

C-GRASP PARA OTIMIZAÇÃO DE UM CONTROLADOR PI COM ANTI-WINDUP APLICADO A UMA VÁLVULA DE RECIRCULAÇÃO DE GASES DE ESCAPE

Júlio Xavier Vianna Neto^{*,+} e Leandro dos Santos Coelho^{*,#}

* Departamento de Engenharia Elétrica, Universidade Federal do Paraná
Pós-Graduação em Engenharia Elétrica, Av. Cel. Francisco H. dos Santos, 210
Centro Politécnico, CEP 81530-970, Curitiba, PR, Brasil

+ Divisão de Sistemas Mecânicos (DVSM), Departamento de Mecânica e Emissões (DPME)
Instituto de Tecnologia para o Desenvolvimento (LACTEC)
Av. Prof. Lothário Meissner, 01, CEP 80210-170, Curitiba, PR, Brasil

Grupo Produtrônica, Pós-Graduação em Engenharia de Produção e Sistemas (PPGEPS)
Pontifícia Universidade Católica do Paraná (PUCPR)
Rua Imaculada Conceição, 1150, CEP 80215-901, Curitiba, PR, Brasil

Emails: julio.neto@lactec.org.br, leandro.coelho@pucpr.br

Resumo – Em motores de combustão interna, há uma técnica para redução de emissão de óxido de nitrogênio, denominada Recirculação de Gases de Escape (RGE). Para que o sistema de RGE funcione de forma adequada, é necessário ter uma válvula que controle o volume de gases de escape realimentados para a admissão. Este artigo apresenta uma metodologia de projeto para o controle de posição da válvula de RGE, através da aplicação de C-GRASP (*Continuous Greedy Randomized Adaptive Search Procedure*) para otimização paramétrica de um controlador PI (proporcional-integral) com *anti-windup*. O controlador otimizado é apropriado para uso na referida aplicação, pois mostrou ser robusto e ter bom desempenho no seguimento de trajetória.

Palavras-chave – Otimização, Controle, Recirculação de Gases de Escape, C-GRASP, *Anti-windup*.

Abstract – In internal combustion engines, there is a nitrogen oxide emission reducing technique, called Exhaust Gas Recirculation (EGR). In order to function properly, the EGR system must have a valve that controls the volume of exhaust gases fed back to the admission. This paper presents a design methodology for the EGR valve position control through the use of C-GRASP (*Continuous Greedy Randomized Adaptive Search Procedure*) for parametric optimization of a PI (proportional-integral) controller with *anti-windup*. The optimized controller is suitable for controlling the EGR valve, as it showed to be robust and to have good performance in following reference.

Keywords – Optimization, Control, Exhaust Gas Recirculation, C-GRASP, *Anti-windup*.

1 Introdução

Em motores de combustão interna, há uma técnica para redução de emissão de óxido de nitrogênio, denominada *Recirculação de Gases de Escape* (RGE) ou *Exhaust Gas Recirculation* (EGR), em sua definição original. RGE é aplicada na maioria dos motores a diesel. RGE reenvia uma parte dos gases de escape de um motor de volta para seus cilindros. Nos motores diesel, os gases de escape substituem parte do excesso de oxigênio na mistura pré-combustão. Estes gases constituem material inerte na combustão, mas elevam a capacidade térmica contida no cilindro. Por isso a temperatura de combustão é reduzida, mantendo a mesma pressão no pistão. Desta maneira RGE diminui a quantidade de NO_x (óxido de nitrogênio) gerado na combustão, pois a formação deste óxido progride muito mais rápida em altas temperaturas.

Para que o sistema funcione de forma adequada, é necessário ter uma válvula que controle o volume de gases de escape realimentados para a admissão. Tal válvula sofre com a alta temperatura e precisa ter uma potência relativamente alta para aguentar a contrapressão de escape. Além disso, o desgaste e a carbonização de componentes mecânicos alteram a dinâmica do processo, fazendo com que o controle da válvula do sistema falhe. O problema é projetar um sistema adequado para o controle de posição da válvula da RGE, que opere de forma ótima, considerando todas as características do sistema.

Existem, em aplicações automotivas, três principais áreas onde os sistemas de controle têm papel vital: redução de emissões, segurança ativa e consumo de combustível [1]. Pesquisas recentes se concentram nestas três áreas, e no parágrafo seguinte serão revisados alguns trabalhos na área de redução de emissões.

O controle e o diagnóstico de catalisadores utilizando sensor universal de gás de escape oxigênio para motores a gasolina são descritos em [2] e [3]. Controle de motores com comando de abertura variável para redução da formação de NO_x é

tratado em [4]-[6]. Uma representação orientada a controle do sistema de pós-tratamento de escape LNT (*lean NO_x trap*) foi inicialmente desenvolvida por Wang, Raman e Grizzle [7]. Em [8] é demonstrado que os parâmetros desta representação podem ser identificados *on-line* utilizando um sensor convencional. Programação dinâmica é utilizada em [9] para explorar os benefícios do perfil da taxa ar-combustível. Em motores diesel, técnicas de controle são aplicadas para determinar as posições da válvula de RGE e do turbocompressor de geometria variável, a citar, controle PID (proporcional-integral-derivativo) [10]-[11], controle LQG (*Linear Quadratic Gaussian*) [12], controle PI (proporcional-integral) linear multivariável (MIMO PI) [13] e controle preditivo baseado em modelo [14].

Este artigo apresenta uma metodologia de projeto para o controle de posição da válvula de RGE, através da aplicação de C-GRASP (*Continuous Greedy Randomized Adaptive Search Procedure*) para otimização paramétrica de um controlador PI com *anti-windup*. O restante do artigo está organizado da seguinte forma. Uma descrição da válvula e do sistema de controle é apresentada na seção 2. Na seção 3 é feita uma exposição do algoritmo C-GRASP. Os resultados obtidos são mostrados na seção 4. Por último, na seção 5 a conclusão é apresentada.

2 Descrição da Válvula e do Sistema de Controle

A válvula do sistema de RGE é acionada por um trem de engrenagens e um motor elétrico. A Figura 1 ilustra os seus componentes de acionamento. O motor elétrico é fixado ao primeiro eixo rotacional e o último eixo rotacional tem um sensor de posição acoplado. O trem de engrenagens transforma o movimento rotacional em um movimento linear através da utilização de um elo similar a uma barra de conexão.

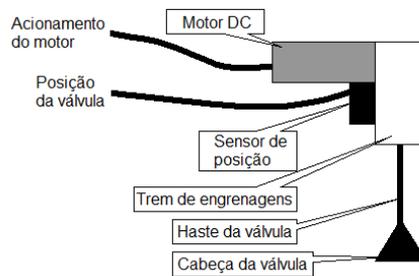


Figura 1 – Componentes da válvula de RGE.

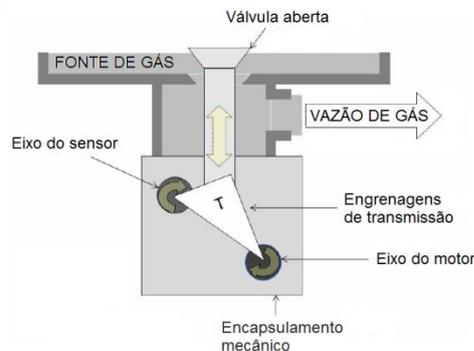


Figura 2 – Máxima vazão da válvula de RGE.

A máxima potência e a máxima velocidade de acionamento são definidas pela taxa de transmissão das engrenagens e pela construção do motor. A fonte de gás é vedada por um elemento cônico que move linearmente. A vazão do gás através da válvula é dada pelo ângulo do cone, diâmetro da cabeça, diâmetro da haste cilíndrica e posição do elemento em relação ao assento da válvula. A Figura 2 mostra a válvula com vazão total.

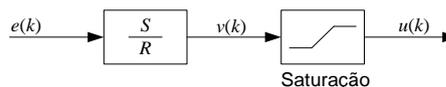


Figura 3 – Controlador sem *anti-windup*.

A configuração utilizada na montagem do sensor evita que a folga mecânica, necessária para um acoplamento correto de engrenagens, afete o sistema de controle. O sensor aplicado é um sensor de efeito Hall, baseado nas propriedades de campo magnético. O correto acoplamento é obtido quando o curso da válvula é totalmente proporcional à tensão de saída no sensor. A partir de testes realizados com quatro novas válvulas em [15], pôde-se constatar que tais válvulas apresentam grande linearidade e repetitividade. Um sistema de controle usado neste dispositivo deve lidar com uma pequena não-

proporcionalidade entre o sinal no sensor e a posição real, e deve prever uma pequena diferença entre as operações de avanço e retorno durante a vida útil da válvula, devido ao aumento da folga causado pelo atrito normal das engrenagens.

O motor utilizado é um motor elétrico DC comum com escovas de carbono, acionado por um módulo PWM (*Pulse Width Modulation*), a fim de controlar a potência e conseqüentemente a posição. Para evitar o *windup* em sistemas em que o processo e/ou o controlador é instável, alguma forma de mecanismo *anti-windup* deve ser implementada. Existem muitas formas de introduzir *anti-windup*, sendo que uma delas é detalhada a seguir. Outras maneiras de implementar técnicas *anti-windup* são vistas em [16]. O controlador da Figura 3 pode ser descrito como:

$$v(k) = e(k) \frac{S}{R} \quad (1)$$

$$u(k) = f(v(k)) \quad (2)$$

onde $e(k)$ é o sinal de erro, $v(k)$ é a saída ilimitada do controlador, $f(v(k))$ é uma função não-linear que representa a limitação do atuador, $u(k)$ é a saída limitada do atuador e S e R são os polinômios do numerador e denominador do controlador. Tal controlador não apresenta nenhum mecanismo de proteção contra *windup*.

Desconsiderando a não-linearidade do atuador, a equação do controlador pode ser reescrita, de forma que esta pareça um sistema dinâmico com duas entradas, o sinal de erro $e(k)$ e o sinal de controle $u(k)$. Adicionando $A_o u(k)$ aos dois lados da Equação 1 e admitindo $u(k) = v(k)$, temos:

$$A_o u(k) = S e(k) + (A_o - R) u(k) \quad (3)$$

onde A_o é o polinômio característico desejado do observador *anti-windup*. Um controlador com compensação *anti-windup* é então dado pelas equações a seguir:

$$A_o v(k) = S e(k) + (A_o - R) u(k) \quad (4)$$

$$u(k) = f(v(k)) \quad (5)$$

Este controlador é equivalente ao da Equação 1 quando não está saturado. Quando a variável de controle satura e a realimentação é quebrada, o controlador responde com a dinâmica do polinômio observador. O diagrama de blocos da modificação não-linear da Equação 1 que evita *windup* é mostrada na Figura 4. Um caso particular simples é quando $A_o^* = 1$, que corresponde a um observador *deadbeat*. O controlador pode então ser escrito como:

$$u(k) = f(S^* e(k) + (1 - R^*) u(k)) \quad (6)$$

O controlador PI com *anti-windup* pode ser visto como o controlador da Figura 4, com os polinômios A_o , S e R possuindo apenas uma raiz, e é esta abordagem que foi aplicada no estudo detalhado neste artigo.

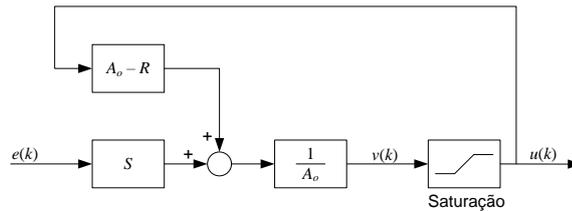


Figura 4 – Controlador com *anti-windup*.

3 Continuous-GRASP

Continuous-GRASP (C-GRASP) foi inicialmente proposto por Hirsch *et al.* [17], como um novo método de otimização global. C-GRASP estende o *Greedy Randomized Adaptive Search Procedure* (GRASP) de Feo e Resende [18] do domínio de otimização discreto para o de otimização global contínua. C-GRASP é um método estocástico de busca local, simples de ser implementado, que pode ser aplicado a uma vasta gama de problemas e que não faz uso de informação derivada. Desta forma este método é adequado para a resolução de problemas de otimização global. Em [17] foi registrado o desempenho do C-GRASP na resolução de problemas teóricos padrões e problemas reais de otimização global.

A metaheurística GRASP é descrita por Feo e Resende [18] como um procedimento de busca local multipartida, e já foi aplicada em inúmeros problemas de otimização combinatória discreta. Cada iteração em GRASP consiste de duas fases, uma fase de construção e uma de busca local. A construção combina voracidade e aleatoriedade para produzir uma série de soluções de boa qualidade a partir de onde se inicia a busca local. A solução final mantida é a melhor dentre todas as iterações. C-GRASP assemelha-se a GRASP, mas difere no sentido de que uma iteração não consiste de apenas uma única fase de construção seguida de uma melhoria local, mas sim de uma série de ciclos de construção e melhoria local.

C-GRASP considera um domínio S em um espaço n -vetorial, onde uma solução x qualquer é composta de n variáveis x_1, \dots, x_n , sendo que $l \leq x \leq u$, onde l e u são os vetores de limite inferior e superior, respectivamente, e $l, x, u \in \mathcal{R}^n$. O problema de minimização consiste em encontrar o mínimo global x^* para a função objetivo $f(x)$, $f: \mathcal{R}^n \rightarrow \mathcal{R}$.

```

procedimento C-GRASP( $n, l, u, f(\cdot), MaxIters, MaxNumIterNoImprov, NumTimesToRun, MaxDirToTry, \alpha$ )
1    $f^* \leftarrow \infty$ ;
2   para  $j = 1, \dots, NumTimesToRun$  fazer
3        $x \leftarrow UnifRand(l, u); h \leftarrow 1; NumIterNoImprov \leftarrow 0$ ;
4       para  $Iter = 1, \dots, MaxIters$  fazer
5            $x \leftarrow ConstructGreedyRandomized(x, f(\cdot), n, h, l, u, \alpha)$ ;
6            $x \leftarrow LocalImprovement(x, f(\cdot), n, h, l, u, MaxDirToTry)$ ;
7           se  $f(x) < f^*$  então
8                $x^* \leftarrow x; f^* \leftarrow f(x); NumIterNoImprov \leftarrow 0$ ;
9           senão
10               $NumIterNoImprov \leftarrow NumIterNoImprov + 1$ ;
11          fim se
12          se  $NumIterNoImprov \geq MaxNumIterNoImprov$  então
13               $h \leftarrow h/2; NumIterNoImprov \leftarrow 0$ ; /*faz a malha mais densa*/
14          fim se
15      fim para
16  fim para
17  retornar( $x^*$ );
fim C-GRASP;
    
```

Figura 5 – Pseudocódigo de C-GRASP.

Na Figura 5 observa-se o pseudocódigo de C-GRASP. $MaxIters$, $MaxNumIterNoImprov$, $NumTimesToRun$, $MaxDirToTry$ e α são parâmetros configuráveis e indicam, respectivamente, o número máximo de ciclos de construção e melhoria local por iteração principal, número máximo de chamadas do procedimento de melhoria local sem que haja melhoria, número máximo de iterações multipartida (ou iterações principais), número máximo de direções avaliadas na fase de melhoria local e condição para formar a lista de candidatos restrita na fase de construção. f^* representa $f(x^*)$, h é o valor de discretização do espaço de busca, $UnifRand(l, u)$ define um ponto aleatório dentro do domínio do problema, e $ConstructGreedyRandomized$ e $LocalImprovement$ são as fases de construção e melhoria local, que têm seus respectivos pseudocódigos ilustrados na Figura 6 e na Figura 7.

Na Figura 6, S representa o conjunto de coordenadas não-fixas de x , que inicialmente contém todas as coordenadas e ao final da fase de construção não contém mais nenhuma. $LineSearch(x, h, i, n, f(\cdot), l, u)$ executa uma busca linear discreta, ao longo da coordenada i de x , pelo valor z_i que minimiza a função objetivo, respeitando a discretização de acordo com h . g_i é o valor da função objetivo para a solução com z_i . min e max armazenam os valores máximo e mínimo de g_i dentre todas as coordenadas não-fixas de x . Nas linhas 12 a 17 é formada uma lista de candidatos restrita RCL , que contém as coordenadas não-fixas que satisfazem a condição da linha 14, sendo que $\alpha \in [0,1]$. $RandomlySelectElement(RCL)$ seleciona aleatoriamente um elemento de RCL , que será a coordenada a ser fixada, na linha 19. Tal procedimento garante a aleatoriedade da fase de construção. Assim que todas as direções forem fixadas, a solução x é retornada pela fase de construção.

```

procedimento ConstructGreedyRandomized( $x, f(\cdot), n, h, l, u, \alpha$ )
1    $S \leftarrow \{1, 2, \dots, n\}$ ;
2   enquanto  $S \neq \emptyset$  fazer
3        $min \leftarrow +\infty; max \leftarrow -\infty$ ;
4       para  $i = 1, \dots, n$  fazer
5           se  $i \in S$  então
6                $z_i \leftarrow LineSearch(x, h, i, n, f(\cdot), l, u)$ ;
7                $g_i \leftarrow f(z_i)$ ;
8               se  $min > g_i$  então  $min \leftarrow g_i$ ;
9               se  $max < g_i$  então  $max \leftarrow g_i$ ;
10          fim se
11      fim para
12       $RCL \leftarrow \emptyset$ ;
13      para  $i = 1, \dots, n$  fazer
14          se  $i \in S$  e  $g_i \leq (1 - \alpha) \cdot min + \alpha \cdot max$  então
15               $RCL \leftarrow RCL \cup \{i\}$ ;
16      fim se
17      para
18           $j \leftarrow RandomlySelectElement(RCL)$ ;
19           $x_j \leftarrow z_j; S \leftarrow S \setminus \{j\}$ ;
20  fim enquanto
21  retornar( $x$ );
fim ConstructGreedyRandomized;
    
```

Figura 6 – Pseudocódigo da fase de construção de C-GRASP.

```

procedimento LocalImprovement( $x, f(\cdot), n, h, l, u, MaxDirToTry$ )
1    $Improved \leftarrow \text{true}; D \leftarrow 0;$ 
2    $x^* \leftarrow x; f^* \leftarrow f(x);$ 
3    $NumDirToTry \leftarrow \min(3^n - 1, MaxDirToTry);$ 
4   enquanto  $Improved$  fazer
5        $Improved \leftarrow \text{false};$ 
6       enquanto  $|D| \leq NumDirToTry$  e não  $Improved$  fazer
7           Gerar  $r \leftarrow \lceil \text{UnifRand}(1, 3^n - 1) \rceil \notin D;$ 
8            $D \leftarrow D \cup \{r\};$ 
9            $d \leftarrow \text{Ternary}'(r); x \leftarrow x^* + h.d;$ 
10          se  $l \leq x \leq u$  então
11              se  $f(x) < f^*$  então
12                   $x^* \leftarrow x; f^* \leftarrow f(x);$ 
13                   $D \leftarrow 0;$ 
14                   $Improved \leftarrow \text{true};$ 
15              fim se
16          fim se
17      fim enquanto
18  retornar  $(x^*);$ 
fim LocalImprovement;
    
```

Figura 7 – Pseudocódigo da fase de melhoria local de C-GRASP.

No pseudocódigo da fase de melhoria local (Figura 7), as variáveis *Improved*, *D*, *r* e *d* representam respectivamente a ocorrência de melhoria na direção avaliada, o conjunto de direções já avaliadas, uma nova direção aleatória e o vetor direcional correspondente a esta nova direção. Existem $3^n - 1$ direções possíveis, e $\min(3^n - 1, MaxDirToTry)$ define o número de direções a serem avaliadas, que corresponde ao menor valor entre o total de direções possíveis e o parâmetro *MaxDirToTry*. $\lceil \text{UnifRand}(1, 3^n - 1) \rceil \notin D$ define um número inteiro uniformemente aleatório entre 1 e $3^n - 1$ não pertencente a *D*. $\text{Ternary}'(r)$ converte *r* da base decimal para a base ternária, em seguida substitui todo algarismo '2' por '-1', gerando um vetor de tamanho *n* que contém elementos '1', '0' ou '-1', o qual corresponde ao vetor direcional *d*.

A fase de melhoria local é uma aproximação do papel do gradiente da função objetivo. A partir de uma solução inicial, o algoritmo de melhoria local gera uma série de direções e determina em qual direção o valor da função objetivo melhora, se é que existe tal direção. A fase de melhoria local segue até que se encontre uma solução que seja melhor do que suas correspondentes em todas as direções avaliadas.

4 Resultados

Como anteriormente citado, a otimização dos parâmetros do controlador PI com *anti-windup* se deu através do algoritmo C-GRASP. As soluções foram representadas pelos ganhos proporcional e integral (*Kp* e *Ki*, respectivamente) e pela raiz de *A_o*. O domínio para *Kp* foi de 0 a 20, para *Ki* de 350 a 530 e para a raiz de *A_o* de -1 a 1. Os parâmetros *MaxIters*, *MaxNumIterNoImprov*, *NumTimesToRun*, *MaxDirToTry* e α utilizados foram, respectivamente, 100, 20, 1, 30 e 0,4.

Para avaliação da função objetivo foi simulada uma rotina de controle e como resultado foi utilizado o desempenho do controlador em questão. Para tanto foram aplicados quatro degraus como referência, de 50, 90, 60 e 8%, em um horizonte de 2000 amostras. A função objetivo foi então definida como:

$$f(x) = \sum_{k=0}^{2000} t.e(k)^2 + \beta \cdot \sum_{k=1}^{2000} (u(k) - u(k-1))^2 \quad (7)$$

onde *e(k)* é o erro no instante da amostra *k*, *t* é o tempo em segundos desde a última mudança de referência e *u(k)* é o sinal de controle aplicado no instante da amostra *k*. β é um parâmetro de projeto que representa o compromisso entre rapidez de resposta e oscilação do sinal de controle. Desta forma quanto menor for β , mais rápido e oscilatório será o sistema com o controlador ótimo encontrado, e quanto maior, mais lento será o sistema e menos oscilatório será o sinal de controle. Para este projeto, o parâmetro β escolhido foi de 0,07. Na simulação foi utilizado o modelo discreto de uma válvula nova apresentado em [19]. Com todos estes quesitos definidos, o melhor controlador encontrado foi:

$$A_o = 1 - 0,7500 z^{-1} \quad (8)$$

$$S = 12,0000 - 11,5771 z^{-1} \quad (9)$$

$$R = 1 - z^{-1} \quad (10)$$

Isto corresponde a *Kp* = 12 e *Ki* = 423,6. A Figura 8 ilustra o desempenho deste controlador durante a simulação, o qual se demonstrou rápido e com sobre-sinal praticamente nulo, e obteve uma avaliação da função objetivo de 7845,5.

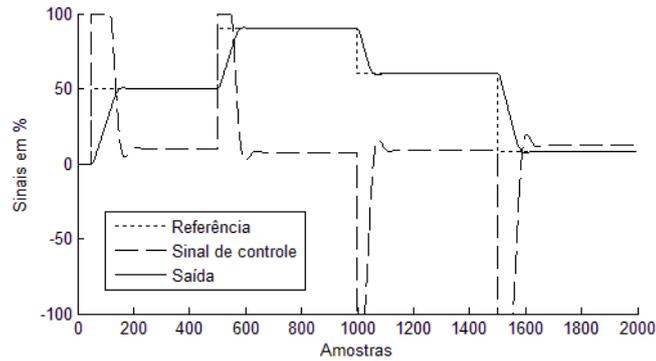


Figura 8 – Desempenho do controlador otimizado.

Para validação do controlador PI com *anti-windup* otimizado (controlador 2) foram simulados alguns testes comparativos utilizando também um controlador de terceira ordem com *anti-windup* (controlador 1). Estes testes tiveram o objetivo de avaliar os controladores em relação a seguimento de trajetória, rejeição a perturbação na entrada do processo, atenuação de ruído no sensor, e robustez.

O controlador de terceira ordem com *anti-windup* (controlador 1) possui a mesma estrutura que o controlador PI com *anti-windup* (controlador 2), sendo a diferença entre eles a ordem dos polinômios A_o , S e R . O controlador 1 foi sintonizado da mesma maneira que o controlador 2, ou seja, através de otimização com o algoritmo C-GRASP, e é descrito pelas equações a seguir:

$$A_o = 1 + 1,6250z^{-1} + 0,6563z^{-2} \quad (11)$$

$$S = 8,8750 + 4,4375z^{-1} - 6,5176z^{-2} - 3,6401z^{-3} \quad (12)$$

$$R = 1 - 0,2500z^{-1} - 0,7656z^{-2} + 0,1914z^{-3} \quad (13)$$

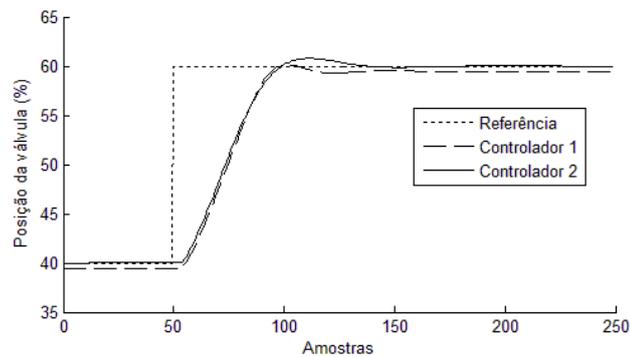


Figura 9 – Seguimento de trajetória (referência em degrau).

Inicialmente foi feita uma comparação entre os dois controladores em relação ao seguimento de trajetória. A Figura 9 mostra o resultado da posição da válvula utilizando os dois controladores na simulação da aplicação de um degrau. O controlador 2 apresenta um sobre-sinal um pouco mais elevado (em torno de 4%) que o controlador 1, mas consegue zerar o erro em regime permanente, enquanto o controlador 1 mantém um pequeno erro em regime.

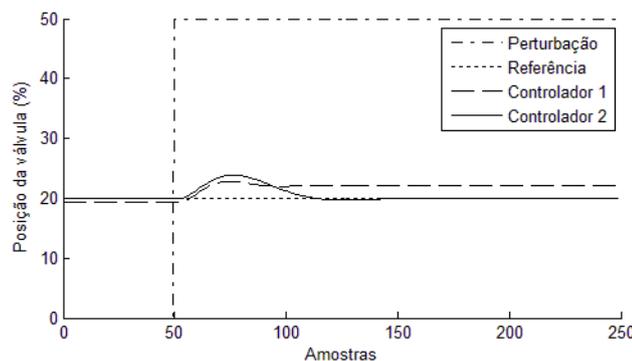


Figura 10 – Rejeição de perturbação em degrau.

Em seguida foi traçado na Figura 10 um comparativo do desempenho dos controladores na rejeição de uma perturbação aplicada na entrada do processo. Na operação da válvula de RGE é importante que o controlador consiga rejeitar apropriadamente as perturbações, pois a contrapressão de escape à qual a válvula está sujeita é vista como uma forte perturbação no processo. O controlador 2 consegue anular o distúrbio em regime permanente, mas o controlador 1 consegue apenas minimizar o efeito deste, mantendo um erro permanente.

A Figura 11 ilustra o ruído no sensor e o efeito deste na saída do processo com os controladores 1 e 2, tendo a referência em 50%. O ruído está sendo ampliado na saída, em 3,46 vezes com o controlador 1 e 4,34 vezes com o 2. Isto é indesejado, por isso a aplicação prática dos controladores aqui analisados deve ser associada ao uso de um filtro no sensor para garantir que este efeito seja minimizado.

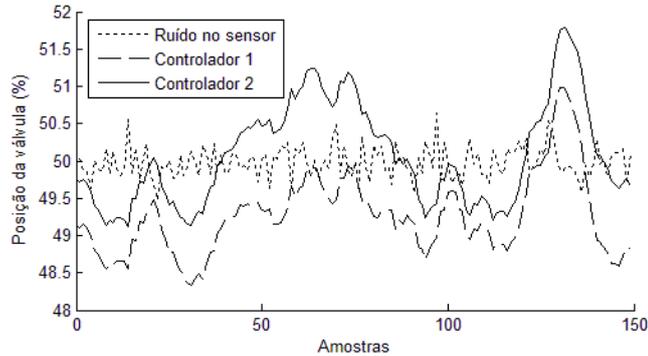


Figura 11 – Efeito do ruído no sensor sobre a saída do processo.

Por fim, a Figura 12 mostra o desempenho do controlador 2 na mesma rotina de controle utilizada na otimização, mas com o uso adicional dos modelos de duas válvulas usadas. Desta forma é avaliada a robustez do controlador em relação ao desgaste da válvula durante sua vida útil. O controlador se demonstrou robusto, sendo que com as válvulas mais desgastadas o sistema ficou um pouco mais lento.

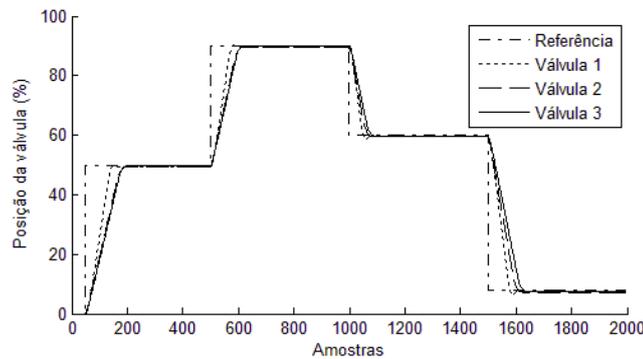


Figura 12 – Análise de robustez.

De acordo com [19], os modelos das 3 válvulas foram obtidos através da aplicação de uma ferramenta de identificação proposta por Madár, Abonyi e Szeifert [20], a qual utiliza Programação Genética e o algoritmo dos Mínimos Quadrados Ortogonais. Tais modelos são descritos nas equações a seguir, respectivamente para as válvulas 1, 2 e 3, onde $u(k)$ é a entrada e $y(k)$ é a saída do modelo no instante k :

$$y(k) = 0,558550y(k-1) + 0,391325y(k-2) + 0,269794y(k-3) - 0,219340y(k-6) - 0,005472u(k-5) - 0,070273 \quad (14)$$

$$y(k) = 0,362635y(k-1) + 0,336296y(k-2) + 0,302216y(k-3) - 0,009115u(k-4) - 0,159495 \quad (15)$$

$$y(k) = 0,394192y(k-1) + 0,335209y(k-2) + 0,271384y(k-3) - 0,008082u(k-6) - 0,086385. \quad (16)$$

5 Conclusão

Recentemente Hirsch *et al.* [17] propuseram C-GRASP, para problemas de minimização, que é um algoritmo de busca estocástica multipartida, onde um procedimento ávido aleatório gera soluções de entrada para um método de melhoria local.

Cada iteração deste algoritmo consiste em uma série de ciclos de construção e melhoria local, e a malha discreta do espaço de busca se torna mais densa à medida que C-GRASP evolui.

Este artigo mostrou que o algoritmo C-GRASP é eficaz na otimização de parâmetros de projeto de controladores. A abordagem utilizada para avaliação da função objetivo foi validada, pois representou de maneira adequada o desempenho do controlador avaliado, e com apenas um parâmetro de projeto (β) pôde definir a dinâmica desejada. O controlador otimizado é apropriado para uso no controle de posição da válvula de RGE, pois mostrou ser robusto e ter bom desempenho no seguimento de trajetória, lembrando que deve ser aplicado um filtro passa-baixa no sinal do sensor, para filtrar o ruído e minimizar seu efeito sobre a saída. Além disso, o controlador PI com *anti-windup* rejeitou com eficiência as perturbações na entrada do processo. No futuro poderá ser realizada uma análise detalhada sobre o desempenho e a convergência de C-GRASP, comparando-o com outros métodos de otimização conhecidos na resolução do problema apresentado neste artigo.

6 Referências

- [1] L. Guzzella, C. Onder, Past, present and future of automotive control. In: B. A. Francis, M. C. Smith, J. C. Willems, Control of uncertain systems: modelling, approximation, and design, **Springer-Verlag**, 329 (2006), 163 – 182.
- [2] M. Ammann, H. P. Geering, C. H. Onder, C. A. Roduner, E. Shafai, Adaptive control of a three-way catalytic converter, **Proceedings of the American Control Conference**, Chicago, USA, 3 (2000), 1561 – 1566.
- [3] A. T. Vemuri, Diagnosis of sensor bias faults, **Proceedings of the American Control Conference**, San Diego, USA, 1 (1999), 460 – 464.
- [4] T. W. Asmus, Perspectives on applications of variable valve timing, **SAE International Congress & Exposition**, Detroit, USA, SAE Paper 910455 (1991).
- [5] Y. Moriya, A. Watanabe, H. Uda, H. Kawamura, M. Yoshiuka, A newly developed intelligent variable valve timing system – continuously controlled CAM phasing as applied to new 3 liter inline 6 engine. **SAE International Congress & Exposition**, Detroit, USA, SAE paper 960579 (1996).
- [6] R. A. Stein, K. M. Galietti, T. G. Leone, Dual equal VCT – a variable camshaft timing strategy for improved fuel economy and emissions, **SAE International**, SAE Paper 950975 (1995).
- [7] Y. Wang, S. Raman, J. W. Grizzle, Dynamic modeling of a lean NOx trap for lean burn engine control, **Proceedings of the American Control Conference**, San Diego, USA, (1999).
- [8] L. Y. Wang, I. V. Kolmanovsky, J. Sun, On-line identification and adaptation of LNT models for improved emission control in lean burn automotive engines, **Proceedings of the American Control Conference**, Chicago, USA, 2 (2000), 1006 – 1010.
- [9] Y. W. Kim, J. Sun, L. Y. Wang, Optimization of lean NOx trap control for fuel economy and exhaust emissions, **Proceedings of the American Control Conference**, Boston, USA, (2004).
- [10] H. J. Dekker, W. L. Sturm, Simulation and control of a HD diesel engine equipped with new EGR technology. **SAE International Congress & Exposition**, Detroit, USA, SAE Paper 960871 (1996).
- [11] R. Buratti, A. Carlo, E. Lanfranco, A. Pisoni, DI diesel engine with variable geometry turbocharger (VGT): a model-based boost pressure control strategy. **Meccanica**, 32 (1997), 409 – 421.
- [12] A. Amstutz, L. R. Del Re, EGO sensor based robust output control of EGR in diesel engines, **IEEE Transactions on Control Systems Technology**, 3 (1995), 39 – 48.
- [13] M. Van Nieuwstadt, I. Kolmanovsky, P. Moraal, A. Stefanopoulou, M. Jankovic, EGR-VGT control schemes: experimental comparison for a high-speed diesel engine, **IEEE Control Systems**, 20 (2000), 63 – 79.
- [14] J. Ruckert, B. Kinoo, M. Kruger, A. Schlosser, H. Rake, S. Pischinger, Simultaneous control of boost pressure and exhaust gas recirculation in a passenger car diesel engine, **MTZ Worldwide**, 62 (2001), 956 – 965.
- [15] L. S. Coelho, C. E. Klein, Identification of a poppet valve using genetic programming method based on adaptive probabilities with chaotic tuning and orthogonal least squares, **Proceedings of the 20th International Congress of Mechanical Engineering (COBEM 2009)**, Gramado, RS, (2009).
- [16] A. Hadade Neto, Técnicas anti-windup em estruturas de controle PID, RST e GPC. **Dissertação (Mestrado em Engenharia Elétrica) – Universidade Federal de Santa Catarina**, Florianópolis, SC, (2005).
- [17] M. J. Hirsch, C. N. Meneses, P. M. Pardalos, M. G. C. Resende, Global optimization by continuous grasp, **Optimization Letters**, 1 (2006), 201 – 212.
- [18] T. A. Feo, M. G. C. Resende, Greedy randomized adaptive search procedures, **Journal of Global Optimization**, 6 (1995), 109 – 133.
- [19] J. X. Vianna Neto, Otimização e análise de diferentes sistemas de controle aplicados à válvula de recirculação de gases de escape presente em motores diesel, **Monografia (Graduação em Engenharia de Controle e Automação) – Pontifícia Universidade Católica do Paraná**, Curitiba, PR, (2010).
- [20] J. Madár, J. Abonyi, F. Szeifert, Genetic programming for the identification of nonlinear input-output models, **Industrial and Engineering Chemistry Research**, 44 (2005), 3178 – 3186.