

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA
CENTRO TECNOLÓGICO
UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

LEANDRO MUNIZ DE LIMA

**FILTRO DE PARTÍCULAS
HIBRIDIZADO COM MÉTODOS DA
COMPUTAÇÃO NATURAL PARA
DETECÇÃO E RASTREAMENTO**

VITÓRIA
2011

LEANDRO MUNIZ DE LIMA

**FILTRO DE PARTÍCULAS HIBRIDIZADO COM MÉTODOS DA
COMPUTAÇÃO NATURAL PARA DETECÇÃO E RASTREAMENTO**

Dissertação apresentada ao Programa de Pós-Graduação em Informática do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Informática, na área de concentração Inteligência Computacional.

Orientador:
Renato A. Krohling

VITÓRIA
2011

Dados Internacionais de Catalogação-na-publicação (CIP)
(Biblioteca Central da Universidade Federal do Espírito Santo, ES, Brasil)

L732f Lima, Leandro Muniz de, 1985-
Filtro de partículas hibridizado com métodos da computação natural para detecção e rastreamento / Leandro Muniz de Lima. – 2011.
57 f. : il.

Orientador: Renato Antônio Krohling.
Dissertação (Mestrado em Informática) – Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Rastreamento automático. 2. Processamento de imagens. 3. Inteligência artificial. 4. Processo estocástico. 5. Otimização matemática. I. Krohling, Renato Antonio. II. Universidade Federal do Espírito Santo. Centro Tecnológico. III. Título.

CDU: 004

Filtro de partículas hibridizado com métodos da computação natural para detecção e rastreamento

Leandro Muniz de Lima

Dissertação submetida ao Programa de Pós-Graduação em Informática da Universidade Federal do Espírito Santo como requisito parcial para a obtenção do grau de Mestre em Informática.

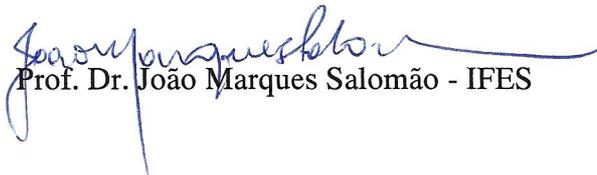
Aprovada em 25/08/2011 por:



Prof. Dr. Renato Antônio Krohling - DI/UFES



Prof^a. Dr^a. Maria Claudia Silva Boeres - DI/UFES



Prof. Dr. João Marques Salomão - IFES

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
Vitória-ES, agosto de 2011

Resumo

Detecção e rastreamento de objetos em seqüências de imagens aparece atualmente em várias situações do nosso cotidiano e se destaca pela sua importância em várias áreas como, por exemplo, na área de segurança (monitoramento de objetos ou indivíduos), dentre outros. Um dos métodos comumente utilizado é o Filtro de Partículas (FP), o principal problema do FP é a degeneração, que pode implicar em um rastreamento pior. Nesta dissertação, serão apresentados dois métodos híbridos baseado no Filtro de Partículas. A hibridização ocorre através da combinação do Filtro de Partículas com um método da computação natural: i) Otimização através de Enxame de Partículas; e ii) Evolução Diferencial. Os métodos propostos foram aplicados para dois estudos de caso: i) para rastreamento de trajetória de um sistema não linear caminhão-reboque, e ii) para detectar e rastrear a face de uma pessoa em uma seqüência de imagens. Os resultados obtidos em termos de qualidade de rastreamento indicam um melhor desempenho dos algoritmos hibridizados quando comparados com o Filtro de Partículas padrão.

Abstract

Detecting and tracking objects in image sequences currently appears in various situations of everyday life and stands out for its importance in many areas, for example, in security (monitoring objects or persons), among others. A commonly used method is the Particle Filter, the main issue of Particle Filter is degeneration, which may imply a worse tracking. In this work, it is presented two hybrid method of Particle Filter. This hybridization occurs combining a Particle Filter and a natural computing: i) Particle Swarm Optimization; and ii) Differential Evolution. That way, aiming to minimize the degeneration problem in Particle Filter, in order to improve the performance of the tracking method. The proposed methods were applied to two case studies: i) for tracking the trajectory of the truck-trailer system, and ii) to detect and track a person's face in an image sequence. The results in terms of tracking quality indicate a better performance of hybridized algorithms when compared with the standard Particle Filter.

Dedicatória

Dedico este trabalho à minha família, pelo amor, segurança e educação.

Declaração de Autoria

Eu declaro que essa dissertação foi escrita por mim. Qualquer ajuda recebida durante o trabalho de pesquisa e na escrita tem sido reconhecida. Além disso, eu certifico que todas as fontes de informação e literatura usadas estão indicadas na dissertação.

Leandro Muniz de Lima

Agradecimentos

Aos professores, por se dedicarem a me ensinar tudo o que sabiam.

Ao professor Renato Krohling, por seu apoio e disponibilidade durante todo o projeto, sempre que foi necessário.

Aos demais membros da banca, professora Maria Claudia Silva Boeres e professor João Marques Salomão, por assumirem a avaliação da dissertação.

À Fundação de Apoio à Ciência e Tecnologia do Espírito Santo (FAPES) pelo fomento ao desenvolvimento científico do Espírito Santo, em especial, pela bolsa de mestrado número 056/2009 a mim concedida.

À todos que ajudaram a desenvolver o conhecimento necessário para que fosse possível a criação deste trabalho.

À minha esposa Manuela Garcia, pelo companheirismo, incentivo e paciência. Sempre me apoiou e me motivou para me dedicar a este trabalho.

Aos meus amigos e familiares, pelo amparo.

Sumário

Lista de Figuras

Lista de Tabelas

1	Introdução	13
1.1	Definição do problema	13
1.2	Metodologia	14
1.3	Objetivo	16
1.4	Estrutura	16
2	Filtro de Partículas	17
2.1	Introdução	17
2.2	Método Sequencial de Monte Carlo	19
2.3	O problema da degeneração	24
3	Algoritmos da computação natural	26
3.1	Otimização através de Enxame de Partículas	26
3.2	Evolução Diferencial	28
4	Filtro de Partículas Híbrido	32
4.1	Introdução	32
4.2	Híbrido com Otimização através de Enxame de Partículas	34
4.3	Híbrido com Evolução Diferencial	34
5	Resultados	36

5.1	Estudo de caso 1: Rastreamento de trajetória do sistema caminhão-reboque . . .	36
5.1.1	Definição do problema	36
5.1.2	Parâmetros do algoritmo	39
5.1.3	Resultados experimentais	39
5.1.4	Discussão	41
5.2	Estudo de caso 2: Detecção e rastreamento em sequência de imagens	41
5.2.1	Definição do problema	42
5.2.2	Parâmetros do algoritmo	44
5.2.3	Resultados experimentais	47
5.2.4	Discussão	51
6	Conclusões	52
	Referências Bibliográficas	54

Lista de Figuras

2.1	Dependência Markoviana entre os estados e as observações	18
2.2	Estimação do sistema com 500 amostras	19
2.3	Estrutura do funcionamento do Filtro de Partículas	21
2.4	Ilustração de duas das etapas do algoritmo FP, mostrando partículas com diferentes pesos	22
2.5	Algoritmo Filtro de Partículas	23
3.1	Estrutura do funcionamento da Otimização através de Enxame de Partículas	27
3.2	Vetores mostrando a composição da nova velocidade da partícula	27
3.3	Algoritmo Otimização através de Enxame de Partículas	28
3.4	Estrutura do funcionamento da Evolução Diferencial	28
3.5	Operadores da Evolução Diferencial	29
3.6	Algoritmo Evolução Diferencial	31
4.1	Algoritmo Filtro de Partículas hibridizado	33
4.2	Estrutura do funcionamento do Filtro de Partículas combinado com Otimização através de Enxame de Partículas	34
4.3	Estrutura do funcionamento do Filtro de Partículas combinado com Evolução Diferencial	35
5.1	Sistema de um caminhão-reboque com três reboques	37
5.2	Resultados da estimação do estado de x_8 . a 100 partículas. b 500 partículas. c 1000 partículas	40
5.3	Rosto da mulher usado como modelo na detecção	45
5.4	Uma amostra da sequência de imagens original	46
5.5	Uma amostra da sequência de imagens com ruído	46

5.6	Rastreamento em imagens, sem ruído (linha superior) e com ruído (linha inferior), utilizando FP (vermelho), FP+OEP (verde) e FP+ED (azul) com detecção através da Interseção de Histograma	47
5.7	Rastreamento em imagens, sem ruído (linha superior) e com ruído (linha inferior), utilizando FP (vermelho), FP+OEP (verde) e FP+ED (azul) com detecção através do Coeficiente de Correlação	48
5.8	Rastreamento em imagens, sem ruído (linha superior) e com ruído (linha inferior), utilizando FP (vermelho), FP+OEP (verde) e FP+ED (azul) com detecção através da Correlação Cruzada Normalizada Híbrida	48
5.9	Gráfico da medida da degeneração ao longo do tempo do teste com melhor desempenho do Filtro de Partículas com Evolução Diferencial usando Interseção de Histograma	50

Lista de Tabelas

5.1	Parâmetros do sistema de caminhão-reboque	38
5.2	MQE médio para o FP, FP+OEP e FP+ED com 100, 500 e 1000 partículas . . .	41
5.3	SEQ médio e mínimo para o FP, FP+OEP e FP+ED com 150 partículas	49
5.4	Degeneração média e mínima do teste com melhor desempenho para o FP, FP+OEP e FP+ED com 150 partículas	49
5.5	SEQ médio e mínimo para o FP, FP+OEP e FP+ED com detecção através da Interseção de Histograma	50

1 *Introdução*

A detecção e rastreamento de objetos em sequências de imagens aparece atualmente em várias situações do nosso cotidiano e se destaca pela sua importância em várias áreas como por exemplo na área de segurança (detecção e rastreamento de objetos ou pessoas suspeitos em portos ou aeroportos), monitoramento de pessoas (crianças perdidas em um shopping center ou estádio), dentre outros. Portanto, métodos para automatizar o processo de detecção e rastreamento são ferramentas importantes para auxiliar seres humanos que realizem este tipo de tarefa, na maioria das vezes cansativa e repetitiva ou mesmo assumir estas tarefas completamente. Métodos convencionais para solucionar este tipo de problema apresentam dificuldades devido ao fato que no mundo real as imagens a serem detectadas e rastreadas podem estar, por exemplo, sujeitas a variado grau de iluminação ou se apresentarem parcialmente ocultas. Devido a isso, se faz necessário o desenvolvimento de novos métodos que levem em conta estes aspectos e que ponderem os critérios de otimalidade e robustez das soluções obtidas. Dentre as abordagens existentes podemos citar o Filtro de Partículas (FP) (ARULAMPALAM et al., 2002) como um algoritmo para solucionar estes tipos de problemas. Porém, mesmo assim essas abordagens podem apresentar dificuldades, como a degeneração presente no FP. Numa tentativa de enfrentar os problemas presente na detecção e rastreamento através do Filtro de Partículas, é proposta a sua hibridização com métodos da computação natural (CASTRO, 2007; KARI; ROZENBERG, 2008).

1.1 **Definição do problema**

No rastreamento, o objeto a ser seguido deve sempre ter sua posição estimada a mais próxima possível da posição real. Algoritmos como o Filtro de Partículas, se baseiam nas observações anteriores para estimar a posição no instante seguinte, tentando, desta forma, restringir a região a ser analisada na detecção.

O Filtro de Partículas é constituído por uma aproximação da distribuição de probabilidade *a posteriori* através de pontos de amostragem, denominada partículas, que são atualizadas con-

forme novos dados que surgem. Partículas são amostras aleatórias, que acompanham a trajetória do estado. O Filtro de Partículas utiliza o método de integração sequencial de Monte Carlo para aproximar a distribuição de probabilidade *a posteriori* criando amostras a partir de uma distribuição de probabilidade conhecida, a fim de calcular os pesos, que representam a importância de cada amostra. Um inconveniente comum ao gerar amostras a partir de uma distribuição é a degeneração, o que significa que, após um certo número de iterações a variância dos pesos aumenta muito. Uma técnica que pode ser usada para tentar evitar a degeneração é o algoritmo Amostrando e Reamostrando pela Importância (identificado pela sigla SIR, do inglês: *Sampling Importance Resampling*) (GORDON; SALMOND; SMITH, 1993; DOUCET; FREITAS; GORDON, 2001; ARULAMPALAM et al., 2002; CRISAN; DOUCET, 2002). Apesar de ajudar a evitar, o SIR não resolve o problema de degeneração.

1.2 Metodologia

Problemas de Engenharia e Computação, em geral, podem ser complexos e difíceis de serem solucionados usando apenas um método ou técnica devido a dinâmica não linear, por serem afetados por ruídos, etc. Abordagens recentes objetivando melhoras no desempenho do Filtro de Partículas usando Otimização através de Enxame de Partículas (OEP) (KENNEDY; EBERHART, 1995), um método de computação natural, foi desenvolvida por Krohling (2004a). Neste caso o problema de degeneração do FP também foi tratado como um problema de otimização.

Neste trabalho, a meta foi o desenvolvimento de um novo algoritmo inspirado numa abordagem híbrida baseada no Filtro de Partículas com a Evolução Diferencial (ED) (KROHLING, 2010; LIMA; KROHLING, 2011), para tratar a degeneração. Além disso, para propósito de comparação será também implementado o método híbrido combinando o algoritmo Filtro de Partículas com otimização através de Enxame de Partículas (KROHLING, 2004a).

O termo computação natural designa a computação inspirada na natureza. Nela, o que ocorre na natureza serve de inspiração para o desenvolvimento de técnicas para solucionar problemas. O objetivo deste ramo é desenvolver ferramentas computacionais para resolver problemas complexos.

Um campo da computação natural é a computação evolutiva (EIBEN; SMITH, 2008), que se inspira na teoria da evolução. Nela uma população de indivíduos capazes de se reproduzir e sujeitos a variação genética seguido por uma seleção dos resultados em uma nova população de indivíduos que de forma incremental se adapta ao ambiente. Um dos algoritmos mais conheci-

dos da computação evolutiva é o Algoritmo Genético (WHITLEY, 2001), mas a Evolução Diferencial (STORN; PRICE, 1997; DAS; KONAR; CHAKRABORTY, 2005) vem se destacando ultimamente. Sua maior diferença para o Algoritmo Genético está na forma que os operadores evolutivos funcionam, com o algoritmo ED utilizando a diferença entre dois indivíduos de sua população para a mutação. Versões aprimoradas do algoritmo Evolução Diferencial tem sido apresentadas em Brest et al. (2006), Rahnamayan, Tizhoosh e Salama (2008) com resultados promissores.

A inteligência coletiva (ENGELBRECHT, 2006; EBERHART; SHI, 2007), outro campo da computação natural, envolve o estudo do comportamento coletivo em sistemas descentralizados. Tais sistemas são compostos por uma população de indivíduos simples que interagem localmente um com o outro e com seu ambiente. Embora não haja tipicamente nenhum controle centralizado que dita o comportamento dos indivíduos, as interações locais entre os indivíduos fazem com que um padrão de comportamento global apareça (solução do problema). Exemplos de tais sistemas podem ser encontrados na natureza, incluindo colônias de formigas, bandos de pássaros, colmeias de abelhas e muitos outros. O termo inteligência coletiva refere-se a solução de problemas que emergem da interação entre indivíduos em tais sistemas e o termo inteligência computacional coletiva se refere a modelos algorítmicos de tais comportamentos. Um algoritmo da inteligência coletiva que vem tendo significativo progresso é a Otimização através de Enxame de Partículas.

Normalmente, para se detectar ou estimar o movimento de um objeto observado através de imagens de uma sequência de vídeo deve-se envolver um processo de otimização. Um das alternativas para se solucionar o problema de otimização são os métodos determinísticos baseados em gradiente (XIAO; LI, 2008). Estes métodos normalmente são eficientes computacionalmente, mas podem cair em mínimos locais ocasionando uma estagnação. Por outro lado, os métodos estocásticos (ZHANG et al., 2008) possuem uma maior probabilidade de alcançar o ótimo global. Dentre os métodos de otimização estocásticos biologicamente inspirados baseados em populações, podemos citar o algoritmo conhecido como Otimização através de Enxame de Partículas (KENNEDY; EBERHART, 1995; SHI; EBERHART, 1998) e o algoritmo Evolução Diferencial (STORN; PRICE, 1997; DAS; KONAR; CHAKRABORTY, 2005). Nestes métodos, a solução do problema não envolve o cálculo de derivadas, mas uma população de possíveis soluções do problema é evoluída em paralelo. Geralmente estes métodos de busca, ou otimização, por trabalharem com uma população são mais custosos computacionalmente do que baseados em gradiente, mas por outro lado fornecem soluções mais robustas.

1.3 Objetivo

O objetivo principal deste trabalho é o desenvolvimento e análise do Filtro de Partículas hibridizado com Evolução Diferencial ou Otimização através de Enxame de Partículas para incrementar a robustez dos métodos de rastreamento. Através deste projeto, pretende-se aperfeiçoar os métodos existentes para detecção e rastreamento de objetos em imagens de uma única câmera de vídeo. Algumas abordagens para o problema existente podem ser encontradas nos trabalhos de Siddiqui e Yousaf (2007), Kao, Maasdam e Sheppard (2008), Lozano e Otsuka (2009).

O Filtro de Partículas pode ser usado para a solução de problemas de rastreamento, porém encontra algumas limitações. A maior delas é a degeneração de suas partículas ao longo do tempo. Para tentar minimizar essa falha, o problema da degeneração foi tratado como um problema de otimização. Desta forma, foi possível o uso dos algoritmos Evolução Diferencial e Otimização através de Enxame de Partículas, este último utilizado para permitir uma melhor comparação de desempenho dos algoritmos.

Neste trabalho é proposto implementar e validar, através de dois estudos de caso, os algoritmos que utilizam o Filtro de Partículas hibridizado com Evolução Diferencial. O primeiro é o problema do caminhão-reboque, onde uma função não linear com nove variáveis é rastreada. Já o segundo estudo de caso consiste em detectar e rastrear um objeto, no caso, o rosto de uma pessoa em uma sequência de imagens.

1.4 Estrutura

Esta dissertação está organizada da seguinte forma.

No Capítulo 2, a teoria do Filtro de Partículas é descrita junto com seu algoritmo em pseudo-código. Também é apresentado o problema de degeneração presente no FP.

O Capítulo 3 apresenta uma descrição dos algoritmos da computação natural: i) Otimização através de Enxame de Partículas; e ii) Evolução Diferencial.

No Capítulo 4, é descrito a hibridização do algoritmo Filtro de Partículas com Otimização através de Enxame de Partículas ou Evolução Diferencial.

O Capítulo 5 mostra os testes que foram feitos e apresenta os resultados dos dois estudos de caso.

Conclusões são apresentadas no Capítulo 6.

2 *Filtro de Partículas*

Estimação de estado é um problema muito importante não só em controle automático, mas também na predição de séries temporais, rastreamento, navegação de robôs, etc. Uma abordagem comum de ser usada para a modelagem é baseada em espaço de estado. Se o processo é linear e Gaussiano, há uma solução ótima na forma fechada como o Filtro de Kalman e o Filtro de Kalman estendido. Por outro lado, nos problemas do mundo real, que apresentam comportamento não linear e são não-Gaussianos, a informação disponível chega sequencialmente e está corrompida por ruído. Para tal tipo de processo não há soluções fechadas, mas os estados ocultos do modelo podem ser estimados usando Filtro de Partículas, que possui sua fundamentação na teoria da estimação Bayesiana (GORDON; SALMOND; SMITH, 1993; DOUCET; FREITAS; GORDON, 2001; ARULAMPALAM et al., 2002).

2.1 Introdução

Considere o modelo de espaço de estado não linear descrito por

$$X_k = f(X_{k-1}, V_{k-1}) \quad (2.1)$$

$$Y_k = g(X_k, U_k) \quad (2.2)$$

onde X_k é a variável de estado no instante de tempo k , Y_k são as observações no instante de tempo k . As funções f e g são não lineares e geram X_k e Y_k , respectivamente. Temos ainda que V_k é o ruído do sistema e U_k é o ruído da medição, que podem ser assumidos como variáveis aleatórias com distribuição normal e covariância Q_k e R_k , respectivamente.

A Equação (2.1) implica a função de densidade de probabilidade $p(X_k|X_{k-1})$, referente à transição. Já a Equação (2.2) implica a função de densidade de probabilidade $p(Y_k|X_k)$, que representa a observação.

O problema considerado aqui consiste em obter a melhor estimativa para a variável de estado X_k quando apenas dados das observações estão disponíveis, por exemplo, $Y_{1:k} = \{Y_1, Y_2, \dots\}$,

$Y_k\}$. Assumindo que X_k segue um processo de Markov, a distribuição *a posteriori* pode ser definida como $p(X_k|Y_{1:k})$. A Figura 2.1 mostra graficamente a dependência entre os estados e as observações.

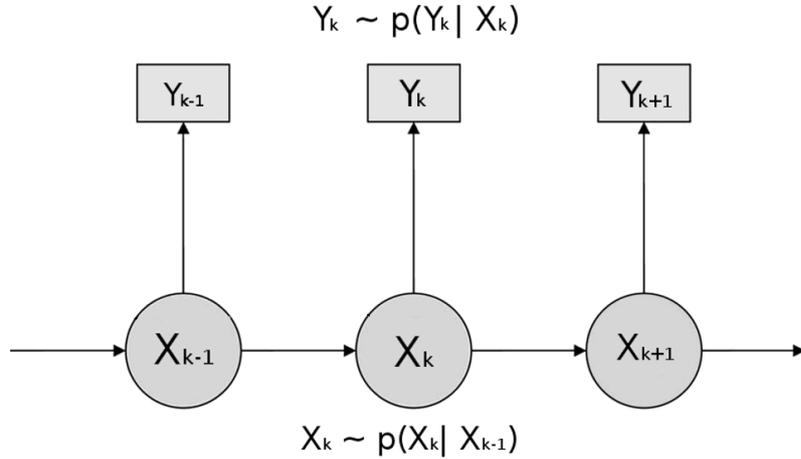


Figura 2.1: Dependência Markoviana entre os estados e as observações

Temos então que o problema de estimação pode ser formulado como a minimização da Média Quadrática do Erro (MQE) estimada, dada por:

$$\hat{X}_k = E[X_k|Y_{1:k}] = \int X_k p(X_k|Y_{1:k}) dX_k \quad (2.3)$$

onde E representa o valor esperado.

Para ilustrar melhor como todo esse processo pode ser utilizado em um problema real, considere o modelo de uma série temporal não linear que pode ser representado pelas equações do espaço de estado:

$$X_k = \frac{X_{k-1}}{2} + 25 \frac{X_{k-1}}{1 + X_{k-1}^2} + 8 \cos(1,2k) + V_k,$$

$$Y_k = \frac{X_k^2}{20} + U_k,$$

onde $V_k \sim N(0, \sigma_V^2)$ e $U_k \sim N(0, \sigma_U^2)$ e considerando $\sigma_V^2 = 10$ e $\sigma_U^2 = 1$ valores fixos e conhecidos. A distribuição do estado inicial é $X_0 \sim N(0, 2)$. Pode-se representar esse problema através das densidades $f(X_k|X_{k-1})$ e $g(Y_k|X_k)$ como sendo

$$f(X_k|X_{k-1}) = N\left(X_k \left| \frac{X_{k-1}}{2} + 25 \frac{X_{k-1}}{1 + X_{k-1}^2} + 8 \cos(1,2k), \sigma_V^2\right.\right)$$

$$g(Y_k|X_k) = N\left(Y_k \mid \frac{X_k^2}{20}, \sigma_U^2\right)$$

Um modelo dinâmico deste tipo pode ser simulado com o pressuposto de Markov em X_k e Y_k , o que implica na densidade de probabilidade composta dos estados e observações

$$\pi_{0:K,0:K} = \pi_0(X_0)g(Y_0|X_0) \times \prod_{k=1}^K f(X_k|X_{k-1})g(Y_k|X_k)$$

Neste modelo, os estados podem ser amostrados um por um sucessivamente sorteando amostras aleatórias a partir das densidades da transição e da observação. Tomando $X_0 = 0,1$; a Figura 2.2 (GORDON; SALMOND; SMITH, 1993) mostra os 50 primeiros passos em X_k . Nela os asteriscos representam o valor real simulado e as linhas sólidas a média das amostras estimadas pelo Filtro de Partículas, que terá seu funcionamento descrito ao longo deste capítulo.

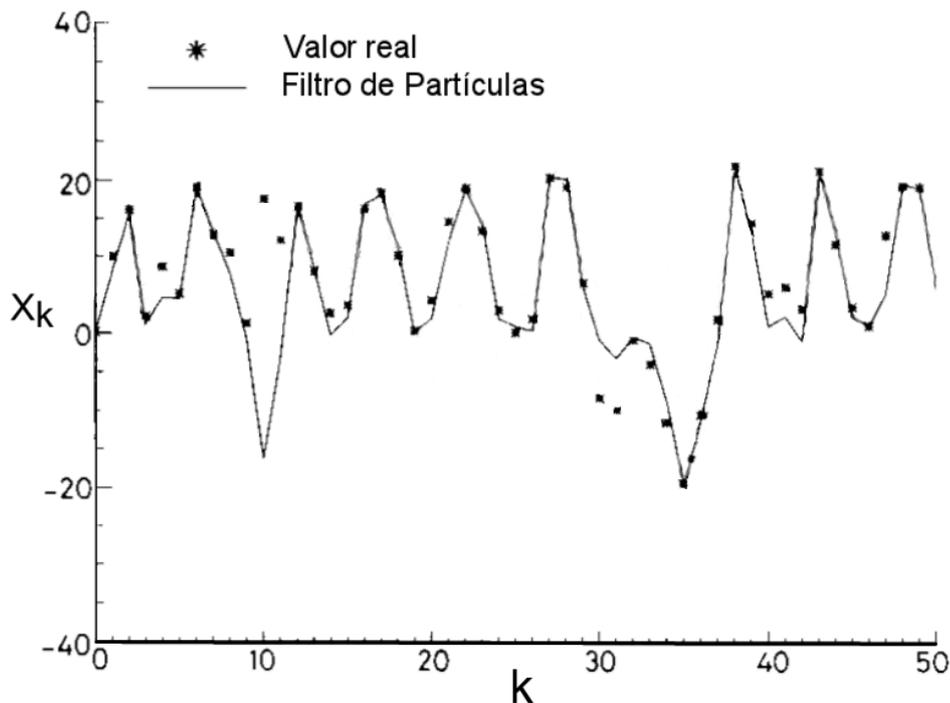


Figura 2.2: Estimação do sistema com 500 amostras

2.2 Método Sequencial de Monte Carlo

O método de Monte Carlo utiliza, para representar a distribuição *a posteriori*, uma coleção de pontos aleatórios chamados amostras ou partículas.

A partir da função densidade do estado inicial X_0 do sistema, juntamente com uma etapa

de *transição* e uma de *observação*, é possível chegar à função densidade do momento seguinte. Dessa maneira, a função densidade *a posteriori* $p(X_k|Y_{1:k})$ pode ser avaliada de forma recursiva. Na etapa de transição, se está interessado em propagar a próxima etapa de tempo k , por meio da densidade de transição dada por

$$p(X_k|Y_{1:k-1}) = \int p(X_k|X_{k-1})p(X_{k-1}|Y_{1:k-1})dX_{k-1}. \quad (2.4)$$

A etapa de observação, envolve a aplicação do teorema de Bayes (BAYES; PRICE, 1763) quando novos dados chegam e é calculado de acordo com

$$p(X_k|Y_{1:k}) = \frac{p(Y_k|X_k)p(X_k|Y_{1:k-1})}{\int p(Y_k|X_k)p(X_k|Y_{1:k-1})dX_k}. \quad (2.5)$$

Estas duas etapas fornecem a solução ótima para o problema de estimação, mas infelizmente a solução da integração multidimensional é difícil de ser obtida analiticamente. Uma abordagem alternativa é obtida através do método Sequencial de Monte Carlo (SMC) (GORDON; SALMOND; SMITH, 1993; DOUCET; FREITAS; GORDON, 2001; ARULAMPALAM et al., 2002; CAPPE; GODSILL; MOULINES, 2007).

A distribuição de probabilidade p , da Equação (2.5), pode ser extremamente complexa. Porém, pode-se amostrar a partir de uma outra distribuição q (distribuição de importância), onde sorteia-se N amostras aleatórias de X_k , $k = 1 \dots N$ a partir de q ao invés de p . Entretanto, para assegurar que o estimador da esperança da Equação (2.3) não seja viciado é preciso fazer uma correção. Este ajuste consiste em definir um peso positivo para cada um dos pontos aleatórios. Se tem que o valor requerido para o peso é proporcional a relação $r = p/q$ avaliada no ponto aleatório. A função r é chamada de *função importância*. Logo, a esperança da Equação (2.3) pode ser estimada como a média ponderada.

$$\begin{aligned} \hat{X}_k &= E[X_k|Y_{1:k}] = \int X_k \frac{q(X_k|Y_{1:k})p(X_k|Y_{1:k})}{q(X_k|Y_{1:k})} dX_k \\ &= \int X_k r(X_k|Y_{1:k})q(X_k|Y_{1:k})dX_k \approx \sum_{k=1}^N \frac{W_k}{\sum_j^N W_j} X_k \end{aligned} \quad (2.6)$$

Dessa forma o resultado da integral através de um conjunto de amostras pode ser calculada como:

$$\hat{X}_k = \sum_{i=1}^N X_k^{(i)} \hat{W}_k^{(i)}, \quad (2.7)$$

onde $\hat{W}_k^{(i)}$ é o peso normalizado da amostra i no momento k . O peso de cada amostra presente nesta equação representa a probabilidade do estado real estar no estado representado por essa amostra.

Para aproximar a função de densidade *a posteriori* foi usado o algoritmo Amostrando e Reamostrando pela Importância. Esse algoritmo tenta contornar o problema da degeneração das partículas do Filtro de Partículas reamostrando as que contribuem pouco para a solução.

O algoritmo SIR consiste em três etapas como mostrado na Figura 2.3: primeiro, todas as partículas são amostradas. Em seguida, o peso da importância é calculado para cada uma delas. Então, as partículas são redefinidas, a fim de descartar partículas com baixa importância e explorar a região onde as partículas têm grande importância.

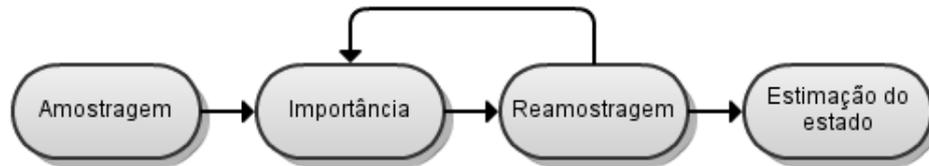


Figura 2.3: Estrutura do funcionamento do Filtro de Partículas

A Figura 2.4 mostra as partículas como círculos e seus pesos são representados pelo tamanho dos círculos, onde os círculos com tamanho maior correspondem a pesos de maior importância. De forma análoga, os círculos menores representam pesos de menor importância. A importância de cada partícula é calculada através da densidade *a posteriori* da partícula.

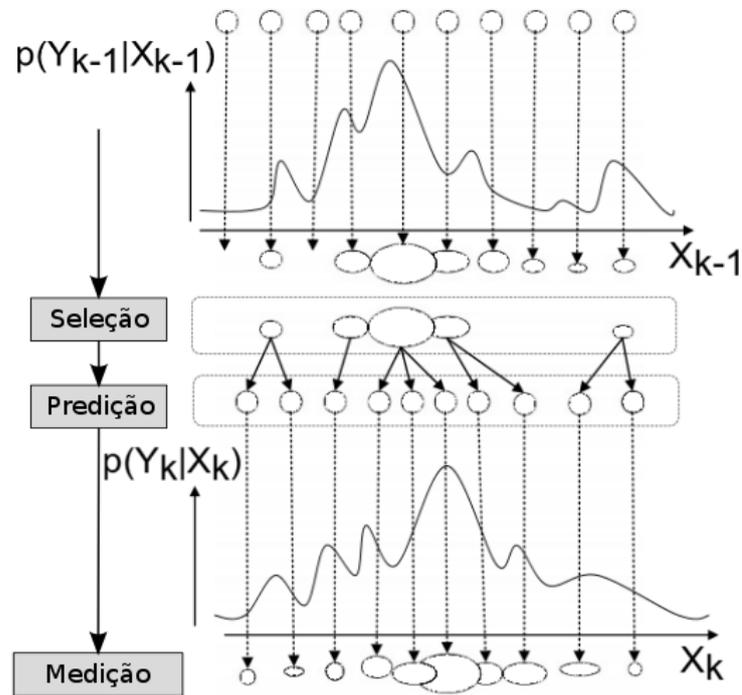


Figura 2.4: Ilustração de duas das etapas do algoritmo FP, mostrando partículas com diferentes pesos

A implementação do Filtro de Partículas pode ser feita de acordo com o pseudo-código mostrado na Figura 2.5. No primeiro laço *For* acontece a inicialização das partículas e seus pesos. Em seguida, o algoritmo entra em um laço onde para cada passo no tempo se atualiza cada uma das partículas e os pesos, além de reamostrá-las, caso necessário. Note que os pesos são normalizados, mantendo assim os pesos sempre com valores no intervalo $[0, 1]$.

```

Input: População de tamanho M
  for cada partícula  $i = 1, \dots, M$  do
    // Criação das partículas (amostras)
     $X_0^{(i)} \sim p(X_0)$ 
    Calcula  $W_0^{(i)}$ 
  5: end for
  repeat
     $X_{orig} = \{X_k^{(i)}, W_k^{(i)}\}_{i=1}^M$ 
    for cada partícula  $i = 1, \dots, M$  do
      // Computa os pesos
      10:  $W_k^{(i)} = W_k^{(i)} p(Y_k | X_k^{(i)})$ 
      // Normaliza os pesos
      
$$W_k^{(i)} = \frac{W_k^{(i)}}{\sum_{i=1}^M W_k^{(i)}}$$

      // Reamostra
      
$$\left\{X_k^{(i)}, \frac{1}{M}\right\}_{i=1}^M = \left\{X_k^{(i)}, W_k^{(i)}\right\}_{i=1}^M$$

      15: Replica as partículas na proporção de seus pesos
    end for
  until número máximo de iterações

```

Figura 2.5: Algoritmo Filtro de Partículas

2.3 O problema da degeneração

Para diminuir a degeneração de um conjunto de partículas deve-se minimizar a variância dos pesos. Logo, para que ocorra essa minimização da variância é preciso (ZHANG et al., 2010):

- Para partículas com peso relativamente grande, diminuir seus pesos.
- Para partículas com peso relativamente pequeno, aumentar seus pesos.
- Manter a ordem dos pesos das partículas, ou seja, os pesos maiores continuam maiores e os pesos menores continuam menores.

Uma forma de minimizar a variância, e conseqüentemente combater a degeneração, é tentando manter sempre um conjunto de partículas com elementos o mais informativo possível, quer dizer, maximizar o peso absoluto das partículas. Assim, todas as partículas tendem a um valor limite superior e ao se normalizar esses pesos a variância tende a zero. Mas, deve-se ter o cuidado de não impedir a exploração adequada do espaço de busca, de modo a não cair em um máximo local.

A reamostragem tentar evitar a degeneração exatamente dessa forma, substituindo as partículas com peso pequeno com novas partículas que possivelmente tem pesos relativamente maiores. Apesar dessa abordagem reduzir o problema, a degeneração continua existindo, pois a nova amostra pode continuar tendo peso relativamente pequeno.

Uma forma de medir a degeneração é através do tamanho efetivo da amostra (N_{eff}) (KONG; LIU; WONG, 1994; AIUBE, 2005), que é calculado de acordo com a seguinte equação:

$$N_{eff} = \frac{N}{1 + Var_{\pi^{(*)|Y_{0:k}}}(W^{*}(X_{0:k}))}, \quad (2.8)$$

onde π é a função densidade usada na tentativa de aproximar a função densidade real do problema.

Porém, de modo geral, é impossível calcular a variância da Equação (2.8). Logo, uma aproximação de N_{eff} é mais interessante para essa medição. A aproximação \hat{N}_{eff} permite uma avaliação do grau de degeneração do algoritmo através da equação

$$\hat{N}_{eff} = \frac{1}{\sum_{i=1}^N (W^{(i)})^2}. \quad (2.9)$$

O tamanho efetivo da amostra aumenta quando a variância do valor dos pesos é reduzida, ou seja, quando a degeneração apresentada pelas partículas é reduzida. De forma simétrica, o tamanho efetivo da amostra diminui quando a degeneração aumenta.

3 *Algoritmos da computação natural*

O termo computação natural (CASTRO, 2007) é uma terminologia relativamente nova introduzida para abranger três classes de métodos: i) aqueles que se inspiram na natureza para o desenvolvimento de novas técnicas para solução de problemas; ii) aqueles que são baseadas no uso de algoritmos para sintetizar fenômenos naturais; e iii) aqueles que empregam materiais naturais (por exemplo, moléculas) para computar.

O campo de estudo que estamos interessados é o das técnicas que são inspiradas na natureza, também conhecidos como computação inspirada biologicamente (inglês: bio-inspired computation). Neste contexto, nós vamos usar dois algoritmos da área de computação inspirada biologicamente, usados para otimização que serão descritos a seguir: i) Otimização através de Enxame de Partículas (KENNEDY; EBERHART, 1995) inspirada pelo movimento dos bandos de pássaros ou cardumes de peixes; e ii) a Evolução Diferencial (STORN; PRICE, 1997) baseado na teoria da evolução.

3.1 **Otimização através de Enxame de Partículas**

Otimização através de Enxame de Partículas (KENNEDY; EBERHART, 1995) se inspira no movimento dos bandos de pássaros para encontrar as possíveis soluções. Cada pássaro do bando representa um elemento da população, que corresponde a uma possível solução que está sendo explorada no espaço de busca.

Assim, como outros algoritmos baseados em população, a Otimização através de Enxame de Partículas tem sua população inicializada aleatoriamente. Cada elemento da população, chamado de partícula, recebe uma velocidade inicial aleatória. Desta forma são definidas as partículas iniciais da população.

Após ser analisada a aptidão de cada elemento da população, é calculada a melhor aptidão encontrada para essa população e cada partícula tem sua velocidade modificada com um fator influenciado pela melhor posição já encontrada da partícula e a melhor posição já encontrada

de toda a população, como mostrado na Figura 3.1.



Figura 3.1: Estrutura do funcionamento da Otimização através de Enxame de Partículas

A etapa de atualização da OEP consiste no cálculo da nova velocidade e da nova posição de cada uma das partículas da população. Dada uma partícula i na geração G sua velocidade e posição é calculada pelas equações:

$$v_G^{(i)} = wv_G^{(i)} + c_1 \text{rand}_p(p_G^{(i)} - x_G^{(i)}) + c_2 \text{rand}_g(p_G^{(g)} - x_G^{(i)}) \quad (3.1)$$

$$x_{G+1}^{(i)} = x_G^{(i)} + v_G^{(i)} \quad (3.2)$$

onde rand_p e rand_g são números aleatórios gerados usando a distribuição de probabilidade uniforme no intervalo $[0, 1]$. A constante w é o peso de inércia das partículas, sendo o fator que define a importância da contribuição da velocidade anterior. Os vetores $p_G^{(i)}$ e $p_G^{(g)}$ representam a melhor posição anteriormente encontrada pela partícula i e por todas as partículas, respectivamente. O termo $c_1 \text{rand}_p(p_G^{(i)} - x_G^{(i)})$ é chamado termo cognitivo da partícula i . Já o termo $c_2 \text{rand}_g(p_G^{(g)} - x_G^{(i)})$ é chamado de termo social. O significado dessas influências na nova velocidade ficam mais claras ao se observar a Figura 3.2.

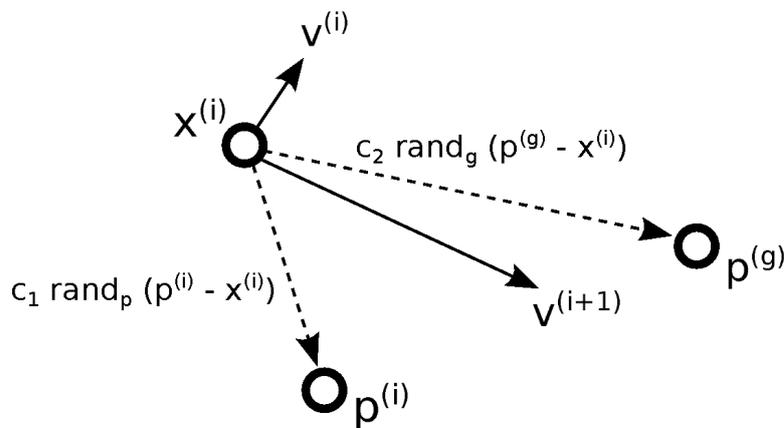


Figura 3.2: Vetores mostrando a composição da nova velocidade da partícula

No pseudo-código da Figura 3.3 está descrito o algoritmo Otimização através de Enxame de Partículas. Nos últimos anos outras versões de OEP foram desenvolvidos, como o En-

xame Gaussiano (KENNEDY, 2003; KROHLING, 2004b) e OEP com salto caótico (MENDEL; KROHLING; CAMPOS, 2011). Nesta dissertação, o algoritmos OEP padrão foi usado para propósito de comparação com o ED padrão a ser descrito a seguir.

```

Input: População de tamanho  $N_P$ 
Inicialização da população
repeat
  for cada indivíduo na OEP  $i = 1, \dots, N_P$  do
    // Atualiza a velocidade
     $v^{(i)} = wv^{(i)} + c_1 \text{rand}_p(p^{(i)} - x^{(i)}) + c_2 \text{rand}_g(p^{(g)} - x^{(i)})$ 
    // Atualiza a posição
     $x^{(i)} = x^{(i)} + v^{(i)}$ 
    if  $f(x^{(i)}) > f(p^{(i)})$  then
       $p^{(i)} = x^{(i)}$ 
      if  $f(x^{(i)}) > f(p^{(g)})$  then
         $p^{(g)} = x^{(i)}$ 
      end if
    end if
  end for
until número máximo de iterações

```

Figura 3.3: Algoritmo Otimização através de Enxame de Partículas

3.2 Evolução Diferencial

Evolução Diferencial é um algoritmo evolutivo composto por uma população de possíveis soluções. Pode ser resumido pelos passos descritos na Figura 3.4.

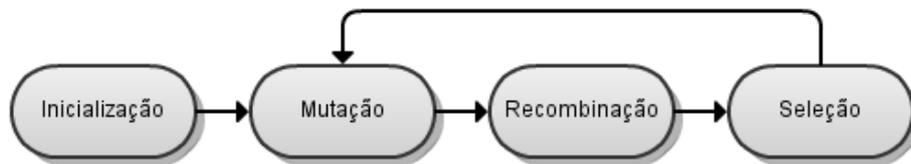


Figura 3.4: Estrutura do funcionamento da Evolução Diferencial

A população inicial é inicializada de forma aleatória, se nenhum conhecimento prévio sobre o espaço de solução estiver disponível. Supondo uma população $P_G^{(i)}$, um vetor de tamanho N_P com componentes $p_{j,G}^{(i)}$, onde i é o índice para cada indivíduo $P_G^{(i)}$, j é a posição no indivíduo D -dimensional e G é a geração que a população pertence. Sucessivas gerações podem ser criadas através da adição de um elemento, escolhido aleatoriamente, à diferença ponderada de dois outros, também selecionados aleatoriamente, como mostrado na Figura 3.5. Para a ED

padrão (DE/rand/1/bin) a recombinação, a mutação e a seleção são definidos diretamente como se segue.

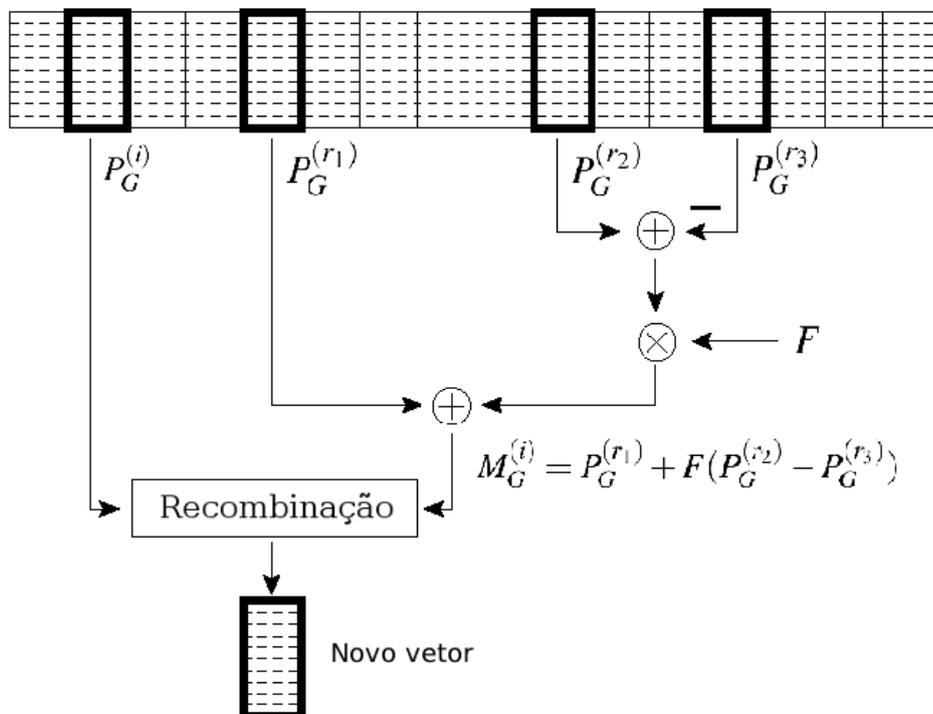


Figura 3.5: Operadores da Evolução Diferencial

Para cada vetor de possíveis soluções $P_G^{(i)}$ na geração G um vetor mutante $M^{(i)}$ é calculado por

$$M_G^{(i)} = P_G^{(r_1)} + F(P_G^{(r_2)} - P_G^{(r_3)}) \quad (3.3)$$

onde $i = 1, 2, \dots, N_p$ e r_1, r_2 e r_3 são índices inteiros mutuamente diferentes e selecionados aleatoriamente a partir de $\{1, 2, \dots, N_p\}$. Isso implica $N_p \geq 4$. O fator de mutação F tipicamente possui valor entre $[0,4; 1]$ (DAS; SUGANTHAN, 2011), é uma constante real, que determina a ampliação da adição da variação diferencial de $(P_G^{(r_2)} - P_G^{(r_3)})$. Valores maiores de F resultam em maior diversidade na população gerada e valores mais baixos causam convergência mais rápida (RAHNAMAYAN; TIZHOOSH; SALAMA, 2008).

A Evolução Diferencial utiliza a operação de recombinação para gerar novas soluções, embaralhando cada vetor com um mutante, e também para aumentar a diversidade da população.

Os elementos do vetor gerado através da operação de recombinação são calculados de acordo com:

$$z_{j,G}^{(i)} = \begin{cases} m_{j,G}^{(i)}, & \text{se } \text{rand}_j(0,1) \leq C_r \text{ ou } j = k \\ p_{j,G}^{(i)}, & \text{caso contrário.} \end{cases} \quad (3.4)$$

A probabilidade de cruzamento $C_r \in [0, 1]$ é a taxa de recombinação, que controla a fração dos valores mutantes que são usados. Para cada j um número aleatório é gerado de acordo com a distribuição uniforme no intervalo $[0, 1]$, chamado aqui $\text{rand}_j(0, 1)$. O índice $k \in \{1, 2, \dots, D\}$ é um índice do parâmetro aleatório, escolhido uma vez para cada i para garantir que pelo menos um parâmetro seja selecionado a partir do vetor mutado $M_G^{(i)}$. A maioria dos valores utilizados para C_r estão dentro do intervalo $[0, 4; 1]$, de acordo com recomendação de Das, Konar e Chakraborty (2005).

Na seleção, o vetor $Z_G^{(i)}$ ou o $P_G^{(i)}$ é selecionado para ser um membro da próxima geração $G + 1$ de acordo com o valor da função objetivo. Em caso de problema de maximização:

$$P_{G+1}^{(i)} = \begin{cases} Z_G^{(i)}, & \text{se } f(Z_G^{(i)}) \geq f(P_G^{(i)}) \\ P_G^{(i)}, & \text{caso contrário.} \end{cases} \quad (3.5)$$

Existem outras variantes com base em estratégias diferentes de mutação e recombinação (STORN; PRICE, 1997). O pseudo-código do algoritmo Evolução Diferencial está mostrado na Figura 3.6.

```

Input: População de tamanho  $N_P$ 
Inicialização da população
repeat
  for cada indivíduo na ED  $i = 1, \dots, N_P$  do
    Seleciona  $P^{(r_1)}, P^{(r_2)}$  e  $P^{(r_3)}$ , onde  $r_1 \neq r_2 \neq r_3 \neq i$ 
    //  $D$  é a dimensão de uma partícula
     $j_{rand} = \text{floor}(D * \text{rand}(0, 1))$ 
    for cada componente de um indivíduo  $j = 1, \dots, D$  do
      if  $\text{rand}(0, 1) \leq C_r$  ou  $j = j_{rand}$  then
         $z_j^{(i)} = p_j^{(r_1)} + F * (p_j^{(r_2)} - p_j^{(r_3)})$ 
      else
         $z_j^{(i)} = p_j^{(i)}$ 
      end if
    end for
  end for
  // seleciona a próxima geração
  for cada indivíduo  $i = 1, \dots, N_P$  do
    if  $f(Z^{(i)}) \geq f(P^{(i)})$  then
       $P^{(i)} = Z^{(i)}$ 
    end if
  end for
until número máximo de iterações

```

Figura 3.6: Algoritmo Evolução Diferencial

4 *Filtro de Partículas Híbridizado*

O Filtro de Partículas consiste de uma aproximação da distribuição de probabilidade *a posteriori* através da amostragem de pontos, chamados partículas, que são atualizadas assim que novos dados chegam. As partículas são amostras aleatórias, que perseguem a trajetória do estado. O Filtro de Partículas utiliza o método de integração Sequencial de Monte Carlo para aproximar a distribuição de probabilidade *a posteriori* através da amostragem a partir de uma distribuição de probabilidade conhecida para que se calcule os pesos. Um ponto negativo comum presente por se amostrar a partir de uma distribuição é a degeneração, que significa que após algumas iterações alguns pesos tendem em se concentrar em um ponto não-ótimo.

4.1 Introdução

Para sobrepor tal problema, é proposta uma modificação do algoritmo Amostrando e Reamostrando pela Importância incorporando a Evolução Diferencial. Tentativas anteriores de hibridizar computação natural com FP já foram apresentadas na literatura, por exemplo, Estratégia Evolutiva (UOSAKI; KIMURA; HATANAKA, 2003), Otimização através de Enxame de Partículas (KROHLING, 2004a), Algoritmo Evolutivo (PARK et al., 2009), e recentemente Otimização através de Colônia de Formigas (XU; ZHU; XU, 2010). Neste trabalho, uma abordagem híbrida combinando Filtro de Partículas com Evolução Diferencial e também Filtro de Partículas com Otimização através de Enxame de Partículas é proposta.

A questão da degeneração em Filtro de Partículas acontece quando há uma pequena quantidade de partículas de importância muito alta e as demais com pouca relevância. Como queremos amostrar a partir da densidade *a posteriori*, a densidade proposta precisa ser parecida com *a posteriori*, o que significa que a variação dos pesos da importância esteja perto de 0 para estimá-los bem.

Uma tentativa de resolver o problema da degeneração do FP é criar um novo grupo de partículas que gerem pesos maiores. Estas novas partículas substituem aquelas que têm menos

relevância no conjunto original, ou seja, com pesos menores. Neste trabalho, este novo grupo de partículas tem o indivíduo mais apto, aquele que tem um peso elevado, encontrado pela Otimização por Enxame de Partículas ou pela Evolução Diferencial.

Na Figura 4.1 é mostrado o pseudo-código do algoritmo Filtro de Partículas hibridizado. Na linha 10, a chamada para a função *Otimizar* representa a chamada para a execução de um dos dois métodos de otimização, OEP ou ED, onde ocorre a hibridização propriamente dita.

```

Input: População de tamanho M
for cada partícula  $i = 1, \dots, M$  do
    // Criação das partículas (amostras)
     $X_0^{(i)} \sim p(X_0)$ 
    Calcula  $W_0^{(i)}$ 
5: end for
    repeat
         $X_{orig} = \{X_k^{(i)}, W_k^{(i)}\}_{i=1}^M$ 
        // Criação de partículas melhores através da OEP ou ED
        // Otimizar chama o Algoritmo 3.3 ou o Algoritmo 3.6
10:  $X_{otim} = \text{Otimizar}(X_{orig})$ 
        Ordena  $X_{orig}$  de forma ascendente
        Ordena  $X_{otim}$  de forma descendente
        // substitui as  $N$  partículas com menores pesos em  $X_{orig}$ 
        // pelas  $N$  partículas com maiores pesos em  $X_{otim}$ 
15:  $X = \{X_{otim}\}_{i=1}^N \cup \{X_{orig}\}_{i=N+1}^M$ 
        for cada partícula  $i = 1, \dots, M$  do
            // Computa os pesos
             $W_k^{(i)} = W_k^{(i)} p(Y_k | X_k^{(i)})$ 
            // Normaliza os pesos
20:  $W_k^{(i)} = \frac{W_k^{(i)}}{\sum_{i=1}^M W_k^{(i)}}$ 
            // Reamostra
             $\left\{X_k^{(i)}, \frac{1}{M}\right\}_{i=1}^M = \left\{X_k^{(i)}, W_k^{(i)}\right\}_{i=1}^M$ 
            Replica as partículas na proporção de seus pesos
        end for
25: until número máximo de iterações

```

Figura 4.1: Algoritmo Filtro de Partículas hibridizado

4.2 Hibridização com Otimização através de Enxame de Partículas

Neste contexto, as partículas do Filtro de Partículas que são passadas como parâmetro são as partículas iniciais utilizado pela OEP e a função de avaliação é a mesma função de avaliação do FP. Podendo o método OEP inserir, ou não, partículas novas para aumentar a diversidade da população. Após os ciclos de otimização as novas partículas encontradas são retornadas para o FP, conforme mostrado na Figura 4.2.

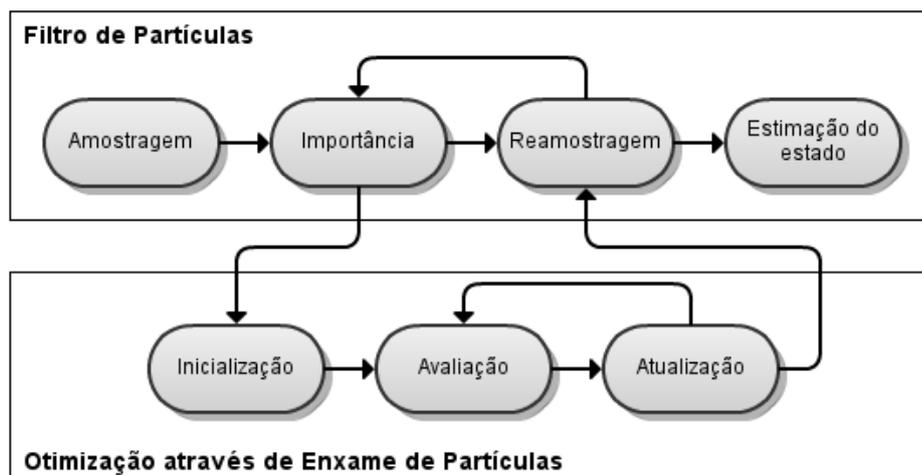


Figura 4.2: Estrutura do funcionamento do Filtro de Partículas combinado com Otimização através de Enxame de Partículas

Nas linhas 7 – 15 do algoritmo da Figura 4.1 é mostrado a mudança necessária no algoritmo do Filtro de Partículas para que seja integrado ao OEP. Para o OEP ser capaz de aproveitar o esforço computacional já gasto, as partículas atuais encontradas pelo Filtro de Partículas (X_{orig}) são utilizados como população inicial, ao invés de inicializar sua população de forma aleatória. Como saída do algoritmo OEP (algoritmo da Figura 3.3) é retornado um conjunto OEP modificado (X_{otim}). Então, esses dois conjuntos são classificados de maneira que a combinação das $(M - N)$ últimas partículas em X_{orig} com as N primeiras em X_{otim} irão substituir as N partículas com os menores pesos pelos N mais aptos encontrado no OEP. Este novo conjunto é utilizado como conjunto de partículas nessa etapa de tempo.

4.3 Hibridização com Evolução Diferencial

A hibridização do Filtro de Partículas com Evolução Diferencial ocorre de forma análoga. Os indivíduos da ED são aquelas partículas estimadas no passo atual do FP, e a função de

avaliação na ED é a função que calcula o peso de uma partícula. A estrutura do funcionamento dessa hibridização é mostrada na Figura 4.3.

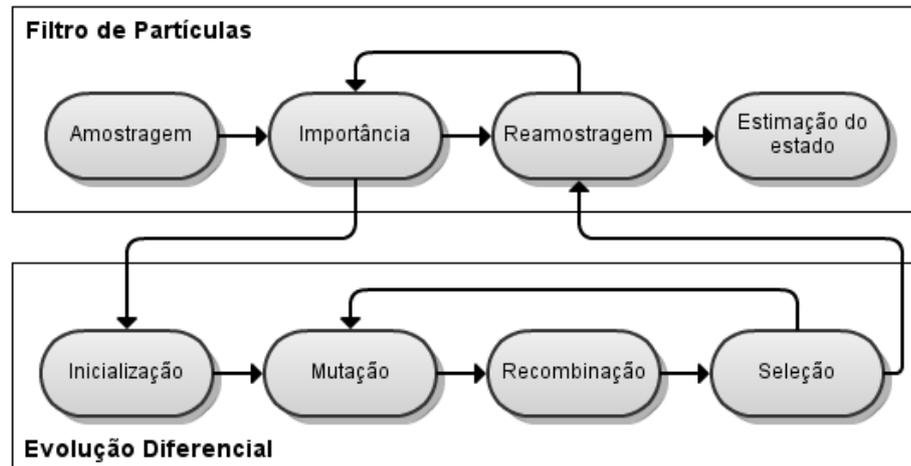


Figura 4.3: Estrutura do funcionamento do Filtro de Partículas combinado com Evolução Diferencial

As alterações no código são idênticas as mudanças descritas na seção 4.2, que ocorrem nas linhas 7 – 15 do algoritmo da Figura 4.1. Apenas o método de otimização das partículas utilizado é realizado pelo algoritmo Evolução Diferencial.

5 *Resultados*

Os métodos de rastreamento Filtro de Partículas, Filtro de Partículas hibridizado com Otimização através de Enxame de Partículas e Filtro de Partículas hibridizado com Evolução Diferencial são testados em dois estudos de caso. O primeiro, consiste no rastreamento da trajetória do sistema caminhão-reboque. O segundo, na detecção e rastreamento em sequência de imagens.

Em todos os estudos de caso os algoritmos foram escritos na linguagem MATLAB[®] e testados utilizando uma máquina AMD Athlon[™] 64 X2 Dual Core Processor 4000+ com 2GB de memória RAM.

5.1 **Estudo de caso 1: Rastreamento de trajetória do sistema caminhão-reboque**

Os algoritmos foram testados no problema do caminhão-reboque (SIMON, 2003; KIYUNA et al., 2003; PARK et al., 2009) com vários reboques. Esse problema consiste em estimar cada um dos nove estados do sistema, um veículo em velocidade constante e com o ângulo da direção do caminhão variando com o tempo.

5.1.1 **Definição do problema**

Como mostrado na Figura 5.1, o problema apresentado aqui consiste de um caminhão ligado a três reboques. O rastreamento da trajetória do caminhão-reboque é obtido através da estimação dos estados utilizando os algoritmos Filtro de Partículas, Filtro de Partículas hibridizado com Otimização através de Enxame de Partículas e Filtro de Partículas hibridizado com Evolução Diferencial. O modelo dinâmico do sistema é descrito pelas seguintes equações:

$$x_0(k+1) = x_0(k) + \frac{vT}{l} \tan(u(k)) + w_0(k) \quad (5.1)$$

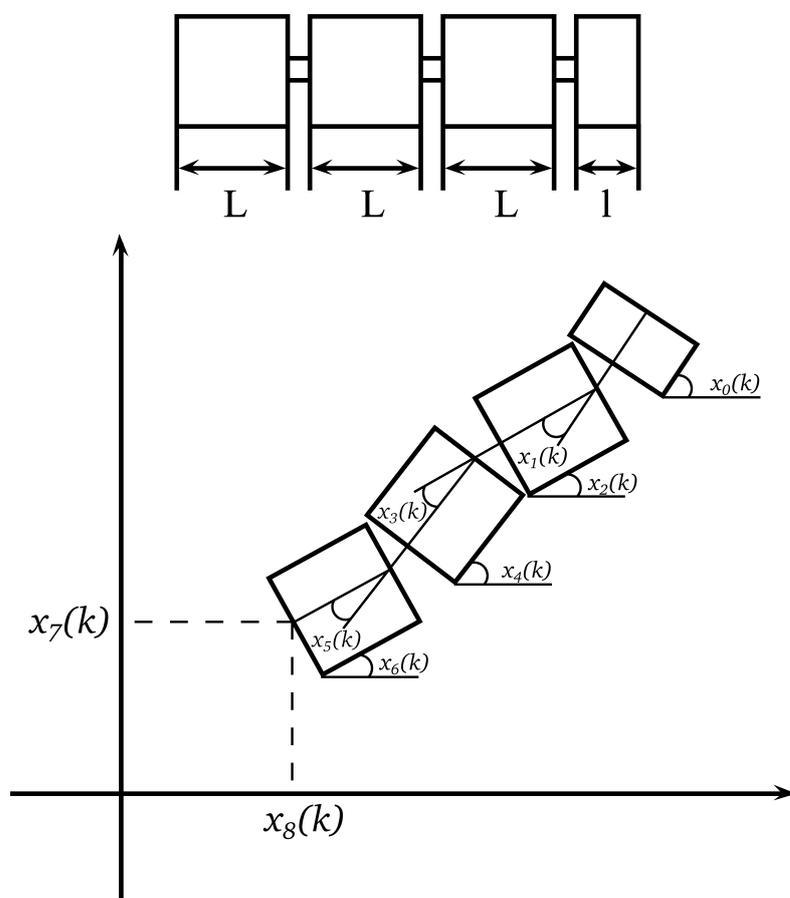


Figura 5.1: Sistema de um caminhão-reboque com três reboques

$$x_1(k) = x_0(k) - x_2(k) \quad (5.2)$$

$$x_2(k+1) = x_2(k) + \frac{vT}{L} \sin(x_1(k)) + w_2(k) \quad (5.3)$$

$$x_3(k) = x_2(k) - x_4(k) \quad (5.4)$$

$$x_4(k+1) = x_4(k) + \frac{vT}{L} \sin(x_3(k)) + w_4(k) \quad (5.5)$$

$$x_5(k) = x_4(k) - x_6(k) \quad (5.6)$$

$$x_6(k+1) = x_6(k) + \frac{vT}{L} \sin(x_5(k)) + w_6(k) \quad (5.7)$$

$$x_7(k+1) = x_7(k) + vT \cos(x_5(k)) \sin\left(\frac{x_6(k+1) + x_6(k)}{2}\right) + w_7(k) \quad (5.8)$$

$$x_8(k+1) = x_8(k) + vT \cos(x_5(k)) \cos\left(\frac{x_6(k+1) + x_6(k)}{2}\right) + w_8(k) \quad (5.9)$$

As variáveis e os parâmetros do sistema do caminhão-reboque (SIMON, 2003) são apresentados na Tabela 5.1. Note que o comprimento do caminhão, o comprimento do reboque, o tempo de amostragem e a velocidade do caminhão são constantes. Temos ainda que $u(k)$ é a variável de entrada e $x_i(k)$, onde $i = 0, \dots, 8$; são as variáveis de saída do problema.

Tabela 5.1: Parâmetros do sistema de caminhão-reboque

Nome	Descrição
l	Comprimento do caminhão
L	Comprimento do reboque
T	Tempo de amostragem
v	Velocidade do caminhão
$u(k)$	Controle de entrada para o ângulo da direção
$x_0(k)$	Ângulo do caminhão
$x_1(k)$	Diferença do ângulo entre o caminhão e o primeiro reboque
$x_2(k)$	Ângulo do primeiro reboque
$x_3(k)$	Diferença do ângulo entre o primeiro reboque e o segundo reboque
$x_4(k)$	Ângulo do segundo reboque
$x_5(k)$	Diferença do ângulo entre o segundo reboque e o terceiro reboque
$x_6(k)$	Ângulo do terceiro reboque
$x_7(k)$	Posição vertical do terceiro reboque
$x_8(k)$	Posição horizontal do terceiro reboque

5.1.2 Parâmetros do algoritmo

Define-se que $w(k) = [w_0(k) w_2(k) w_4(k) w_6(k) w_7(k) w_8(k)]^T \sim N(0, Q)$ e $v(k) \sim N(0, R)$ como sendo distribuições Gaussianas independentes entre si. As matrizes de covariância do ruído do sistema e do ruído da medição é $Q = \text{diag}(1^2, \dots, 1^2)$ e $R = \text{diag}(1^2, \dots, 1^2)$, respectivamente, com as devidas dimensões. Considera-se que o sinal de medição está ausente por um período de tempo e que o caminhão-reboque se move rápido durante a falta de medição.

Nesta simulação, o tempo de amostragem é de 0,5 segundos e o número de partículas utilizadas nos experimentos foi ajustado para 100, 500 e 1000. Além disso, os reboques têm 5,0m cada, o caminhão tem 2,8m e sua velocidade é de 2,0m/s. Na Evolução Diferencial, são usadas 200 gerações, a taxa de recombinação C_r foi 0,8 e o fator de mutação F foi 0,75. Na Otimização através de Enxame de Partículas, as constantes de aceleração c_1 e c_2 possuem valor 2 e o peso da inércia w tem valor 0,9 (EBERHART; SHI, 2007). Durante a simulação, o ruído branco Gaussiano foi a entrada de controle do ângulo de direção, $u(k)$. O valor inicial do vetor $\vec{x}(0) = [x_0(0) x_1(0) \dots x_7(0) x_8(0)]^T$ é $[00000001010]^T$ (PARK et al., 2009).

Para simular o problema de oclusão, equivalente a uma falha na leitura da medição do sistema, as 10 observações seguintes à observação de número 49 foram removidas. Portanto, a observação de número 50 nos gráficos corresponde na verdade ao passo do tempo de número 60 e assim sucessivamente.

5.1.3 Resultados experimentais

Na Figura 5.2 é mostrado o resultado da simulação de x_8 para um conjunto específico de valores de estado. Por simplicidade, uma vez que este sistema tem nove estados, apenas x_8 é mostrado (PARK et al., 2009). Note que x_8 depende de todas os outros estados, exceto x_7 .

Os dez primeiros passos foram omitidos pois estão muito influenciados pela posição inicial definida de forma heurística. Como o problema possui muitas dimensões, o gráfico de apenas uma dimensão pode não ser muito confiável para analisar o desempenho global dos algoritmos. Mesmo assim, pode-se notar que quanto maior o número de partículas menor a diferença entre os métodos de rastreamento.

Uma maneira de analisar melhor o desempenho dos algoritmos híbridos é através da Tabela 5.2, que mostra o desempenho de todos os estados de forma resumida. Os valores apresentados na Tabela 5.2 são as Médias Quadrática do Erro (MQE) de 10 execuções.

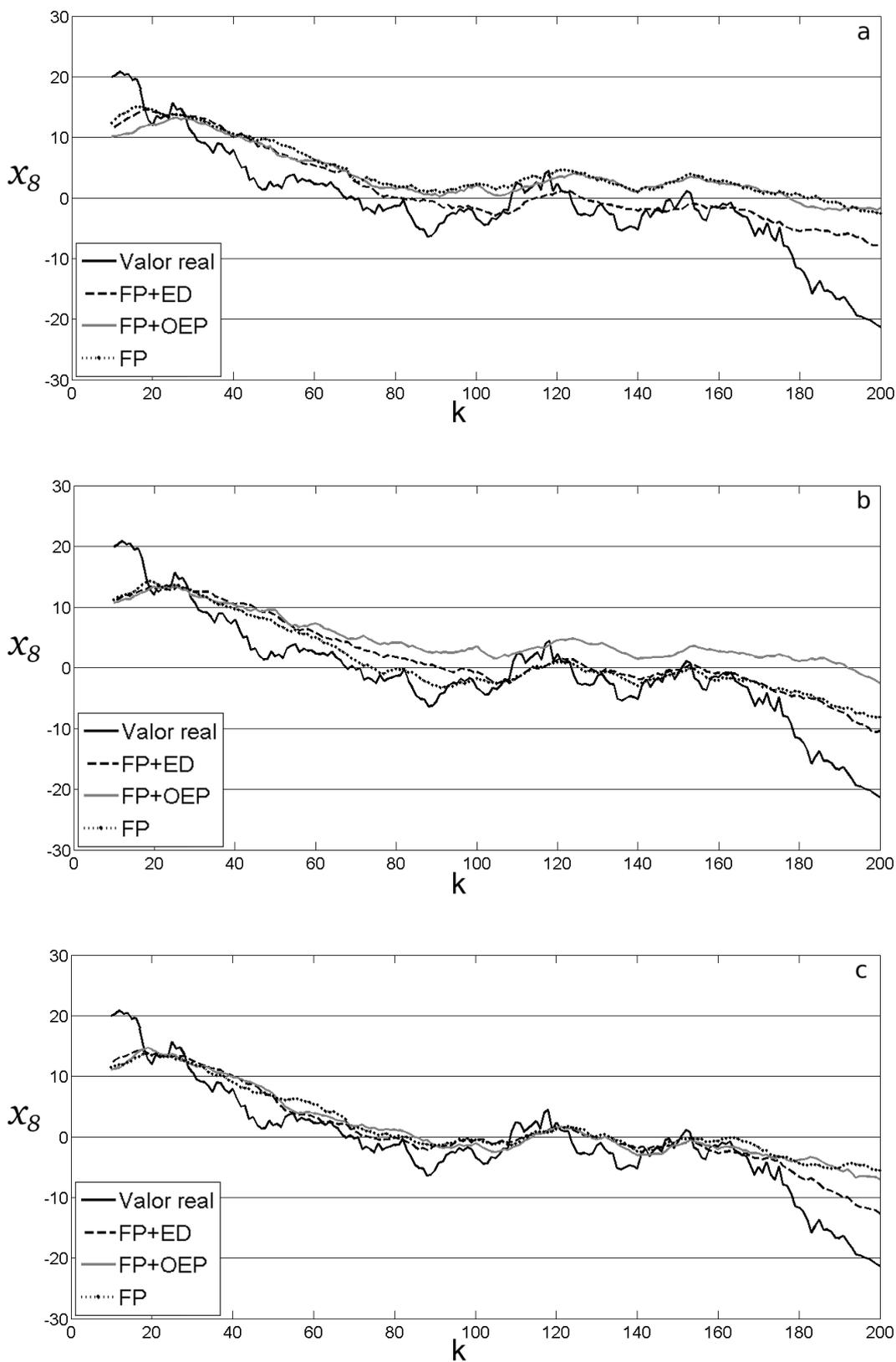


Figura 5.2: Resultados da estimação do estado de x_8 . **a** 100 partículas. **b** 500 partículas. **c** 1000 partículas

Tabela 5.2: MQE médio para o FP, FP+OEP e FP+ED com 100, 500 e 1000 partículas

Partículas	100			500			1000		
	Algoritmos FP	FP+OEP	FP+ED	FP	FP+OEP	FP+ED	FP	FP+OEP	FP+ED
$x_0(k)$	4,39	2,36	3,88	86,65	4,37	3,33	33,29	5,87	5,98
$x_1(k)$	48,09	2,36	6,26	85,56	22,94	2,09	33,81	9,03	14,55
$x_2(k)$	47,24	5,01	5,00	7,74	20,98	6,48	9,52	5,66	10,60
$x_3(k)$	44,95	5,73	17,90	53,67	20,24	5,20	11,54	18,80	13,67
$x_4(k)$	1,13	1,28	7,30	57,15	1,11	0,92	1,59	18,72	1,01
$x_5(k)$	217,94	109,90	46,45	262,43	70,33	267,24	85,68	65,13	85,85
$x_6(k)$	214,94	107,31	37,82	217,95	68,56	261,13	84,35	51,80	83,69
$x_7(k)$	67,88	5,07	117,90	81,89	105,65	68,85	25,11	39,18	4,23
$x_8(k)$	487,99	639,74	240,68	208,38	563,89	167,61	323,26	235,75	95,02
Soma	1.134,55	878,74	483,20	1.061,43	878,05	782,86	608,15	449,94	314,62

5.1.4 Discussão

Como pode ser visto na Figura 5.2 e na última linha da Tabela 5.2, que representa a soma de todas as dimensões MQE, a abordagem híbrida que combina Filtro de Partículas com Otimização através de Enxame de Partícula obteve uma melhoria em relação ao Filtro de Partículas. Entretanto, o Filtro de Partículas com Evolução Diferencial foi o algoritmo que conseguiu o melhor desempenho. Esta relação pode-se notar para todos os números de partículas testados. O desempenho do algoritmo FP+ED com 100 partículas foi melhor que o desempenho do FP padrão com 1000, ficando evidente a melhora obtida com a hibridização.

5.2 Estudo de caso 2: Detecção e rastreamento em sequência de imagens

O objetivo desse problema é reconhecer o rosto de uma determinada pessoa e estimar a sua posição durante uma sequência de imagens.

Para o ser humano a tarefa de visualizar e detectar um objeto se realiza de forma instintiva, quase imperceptível e instantânea. Uma forma de detectar um determinado objeto é utilizando uma imagem modelo, uma lembrança de como é o objeto ou um parecido, e procurar por imagens semelhantes na imagem sendo analisada. Essa semelhança deve ser definida baseada em pelo menos uma característica do objeto, como cor ou formato.

Desenvolver um programa de computador que reproduza alguma função do corpo humano não costuma ser simples. Reproduzir a tarefa de detectar um objeto não é diferente. Apesar disso, já foram criadas várias formas de se realizar detecção (ZHOU; MA; CHELBERG, 2006),

baseado em uma ou várias características da imagem. A semelhança entre uma imagem modelo e a analisada pode ser definida usando dados estatísticos de alguma dessas características, como histograma e coeficiente de correlação. Esse tipo de técnica de casamento de padrão (do inglês: *Template Matching*), é usado largamente para detecção de faces.

O rastreamento em vídeo, ou em uma sequência de imagens, consiste em associar um objeto-alvo em quadros ou em imagens consecutivas. Há uma variedade de usos, como segurança e vigilância, realidade aumentada, controle de tráfego, imagens médicas, edição de vídeo.

O processo de rastreamento pode ser dificultado se o objeto analisado se move rapidamente, muda de direção nos quadros, sofre oclusão de um outro elemento da cena, mudança de iluminação ou deformação do objeto.

5.2.1 Definição do problema

São apresentadas três formas de Casamento de Padrão para detecção neste trabalho: Correlação Cruzada Normalizada Híbrida (CCNH), Interseção de Histograma (IH) e Coeficiente de Correlação (CC).

A primeira delas, a CCNH (LI; DUAN; ZHANG, 2010), utiliza duas funções usadas para comparar a similaridade entre uma imagem de teste $t(i, j)$ e a imagem de modelo $m(i, j)$, onde i e j são as coordenadas que representam a posição de um pixel na imagem, considerando a imagem uma matriz onde cada elemento da matriz possui informação das cores do pixel. Essas funções são a SSD, do inglês *Sum of Squared Differences*, e a NCC, do inglês *Normalized Cross Correlation*, a partir delas é definida uma terceira função híbrida, mais robusta, que define o valor usado nesse trabalho para medir a similaridade entre duas imagens.

$$SSD(m, n) = \sum_i \sum_j (t(p+i, q+j) - m(i, j))^2 \quad (5.10)$$

$$NCC(m, n) = \frac{\sum_i \sum_j t(p+i, q+j) \cdot m(i, j)}{\sqrt{\sum_i \sum_j t(p+i, q+j)^2} \cdot \sqrt{\sum_i \sum_j m(i, j)^2}} \quad (5.11)$$

Nas equações, os valores p e q equivalem a dizer que o centro da imagem modelo foi colocado na posição (p, q) da imagem de teste. Sabe-se ainda que $SSD(p, q)$ é não negativo. $NCC(p, q)$ varia no intervalo $[0, 1]$. Quando este valor for igual a 1 o melhor casamento possível foi encontrado. A função híbrida é definida como

$$F(p, q) = \frac{1}{\sqrt{2\pi}\delta} \cdot e^{\frac{-(SSD/NCC)}{2\delta^2}}, \quad (5.12)$$

onde δ é um parâmetro que pode variar entre 1 e 10. Nos experimentos realizados foi adotado o valor 1 (LI; DUAN; ZHANG, 2010).

Outra forma de detectar um objeto numa imagem colorida é através de seu histograma (SWAIN; BALLARD, 1991). O histograma, em processamento de imagem, indica a informação da quantidade de vezes que cada uma das cores se repete na imagem. Um histograma com k intervalos significa que o histograma possui o número de repetições de k cores distintas. O método da Interseção de Histograma (SMITH, 1997) calcula a distância da semelhança entre a imagem de teste e a imagem modelo, sendo que a distância 0 significa imagens idênticas para o método. Assim, quanto mais distintas as imagens, maior o módulo da distância. Essa distância pode ser calculada para histogramas com k intervalos como:

$$d'_{m,t} = 1 - \frac{\sum_{k=0}^{K-1} \min(h_m[k], h_t[k])}{\min(|h_m|, |h_t|)}, \quad (5.13)$$

onde

$$|h_x| = \sum_{k=0}^{K-1} h_x[k] \quad (5.14)$$

$h_m[k]$ e $h_t[k]$ indica o valor da k -ésima frequência do histograma da imagem modelo e de teste, respectivamente.

O método de casamento de padrões com Coeficiente de Correlação (SUSSMAN; WRIGHT, 1999) usa uma função de distância para ser minimizada baseada no método dos mínimos quadrados. Considerando que a imagem modelo $m(i, j)$ tem tamanho $M \times N$:

$$d'^2(p, q) = 2 - 2 \sum_{i=1}^M \sum_{j=1}^N t'(i-p, j-q) m'(i, j) \quad (5.15)$$

$$t'(i, j) = [t(i, j) - \bar{t}] / \sigma_t \quad (5.16)$$

$$m'(i, j) = [m(i, j) - \bar{m}] / \sigma_m \quad (5.17)$$

onde \bar{t} e σ_t são a média e o desvio padrão de $t(i, j)$ para a imagem de teste e, de forma similar, \bar{m} e σ_m são a média e o desvio padrão de $m(i, j)$ para a imagem modelo.

Neste trabalho o rastreamento é feito utilizando o Filtro de Partículas híbrido para processar cada um dos quadros do vídeo que forma a sequência de imagens, de forma similar Vidal (2011) usou o Filtro de Kalman híbrido com OEP. Esse método é utilizado para tentar minimizar a perda da qualidade do rastreamento quando encontra algumas das dificuldades comuns no processo, comentadas anteriormente, mesmo com um mínimo de informação possível e pré-processamento. Um quadro equivale a um passo no Filtro de Partículas, dessa maneira o número total de passos será equivalente ao número de quadros a ser analisado.

O Filtro de Partículas estima a probabilidade da presença do objeto na imagem analisada baseado no histórico de observações até o presente momento. Desta maneira, pode-se conseguir resultados melhores em problemas como o de oclusão e a movimentação brusca do objeto.

Inicialmente não se possui nenhuma informação sobre a imagem. Portanto, as partículas iniciais são posicionadas aleatoriamente dentro do primeiro quadro analisado. A função de avaliação nesse problema é a distância entre a imagem teste e a imagem modelo em um dos métodos de detecção apresentados na subseção anterior. O algoritmo Filtro de Partículas híbrido é o mesmo da Figura 4.1.

O modelo do movimento se baseia em Zotkin, Duraiswami e Davis (2002). A posição a ser analisada em um determinado quadro é definido como o par $(pos_x^{(i)}, pos_y^{(i)})$ para o quadro i . A sua velocidade (vel_x, vel_y) é calculada de acordo com a variação de (pos_x, pos_y) entre i e $(i - 1)$. O movimento que uma determinada partícula realizará no eixo x é definido por:

$$pos_x^{(i+1)} = F_{rand} + vel_x^{(i)} + pos_x^{(i)}, \quad (5.18)$$

onde a força de excitação aleatória F_{rand} permite mudanças no estado do objeto. Essa força é um número aleatório Gaussiano com média zero e desvio padrão $\sigma = 1$. O movimento no eixo y é calculado de forma similar.

5.2.2 Parâmetros do algoritmo

Para cada um dos algoritmos testados, Filtro de Partículas, Filtro de Partículas híbrido com Otimização através de Enxame de Partículas e Filtro de Partículas híbrido com Evolução Diferencial, foram usados três tipos de detecção: Correlação Cruzada Normalizada Híbrida (CCNH), Interseção de Histograma (IH) e Coeficiente de Correlação (CC), para rastreamento em uma sequência de imagem com 100 quadros. Essa sequência são os 100 primeiros quadros do vídeo *Girl* (disponível em <http://www.ces.clemson.edu/~stb/research/headtracker/>), sem a elipse vermelha presente no vídeo original.

Os quadros do vídeo possuem largura de 320 pixels por 240 pixels de altura. Além disso, para uso neste trabalho, a sequência de imagens coloridas possui duas versões uma com os quadros originais e outra criada com os quadros modificados inserindo ruído branco Gaussiano com média zero e desvio padrão 0,1.

Em todos os algoritmos foram usadas 150 partículas no Filtro de Partículas para estimar a posição. Para os algoritmos hibridizados foram usadas 5 gerações para otimização. Na Evolução Diferencial, a taxa de recombinação C_r foi 0,8 e o fator de mutação F foi 0,75. Já na Otimização através de Enxame de Partículas, as constantes de aceleração c_1 e c_2 possuem valor 2 e o peso da inércia w tem valor 0,9 (EBERHART; SHI, 2007).

Foram feitos três testes para cada um dos casos analisados. Como imagem modelo para a detecção foi usado um recorte do rosto da mulher que aparece no vídeo conforme mostrado na Figura 5.3. Essa imagem possui altura de 85 pixels e largura de 94 pixels. Neste trabalho não foi feito qualquer pré-processamento na sequência de imagens.



Figura 5.3: Rosto da mulher usado como modelo na detecção

A sequência de imagens presente na Figura 5.4 mostra alguns quadros do conjunto extraídos do vídeo original. Essa sequência tem variação na posição, tamanho e forma da imagem modelo, além da rotação da mulher, o que provoca uma oclusão total. Na Figura 5.5, é apresentada a sequência criada com ruído, logo, os mesmos problemas aparecem acrescidos de ruído.

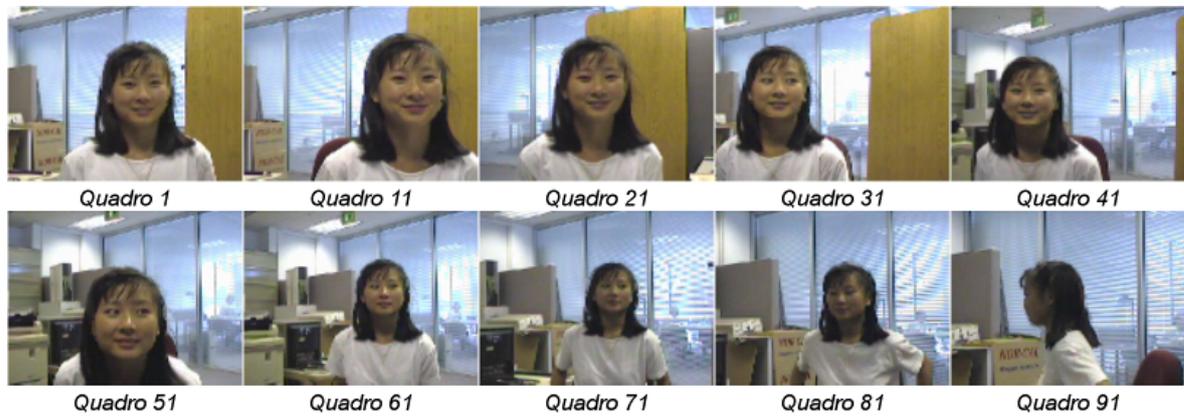


Figura 5.4: Uma amostra da sequência de imagens original

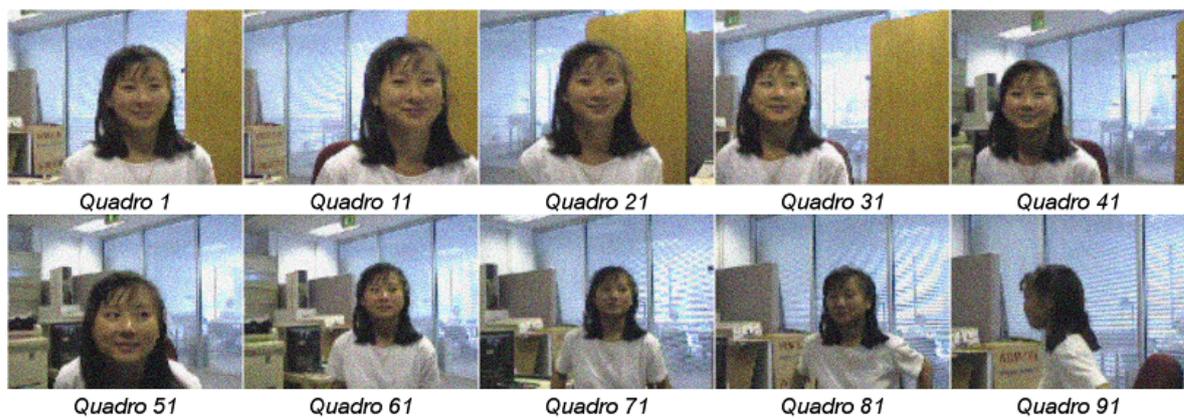


Figura 5.5: Uma amostra da sequência de imagens com ruído

5.2.3 Resultados experimentais

Para uma análise qualitativa, foram criadas figuras que mostram alguns quadros da sequência de imagens marcados onde cada método aponta ser o local onde o rosto da mulher está presente. O quadrado de cor vermelha representa o local encontrado pelo algoritmo Filtro de Partículas padrão. Na cor verde, temos o quadrado que representa o local encontrado pelo algoritmo Filtro de Partículas hibridizado com Otimização através de Enxame de Partículas. Por último, há o quadrado na cor azul identificando a posição apontada pelo algoritmo Filtro de Partículas combinado com a Evolução Diferencial. Na linha de quadros superior de cada imagem são mostrados os quadros da sequência de imagens original com as marcações, da mesma maneira, a linha de quadros inferior mostra a sequência de imagens com ruído.

Na Figura 5.6, os métodos de rastreamento utilizaram a detecção por Interseção de Histograma. No quadro 1, as partículas ainda estão com forte influência da distribuição inicial, por esse motivo podem se encontram tão afastadas do rosto. Também pode ser notado em todos os quadros que o rastreamento com Filtro de Partículas não conseguiu rastrear o rosto com os parâmetros propostos. Ainda podemos ver através dos quadros a perceptível degradação do desempenho quando o ruído foi acrescentado na imagem.

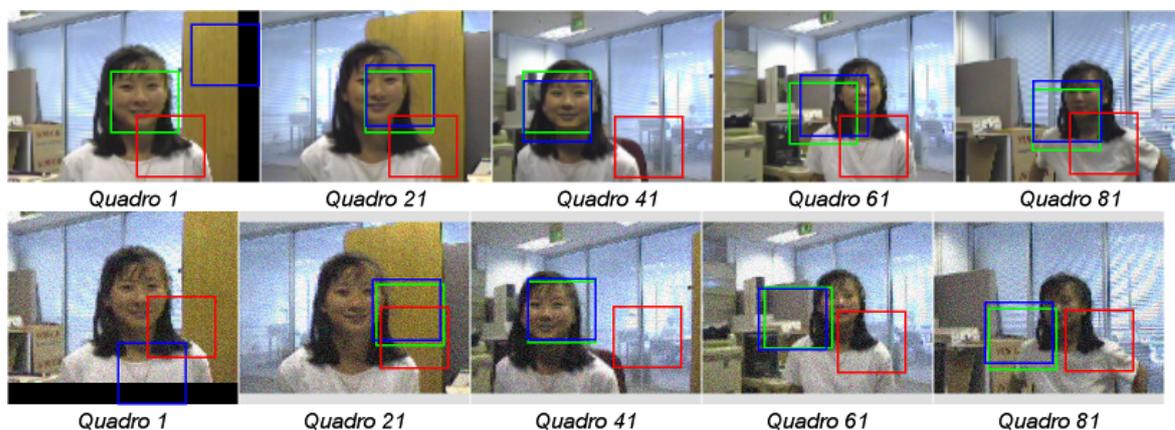


Figura 5.6: Rastreamento em imagens, sem ruído (linha superior) e com ruído (linha inferior), utilizando FP (vermelho), FP+OEP (verde) e FP+ED (azul) com detecção através da Interseção de Histograma

Os quadros da sequência de imagens presentes na Figura 5.7 mostra os algoritmos de rastreamento em funcionamento com o método de detecção por Coeficiente de Correlação. Da mesma forma que ocorreu na imagem anterior, o primeiro quadro tem sua performance prejudicada pela definição de uma posição inicial não muito próxima da real. Utilizando essa forma de detecção com os parâmetros apresentados também não permitiu um rastreamento razoável usando o Filtro de Partículas. A movimentação brusca presente nos quadros anteriores ao qua-

dro 61 fez com que todos os métodos de rastreamento tivessem um resultado muito ruim nesse quadro. Além disso, mais uma vez o ruído causou uma piora perceptível no desempenho de todos os métodos de rastreamento.

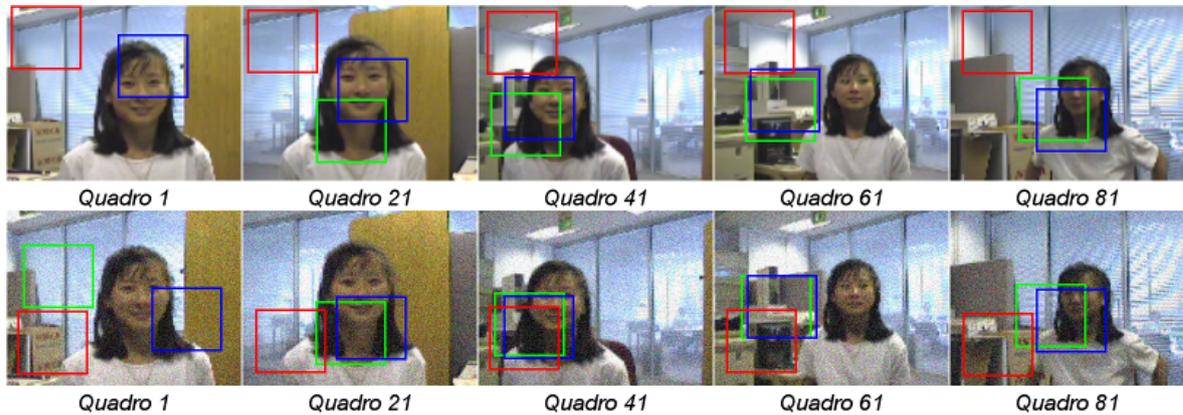


Figura 5.7: Rastreamento em imagens, sem ruído (linha superior) e com ruído (linha inferior), utilizando FP (vermelho), FP+OEP (verde) e FP+ED (azul) com detecção através do Coeficiente de Correlação

Por último, temos a Figura 5.8 que possui os quadros da sequência de imagens com a detecção sendo feita através da Correlação Cruzada Normalizada Híbrida. Pode-se notar que os testes realizados com esse tipo de detecção obtiveram resultados muito ruins, ficando quase que o período todo sem nenhum dos algoritmos conseguindo rastrear o rosto da mulher. O que pode indicar que as condições utilizadas no problema não são adequados para esse tipo de detecção.

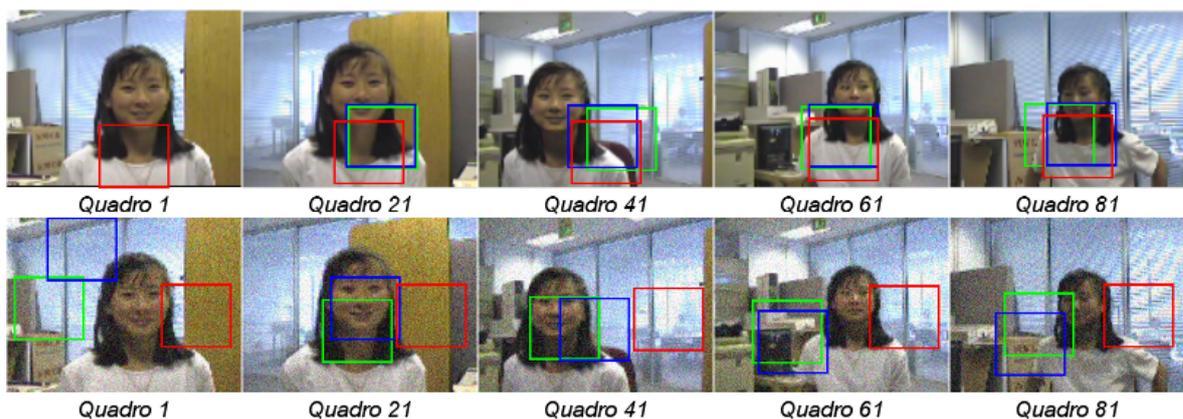


Figura 5.8: Rastreamento em imagens, sem ruído (linha superior) e com ruído (linha inferior), utilizando FP (vermelho), FP+OEP (verde) e FP+ED (azul) com detecção através da Correlação Cruzada Normalizada Híbrida

Para fins da medida de distância em pixels, cada quadro foi marcado onde se localiza, ou onde se supõe que esteja nos casos em que há oclusão, a ponta do nariz da mulher. Desta forma, consideramos o centro do rosto como sendo a ponta do nariz. A partir dessa marcação foi calculado, em cada quadro, a distância entre a posição em que o algoritmo considerou o

centro do rosto e o marcado. Essa distância, em pixels, representa o erro do algoritmo para essa sequência.

A Tabela 5.3 mostra o Somatório do Erro Quadrático !!Somatório Quadrático do Erro!! (SEQ) de todos os quadros da sequência de imagens testados. Nesta tabela fica evidente que o rastreamento com Filtro de Partículas não conseguiu alcançar o objetivo com os parâmetros testados. Outro fato que se destaca é a evidente superioridade da detecção usando Interseção de Histograma. Além disso, os métodos de detecção baseados em uma imagem modelo mostraram sua deficiência ao tratar imagens com ruído.

Tabela 5.3: SEQ médio e mínimo para o FP, FP+OEP e FP+ED com 150 partículas

Algoritmos		FP		FP+OEP		FP+ED	
Detecção	Ruído	Média	Mínimo	Média	Mínimo	Média	Mínimo
IH	Não	3.109.797	987.696	230.724	197.022	195.813	166.988
	Sim	2.052.464	948.826	717.105	653.751	731.427	639.830
CC	Não	2.249.436	880.024	608.161	553.769	562.110	416.307
	Sim	3.250.309	1.409.232	639.659	629.475	538.538	504.243
CCNH	Não	3.008.815	684.750	664.381	592.750	638.692	485.497
	Sim	3.207.483	1.280.753	665.689	649.367	783.240	742.304

Na tabela 5.4 é mostrado a degeneração, que foi medida através de uma aproximação do tamanho efetivo da amostra, definida pela Equação (2.9). Vale destacar que quanto menor o tamanho efetivo, maior a variância dos pesos e, conseqüentemente, a degeneração do conjunto de partículas. A média da degeneração foi bem similar na grande maioria das combinações de algoritmos testados. Um fato que se destaca nessa tabela é que o Filtro de Partículas hibridizado com Evolução Diferencial usando detecção através da Interseção de Histograma tem a segunda pior média de degeneração, mas obteve a melhor média de erro na Tabela 5.3.

Tabela 5.4: Degeneração média e mínima do teste com melhor desempenho para o FP, FP+OEP e FP+ED com 150 partículas

Algoritmos		FP		FP+OEP		FP+ED	
Detecção	Ruído	Média	Menor	Média	Menor	Média	Menor
IH	Não	146,95	137,94	146,35	104,78	143,96	1,00
	Sim	148,91	145,60	145,50	1,00	145,55	1,00
CC	Não	144,88	90,99	146,02	1,00	146,66	140,13
	Sim	148,71	143,18	146,47	137,25	146,80	141,18
CCNH	Não	143,62	4,00	147,75	1,00	147,81	1,00
	Sim	149,32	146,95	149,24	147,42	146,91	105,97

Se analisarmos ainda os mínimos das degenerações na Tabela 5.4, pode ser identificado que os métodos híbridos possuem os piores mínimos. Porém, as médias do tamanho efetivo da

amostra na Figura 5.9, que apresenta o valor da aproximação do tamanho efetivo da amostra ao longo do tempo, mostra que esses mínimos tão baixos são apenas grandes vales. Desta maneira, rapidamente o tamanho efetivo da amostra volta a ficar por volta de valores bem maiores, ou seja, com degeneração menor.



Figura 5.9: Gráfico da medida da degeneração ao longo do tempo do teste com melhor desempenho do Filtro de Partículas com Evolução Diferencial usando Interseção de Histograma

Para se avaliar a influência do número de partículas no desempenho, pode-se analisar a Tabela 5.5. Nela fica evidente que o aumento do número de partículas não melhora o desempenho do algoritmo, exceto no algoritmo Filtro de Partículas. A melhora no Filtro de Partículas pode ser notada principalmente ao se observar os SEQ mínimos.

Tabela 5.5: SEQ médio e mínimo para o FP, FP+OEP e FP+ED com detecção através da Interseção de Histograma

Algoritmos		FP		FP+OEP		FP+ED	
Partículas	Ruído	Média	Mínimo	Média	Mínimo	Média	Mínimo
150	Não	3.109.797	987.696	230.724	197.022	195.813	166.988
	Sim	2.052.464	948.826	717.105	653.751	731.427	639.830
300	Não	1.690.587	316.973	223.152	197.399	211.027	194.860
	Sim	1.823.255	322.946	744.600	653.315	759.645	696.177
600	Não	3.185.971	652.794	227.338	205.829	227.210	206.640
	Sim	2.760.235	235.212	801.509	740.035	792.936	702.227

5.2.4 Discussão

O algoritmo que obteve o melhor desempenho foi o rastreamento com Filtro de Partículas utilizando a detecção através da Interseção de Histograma, com a média da soma quadrática do erro de 195.813 pixels. Apesar do bom desempenho, essa combinação de rastreamento e detecção obteve a pior média de degeneração, com 143,96.

A explicação para isso se deve ao fato de se substituir apenas a pior partícula pela melhor encontrada na otimização, pois a diferença do peso da pior partícula para a média dos pesos pode ser menor que a diferença da nova partícula, que pode ter um peso muito grande, logo uma diferença grande. Para contornar esse problema pode-se tentar substituir um número maior de partículas ruins.

Mesmo com degeneração piorando, o desempenho do algoritmo se mostrou muito melhor com a hibridização. O rastreamento com o Filtro de Partículas sequer conseguiu rastrear o rosto na sequência de imagens com a quantidade de partículas investigada.

Como as partículas iniciais provavelmente não são muito boas, normalmente quando as partículas sofrem a primeira otimização através do método híbrido a partícula substituída possui um peso muito maior que as demais. Desta forma, no segundo quadro temos um grau de degeneração severo, mas que logo retorna a um patamar aceitável.

O aumento no número de partículas não necessariamente melhora o desempenho dos algoritmos. O limite da eficiência do método de detecção foi alcançado para os métodos de rastreamento Filtro de Partículas hibridizado com Evolução Diferencial e Filtro de Partículas hibridizado com Otimização através de Enxame de Partículas, portanto não se obtém uma melhora no desempenho ao se aumentar o número de partículas. Entretanto, o método Filtro de Partículas possui um desempenho melhor ao se aumentar o número de partículas, pois para esse método não se havia atingido o limite da eficiência do método de detecção.

6 Conclusões

Neste trabalho, foi proposto uma alteração no Filtro de Partículas usando algoritmos da computação natural: i) Otimização através de Enxame de Partículas; e ii) Evolução Diferencial, para diminuir o erro de rastreamento. A abordagem está focada em substituir as partículas que possuem baixa qualidade por uma melhor encontrada pelo algoritmo de otimização.

Inicialmente se pretendia diminuir também a degeneração, mas os testes mostraram que a degeneração não necessariamente reduz com essa abordagem. Uma vez que a partícula otimizada em muitos casos possui uma diferença da média dos pesos das partículas maior, em módulo, que a diferença do pior peso com a média. Esse fato ocorre, por exemplo, nas primeiras iterações do Filtro de Partículas nos problemas em que a escolha da posição inicial não foi muito boa, pois todas as partículas tendem a ter uma importância muito pequena, como se pode detectar no gráfico da medida de degeneração com o tempo.

Nas simulações, uma melhoria foi observada nas versões modificadas do Filtro de Partículas, como mostrado nos resultados obtidos nos dois estudos de caso. A implementação de um Filtro de Partículas hibridizado permite, por exemplo, tratar um problema com um número bem menor de partículas e obter resultados similares aos de um com o Filtro de Partículas padrão com mais partículas.

Entretanto, ao comparar o tempo de execução do rastreamento das três versões estudadas com um mesmo número de partículas sempre se obteve tempos melhores com o Filtro de Partículas padrão. Esse fato pode ser explicado facilmente pela ausência do passo de hibridização, que em série aumenta o tempo computacional consideravelmente em relação ao Filtro de Partículas.

Os métodos de detecção utilizados não alcançaram um resultado satisfatório quando testados em uma imagem ruidosa. Isso indica a dificuldade que os algoritmos de detecção baseados em imagem modelo (*template matching*) possuem para esse tipo de problema.

Para trabalhos futuros, pode-se experimentar aumentar o número de partículas a serem substituídas para tentar manter a degeneração baixa. Além disso, há espaço para reduzir o tempo

computacional paralelizando o Filtro de Partículas (HENDEBY; KARLSSON; GUSTAFSSON, 2010), a Evolução Diferencial (VERONESE; KROHLING, 2010) e a Otimização através de Enxame de Partículas (VERONESE; KROHLING, 2009).

Há interesse também em uma análise utilizando outras combinações de hibridizações de algoritmos, como híbridos usando modificações da Evolução Diferencial (DAS; SUGANTHAN, 2011) e da Otimização através de Enxame de Partículas (SONG; GU, 2004; VALLE et al., 2008) não testadas neste trabalho.

Como o desempenho do rastreamento está intimamente ligado ao desempenho da detecção, análise e testes de outros algoritmos de detecção, como em Viola e Jones (2004), em conjunto com os rastreamentos propostos neste trabalho se mostram interessantes para estudos posteriores.

Referências Bibliográficas

- AIUBE, F. *Modelagem dos Preços Futuros de Commodities: Abordagem pelo Filtro de Partículas*. Tese (Doutorado) — Pontifícia Universidade Católica do Rio de Janeiro, 2005.
- ARULAMPALAM, M. S. et al. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, v. 50, p. 174–188, 2002.
- BAYES, M.; PRICE, M. An essay towards solving a problem in the doctrine of chances. By the Late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S. *Philosophical Transactions*, v. 53, p. 370–418, 1763.
- BREST, J. et al. Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE Transactions on Evolutionary Computation*, v. 10, n. 6, p. 646–657, nov. 2006.
- CAPPE, O.; GODSILL, S. J.; MOULINES, E. An overview of existing methods and recent advances in sequential monte carlo. *Proceedings of the IEEE*, v. 95, n. 5, p. 899–924, jul. 2007.
- CASTRO, L. de. Fundamentals of natural computing: an overview. *Physics of Life Reviews*, v. 4, n. 1, p. 1–36, mar. 2007.
- CRISAN, D.; DOUCET, A. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, v. 50, n. 3, p. 736–746, mar 2002.
- DAS, S.; KONAR, A.; CHAKRABORTY, U. Improved differential evolution algorithms for handling noisy optimization problems. *IEEE Congress on Evolutionary Computation*, v. 2, p. 1691–1698, 2005.
- DAS, S.; SUGANTHAN, P. Differential evolution: A survey of the state-of-the-art. *IEEE Transactions on Evolutionary Computation*, v. 15, n. 1, p. 4–31, fev. 2011.
- DOUCET, A.; FREITAS, N. de; GORDON, N. *Sequential Monte Carlo Methods in Practice*. Berlin: Springer-Verlag, 2001.
- EBERHART, R.; SHI, Y. *Computational Intelligence: Concepts to Implementations*. [S.l.]: Morgan Kaufman, 2007.
- EIBEN, A. E.; SMITH, J. E. *Introduction to Evolutionary Computing*. Berlin: Springer-Verlog, 2008. (Natural Computing).
- ENGELBRECHT, A. P. *Fundamentals of Computational Swarm Intelligence*. [S.l.]: John Wiley and Sons, 2006.
- GORDON, N. J.; SALMOND, D. J.; SMITH, A. F. M. Novel approach to nonlinear/non-gaussian bayesian state estimation. In: *IEE Proceedings F: Radar and Signal Processing*. [S.l.: s.n.], 1993. v. 140, n. 2, p. 107–113.

- HENDEBY, G.; KARLSSON, R.; GUSTAFSSON, F. Particle filtering: The need for speed. *EURASIP Journal on Advances in Signal Processing*, Hindawi, v. 2010, p. 1–10, 2010.
- KAO, E.; MAASDAM, P. van; SHEPPARD, J. Image-based tracking with particle swarms and probabilistic data association. In: *Proceedings of the IEEE Swarm Intelligence Symposium*. Saint Louis, MO: [s.n.], 2008. p. 1–8.
- KARI, L.; ROZENBERG, G. The many facets of natural computing. *Commun. ACM*, ACM, New York, NY, USA, v. 51, n. 10, p. 72–83, 2008.
- KENNEDY, J. Bare bones particle swarms. In: *Proceedings of the IEEE Swarm Intelligence Symposium*. Piscataway, NJ, USA: [s.n.], 2003. p. 80–87.
- KENNEDY, J.; EBERHART, R. Particle swarm optimization. In: *Proceedings of the IEEE International Conference on Neural Networks*. Piscataway, NJ: IEEE Press, 1995. v. 4, p. 1942–1948.
- KIYUNA, A. et al. Simulation study of multitrailer control system using neurocontrollers evolved by a modified genetic algorithm. *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, v. 3, p. 1364–1368, 2003.
- KONG, A.; LIU, J. S.; WONG, W. H. Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Association*, v. 89, n. 425, p. 278–288, mar. 1994.
- KROHLING, R. A. Gaussian particle swarm and particle filter for nonlinear state estimation. In: *Proceedings of the International Conference on Artificial Intelligence and Soft Computing*. Spain: [s.n.], 2004. p. 399–404.
- KROHLING, R. A. Gaussian swarm: a novel particle swarm optimization algorithm. In: *Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems*. Singapore: [s.n.], 2004. v. 1, p. 372–376.
- KROHLING, R. A. Comunicado privado. *PPGI/UFES*, 2010.
- LI, H.; DUAN, H.-B.; ZHANG, X.-Y. A novel image template matching based on particle filtering optimization. *Pattern Recognition Letters*, Elsevier Science Inc., New York, NY, USA, v. 31, p. 1825–1832, October 2010.
- LIMA, L. M. de; KROHLING, R. A. Particle filter with differential evolution for trajectory tracking. In: GASPAR-CUNHA, A. et al. (Ed.). *Soft Computing in Industrial Applications*. Berlin: Springer-Verlag, 2011, (Advances in Intelligent and Soft Computing, v. 96). p. 209–219.
- LOZANO, O. M.; OTSUKA, K. Real-time visual tracker by stream processing. *Journal of Signal Processing Systems*, Kluwer Academic Publishers, Hingham, MA, USA, v. 57, n. 2, p. 285–295, November 2009.
- MENDEL, E.; KROHLING, R. A.; CAMPOS, M. Swarm algorithms with chaotic jumps applied to noisy optimization problems. *Information Sciences*, v. 181, p. 4494–4514, 2011.
- PARK, S. et al. A new evolutionary particle filter for the prevention of sample impoverishment. *IEEE Transactions on Evolutionary Computation*, v. 13, p. 801–809, 2009.

- RAHNAMAYAN, S.; TIZHOOSH, H. R.; SALAMA, M. M. A. Opposition-based differential evolution. *IEEE Transactions on Evolutionary Computation*, v. 12, p. 64–79, 2008.
- SHI, Y.; EBERHART, R. A modified particle swarm optimizer. In: *Proceedings of IEEE International Conference on Evolutionary Computation*. [S.l.: s.n.], 1998. p. 69–73.
- SIDDIQUI, M. N.; YOUSAF, B. Bare bones strategy for human detection and tracking. In: *Proceedings of the IEEE Symposium on Computational Intelligence in Image and Signal Processing*. Honolulu, HI, USA: [s.n.], 2007. p. 30–35.
- SIMON, D. Kalman filtering for fuzzy discrete time dynamic systems. *Applied Soft Computing*, v. 3, n. 3, p. 191–207, 2003.
- SMITH, J. *Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression*. Tese (Doutorado) — Columbia University, 1997.
- SONG, M.-P.; GU, G.-C. Research on particle swarm optimization: a review. In: *Proceedings of 2004 International Conference on Machine Learning and Cybernetics*. [S.l.: s.n.], 2004. v. 4, p. 2236–2241.
- STORN, R.; PRICE, K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, Kluwer Academic Publishers, Hingham, MA, USA, v. 11, n. 4, p. 341–359, 1997.
- SUSSMAN, M.; WRIGHT, G. The correlation coefficient technique for pattern matching. In: *Proceedings of the 7th Scientific Meeting and Exhibition of the International Society for Magnetic Resonance in Medicine*. Philadelphia, Pennsylvania, USA: [s.n.], 1999.
- SWAIN, M. J.; BALLARD, D. H. Color indexing. *International Journal of Computer Vision*, v. 7, p. 11–32, 1991.
- UOSAKI, K.; KIMURA, Y.; HATANAKA, T. Nonlinear state estimation by evolution strategies based particle filters. *IEEE Congress on Evolutionary Computation*, v. 3, p. 2102–2109, 2003.
- VALLE, Y. del et al. Particle swarm optimization: Basic concepts, variants and applications in power systems. *IEEE Transactions on Evolutionary Computation*, v. 12, n. 2, p. 171–195, april 2008.
- VERONESE, L. P.; KROHLING, R. A. Swarm’s flight: Accelerating the particles using c-cuda. *Congress on Evolutionary Computation*, p. 3264–3270, 2009.
- VERONESE, L. P.; KROHLING, R. A. Differential evolution algorithm on the gpu with c-cuda. *IEEE Congress on Evolutionary Computation*, p. 1–7, 2010.
- VIDAL, F. Rastreamento visual de objetos utilizando otimização por enxame de partículas. *Journal of the Brazilian Neural Network Society*, v. 9, p. 52–65, 2011.
- VIOLA, P.; JONES, M. J. Robust real-time face detection. *International Journal of Computer Vision*, Kluwer Academic Publishers, Hingham, MA, USA, v. 57, p. 137–154, May 2004.
- WHITLEY, D. An overview of evolutionary algorithms: practical issues and common pitfalls. *Information and Software Technology*, v. 43, n. 14, p. 817–831, 2001.

- XIAO, L.; LI, P. Improvement on mean shift based tracking using second-order information. In: *Proceedings of the 19th International Conference on Pattern Recognition*. Tampa, USA: [s.n.], 2008. p. 1–4.
- XU, B.; ZHU, J.; XU, H. An ant stochastic decision based particle filter and its convergence. *Signal Processing*, Elsevier, v. 90, n. 8, p. 2731–2748, 2010.
- ZHANG, G.-Y. et al. Design of an adaptive particle filter based on variance reduction technique. *Acta Automatica Sinica*, v. 36, p. 1020–1024, 2010.
- ZHANG, X. et al. Sequential particle swarm optimization for visual tracking. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. USA: [s.n.], 2008. p. 1–8.
- ZHOU, Q.; MA, L.; CHELBERG, D. Adaptive object detection and recognition based on a feedback strategy. *Image Vision Computing*, Butterworth-Heinemann, Newton, MA, USA, v. 24, p. 80–93, January 2006.
- ZOTKIN, D. N.; DURAISWAMI, R.; DAVIS, L. S. Joint audio-visual tracking using particle filters. *EURASIP Journal on Applied Signal Processing*, Hindawi Publishing Corp., v. 2002, n. 1, p. 1154–1164, 2002.