International Journal of Automation and Computing 1 (2004) 76-88

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

Yun Li*, Kiam Heong Ang, Gregory C.Y. Chong, Wenyuan Feng, Kay Chen Tan
Department of Electronics and Electrical Engineering, University of Glasgow, Glasgow G12 8LT, UK
Hiroshi Kashiwagi
Department of Mechanical Engineering and Material Science, Kumamoto University, Japan

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 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 at relieving practising engineers from having to select a particular control scheme and from sacrificing certain performance goals resulting from pre-commitment to such schemes. With recent progress in evolutionary computing based extra-numeric, multi-criterion search and optimisation techniques, such unification of LTI control schemes becomes feasible, analytical and practical, 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, and meets multiple objectives in the design of an LTI controller for a non-minimum phase plant and offers a high-performance LTI controller network for a non-linear chemical process.

Keywords: Linear time-invariant (LTI), proportional plus integral plus derivative (PID), control system design (CSD), computer-aided control system design (CACSD), performance index, genetic algorithms (GA), evolutionary computation (EC), 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 everything that a practising engineer desires[1,2]. A single control scheme is often restricted ad hoc to a 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 them may also 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 computable with the help of an optimiser. However, such schemes impose some theoretical conditions and restrictions in order that optimisation may be carried out. It is therefore difficult to accommodate practical constraints or any structural optimisation of controllers[3,4]. 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[1,2].

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. Recent progress in evolutionary and soft computing techniques has enabled the replacement of human trial-and-error based iterative processes with computer-automated ones[3,5,6,7]. 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 design of LTI control systems might now be unified under one banner: "*performance satisfaction*"[3,7].

Manuscript received April 23, 2004; revised June 23, 2004.
*Corresponding author. *E-mail address*: Y.Li@elec.gla.ac.uk

Therefore, 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 might such unification be achieved?

(c) Is it viable to unify them with multiple design criteria or 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 also addressed. Section 3 proposes a way to unify and shows how EC's 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 been used together with time-domain specifications to form a composite design objective[1~3]. These are summarised in Section 4, while 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 be much more flexible and creative. The first step towards unifying LTI controllers is to consider how a design meets practical performance requirements, instead of how the design is applied to a specific scheme or particular domain. Such a unified design approach aims to eliminate the need for the 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 to incorporate 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 for a given plant $G(s)$ with controller $H(s)$. With reference to Fig.1 for notations, without loss of generality, for the case $F(s) = 1$,

$$E(s) = R(s) - Y(s)$$

$$= \frac{1}{1 + H(s)G(s)}[R(s) - G(s)D(s)] \qquad (1)$$

where $D(s)$ represents a disturbance, which may be coloured and also modelled to include plant uncertainty.

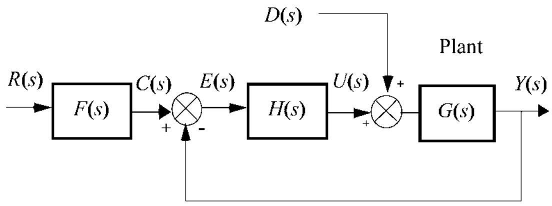


Fig.1 Generic feedback control system with a model-following command

The ultimate objective of a control system design is hence to find an $H(s)$ such that:

$$E(s) = 0, \ \forall s, \ D(s) \qquad (2)$$

or:

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

This ultimate goal means that condition (2) or (3) needs to be satisfied under plant and environmental uncertainties. This is impossible in a 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. the current limit). In fact, should (2) or (3) be met regardless of the time and the 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, $J : R^n \to R^+$, must be devised to measure *how closely* 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[1~3]:

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);

Spec 5. Robustness to 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, $f: R^+ \to R^+$,

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

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 adopted at an early stage of design[8]. 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:

$$J_f(H) = \| E(j\omega) \|_x \tag{5}$$

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 however, that 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 $e(0)$ in step tracking, as often happens in any reasonable design.

Special cases of the fundamental index (FI) are two commonly used indices listed below:

(a) Integral of absolute error (IAE)[2]:

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

(b) Integral of square error (ISE)[2]:

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

In the frequency domain, this is equivalent to the:

(c) Frequency integral of square error (FISE):

$$J_{\text{FISE}} = \sum_\omega |E(j\omega)|^2 = \| E(j\omega) \|_2^2 = N \sum_t e^2(t) \tag{9}$$

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

Remark 3. Equations (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 a single domain.

Remark 4. An ad hoc LTI control scheme may be represented by a uniform scheme through the modification of the FI by (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 its norm equivalence.

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

1) Multiplicative indexing building blocks

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

(a) Integral of time weighted absolute error (ITAE)[2]:

$$J_{\text{ITAE}} = \sum_t t|e(t)| = \| te(t) \|_1 \tag{10}$$

(b) Integral of time weighted square error (ITSE)[2]:

$$J_{\text{ITSE}} = \sum_t te^2(t) = \| \sqrt{t}e(t) \|_2^2 \tag{11}$$

(c) Integral of square time weighted square error (ISTSE)[9]:

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

which provides a double emphasis to steady-state suppression.

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

(a) Integral of inverse frequency weighted absolute error (IIFAE):

$$J_{\mathrm{IIFAE}} = \sum_\omega \frac{1}{\omega}|E(j\omega)| = \left\| \frac{E(j\omega)}{\omega} \right\|_1 \tag{13}$$

(b) Integral of inverse frequency weighted square error (IIFSE):

$$J_{\mathrm{IIFSE}} = \sum_\omega \frac{1}{\omega}|E(j\omega)|^2 = \left\| \frac{E(j\omega)}{\sqrt{\omega}} \right\|_2^2 \tag{14}$$

(c) Integral of square inverse frequency weighted square error (ISIFSE):

$$J_{\mathrm{ISIFSE}} = \sum_\omega |\frac{1}{\omega}E(j\omega)|^2 = \left\| \frac{E(j\omega)}{\omega} \right\|_2^2 \tag{15}$$

2) Additive indexing building blocks

Without loss of generality, for a unit step command $r(t)$, steady-state error can be represented in both the time and the 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 in suppression 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 the 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 non-minimum 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 $\forall\omega \to \infty$, 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:

(a) Integral of absolute error derivative (IAED):

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

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

(b) Integral of frequency weighted absolute error (IFAE):

$$J_{\mathrm{IFAE}} = \sum_\omega \omega|E(j\omega)| = \| \omega E(j\omega) \|_1 \tag{19}$$

(c) Integral of square error derivative (ISED):

$$J_{\mathrm{ISED}} = \sum_t \dot{e}^2(t) = \| \dot{e}(t) \|_2^2 = \frac{1}{N} \| \omega E(j\omega) \|_2^2 \tag{20}$$

(d) Integral of time weighted absolute error derivative (ITAED):

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

(e) Integral of time weighted square error derivative (ITSED):

$$J_{\mathrm{ITSED}} = \sum_t t\dot{e}^2(t) = \| \sqrt{t}\dot{e}(t) \|_2^2 \tag{22}$$

(f) Integral of square time weighted square error derivative (ISTSED):

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

2.2.5 FI implicit to disturbance rejection

Referring to Fig.1 again, the magnitude of disturbance transfer to the closed-loop output is given by:

$$\left\| \frac{Y(j\omega)}{D(j\omega)} \right\| = \left\| \frac{1}{1 + H(j\omega)G(j\omega)} \right\| \| G(j\omega) \| \qquad (24)$$

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 however, that the best set-point following does not necessarily mean the best load disturbance rejection[10].

2.2.6 FI implicit to robustness against plant uncertainty

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

$$\begin{aligned} \| S_G^{G_c} \| &= \left\| \lim_{\triangle G \to 0} \frac{\triangle G_c(j\omega)/G_c(j\omega)}{\triangle G(j\omega)/G(j\omega)} \right\| \\ &= \left\| \frac{1}{1 + H(j\omega)G(j\omega)} \right\| \end{aligned} \qquad (25)$$

Remark 17. Closed-loop sensitivity to 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 often characterised by a second-order system. Suppose that a design result 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 ζ is regarded as "good" if it is of a value between the resulting in a critically-damped system (ζ=1.0) and resulting in a resonance (ζ=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 [9,11] 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 plotted in Fig.2 against selectivity in terms of damping ratio. It can be seen that, if the resultant closed-loop system is of 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 infinity.

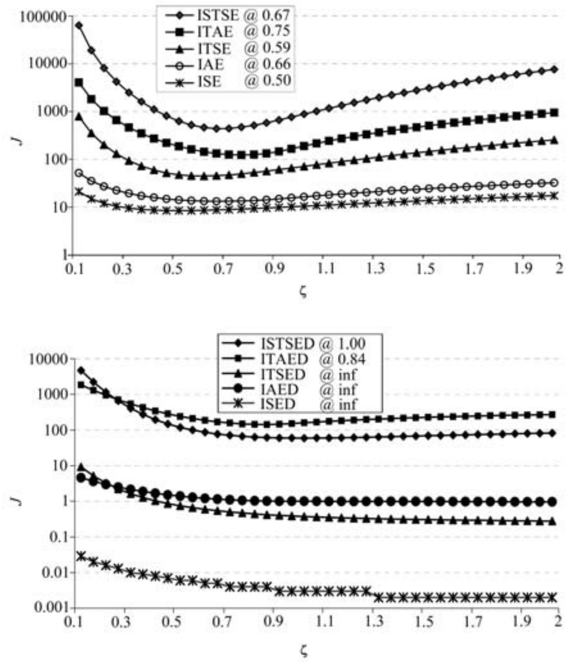


Fig.2 Selectivity of performance metrics in terms of a damping ratio

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

2.3.2 Soft-start and selectivity

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

In practical applications, 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 a closed-loop system is desired to follow that of $F(s)$. This "model-following" control strategy[2] is used to avoid sharp acceleration in course-keeping or aircraft control for example, and to minimise changes of actuator saturation. In practice, an infinite current is not available to guarantee perfect hard-command following.

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

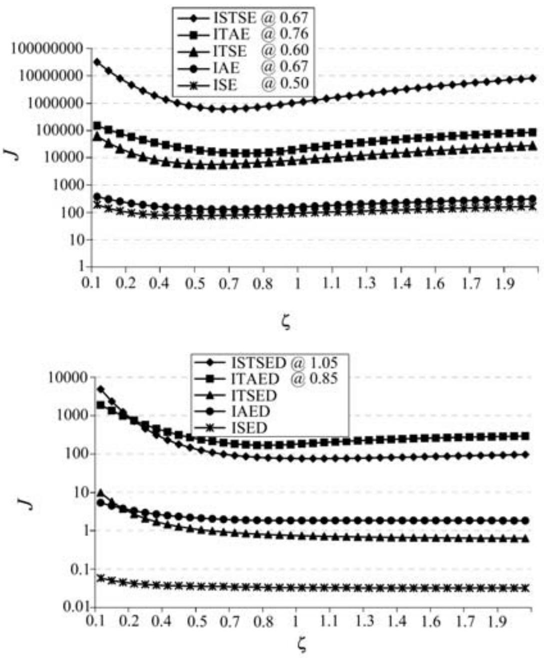


Fig.3 Selectivity of performance metrics for soft-command following

2.4 Reconciling accuracy and chattering with hybrids

As observed earlier, frequency weighting may be replaced by derivatives. One such simple hybrid that places emphasis on both the tracking accuracy and the 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 computation[3,5,6].

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

$$\begin{aligned} J_{\mathrm{Notch}} &= \sum_{\omega} \Big(\omega + \frac{1}{\omega}\Big) E(j\omega)E(-j\omega) \\ &= \Big\| \sqrt{\omega + \frac{1}{\omega}} E(j\omega) \Big\|_2^2 \end{aligned} \tag{27}$$

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:

$$\begin{aligned} J_{\mathrm{TF}} &= \sum_{t} te^2(t) + \sum_{\omega} \omega |E(\omega)|^2 \\ &= \| \sqrt{t} e(t) \|_2^2 + \| \sqrt{\omega} E(j\omega) \|_2^2 \end{aligned} \tag{28}$$

If needed, another hybrid index may also be formed to offer a selecting point on ζ, in addition to the existing ones. However, the ten indices shown in Fig.2 already cover 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 coefficients of the transfer function, for example, vary with the design objective or specific pre-selected control law. For instance, a controller designed out of an LQR scheme tends to offer 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 the 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 the 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, a single-input and single-output can be considered. As shown in Fig.1, the controller has a unified structure and can be 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 $p_i \in R^+$, $\forall i \in \{0, 1, \ldots, n\}$ 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 which usually has a hard-constraint saturation range in practice, such as a limited drive voltage (or current), subject to a plant's dead-zone, backlash and hysteresis. The error input to the controller, $L^{-1}[E(s)] = e(t)$ is calculated by subtracting a reference from the plant output, which is often subject to the delayed measurement, the sensor nonlinearity and the restricted amplitude of the A/D converter used.

While these practical issues can hardly be addressed all together using analytical design tools, they can all be simulated on computers now. It should also be possible to 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 simulations. Further, this treatment is more practice-oriented than mathematically penalising $u(t)$ for the sake of solving optimisation by conventional means.

3.2 Evolution enabled CAutoCSD

Evolutionary computation based design techniques make use of simulation results just as a human designer "intelligently" transforms a 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 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 an application engineer's library of existing designs or from an initial population of completely 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 "breed" offspring candidates for future improvement, 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.

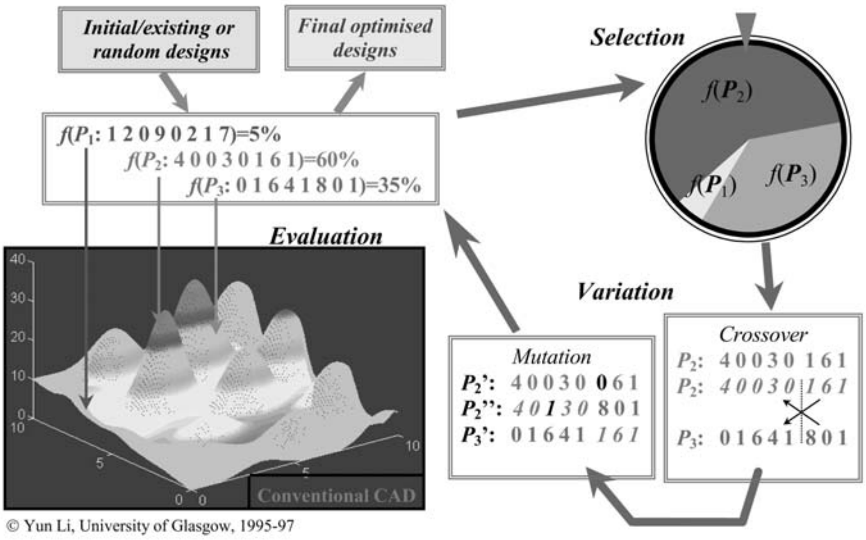


Fig.4 Computer-automated design through artificial evolution

Taking actuator saturation into account, a conventional CACSD package that provides simulation results, 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 precision slightly using nondeterministic adjustments, an EA exponentially reduces the search time compared to exhaustive search and thus provides much improved tractability and efficiency in design automation[3~6].

Such an algorithm evaluates the performance of candidate solutions at multiple points simultaneously and thus efficiently approaches a 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 powers facilitate design automation, transforming a manual iterative tuning process based on existing CAD or CACSD packages into CAutoCSD[7,12]. The advantage of such CAutoCAD over traditional CACSD approaches includes the ability to meet multiple design objectives, to offer design quality improvements beyond present performance bounds, and to dramatically reduce the design cycle and time-to-market.

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-delay may appear. The velocity of a motor is difficult to control, as it is a Type 0 system, where no integral element is apparent and hence a steady-state error will result when following a step command. The LTI model of this system is given by the second order differential equation:

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

where $v(t)\in[0V,5V]$, is the field control voltage with a saturation limit allowing no braking voltage; $\omega(t)\in R$ is the angular velocity calculated from a Gray-code shaft encoder; $K_T = 13.5\text{NmA}^{-1}$, is the torque constant for a fixed armature current; $R=9.2\Omega$, is the resistance of winding; $L=0.25\text{H}$, is winding inductance; $J=0.001\text{kgm}^2$ is the inertia of the motor shaft combined with a load; and $B=2.342\times10^3\text{Nms}$, is 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 (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, to offer zero steady-state error. The numerator indicates that a differentiator is also recommended for a good transient response. Closed-loop performance is shown in Fig.5, where a step-down 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, hence the motor is only slowed down by friction. To test the robustness of the control system designed from 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 Subsection 2.2.2.

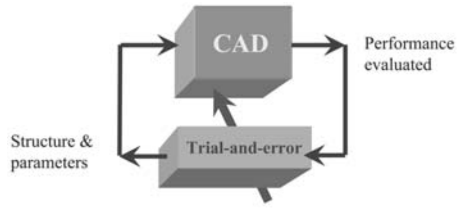


Fig.5 Evolutionary computing transforming manual CACSD to CAutoCSD

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 an evolutionary design not only yields a practical control signal that offers optimised performance, but also eliminates the need to artificially minimise the control energy in scheme-dependent modern control approaches, such as the LQR.

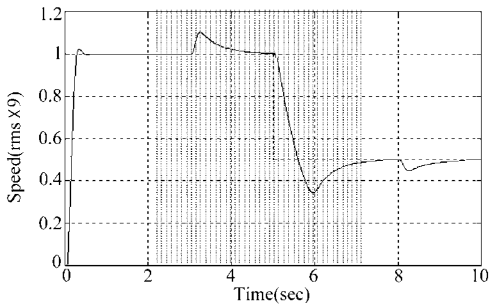


Fig.6 Performance of the automatically evolved unified LTI controller (where the brake was released at $t=3$s and re-applied at $t=8$s)

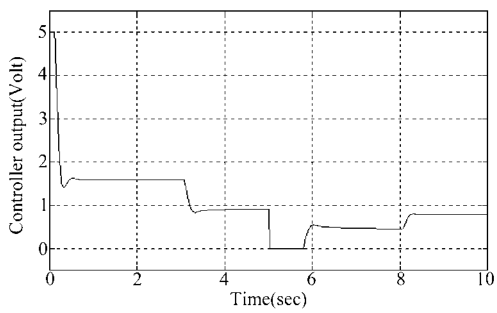


Fig.7 Actuator constraint control action of the unified-scheme controller

4.2 A multi-objective scheme for robust control

Consider the well-studied non-minimal phase plant[13]:

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

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.

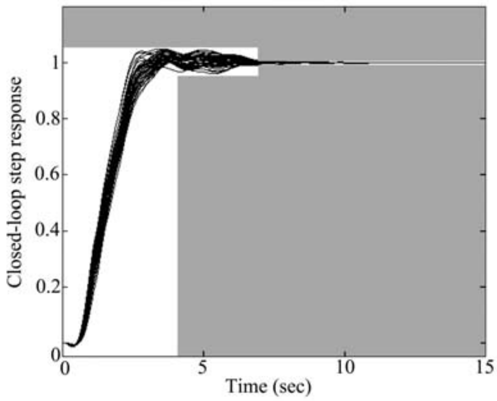


Fig.8 Graphical requirement on the response and design results

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

Customer specifications		Objectives	Goals	Priority
Frequency domain	Stability (Closed-loop poles)	$Re[p_i] > 0, \forall i$	0 on RHP	4
	Closed-loop sensitivity or disturbance rejection	$S(j\omega)$	< 1	2
	Plant uncertainty	$T(j\omega)$	< 1	2
Time domain	Actuator saturation	Act	$u \leqslant 0.5$V	3
	Rise time	T_{rise}	4s	1
	Overshoots	O_{shoot}	0.05	1
	Settling time (5%)	T_{settling}	7s	1
	Steady-state error	SS_{error}	0.01s	1

The design requirement in this example also includes that for explicit robustness against disturbance and unstructured plant uncertainty under certain levels of tolerance defined by the weighting functions of W_1 and W_2, as shown in Fig.9. Although determination of the objectives and priorities vector may be a subjective matter and depends on the performance requirement, ranking the priorities may be unnecessary and can be ignored for a "minimum-commitment" design[8]. If, however, the engineer commits himself to prioritising 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 and 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.

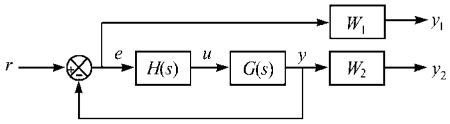
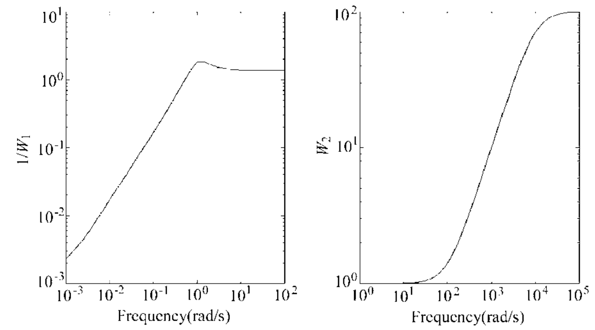


Fig.9 Tolerance added to the robust controller design in the unified scheme

The tolerances in terms of the obtained sensitivity functions, W_1 and W_2, are plotted in Fig.10, which show the time domain performance of 69 controllers, resulting from the unified design scheme enabled by evolutionary computation that satisfies all of 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.



(a) Inverse sensitivity function (b) Complementary sensitivity function

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 design specifications including constraints, and even zoom into a region of interesting points before selecting a 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 cross marks indicating design goals.

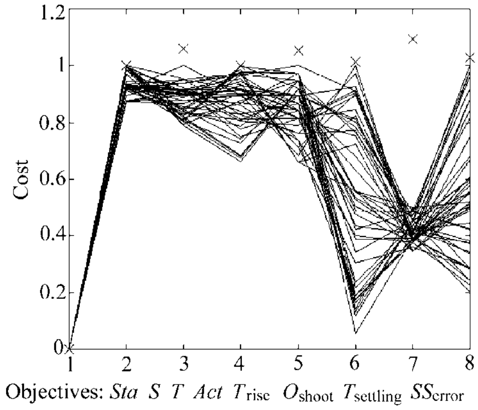


Fig.11 Trade-offs between the objectives of resultant controllers

Trade-offs between adjacent objectives result in lines being crossed between them, whereas concurrent lines indicate that 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 were 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 EA enabled design unification is that priorities or goals can be changed at any time during the evolution process if desired.

4.3 A trajectory PID network for a non-linear chemical process

In this application, a non-linear chemical process at Mitsubishi Chemicals is considered. The dynamics of the constant-temperature reaction are modelled by[1]:

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

where $y(t)$ is the concentration in the outlet stream (mol l^{-1}), $u(t)$ is the flow rate of the feed stream (l h^{-1}), K is the rate of reaction (mol^{-1} l^{-1} h^{-1}), V is the reactor volume (l), d is the 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 non-linear process. The model can be used to determine the range of control signals, the sizing of actuators and the resolution of selected sensors. In practice, such a static model can be obtained either from closed-loop or from open-loop tests, which have a physical interpretation in stable processes[10].

The process is expected to operate within an output range [0 1] mol l^{-1} and a 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. Transient responses are shown in Fig.12.

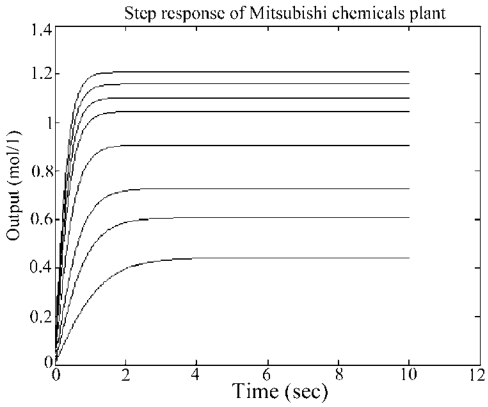


Fig.12 Open-loop tests of the non-linear process within its operating envelope

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

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

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

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 its severely non-linear region.

[1] The autors are grateful to Mitsubishi Chemicals Crop., Japan, for providing this case study.

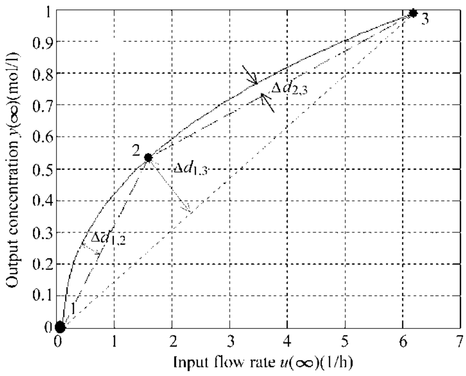


Fig.13 The equilibrium manifold of a non-linear process within its operating envelope

Apparently, to control such a non-linear process, a non-linear controller could 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[10]. Hence, the use and design of a simple PID control system is desirable. The simplest PID structure is given by:

$$\frac{U(s)}{E(s)} = K_p + K_i\frac{1}{s} + K_d s = K_p\Big(1 + \frac{1}{T_i s} + T_d s\Big) \quad (35)$$

Based on the analysis of a non-linear process and its model, however, the use of straightforward PID control would be inadequate. Fortunately, since a static model or open-loop step response data are often available in an existing plant, the use of a trajectory controller network (TCN) appears to be the most appropriate method[12]. 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 during the automated design process. The simplest way of adding new nodes is to add one pre-fixed at the desired set point of $y(\infty) = 0.53$ 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 the location where maximum distance $\triangle d_{1,3}$ occurs. If $\triangle d_{1,3\,\max} > \delta$, where δ is tolerance that the application engineer may specify, more nodes can be added in a bi-sectional search manner. According to the maximum sizes of $\triangle d_{1,2}$ and $\triangle d_{2,3}$, this process continues until the tolerance δ 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 $y(\infty) = 0.553$ mol l^{-1}, which happens to be close to the desired set point. 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 in a fuzzy control system[14]. In this case study, simple input activation is used. Through a CAutoCSD process, an overall controller yields:

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

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. Close-loop control results for a set point of 0.53 mol l^{-1}, desired by Mitsubishi Chemicals, are shown in Fig.14. Its 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 than the single PID controller based methods.

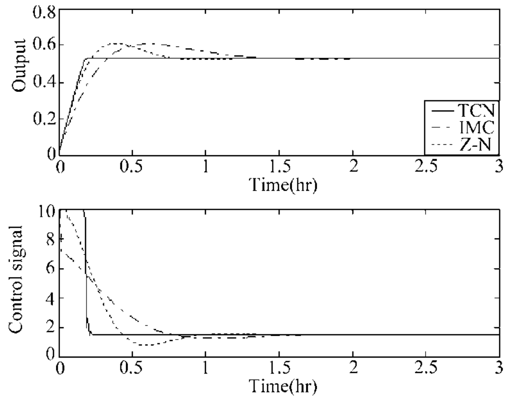


Fig.14 Performance of a three-PID TCN regulator compared to single PID based methods

In essence, an LTI building block based TCN is a non-linear controller. Switching between nodes occurs through soft activation and hence imposes no threat to actuator damage. To thoroughly test the performance of the PID network designed for this non-linear 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 a hard constraint on the input flow rate limited by the range $[0 \quad 10]\,\mathrm{l\,h^{-1}}$ is automatically handled by CAutoCSD and, even with this constraint, extremely high performance is achieved throughout the entire operating envelope.

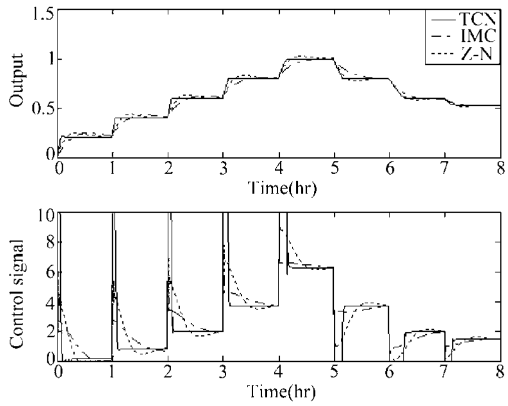


Fig.15 Performance reliability of the PID network in its entire operating envelope

To test the robustness of this TCN regulator, a 20% load disturbance was added between $t = 2$h and $t = 4$h and between $t = 6$h 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 of 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[10].

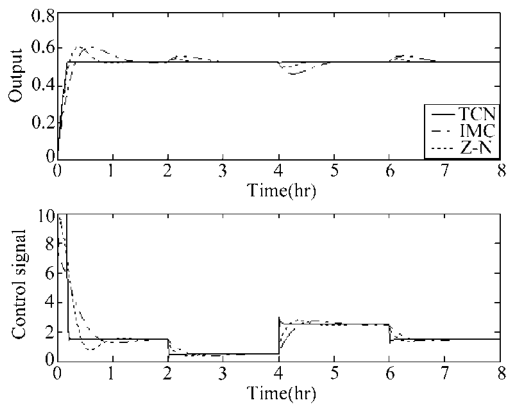


Fig.16 Performance of the TCN regulator subject to a 20% load disturbance

5 Conclusion

This paper attempted to set LTI control system design within a unified scene by formulating various design schemes under index-based optimal design, hence transforming existed CACSD into CAutoCSD. Specifications and objectives in control system design were first analysed and assessed. Different merits and selectivity of some commonly used indices were analysed and compared, together with some new proposed indices. Issues concerning interpreting human engineers' perception of control system performance into a form that may be utilised for CACSD automation were also addressed. The advantage of using EA based global optimising and searching tools for automating CACSD was discussed.

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

Performance-prioritised unification was also shown to be able to relieve practising engineers from having to select particular control schemes and from sacrificing certain performance goals resulting from precommitment to a scheme. Through the studies reported in this paper, we hope to have answered positively the questions raised in the Introduction Section.

References

[1] H. Kashiwagi, *Automatic Control*, Asakura, Tokyo, 1983.
[2] W. S. Levine(ed), *The Control Handbook*, CRC Press/IEEE Press, 1996.
[3] Y. Li, K. C. Tan, K. C. Ng, D. J. Murray-Smith, Performance based linear control system design by genetic evolution with simulated annealing, *Proc. 34th IEEE CDC*, New Orleans, pp. 731−736, 1995.
[4] Y. Li, A. Haeussler, Artificial evolution of neural networks and its application to feedback control, *Int. J. Artificial Intelligence in Engineering*, vol. 10, no. 2, pp. 143−152, 1996.
[5] K. C. Ng, *Switching Control Systems and Their Design Automation via Genetic Algorithms*, Ph.D. Thesis, Dept. of Electronics and Electrical Engineering, University of Glasgow, 1995.
[6] K. C. Tan, *Evolutionary Methods for Modelling and Control of Linear and Non-linear Systems*, Ph.D. Thesis, Dept. of Electronics and Electrical Engineering, University of Glasgow, 1997.
[7] K. C. Tan, Y. Li, Performance-based control system design automation via evolutionary optimization, *Int J Eng Applications of Artificial Intelligence*, vol. 14, no. 4, pp. 473−486, 2001.
[8] X. Guan, K. J. MacCallum, Adopting a minimum commitment principle for computer aided geometric design systems, *Artificial Intelligence in Design*, J. S. Gero, F. Sudweeks(eds), Kluwer Academic Publishers, pp. 623−639, 1996.
[9] M. Zhuang, D. P. Atherton, Automatic tuning of optimum PID controllers, *IEE Proceeding – Part D*, vol. 140, no. 3, 1993.
[10] K. J. Åström, T. Hagglund, *PID Controllers: Theory, Design and Tuning*, Instrument Society of America, 1995.

[11] D. Graham, R. C. Lathrop, The synthesis of optimum response: Criteria and standard forms, Part 2, *Trans AIEE*, vol. 72, pp. 273−288, 1953.

[12] C. Chong, Y. Li, Multi-objective design of networking control systems for non-linear plants and stability analysis, *Proc 5th Asia-Pacific Conference on Measurement and Control*, Dali, China, 185−190, 2002.

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

[14] M. Chowdhury, Y. Li, Learning fuzzy control systems directly from the environment, *Int. J. Intelligent Systems*, vol. 13, no. 10−11, pp. 949−974, 1998.

Yun Li received a BSc from University of Sichuan in 1984, an MSc from UESTC in 1987 and a PhD from University of Strathclyde, UK, in 1990. During 1989-1990, he worked at National Engineering Laboratory, East Kilbride, and for Industrial Systems and Control Ltd, Glasgow. He became a Lecturer at University of Glasgow in 1991, establishing his Intelligent Systems research group, and has since led the group actively to solve systems and control problems using evolutionary computing and modern computational intelligence techniques. Later he established and chaired the IEEE CACSD Evolutionary Computation Working Group and the European Network of Excellent in Evolutionary Computing Workgroup on Systems, Control and Drives Industry.

He is currently a Senior Lecturer at University of Glasgow, UK, and a Visiting Professor to University of Electronic Science and Technology of China (UESTC). He had served on the IEE Scotland South West Committee for six years and on 24 international conference committees. He also served as a Visiting Professor to Kumamoto University, Japan, during July-September 2002. He has over 125 scientific publications. Dr Li is a member of the IEE and the IEEE and is a Chartered Engineer.

Kiam Heong Ang received his B.Eng. degree with First Class Honours in Electronics and Electrical Engineering from the University of Glasgow, U.K., in 1996. From 1997 to 2000, he was a software engineer with the Advanced Process Control group in Yokogawa Engineering Asia Pte. Ltd., Singapore. He is currently working toward the Ph.D. degree at the University of Glasgow, U.K.

His current research interests include evolutionary multi-objective optimisation, computational intelligence and engineering design optimisation.

Gregory Chong received his B.Eng. degree with First-Class Honors in Electronics and Electrical Engineering from University of Glasgow, UK, in 1999. He is currently working towards the Ph.D degree at the same university. His current research interests include evolutionary multi-objective and intelligent control for nonlinear systems.

Kay Tan received his B.Eng. degree with First-Class Honors in Electronics and Electrical Engineering and his Ph.D. degree from the University of Glasgow, UK, in 1994 and 1997, respectively. He is currently an Assistant Professor at the Department of Electrical and Computer Engineering, National University of Singapore.

He has authored and coauthored more than 100 journal and conference publications, and has served as a program or organizing committee member for many international conferences. He currently holds an Associate Editorship for IEEE Transactions on Evolutionary Computation. His current research interests include computational intelligence, evolutionary computing, intelligent control, and engineering design optimization.

Hiroshi Kashiwagi received B.E, M.E. and Ph.D. degrees in measurement and control engineering from the University of Tokyo, Japan, in 1962, 1964 and 1967 respectively.

In 1967 he became an Associate Professor and in 1976 a Professor at Kumamoto University. From 1973 to 1974, he served as a visiting Associate Professor at Purdue University, Indiana, USA. From 1990 to 1994, he was the Director at Computer Center of Kumamoto University. He has also served as a member of Board of Trustees of Society of Instrument and Control Engineers (SICE), Japan, Chairman of Kyushu Branch of SICE and General Chair of many international conferences held in Japan, Korea, China and India.

In 1994, Dr. Kashiwagi was awarded SICE Fellow for his contributions to the field of measurement and control engineering through his various academic activities. He also received the Gold Medal Prize at ICAUTO'95 held in India. In 1997, he received the "Best Book Award" from SICE for his new book entitled "M-sequence and its application" written in Japanese and published in 1996 by Shoukoudou Publishing Co. in Japan. In 1999, he received the "Best Paper Award" from SICE for his paper "M-transform and its application to system identification". His research interests include signal processing and applications, especially pseudorandom sequence and its applications to measurement and control engineering.