algorithms proposed by Zhuang and Atherthon [15] for the $\text{ISE}$ criterion, Shinskey [13] for the $\text{IAE}$ criterion and Murrill [9] for the $\text{ITAE}$ criterion were compared with the GA based technique. The figures of merit defined by equations 4 and 5 were again utilised as the basis for comparison with the results listed in table 2.

Table 2 - Regulator optimised FOLPD

<table>
<thead>
<tr>
<th>Tuning Rule</th>
<th>Error Criterion</th>
<th>$K_p$</th>
<th>$K_i$</th>
<th>$K_d$</th>
<th>$E_{\text{TOTAL}}$</th>
<th>$P_{OS}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>IAE</td>
<td>6.32</td>
<td>18.52</td>
<td>0.49</td>
<td>0.07</td>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>ISE</td>
<td>7.02</td>
<td>26.29</td>
<td>0.84</td>
<td>0.01</td>
<td>0.18</td>
<td>0.18</td>
</tr>
<tr>
<td>ITAE</td>
<td>6.23</td>
<td>17.20</td>
<td>0.48</td>
<td>0.04</td>
<td>0.19</td>
<td>0.19</td>
</tr>
</tbody>
</table>

| GA          | IAE            | 6.07 | 16.70| 0.50 | 0.07              | 0.19   |
|            | ISE            | 6.45 | 28.45| 0.83 | 0.01              | 0.18   |
|            | ITAE           | 5.68 | 14.85| 0.46 | 0.04              | 0.19   |

Following on from these examples, extensive computer based simulation has suggested that few real benefits accrue by applying GA to the task of optimising PID controllers for FOLPD processes. Simple and highly accurate tuning rules for error criteria are in abundance, with results such as displayed in tables 1 and 2 being the norm rather than the exception.

While the presumption of a first-order lag plus dead-time model is sufficient for many industrial processes there are many examples of where this supposition is too limiting. In such cases a higher order model, typically second-order system plus delay (SOSPD), is required with the model subsequently being utilised in the controller-tuning algorithm. Such systems are considerably more
difficult to tune, not merely because of the more complex dynamics but also because the number of
suitable tuning rules diminishes. For example the authors are not aware of the existence of any rules
which minimise any of the error criterion for the servo response of a PI controlled SOSPD. Simple
rules to minimise the IAE regulator response are proposed by Shinskey [13] for the PI control of a
SOSPD based on the ultimate gain and ultimate cycle or the time-delay and process time constants.
Similar rules for the ISE and ITAE regulator responses are conspicuous only by their absence. One
possible reason for this is that simple, yet sufficiently general tuning rules for second-order or higher
models are extremely difficult to arrive at. Many of the tuning relations are derived from extensive
computer simulations, where the PID parameters are obtained by solving an optimisation problem for
various parameterisations of the plant model. The data resulting from repeated optimisations is then
empirically fitted into correlated equations using, for example, a least squares method. This approach
has lead to the development of relatively simple relations for FOLPD models e.g. Zhuang & Atherton
[15], but for more complex models tends to result in cumbersome expressions, e.g. Hwang [7], Huang
et al. [6], the application of which are both tedious and error-prone.
The foregoing depicts one situation where the use of the GA based optimisation is advantageous - the
case where tuning rules do not exist, or - more likely - cannot be readily located. However, there are
other reasons why the GA based technique might be preferred, principal among these is that the rules
available for SOSPD may yield less accurate results than was found with their FOLPD counterparts.
To demonstrate this consider the following second-order model

\[ G_{SO}(s) = \frac{e^{-\tau_m s}}{(s+1)(0.1s+1)} \]

Eq.6

where \( \tau_m = 1 \) sec. A PI controller was designed, using the rules proposed by Shinskey [13], to
minimise the IAE regulator response. The application of this tuning algorithm leads to the controller
coefficients of table 3 and the associated response illustrated in figure 1. This figure depicts the
controlled response to a unit step input which was applied at time, \( t=0 \), and also to a unit step load
disturbance which occurred at time \( t=30 \). In table 3 the controller gains which resulted when the GA
was used to minimise the IAE regulator response are also tabulated. Clearly the true optimum is
achieved by the GA technique.

<table>
<thead>
<tr>
<th>SOSPD</th>
<th>Genetic Algorithm</th>
<th>Tuning Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>( K_p )</td>
<td>0.97</td>
<td>1.10</td>
</tr>
<tr>
<td>( K_i )</td>
<td>0.63</td>
<td>0.29</td>
</tr>
<tr>
<td>( E_{TOTAL} )</td>
<td>1.69</td>
<td>3.28</td>
</tr>
<tr>
<td>( P_{OS} )</td>
<td>0.71</td>
<td>0.70</td>
</tr>
</tbody>
</table>

Another issue associated with the use of tuning rules that the GA approach effectively overcomes is the
existence of artificial limits on many rules that do not coincide with the limitations of the ubiquitous
PID controller. Consider the situation where a PID controller is to be designed for the process defined
by equation 6, to yield minimum ITAE regulator response. In [14] Sung et al present tuning relations
for such a situation. The tuning algorithm stipulates that the maximum time-delay be restricted by the
following relation:

\[ 0.05 < \frac{\tau_m}{T_{ml}} \leq 2 \]

Eq.7

where \( \tau_m \) and \( T_{ml} \) are defined as follows:

\[ G_{SO}(s) = \frac{K_m e^{-s\tau_m}}{T_{ml}^2 s^2 + 2\xi_m T_{ml} s + 1} \]

Eq.8
For the assumed model of equation 8 this implies that the maximum permissible dead-time is approximately 0.6sec. However, as is evident from the results of figure 2 this does not correspond to the physical limitations of the PID controller and reasonable performance is achieved by the GA tuned PID controller for values of $\tau_m$ up to 1.5 sec.

| Table 4 - ITAE regulator optimised SOSPD |  
| $\tau_m$ | 0.6 | 1.0 | 1.5 |  
| $K_p$ | 2.03 | 1.26 | 0.98 |  
| $K_i$ | 2.09 | 1.03 | 0.68 |  
| $K_d$ | 0.57 | 0.53 | 0.55 |  
| $E_{TOTAL}$ | 0.91 | 2.73 | 4.8 |  
| $P_{OS}$ | 0.47 | 0.64 | 0.77 |  

The real power of the GA methodology comes to the fore when tuning processes which are traditionally characterised as 'difficult to tune'. Examples include, but are not limited to, the following:

$$G_{HO}(s) = \frac{e^{-0.5s}}{(s+1)(\alpha s + 1)(\alpha^2 s + 1)(\alpha^3 s + 1)}; \quad \alpha = 0, 1$$

$$G_{NMP}(s) = \frac{(-2s+1)e^{-0.5s}}{(s+1)^3}$$

$$G_{US}(s) = \frac{e^{-0.5s}}{(-s+1)}$$

Eq.9

Since the GA utilises a Simulink scheme to evaluate the fitness of each gain set in the population, optimisation of any of the above process models simply consists of modifying the LTI System Block in the Simulink scheme to reflect the current model. Beyond specifying the desired error criterion no further adjustments to the GA are required. This is in stark contrast to the use of conventional tuning rules where a laborious and perhaps unfruitful search for a suitable tuning algorithm must be undertaken. In the unlikely event of one being found, the user has no guarantee that the resultant parameters are the true optimum ones.

Figure 3 opposite illustrates the performance of the PID controller when applied to each of the process models above. As before the reference is a unit step input and a unit step is applied to the process input at time $t=30$sec. The PID parameters were tuned by utilising the GA to minimise the ITAE regulator response. While the results are not ideal, they must be viewed within the context of the PID controller's capabilities. Classic limitations regarding optimal servo vs optimal regulator tracking are clearly visible for the open-loop unstable time-delayed system. However, considering the nature of the process the regulator response is quite acceptable. The performance of the high-order process, $G_{HO}(s)$, is good on both counts. The performance of the non-minimum phase time-delayed process is likewise poor on both counts. One possible reason for this is the use of a single criterion in the optimisation function which is frequently criticised in the literature as the results may be misleading. For example, the exact minimisation of, say, the IAE may result in a transient that is too oscillatory or a system that lacks robustness. To minimise such eventualities multiple closed-loop criteria are frequently specified e.g. Harris & Mellichamp [5], however such designs were not investigated at this juncture.
CONCLUSION

This paper presents a methodology for tuning PI and PID controllers using Genetic Algorithms and the inherent advantages of such a technique. Fundamentally, it was demonstrated that the GA technique is independent of the process model and consequently can be used to obtain consistent performance for a wide variety of process models. This is in stark contrast to the use of tuning rules which

a) tend to be specific to a particular class of process models
b) may generate sub-optimal solutions, and
c) place artificial limits on the application of these rules

The implications of the latter are that the tuning rule may fail, typically when applied to processes with significant time-delays. Furthermore, while these rules may be in abundance for FOLPD models, fewer rules are available for SOSPDP and fewer still for integrating or unstable processes. The use of a GA avoids the necessity to maintain a 'database' of potentially useful rules and overcomes the difficulties associated with finding them in the first place. In addition the complexity of the approach is greatly reduced through the combination of MATLAB and the availability of quality source code and specialised Toolboxes. Refinedness to the genetic algorithm, such as the ability to utilise small populations, and the advent of fast computing, results in at worst a modest computational burden that can no longer be regarded as an issue. The generality of the technique, combined with its intuitiveness, fast convergence, modest processing requirements and perhaps most importantly minimal system specific information should result in increased use of this technique for both off-line and on-line controller tuning. In any case it is recommended that the GA approach be included in all control engineers 'box of tricks' as an invaluable aid for parameter optimisation.

ACKNOWLEDGEMENT

The authors wish to express their sincere thanks to Dr. Aidan O'Dwyer of Dublin Institute of Technology, who provided them with some very useful references regarding the tuning of PI and PID controllers via conventional tuning rules. The authors further wish to acknowledge the support of the CIT Postgraduate Scholarship Scheme.

REFERENCES