Li, Y. and Ang, K.H. and Chong, G.C.Y. and Feng, W. and Tan, K.C. and Kashiwagi, H. (2004) CAutoCSD–evolutionary search and optimisation enabled computer automated control system design. *International Journal of Automation and Computing* 1(1):pp. 76-88.

CAutoCSD – Evolutionary Search and Optimisation Enabled Computer Automated Control System Design

Yun Li[1], Kiam Heong Ang, Gregory C.Y. Chong, Wenyuan Feng[2], Kay Chen Tan[3]

Department of Electronics and Electrical Engineering
University of Glasgow, Glasgow G12 8LT, UK

Hiroshi Kashiwagi

Department of Mechanical Engineering and Material Science
Faculty of Engineering, Kumamoto University, Japan

[1] Corresponding author email: Y.Li@elec.gla.ac.uk
[2] Present Address: Xinjiang Esquel Textile Co. Ltd., 6 Yinchuan Road, Urumqi 830054, China
[3] Present Address: Department of Electrical Engineering, National University of Singapore, Singapore

Abstract
This paper attempts to set a unified scene for various linear time-invariant (LTI) control system design schemes, by transforming the existing concept of 'Computer-Aided Control System Design' (CACSD) to the novel 'Computer-*Automated* Control System Design' (CAutoCSD). The first step towards this goal is to accommodate, under practical constraints, various design objectives that are desirable in both time and frequency-domains. Such performance-prioritised unification is aimed to relieve practising engineers from having to select a particular control scheme and from sacrificing certain performance goals resulting from pre-committing to the adopted scheme. With the recent progress in evolutionary computing based extra-numeric, multi-criterion search and optimisation techniques, such unification of LTI control schemes becomes feasible, analytically and practically, and the resultant designs can be creative. The techniques developed are applied to, and illustrated by, three design problems. The unified approach automatically provides an integrator for zero-steady state error in velocity control of a DC motor, meets multiple objectives in designing an LTI controller for a non-minimum phase plant and offers a high-performing LTI controller network for a nonlinear chemical process.

Keywords: LTI, PID, Control System Design, CACSD, Performance Index, Genetic Algorithms, Evolutionary Computation, Process Control, Robust Control.

1. INTRODUCTION

Before any design actually takes place in control engineering practice, an applications engineer needs to choose a control scheme to suit his/her application. At present, a single control scheme does not offer all what a practising engineer desires to obtain (Kashiwagi 1983; Levine 1996). One control scheme is often restricted ad hoc to one particular problem and only addresses a subset of performance issues. Further, the design of each different scheme often requires a different design technique.

With the rapid progress in computer-aided control system design (CACSD), the task of design simulation is now tremendously eased. However, a question a practising engineer continues to ask is: *Can the problem of pre-selecting a control scheme also be solved with the power of modern CACSD?*

Unfortunately, the answer is *No*. This is mainly because design specifications and objectives are often mixed, some of which may be weakly defined and hard to quantify. Existing CACSD tools are mostly simulators, but design is the reverse problem of simulation.

Fortunately, the design of an optimal linear quadratic (LQR/LQG), an H_∞ or a μ-synthesis based control system is associated with a pre-defined and more quantified objective. Hence, the design is more computerised with the help of an optimiser. However, these schemes impose some theoretical conditions and restrictions in order that the optimisation may be carried out. It is thus difficult to accommodate practical constraints or any structural optimisation of the controllers (Li *et al* 1995; Li and Haeussler 1996). Conventional optimisers are far from capable of delivering a global, high-dimensional or multi-objective solution. Hence, in control system design, a manual interactive and iterative tuning process is still necessary (Kashiwagi 1983; Levine 1996).

The simulation power of a modern CACSD package can, however, be utilised to achieve design automation if the simulator is interfaced with evolutionary computing (EC) based search and machine learning tools. The recent progress in evolutionary and soft computing techniques has enabled the replacement of the human trial-and-error based iterative process with a computer-automated one (Li *et al* 1995; Ng 1995; Tan 1997; Tan and Li 2001). More importantly, EC reshapes the way we think in designing and modelling engineering systems, and unleashes the uncharted potential of design engineering. With the help of EC techniques, mixed or multi-criterion objectives and hence the designs of linear time-invariant (LTI) control systems might now be unified under one banner: "*performance satisfaction*" (Li *et al* 1995; Tan and Li 2001).

Hence, in this paper, we seek to answer the following questions:

(a) Is there a need to unify LTI control laws and schemes?
(b) Is there a need to unify various design approaches and how such unification may be achieved?
(c) Is it viable to unify them with multiple design criteria best to meet all target specifications?
(d) Is it practical for computers to relieve human designers from tedious iterative tasks and also automatically to evolve practical solutions that may exceed existing performance bounds?

In Section 2 of this paper, specifications and objectives in control system design are assessed. Issues of interpreting human engineers' perception of merit into a form that may be utilised for CACSD automation are addressed. Section 3 proposes a way to unify and shows how that EC may be employed to achieve 'Computer-*Automated* Control System Design' (CAutoCSD). Indices concerning frequency-domain terms such as stability margins and sensitivity for robustness measurements are not used alone, but have also been used together with time-domain specifications to form a composite design objective (Kashiwagi 1983; Levine 1996; Li *et al* 1995). These are summarised in Section 4, where conclusions are drawn in Section 5.

2. PERFORMANCE BASED LTI DESIGN UNIFICATION

Right from the conceptual design, practical system constraints should be taken into account, as these can now be incorporated in evolutionary design unification. Since evolutionary computation does not require direct gradient-guidance, the choice of indices and weighting functions can become much more flexible and creative. The first step towards unifying LTI controllers is to consider the design by meeting practical performance requirements, instead of by a specific scheme or in a particular domain. Such a unified design approach aims to eliminate the need for pre-selection of a control scheme, so as to take a performance-prioritised approach that is easily understood and meaningful to the application engineer. This also aims at incorporating those performance terms that engineers are familiar with in both the time and the frequency domains.

2.1 Core Criteria

Consider a generic unity negative feedback control system of a given plant $G(s)$ with controller $H(s)$. Refer to Fig. 1 for notations. Without loss of generality, for the case $F(s) = 1$,

$$E(s) = R(s) - Y(s) = \frac{1}{1 + H(s)G(s)}\left[R(s) - G(s)D(s)\right]. \tag{1}$$

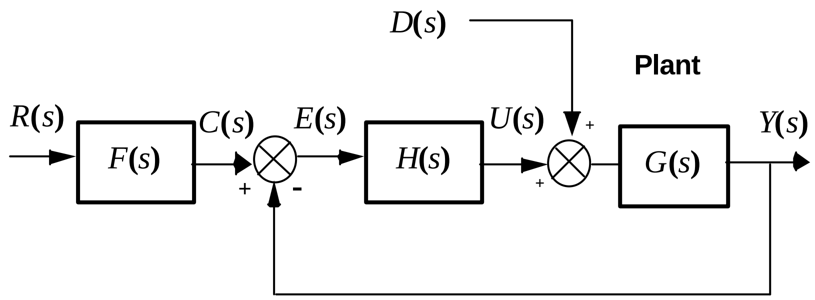


Fig. 1 Generic feedback control system with a model-following command

where $D(s)$ represents a disturbance, which may be coloured and also be modelled to include the plant uncertainty. The ultimate objective of a control system design is hence to find an $H(s)$ such that

$$E(s) = 0, \quad \forall s, D(s), \tag{2}$$

or

$$e(t) = L^{-1}\{E(s)\} = 0, \quad \forall t, d(t). \tag{3}$$

This ultimate goal means that Condition (2) or (3) needs to be satisfied under plant and environmental uncertainties, which is impossible in practical control system design due to control signal or actuator saturation (e.g., voltage limit) and constraints on the rate of change of the control signal (e.g., current limit). In fact, should (2) or (3) be met regardless of time and frequency, the feedback system would become open-loop and this, in turn, would not guarantee a zero $e(t)$ or $E(s)$ with the presence of disturbance or model uncertainty.

Hence, a performance index, $J : \mathbf{R}^n \to \mathbf{R}^+$, must be devised to measure *how close* the above ultimate objective is met, where n is the number of parameters that need to be determined in the design. For this, performance indices and specifications need to reflect the following qualitative specification requirements (Kashiwagi 1983; Levine 1996; Li *et al* 1995):

Spec. 1. Good relative stability (e.g., good gain and phase margins);
Spec. 2. Excellent steady-state accuracy (e.g., minimal or no steady-state errors);
Spec. 3. Excellent transient response (e.g., minimal rise-time, settling-time, overshoots and undershoots);
Spec. 4. Robustness to the environment (e.g., maximal rejection of disturbances); and
Spec. 5. Robustness to the modelling and plant uncertainties (e.g., minimal sensitivities to parametric and structural variations).

In the context of evolutionary computation, a performance index is often termed a 'fitness function', where 'maximising a fitness function' is more commonly encountered than 'minimising a cost function', although an evolutionary algorithm (EA) can do both maximisation and minimisation in one process. For convenience, a cost function can be converted easily into a normalised fitness function by, for example, f: $\mathbf{R}^+ \to \mathbf{R}^+$,

$$f(H) = \frac{1}{1 + J(H)} \in (0,1]. \tag{4}$$

2.2 Performance Metrics for Design Unification

Performance indices should reflect all specifications that need to be considered in practice. They can be in the form of an overall composite objective or cost function, as commonly adopted by control engineers. They can also, preferably, be in the form of multiple independent criteria, if a 'least commitment' principle is to be adopted at an early stage of design (Guan and MacCallum 1996). Thus, for a given application, a control system can be automatically designed or invented if a search, machine learning or optimisation algorithm can accommodate these objectives under practical constraints.

2.2.1 The Fundamental Index

In general, the closed-loop performance of a control system under design may be assessed by an inverse-indexed 'cost function' or metric norm either in the frequency domain:

$$J_f(H) = \|E(jw)\|_x , \tag{5}$$

or in the time domain:

$$J_t(H) = \|e(t)\|_x . \tag{6}$$

Remark 1 This implies that control system design may be carried out by search and optimisation using either time or frequency domain based performance indices.

Remark 2 A performance index based design may be assessed using any one of the common norms, as all linear metrics are equivalent, i.e., they are bounded linearly by one another.

Note that, however, different norms' selectivity in indexing can be different. For example, an index based on L_∞ loses selectivity completely if the maximum error is not greater than $e(0)$ for a step tracking, as often happens to any reasonable design.

Special cases of the Fundamental Index (FI) are two commonly used indices as listed below.

Integral of Absolute Error (IAE) (Levine 1996):

$$J_{\mathrm{IAE}} = \sum_t |e(t)| = \|e(t)\|_1 . \tag{7}$$

Integral of Square Error (ISE) (Levine 1996):

$$J_{\mathrm{ISE}} = \sum_t e^2(t) = \|e(t)\|_2^2 . \tag{8}$$

In the frequency-domain, this is equivalent to the

Frequency Integral of Square Error (FISE):

$$J_{\mathrm{FISE}} = \sum_w |E(jw)|^2 = \|E(jw)\|_2^2 = N \sum_t e^2(t) , \tag{9}$$

where N denotes the number of samples in both the time and the frequency domains. Eq. (9) is obtained from Parseval's energy equivalence theorem in time and frequency domains.

Remark 3 Eqs. (8) and (9) imply that time and frequency domain indices can be equivalent and hence the design of an LTI control system under this index can be unified into any one domain.

Remark 4 An ad hoc LTI control scheme may be represented by a uniform scheme through the modification to the FI of (5) and/or (6).

2.2.2 FI Implicit to Robust Stability

For a linear control system, if the open-loop system is stable, then the Nyquist plot of the denominator in (1) will not encircle its origin in any way. This means that for relatively large stability margins, the denominator plot should be relatively far away from its origin and its magnitude should have a relatively large value.

Remark 5 Minimising the FI leads indirectly to robust stability and hence largely meets Spec. 1, owing to the norm equivalence.

Remark 6 In an infinity norm based uniform robust control system design, L_∞ stable implies bounded-input and bounded-output stable.

For cases where gain and phase margins are specifically required (although unnecessary, given Remark 5), these can be added to a composite index or can form a second, independent, index in non-committal multi-objective optimisation (See Section 3).

2.2.3 Modifying Indices to Stress Steady-State Errors

(i) Multiplicative Indexing Building Blocks

The time itself forms a simple gradual, ramp weighting function. This has been adopted in the following indices.

Integral of Time Weighted Absolute Error (ITAE) (Levine 1996):

$$J_{\mathrm{ITAE}} = \sum_t t|e(t)| = \| t\, e(t)\|_1 . \tag{10}$$

Integral of Time Weighted Square Error (ITSE) (Levine 1996):

$$J_{\mathrm{ITSE}} = \sum_t t\, e^2(t) = \left\|\sqrt{t}\, e(t)\right\|_2^2 . \tag{11}$$

Integral of Square Time Weighted Square Error (ISTSE) (Zhuang and Atherton 1993):

$$J_{\mathrm{ISTSE}} = \sum_t t^2\, e^2(t) = \| t\, e(t)\|_2^2 . \tag{12}$$

which presents a double emphasis on steady-state suppression.

Remark 7 Inversely, dividing a frequency-domain index by frequency itself stresses a similar weighting on the steady-state.

Integral of Inverse Frequency Weighted Absolute Error (IIFAE):

$$J_{\mathrm{IIFAE}} = \sum_w \frac{1}{w}|E(jw)| = \left\|\frac{E(jw)}{w}\right\|_1 . \tag{13}$$

Integral of Inverse Frequency Weighted Square Error (IIFSE):

$$J_{\mathrm{IIFSE}} = \sum_w \frac{1}{w}|E(jw)|^2 = \left\|\frac{E(jw)}{\sqrt{w}}\right\|_2^2 . \tag{14}$$

Integral of Square Inverse Frequency Weighted Square Error (ISIFSE):

$$J_{\mathrm{ISIFSE}} = \sum_w \left|\frac{1}{w}E(jw)\right|^2 = \left\|\frac{E(jw)}{w}\right\|_2^2 . \tag{15}$$

(ii) Additive Indexing Building Blocks

Without loss of generality, for a unit step command $r(t)$, the steady-state error can be represented in both the time and frequency domains as:

$$|e(\infty)| = \left|\frac{1}{1+H(0)G(0)}\right| . \tag{16}$$

Remark 8 If the design of a control system requires emphasis on suppressing against steady-state errors, building block (16) may be added to the FI in either the time or the frequency domain.

Note that if the L_∞ norm is used to replace L_2, an emphasis will be placed on the maximum magnitude of the spectrum that occurs near the dc frequency, where static steady-state errors contribute most. Similarly, the time domain cost can be in L_1, which tends to accumulate the absolute values of errors that are significantly contributed $\forall t \to \infty$.

Remark 9 In the time domain, another additive 'weighting' block against steady-state errors is the L_1 norm added to the FI.

Remark 10 In the frequency domain, a simple 'weighting' block against steady-state errors is the L_∞ norm added to the FI.

2.2.4 Modifying Indices to Emphasise Transients Performance

Without loss of generality, for a unit step command $r(t)$, the initial transient may be represented in both the time and frequency domains as:

$$|e(0)| = \left|\frac{1}{1+H(\infty)G(\infty)}\right|. \tag{17}$$

Remark 11 If fast rising and suppressing overshoots and undershoots are required, weighting against the transient may be realised in either the time or the frequency domain by adding to the FI.

Remark 12 In the time domain, another additive 'weighting' block highlighting transient is the L_∞ norm added to the FI. The L_∞ norm in the time domain places an emphasis on the maximum amplitude of errors, which often occurs at $t \to 0$ for a 'hard-start' command such as a step (unless the controller is too poorly designed or the closed-loop system is of a nonminimum-phase).

Remark 13 In the frequency domain, a simple 'weighting' block highlighting transient is the L_1 norm added to the FI. This is to accumulate frequency response values significantly $\forall w \to \infty$, which are contributed most at transients.

Remark 14 For a 'hard' command, transients already contribute a relatively large amount of error and are, hence, seldom weighted in practice.

Remark 15 The change of error instead of the error itself may be used to highlight the transient performance and/or to penalise chattering. This is equivalent to multiplying by frequency, as transients constitute high frequencies.

Such indices are proposed below.

<u>Integral of Absolute Error Derivative (IAED)</u>:

$$J_{\text{IAED}} = \sum_t |\dot{e}(t)| = \|\dot{e}(t)\|_1 . \tag{18}$$

This is in effect equivalent to weighting the error by frequency:

<u>Integral of Frequency Weighted Absolute Error (IFAE)</u>:

$$J_{\text{IFAE}} = \sum_w w|E(jw)| = \|wE(jw)\|_1 . \tag{19}$$

<u>Integral of Square Error Derivative (ISED)</u>:

$$\begin{aligned} J_{\text{ISED}} &= \sum_t \dot{e}^2(t) = \|\dot{e}(t)\|_2^2 \\ &= \frac{1}{N}\|wE(jw)\|_2^2 . \end{aligned} \tag{20}$$

<u>Integral of Time Weighted Absolute Error Derivative (ITAED)</u>:

$$J_{\text{ITAED}} = \sum_t t|\dot{e}(t)| = \| t\,\dot{e}(t)\|_1 . \tag{21}$$

<u>Integral of Time Weighted Square Error Derivative (ITSED)</u>:

$$J_{\text{ITSED}} = \sum_t t\,\dot{e}^2(t) = \left\|\sqrt{t}\,\dot{e}(t)\right\|_2^2 . \tag{22}$$

<u>Integral of Square Time Weighted Square Error Derivative (ISTSED)</u>:

$$J_{\text{ISTSED}} = \sum_t t^2\ \dot{e}^2(t) = \| t\,\dot{e}(t)\|_2^2 . \tag{23}$$

2.2.5 FI Implicit to Disturbance Rejection

Refer to Fig. 1 again. The magnitude of the disturbance transfer to the closed-loop output is give by

$$\left\|\frac{Y(jw)}{D(jw)}\right\| = \left\|\frac{1}{1+H(jw)G(jw)}\right\| \left\|G(jw)\right\| \tag{24}$$

Comparing this with (1), the following can be inferred.

Remark 16 Load disturbance rejection is maximised if the FI is minimised, largely meeting Spec. 4.

Note that, however, the best set-point following does not necessarily mean the best load disturbance rejection (Åström and Hagglund 1995).

2.2.6 FI Implicit to Robustness Against Plant Uncertainty

In Fig. 1, the magnitude of the sensitivity of the closed-loop transfer function to the plant transfer function is given by

$$\left\|S_G^{G_c}\right\| = \left\|\lim_{\Delta G \to 0} \frac{\Delta G_c(jw)/G_c(jw)}{\Delta G(jw)/G(jw)}\right\| = \left\|\frac{1}{1+H(jw)G(jw)}\right\| \tag{25}$$

Remark 17 The closed-loop sensitivity to the plant uncertainty is minimised if the FI is minimised, largely meeting Spec. 5. Often, the L_∞ norm may used here to represent maximum sensitivity.

2.3 Merit and Selectivity of Metrics

As an LTI system can generally be decomposed into first and second-order subsystems, its dominant dynamics are hence often characterised by a second-order system. Suppose that a design results in an overall closed-loop system that behaves close to a unity-gain second-order system. Then the performance of the closed-loop system will be regarded as too sluggish if it behaves 'over-damped'. If it is too 'under-damped', however, the transient performance will be unsatisfactory. Often, the damping factor, z, is regarded as 'good' if it is of a value between that resulting in a critically-damped system (ζ = 1.0) and that resulting in a resonance (z = 0.707).

2.3.1 Selectivity for Hard-Start Command Following

Controllers obtained by minimising different indices could result in different damping ratios. Hence, the ability of a design criterion in selectivity for optimisation should be assessed. Refer to (Graham and Lathrop 1953; Zhuang and Atherton 1993) for IAE, ISE and ITAE indices, their selectivity and those of some proposed in this paper are illustrated in this section.

Index values resulting from step following assessment are studied and are plotted in Fig. 2 against the selectivity in terms of the damping ratio. It can be seen that, if the resultant closed-loop system is of a second-order dominance, as found in most practical control systems, the use of different indices would result in a damping ratio ranging from 0.50 (ISE) to 1.00 (ISTED), extending to the infinity.

In optimisation, clearly, the use of the ISTSE and ITAE indices would hence offer their sharpest selectivity at $\zeta = 0.67$ and 0.75, respectively. An ITAE-selected controller should offer a high and near-resonant damping. Nevertheless, combining different indices together should provide a composite one that meets different needs a design.

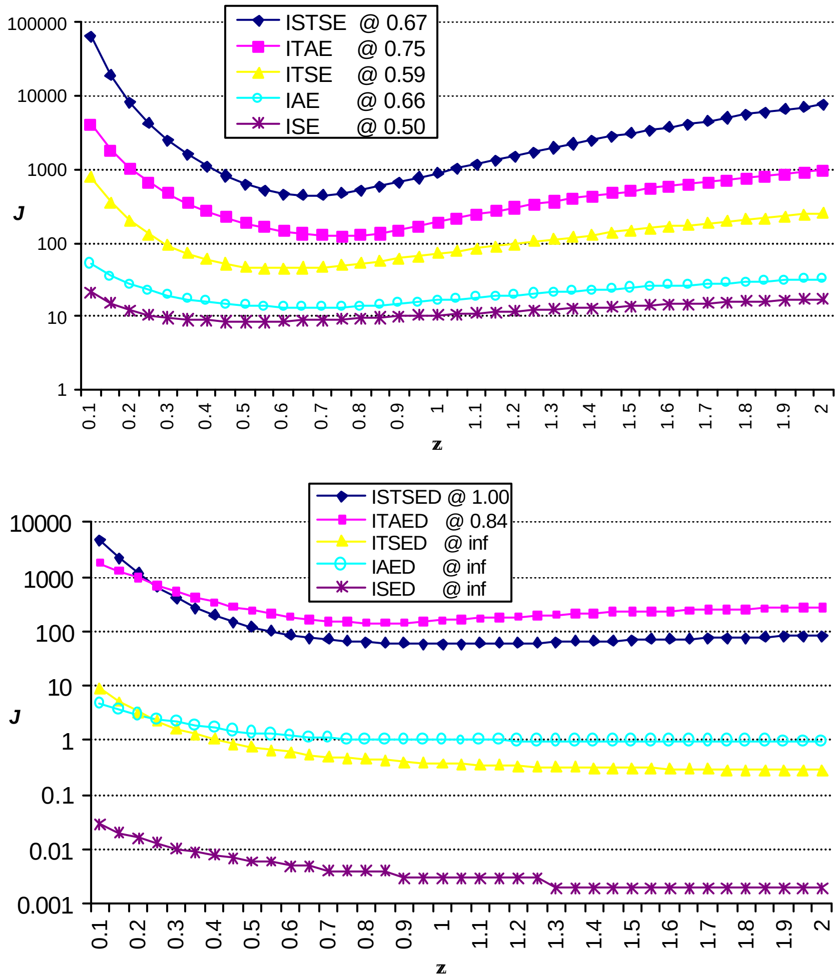


Fig. 2 Selectivity of performance metrics in terms of a damping ratio

2.3.2 Soft-Start and Selectivity

Refer to $F(s)$ and $C(s)$ in Fig. 1. The pre-filter $F(s)$ outside the loop is for robust considerations and model-following. It is often a unity gain first-order low-pass with a relatively small time-constant or a critically damped second-order filter with a relatively high natural frequency.

For practical application, a step response $C(s)$ to the hard command $R(s)$ can be used as a 'soft-start' command for the control system to follow, i.e., the dynamics of the closed-loop system is desired to follow that of $F(s)$. This 'model-following' control strategy (Levine 1996) is to avoid sharp acceleration in course-keeping or aircraft control, for example, and to minimise changes of actuator saturation. In practice, infinite current is not available to guarantee a perfect hard-command following.

For model-following applications, study the selectivity of some metrics. Without loss of generality, suppose that the natural frequency of the model to follow is ten times higher than that of the plant to be controlled. The analysis of the selectivity of some indices is are shown in Fig. 3. As can be seen, the selectivity almost remains the same as those for a hard-start command following.

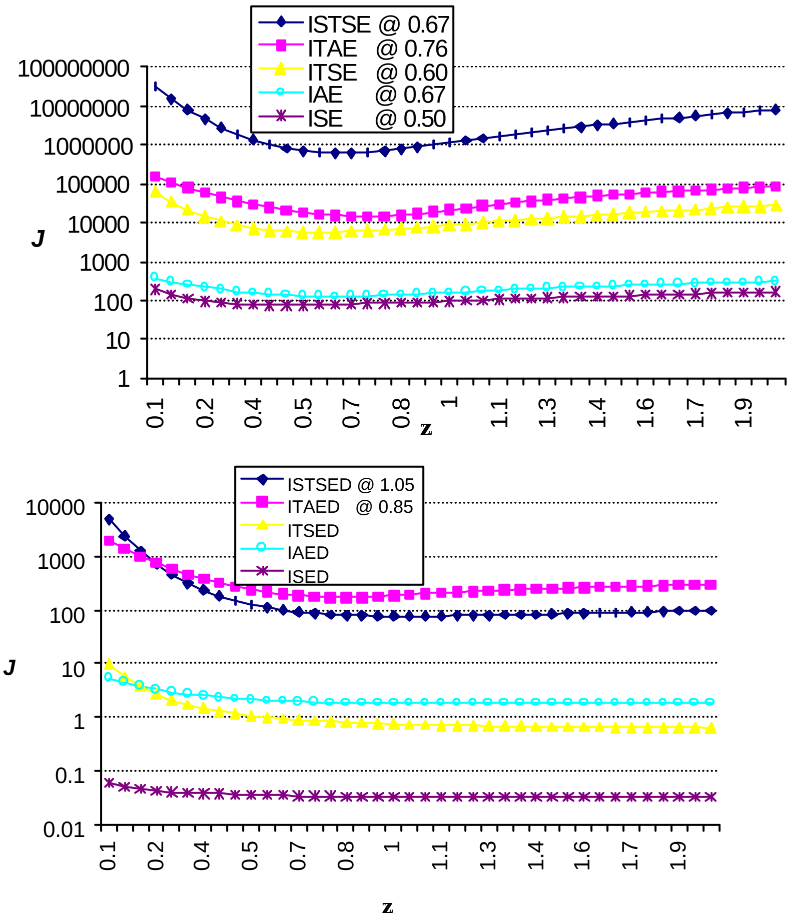


Fig. 3 Selectivity of performance metrics for soft-command following

2.4 Reconciling Accuracy and Chattering with Hybrids

As observed earlier, frequency weighting may be replaced by derivatives. A simple such hybrid that places an emphasis on both tracking accuracy and actuator chattering is given by:

$$J_{\mathrm{H}} = J_{\mathrm{ITSE}} + J_{\mathrm{ITSED}} \tag{26}$$

which is shown to be very effective in control system design automation enabled by evolutionary computation (Li *et al* 1995, Ng 1995, Tan 1997).

Another hybrid example may be constructed in the same domain by multiplying the basic index with a 'notch filter':

$$J_{\mathrm{Notch}} = \sum_{w}\left(w+\frac{1}{w}\right)E(jw)E(-jw) = \left\|\sqrt{w+\frac{1}{w}}\,E(jw)\right\|_2^2 . \tag{27}$$

Further, it has been discussed that steady-state is emphasised by time weighting and transients by frequency. These two weightings can hence be combined together to tackle both steady-state and transient problems. An example of such a hybrid index is given by:

$$J_{\mathrm{TF}} = \sum_{t} te^2(t) + \sum_{w} w|E(w)|^2 = \left\|\sqrt{t}\,e(t)\right\|_2^2 + \left\|\sqrt{w}\,E(jw)\right\|_2^2 . \tag{28}$$

If need, another hybrid index may also be formed to offer a selecting point on z, new from the existing ones. However, the ten indices shown in Fig. 2 have already covered a large range.

3. UNIFICATION AND CACSD AUTOMATION

3.1 Unified LTI Control Scheme

Almost all types of LTI controllers are in the form of a transfer function matrix or its corresponding state space equation when the design is eventually complete. The order and the coefficients of the transfer function, for example, vary with the design objective or specific control law pre-selected. For instance, a controller designed out of an LQR scheme tends to offer a minimised quadratic error with some minimal control effort, whilst an H_∞ controller offers robust performance with a minimal mixed sensitivity function. Although the obtained coefficients and orders of these two types of controllers may be different, the common purpose of both control laws is to devise an LTI controller to offer a closed-loop system that meets certain customer specifications in either the frequency or time domain.

Regardless of a pre-selected control scheme, the design of an LTI controller can be unified under performance satisfaction with objectives that an application engineer wishes to achieve. Without loss of generality, consider single-input and single-output for argument. Shown in Fig. 1, the controller is of a structure unified and universally described by

$$H(s) = \frac{U(s)}{E(s)} = \frac{p_n s^{n-m-1} + \cdots + p_{m+2}s + p_{m+1}}{p_m s^m + \cdots + p_2 s + p_1} \tag{29}$$

where $p_i \in \mathbf{R}^+, \forall i \in \{0,1,\ldots,n\}$, are the coefficients to be determined in the design together with the controller order, under satisfaction of multiple design objectives.

Here $L^{-1}[U(s)] = u(t)$ is the controller output voltage with usually a hard-constraint saturation range in practice, such as the limited drive voltage (or current), subject to the plant's dead-zone, backlash and hysteresis. The error input to the controller, $L^{-1}[E(s)] = e(t)$, is calculated by subtracting the reference from the plant output, which is often subject to a delayed measurement, sensor nonlinearity and restricted amplitude of the A/D converter used.

While these practical issues can hardly be addressed all together using analytical design tools, they can now all be simulated on computers. It should also address the issue of interpreting human engineers' perception of merit into a form that may be utilised for use with a CACSD package. Also, suppressing $u(t)$ in LQG and H_∞ schemes, for example, becomes unnecessary, as the control signal is automatically limited by actuator constraints and its rate of change can also be incorporated as constraints in the simulations. Further, this treatment is more practice-oriented that mathematically penalising $u(t)$ for the sake of the optimisation to be solvable by conventional means.

3.2 Evolution Enabled CAutoCSD

Evolutionary computation based design techniques make use of simulation results just like a human designer to 'intelligently' transform the simulation problem into its reverse problem of design. A multiple coefficient design space characterised by a performance index is usually multi-modal, which is hard to accommodate by traditional optimisation methods. In certain applications, multiple indices may also need to be considered independently. Often, practical system constraints need to be taken into account in the design. These make it almost impossible to use analytical or conventional optimisation or search techniques to automate the design.

Emulating the Darwinian-Wallace principle of 'survival-of-the-fittest' in natural selection and genetics, evolutionary algorithms have been found to be very effective and efficient in searching a poorly understood, irregular and complex space for optimisation and machine learning. The EA can start designs from the application engineer's library of existing designs or from an initial population completely of random candidates. As summarised in Fig. 4, an EA encodes a candidate design in artificial 'chromosomes' P_1, P_2 and P_3, and then varies these chromosomes by 'crossover' and 'mutation' operators to 'bread' offspring candidates for future improvements, generation by generation, in a similar way to natural evolution. This is in effect a parallel search and machine learning process, in which the EA makes use of past trial information in a similar, intelligent, manner to human designers.

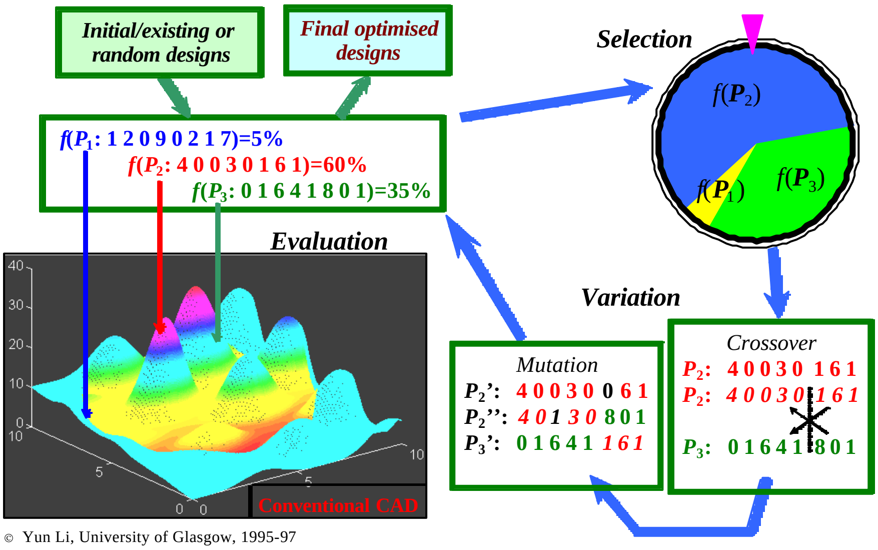


Fig. 4 Computer-automated design through artificial evolution

A conventional CACSD package that provides simulation results, taking into account actuator saturation, is used to evaluate the performance of candidate controllers in terms of plant outputs, closed-loop errors and control signal provision. Artificial evolution then enables CACSD to become CAutoCSD. By trading off precision slightly using nondeterministic adjustments, the EA exponentially reduces the search time compared with exhaustive search and thus provides much improved tractability and efficiently in design automation (Li *et al* 1995; Li and Haeussler 1996; Ng 1995; Tan 1997).

Such an algorithm evaluates performances of candidate solutions at multiple points simultaneously and thus efficiently approaches the global optimum. Evolutionary computation can search multi-objective, globally optimised solutions to many practical engineering problems that cannot be solved by conventional means. A number of automatically 'evolved' top-performing candidates will finally merge as optimal designs. Its unique search and adaptive learning power has facilitated design automation, transforming a manual iterative tuning process based on existing CAD or CACSD packages into CAutoCSD (Chong and Li 2002; Tan and Li 2001). The advantages of such CAutoCAD over traditional CACSD approaches include meeting multiple design objectives, offering design quality improved beyond the present performance bounds, and dramatically reducing design cycle and time-to-market.

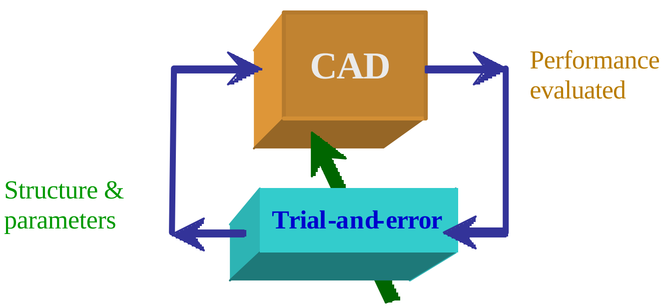


Fig. 5 Evolutionary computing transforming manual CACSD to CAutoCSD

4. APPLICATIONS

4.1 Application to a DC Servomechanism

A DC motor can often be modelled simply as an LTI plant where a small time-delayed may appear. The motor is more difficult for velocity control, as it is a Type 0 system, where no integral element is apparent in the system, and hence it will result in a steady-state error when following a step command. The LTI model of this system is given by the second-order differential equation:

$$\ddot{w}(t-0.06)+\left(\frac{JR+LB}{LJ}\right)\dot{w}(t-0.06)+\left(\frac{RB}{LJ}\right)w(t-0.06)=\left(\frac{K_T}{LJ}\right)v(t) \tag{30}$$

where

$v(t) \in [0\text{V}, 5\text{V}]$	: the field control voltage with a saturation limit and allowing no braking voltage,
$w(t) \in \mathbf{R}$	: the angular velocity calculated from a Gray-code shaft encoder,
$K_T = 13.5$ NmA^{-1}	: the torque constant for a fixed armature current,
$R = 9.2\ \Omega$	: the resistance of the winding,
$L = 0.25$ H	: the winding inductance,
$J = 0.001$ kgm^2	: the inertia of the motor shaft combined with a load, and
$B = 2.342 \times 10^3$ Nms	: the friction coefficient of the shaft, changing to 1.34×10^3 Nms when an eddy current brake is released.

Although it is unnecessary to use a third-order controller for a second-order plant, it is used here to test the ability of the unified scheme and the EA in finding an appropriate and optimal controller. Thus, a third-order uniform controller for the hybrid time and frequency index of Eq. (26) has been designed automatically using an EA. The resulting transfer function of the best controller is given by

$$H(s)=\frac{7.2s^3+153.2s^2+426.9s+293.8}{1.0s^3+27.6s^2+29.2s+0.0} \tag{31}$$

The coefficients in the denominator clearly indicate that an integrator is recommended by the EA automatically, for offering a zero steady-state error. The numerator indicates that a differentiator is also recommended for a good transient response. The closed-loop performance is shown in Fig. 5, where a step-down following was commanded 5 seconds after the initial rising step. Note that the lower limit of $v(t)$ is 0 V, which allows no braking voltage, and hence the motor can only slow down by friction. To test the robustness of the control system designed out of the performance index, the plant parameter value, B, was varied at t = 3 s and t = 8 s. Note that this uncertainty in plant parameters was not modelled and a robust controller was not explicitly requested in the design. However, the automated design procedure managed to achieve the implicit robustness as remarked in Section 2.2.2.

The control action that provides the above closed-loop response is shown in Fig. 6, subject to the hard voltage limits of [0 5] V. It can be seen that the feasibility of incorporating such a practical constraint in the evolutionary design not only yields a practical control signal that offers the optimised performance, but also eliminates the need for artificially minimising the control energy in scheme-dependent modern control approaches, such as the LQR.

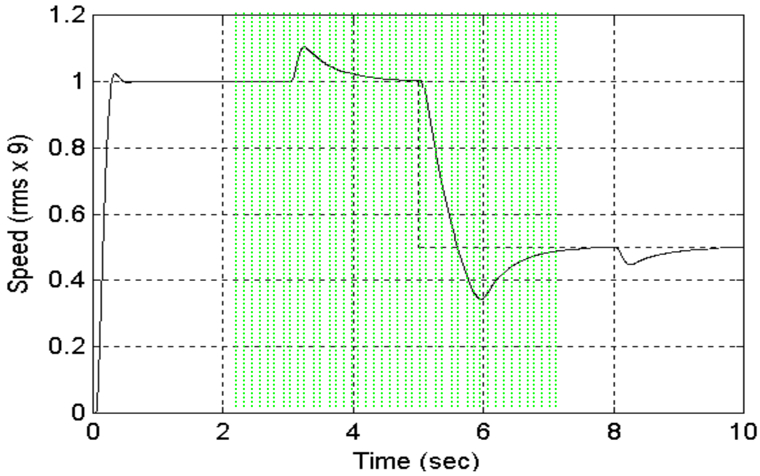


Fig. 6 Performance of the automatically evolved unified LTI controller (where the brake was released at t = 3 and re-applied at t = 8 s)

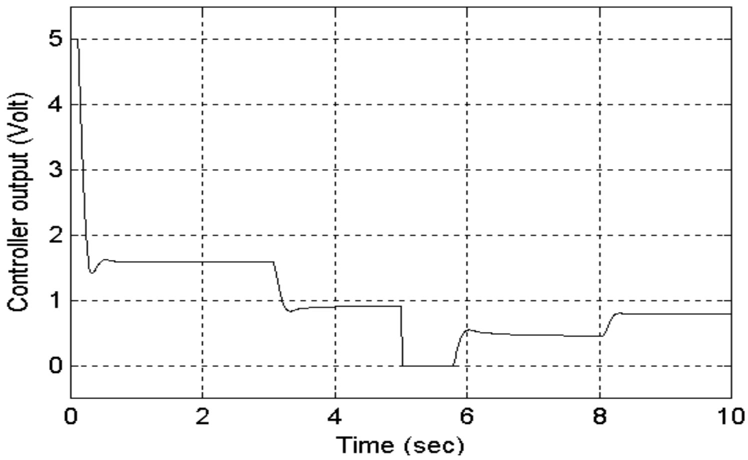


Fig. 7 Actuator constraint control action of the unified-scheme controller

4.2 Multi-Objective Scheme for Robust Control

Consider the well-studied non-minimal phase plant (Doyle *et al* 1992):

$$G(s) = \frac{-6.475s^2 + 4.0302s + 175.77}{5s^4 + 3.5682s^3 + 139.5021s^2 + 0.0929s} \tag{32}$$

For such a non-minimum phase plant, the design of a robust controller using an ad hoc scheme can be difficult. Here, with the unified approach, the design task is to obtain a linear controller that satisfies a multiple number of time and frequency domain specifications as detailed in Table 1. These specifications can be conveniently made by a practising engineer, including the graphical boundaries of an acceptable response as shown by the clear area in Fig. 8.

Table 1. Time and frequency domain design objectives with their targeted values and priority

Customer specifications		Objectives	Goals	Priority
Frequency	1. Stability (Closed-loop poles)	$\mathrm{Re}[p_i] > 0, \forall i$	0 on RHP	4
	2. Closed-loop sensitivity or	$S(jw)$	<1	2

domain	disturbance rejection			
	3. Plant uncertainty	$T(jw)$	<1	2
Time domain	4. Actuator saturation	Act	$u \le 0.5$ V	3
	5. Rise time	T_{rise}	4 s	1
	6. Overshoots	O_{shoot}	0.05	1
	7. Settling time (5%)	T_{settling}	7 s	1
	8. Steady-state error	SS_{error}	0.01 s	1

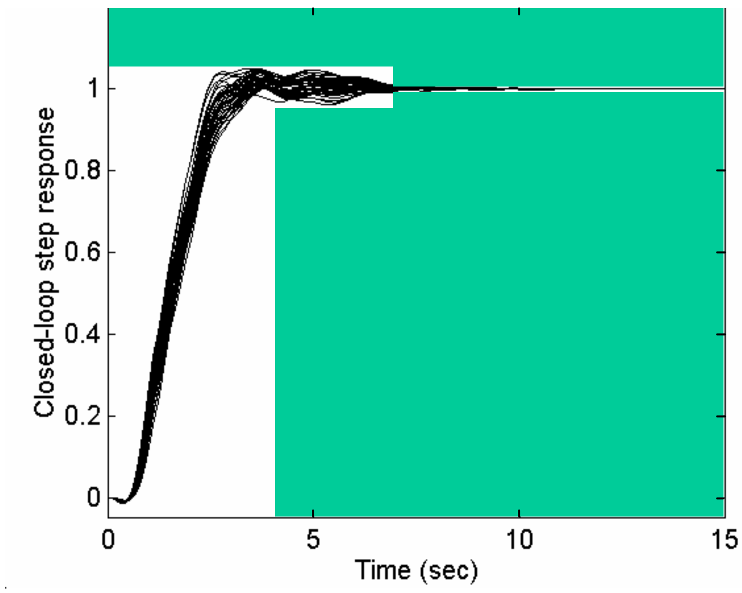


Fig. 8 Graphical requirement on the response and design results

The design requirement in this example also includes that for explicit robustness against disturbance and unstructured plant uncertainty under certain level of tolerances defined by the weighting functions of W_1 and W_2, as shown in Fig. 9. Although determination of the objectives and the priorities vector may be a subjective matter and depends on the performances requirement, ranking the priorities may be unnecessary and can be ignored for a 'minimum-commitment' design (Guan and MacCallum 1996). If, however, the engineer commits himself to prioritising the objectives, it is a much easier task than weighting the objectives, which is somewhat guesswork in conventional optimisation. It is obvious that other design specifications such as gain margins, phase margins and noise rejection (which can be quantified by distinctive LQG or H_2 norms) may also be added to the design if necessary.

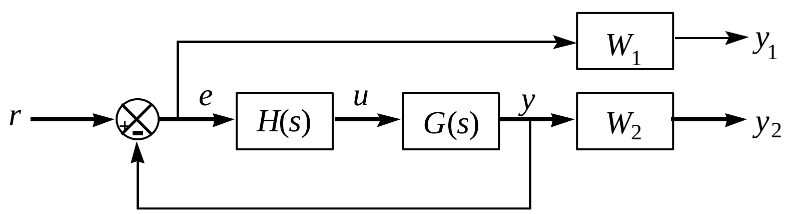


Fig. 9 Tolerance added to the robust controller design in an the unified scheme

The tolerance in terms of sensitivity functions obtained, W_1 and W_2, are plotted in Fig. 10, which shows the time domain performance of 69 controllers, resulting from the unified design scheme enabled by evolutionary computation, that satisfy all the requirements listed in Table 1. While the orders of candidate controllers are free to vary in the evolution, an order up to three was preferred in the application and hence this requirement was accommodated in the design objectives as well.

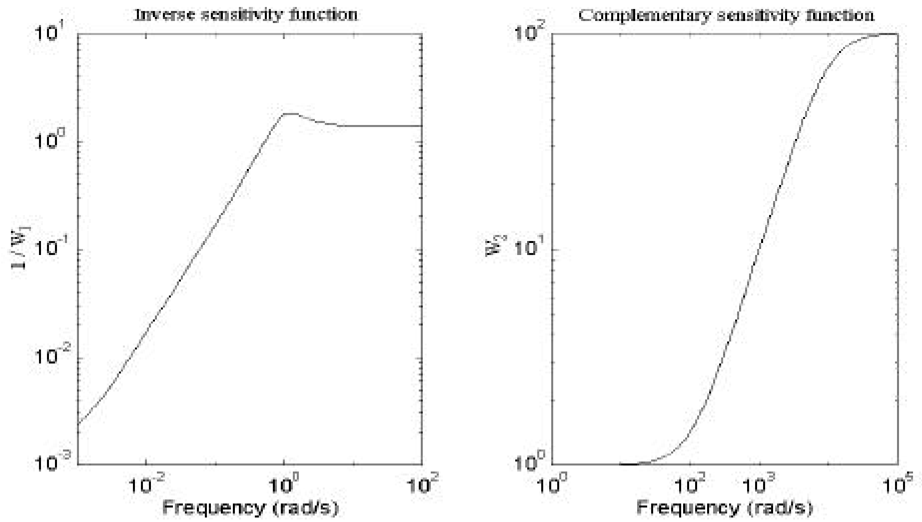
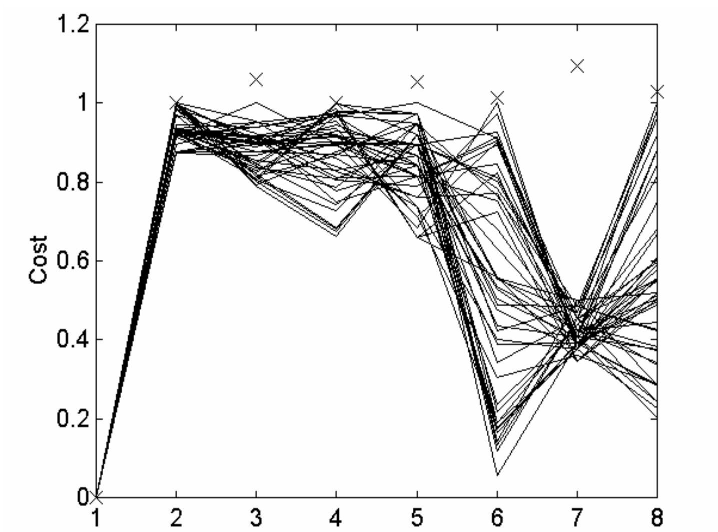


Fig. 10 Tolerance in terms of sensitivity functions obtained, W_1 and W_2

At the end of the evolutionary CAutoCAD process, the applications engineer can transparently examine trade-offs between the design specifications, including constraints and even zoom into the region of interested points before selecting one final controller for on-line test and commissioning. The trade-off between the multiple objectives for the 69 resultant controllers is depicted in Fig. 11, where each line represents one candidate controller recommended by the EA. The abscissa shows the design objectives and the coordinate shows the normalised cost of controllers in each objective domain, with the cross marks indicating the design goals.



Objectives: *Sta* *S* *T* *Act* T_{rise} O_{shoot} T_{settling} SS_{error}

Fig. 11 Trade-offs between the objectives for the resultant controllers

Trade-offs between adjacent objectives result in crossing lines between them, whereas concurrent lines indicate that the specifications do not compete with one another. For example, the stability (*Sta*) and actuator saturation (*Act*) specifications appear not to compete with each other. As anticipated, the sensitivity function (*S*) and the complementary sensitivity (*T*) do not appear to be reconcilable. It is also found that the goal values for the settling time was set too high. The information contained in the trade-off graph suggests that a more stringent goal setting for settling time, overshoot and steady-state error can be achieved. An additional merit of the EA enabled design unification is that the priorities or goals can be changed at any time during the evolution process if so desired.

4.3 Trajectory PID Network for a Nonlinear Chemical Process

In this application, a nonlinear chemical process at Mitsubishi Chemicals is considered. The dynamics of the constant-temperature reaction is modelled by*:

$$\frac{dy(t)}{dt} = -Ky^2(t) + \frac{1}{V}[d - y(t)]u(t) \tag{33}$$

where

$y(t)$ = concentration in the outlet stream (mol l^{-1})
$u(t)$ = flow rate of the feed stream (l h^{-1})
K = rate of reaction (mol^{-1} l^{-1} h^{-1})
V = reactor volume (l)
d = concentration in the inlet stream (mol l^{-1}).

A static model for various operating points or equilibria is often the first step in investigating a nonlinear process. The model can be used to determine the range of control signals, the sizing of the actuators and the resolution of selected sensors. In practice, such a static model can be obtained either from closed-loop or open-loop tests, which have a physical interpretation for a stable process (Åström and Hagglund 1995).

The process is expected to operate within an output range [0 1] mol l^{-1} and the desired output level is 0.53 mol l^{-1}. Using equi-increment step inputs, open-loop static tests of this process are obtained and listed in Table 2. The transient responses are shown in Fig. 12.

Table 2. Equi-increment step inputs and corresponding static responses

Input $u(\infty)$ (l h^{-1})	0	1	2	3	4	5	6	7	8	9	10
Output $y(\infty)$ (mol l^{-1})	0	0.44	0.61	0.73	0.83	0.91	0.98	1.05	1.11	1.16	1.21

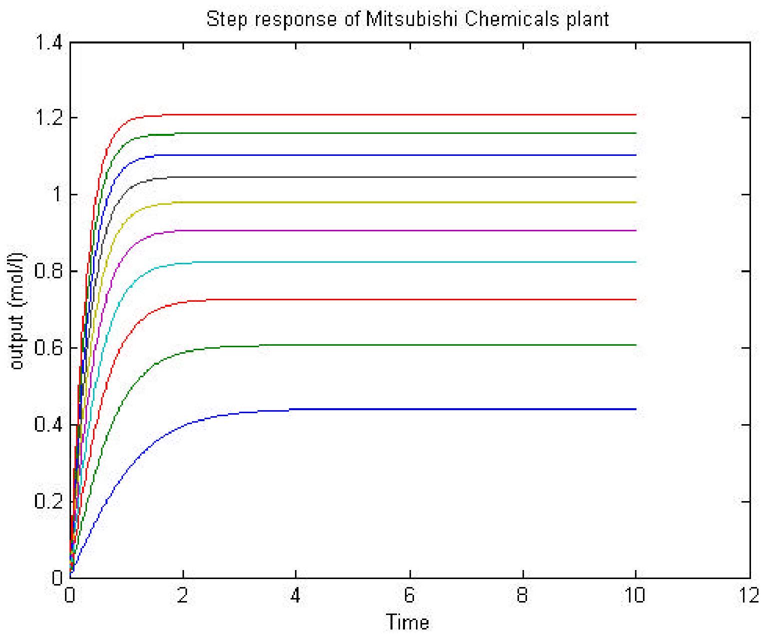


Fig. 12 Open-loop tests of the nonlinear process within its operating envelope

* The authors are grateful to Mitsubishi Chemicals Corp., Japan, for providing this case study.

Static model data shown in Table 2 are often readily available for an established plant. If a first-principles model is available, however, the plant's nonlinearity may be illustrated analytically. For example, corresponding to Eq. (32), a given level of $u(\ )$ determines a $y(\ )$ by the parabolic equation:

$$Ky^2 + \frac{1}{V}u\,y - \frac{d}{V}u = 0 \tag{34}$$

which is often termed an 'equilibrium manifold', as illustrated by the solid curve in Fig. 13, which agrees with Table 2 and Fig. 12, but reveals more details in the severely nonlinear region.

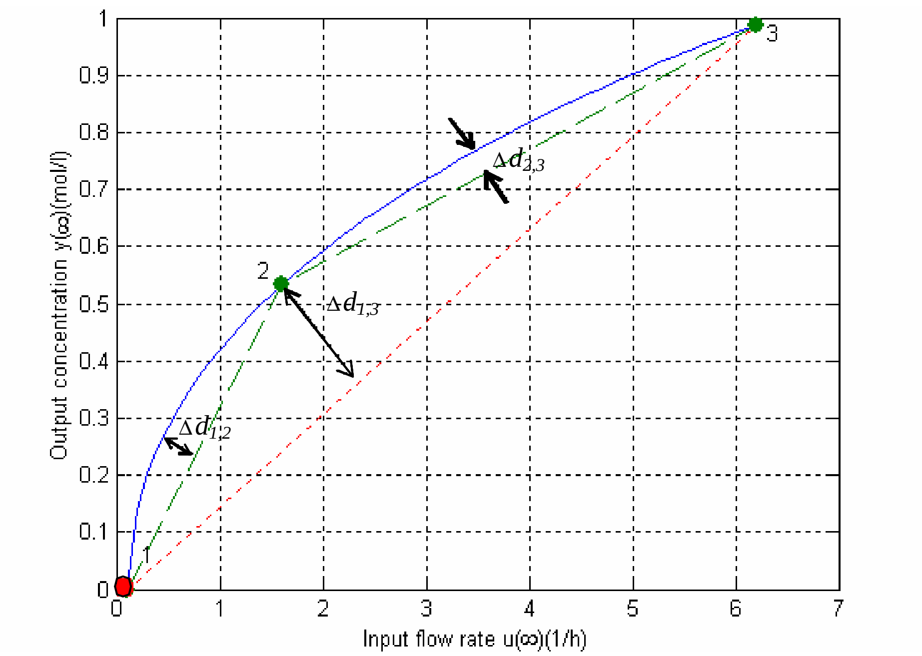


Fig. 13 The equilibrium manifold of the nonlinear process within its operating envelope

Apparently, to control such a nonlinear process, a nonlinear controller may be used, but this lacks the transparency on stability and familiarity that a practising engineer would be confident with. According to a recent survey, over 90% of industrial control systems in use are realised in various forms of proportional plus integral plus derivative (PID) control (Åström and Hagglund 1995). Hence, the use and design of a simple PID control system is desirable. The simplest PID structure is given by

$$\frac{U(s)}{E(s)} = K_p + K_i\frac{1}{s} + K_d s = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) \tag{35}$$

Based on the analysis of the nonlinear process and its model, however, the use of a straightforward PID control would be inadequate. Fortunately, since a static model or open-loop step response data are often available for an existing plant, the use of a Trajectory Controller Network (TCN) appears to be the most appropriate (Chong and Li 2002). In this application, each node of the TCN is a straightforward three-term PID controller, placed along the operating trajectory as shown in Fig. 13.

Such a TCN is easily designed. Two initial TC nodes, 1 and 3, can be placed to bracket the operating envelope for the anticipated output range [0 1] mol l^{-1}. Then, more nodes are added in between during the automated design process. The simplest way of adding new nodes is to add one pre-fixed at the desired set point at $y(\ )$ = 0.53 mol l^{-1}, as depicted in Fig. 13.

Alternatively, a new node can be inserted automatically during CAutoCSD along the trajectory between the initial nodes, at where the maximum distance $\Delta d_{1,3}$ occurs. If $\Delta d_{1,3\ max} > d$, where d is the tolerance that the application engineer may specify, more nodes can be added in a bi-sectional search manner. According to the maximum sizes of $\Delta d_{1,2}$ and $\Delta d_{2,3}$, this process continues until the tolerance d is satisfied.

In this example, a simple three-node network is designed automatically, under a hard constraint on the input flow rate limited by the range [0 10] l h^{-1}. The third node optimally found is at at $y(\;)$ = 0.553 mol l^{-1}, which happens to be close to the desired set point. The scheduling between the three PID controller nodes is determined by either the input or the output levels, using a simple stepped switch or triangular-shaped activation functions $S_1(y)$, $S_2(y)$ and $S_3(y)$. This is quite similar to assigning the degree of memberships in a fuzzification process if a fuzzy control system (Chowdhury and Li 1998). In this case study, simple input activation is used. Through a CAutoCSD process, an overall controller yields:

$$u(t) = [S_1 \quad S_2 \quad S_3]\begin{bmatrix} 176.5 & 8666 & 0.07968 \\ 341.4 & 3675 & 0.4789 \\ 422.3 & 27901 & 0.05248 \end{bmatrix}\begin{bmatrix} 1 \\ p^{-1} \\ p \end{bmatrix} e(t) \tag{36}$$

where p is the differentiation operator. It can be seen that the PID controllers become more aggressive when the operating trajectory approaches the top end of the equilibrium manifold, with a decreasing plant 'gain'.

To test another index and the ability of CAutoCSD, a composite IAE and IAED index is used in the design automation with actuator constraints easily accommodated. The closed-loop control results for the set point of 0.53 mol l^{-1}, desired by Mitsubishi Chemicals, are shown in Fig. 14. The performance is compared with PID controllers designed for this operating point using the Internal Model Control (IMC) and Ziegler-Nichols (Z-N) methods. It can be see that the TCN performs significantly better that single PID controller based methods.

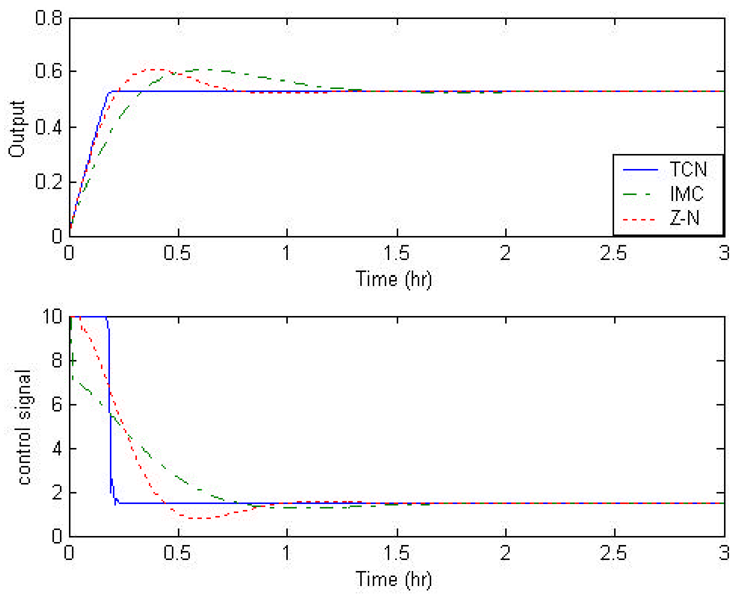


Fig. 14 Performance of the three-PID TCN regulator compared with single PID based methods

In essence, an LTI building block based TCN is a nonlinear controller overall. The switching between the nodes is through soft activation and hence imposes no threat of actuator damage. To thoroughly test the performance of the PID network designed for the nonlinear problem, the system was driven throughout the allowed operating trajectory. Results are shown in Fig. 15. It can be seen that the TCN constantly outperforms a single PID controller in both step up and step down. Note also that the hard constraint on the input flow rate limited by the range [0 10] l h^{-1} is automatically handled by CAutoCSD and, even with this constraint, extremely high performance is achieved throughout the entire operating envelope.

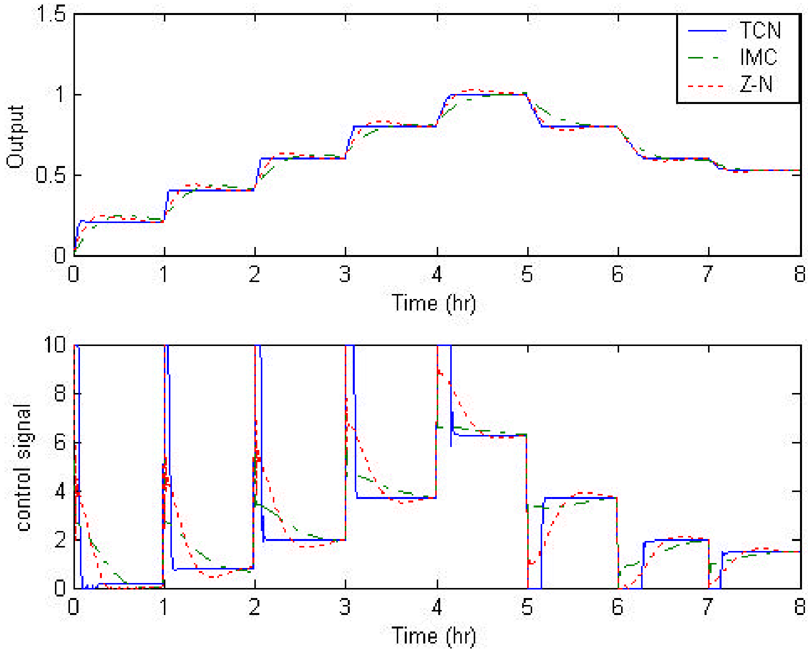


Fig. 15 Performance reliability of the PID network in the entire operating envelope

To test the robustness of this TCN regulator, a 20% load disturbance has been added between *t* = 2 and *t* = 4 h and between *t* = 6 and *t* = 8 h. The results are shown in Fig. 16. It can be seen that the TCN offers a much better performance with a fast rise, no overshoots, and an extremely good rejection to the load disturbance. This is significant in that excellent rejection and set-point following are both achieved at the same time, which are often irreconcilable in many control system designs (Åström and Hagglund 1995).

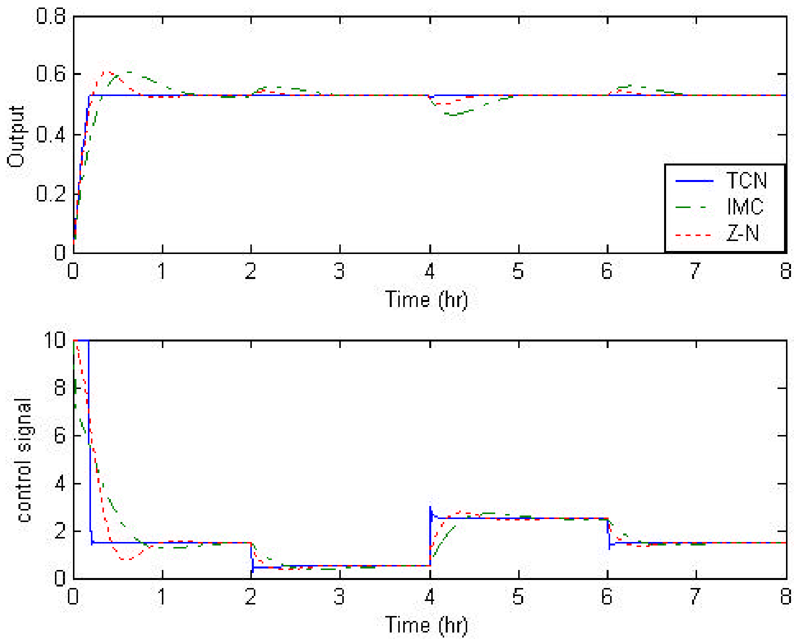


Fig. 16 Performance of the TCN regulator subject to a 20% load disturbance

5. CONCLUSION

The paper has attempted to set LTI control system design to a unified scene by formulating various design schemes under index-based optimal design, hence transforming the present CACSD into

CAutoCSD. Specifications and objectives in control system design are first analysed and assessed. Different merit and selectivity of some commonly used indices are analysed and compared, together with some new indices proposed. Issues concerning interpreting human engineers' perception of control system performance into a form that may be utilised for CACSD automation are also addressed. The advantage of using the EA based global optimising and searching tool for automating CACSD is discussed.

The techniques developed in this paper have been applied to and illustrated by three design problems. It has been shown that such unification is feasible analytically and is practical with the recent progress in evolutionary computing based extra-numeric, multi-criterion search and optimisation techniques. The CAutoCAD provides an integrator for velocity control of the DC motor, meets multiple objectives in designing an LTI controller for the non-minimum phase plant and offers a high-performing LTI network for the nonlinear chemical process.

The performance-prioritised unification is also shown to be able to relieve practising engineers from having to select a particular control scheme and from sacrificing certain performance goals resulting from pre-committing to the scheme. Through the studies reported in this paper, we hope to have answered positively to the questions raise in the Introduction section.

6. REFERENCES

Åström, K.J., and T. Hagglund, 1995. *PID Controllers: Theory, Design and Tuning*, Instrument Society of America.

Chong, C., and Y. Li, 2002. Multi-objective design of networking control systems for nonlinear plants and stability analysis, Proc 5th Asia-Pacific Conference on Measurement and Control, Dali, China, 8-12 July, 185-190.

Chowdhury, M., and Y. Li, 1998. Learning fuzzy control systems directly from the environment, *Int. J. Intelligent Systems*, **13**(10-11), 949-974.

Doyle, J.C., B. Francis and A. Tannenbaum, 1992. *Feedback Control Theory,* Macmillan Publishing Company, New York.

Graham, D., and R.C. Lathrop, 1953, The synthesis of optimum response: Criteria and standard forms, Part 2, *Trans AIEE*, **72**, 273-288.

Guan, X., and K. J. MacCallum, 1996. Adopting a minimum commitment principle for computer aided geometric design systems, in *Artificial Intelligence in Design*, Gero, J. S. and Sudweeks, F., eds, Kluwer Academic Publishers, 623-639.

Kashiwagi, H., 1983, *Automatic Control*, Asakura, Tokyo.

Levine, W.S., (ed), 1996. *The Control Handbook*, CRC Press/IEEE Press.

Li, Y., K.C. Tan, K.C. Ng and D.J. Murray-Smith, 1995. Performance based linear control system design by genetic evolution with simulated annealing, Proc. 34th IEEE CDC, New Orleans, 731-736.

Li, Y., and A. Haeussler, 1996. Artificial evolution of neural networks and its application to feedback control, *Int. J. Artificial Intelligence in Engineering*, **10**(2), 143-152.

Ng, K.C., 1995. *Switching Control Systems and Their Design Automation via Genetic Algorithms*, Ph.D. Thesis, Dept. of Electronics and Electrical Engineering, University of Glasgow.

Tan, K.C., 1997. *Evolutionary Methods for Modelling and Control of Linear and Nonlinear Systems*, Ph.D. Thesis, Dept. of Electronics and Electrical Engineering, University of Glasgow.

Tan, K.C., and Y. Li, 2001. Performance-based control system design automation via evolutionary optimization, *Int J Eng Applications of Artificial Intelligence*, **14**(4), 473-486.

Zhuang, M., and D.P. Atherton, 1993. Automatic tuning of optimum PID controllers, *IEE Proceeding – Part D*, **140**(3), May.