

4. Estudo de Casos

4.1. Configuração

Neste capítulo, comparam-se os resultados e o desempenho obtidos pela PGLIQ com a extensão do modelo proposto GPU-PGLIQ-I que foi desenvolvido nesta dissertação. Apresentam-se dois casos de regressão simbólica que foram analisados anteriormente como *benchmarks* pela PGLIQ [Douglas 2010]. Também é apresentado um caso de estudo de otimização no caminhar de um robô hexapode, com a finalidade de avaliar seu desempenho em aplicações de robótica.

Para o estudo de caso de regressão simbólica, o valor da aptidão de um indivíduo é determinado pela computação do seu erro absoluto médio (MAE), que mede o quão próximas as previsões estão em comparação aos resultados padrão. A métrica MAE é dada pela equação:

$$MAE = \frac{1}{n} \sum_{i=1}^n |t_i - V[0]_i| \quad (4.1)$$

Na qual, n é a quantidade de amostras, t_i o valor do resultado padrão para a amostra i e $V[0]_i$ é a saída do melhor indivíduo. A configuração de parâmetros da PGLIQ é apresentada na Tabela 2.

| Parâmetro | Valor (PGLIQ) |
|--|-------------------|
| Número de gerações (G) | 300.000 a 350.000 |
| Tamanho da população (M) | 6 |
| Número máximo de gerações sem melhoria (gsmmáximo) | 20000 |
| Probabilidade inicial de NOP (p0;0) | 0,9 |
| Tamanho do passo de reinício (sr) | 1,0 |
| Tamanho do passo do operador P (s) | 0,004 |
| Comprimento máximo do programa | 128 |
| Número de registradores da FPU | 8 |

Tabela 2. Configuração do modelo PGLIQ para os experimentos

Para cada experimento foram realizadas 100 repetições. A GPU-PGLIQ-I tem a mesma configuração, mas o número máximo de gerações sem melhoria antes de realizar a etapa de *reset* é de 6,000 gerações. O número de indivíduos avaliados por execução em um algoritmo evolutivo é, geralmente, calculado de maneira trivial, através do produto do número de indivíduos da população (M) pelo número de gerações (G). Para realização deste estudo, foi utilizada uma GPU Tesla C2070, a qual possui 448 elementos de processamento (1.15 GHz) e 6 GB de memória RAM com largura de banda de acesso à memória de 144 GB/segundo.

O computador utilizado possui quatro destas GPUs e um processador Intel(R) Xeon(R), trabalhando na frequência de 3.47 GHz, com 12 GB de memória RAM.

4.2. Distância Euclidiana

O primeiro caso representa um problema com seis variáveis de entrada, conforme mostra a equação:

$$f_{dist}(a_x, b_x, a_y, b_y, a_z, b_z) = \sqrt{(a_x - b_x)^2 + (a_y - b_y)^2 + (a_z - b_z)^2} \quad (4.2)$$

Cada um dos conjuntos de casos de aptidão (treinamento e validação) é composto por 300 amostras, onde os valores das coordenadas dos pontos ($\in \mathfrak{R}^3$) são aleatórios e se situam no intervalo [0; 1], com distribuição uniforme.

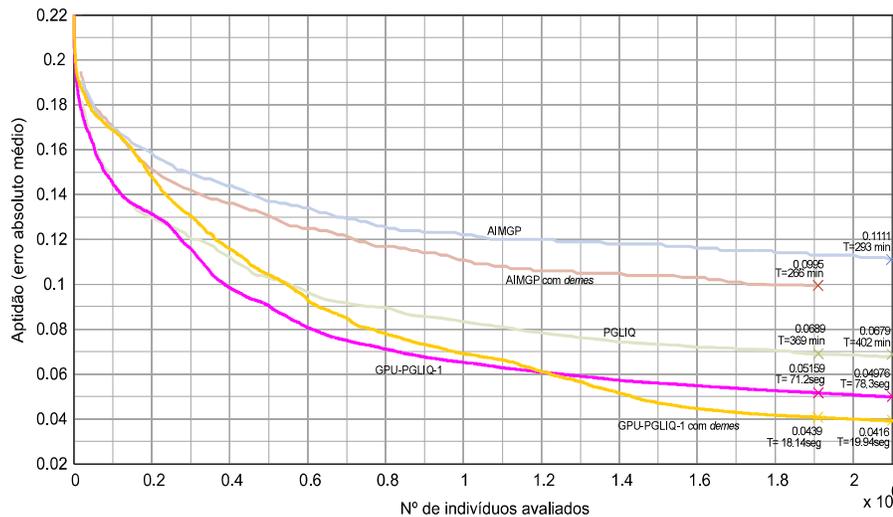


Figura 31. Gráfico evolutivo do estudo de caso “Distância Euclideana” em função do tempo médio de cada execução dos modelos.

Este estudo de caso utiliza um conjunto de funções reduzido, composto pelas seguintes instruções (vide Tabela 2): FADD, FSUB, FMUL, FXCH, FABS e FSQRT. A Figura 31 mostra as curvas evolutivas referentes à média do melhor indivíduo por geração para os modelos AIMGP, com e sem uso de *demes*, PGLIQ, simulados anteriormente por [2], e a GPU-PGLIQ-I com e sem uso de *demes*. Toma-se como referência para a comparação o número de indivíduos avaliados pelos modelos. A Tabela 3 e Figura 32 concentram os resultados obtidos a partir dos dados ilustrados pela Figura 31, apresentando também o desvio padrão σ relativo às aptidões dos melhores indivíduos por geração, para as 100 execuções de cada modelo.

| Demes | Aptidão | | | | | |
|-------|---------|----------|--------|----------|-------------|----------|
| | AIMGP | | PGLIQ | | GPU-PGLIQ-I | |
| | Media | Σ | Media | σ | Media | σ |
| Não | 0.1111 | 0.049 | 0.0679 | 0.0471 | 0.0497 | 0.0147 |
| Sim | 0.0995 | 0.0495 | 0.0689 | 0.0476 | 0.0416 | 0.0256 |

Tabela 3. Comparação de média e desvio padrão das aptidões dos melhores indivíduos por geração para o estudo de caso “Distância Euclideana”.

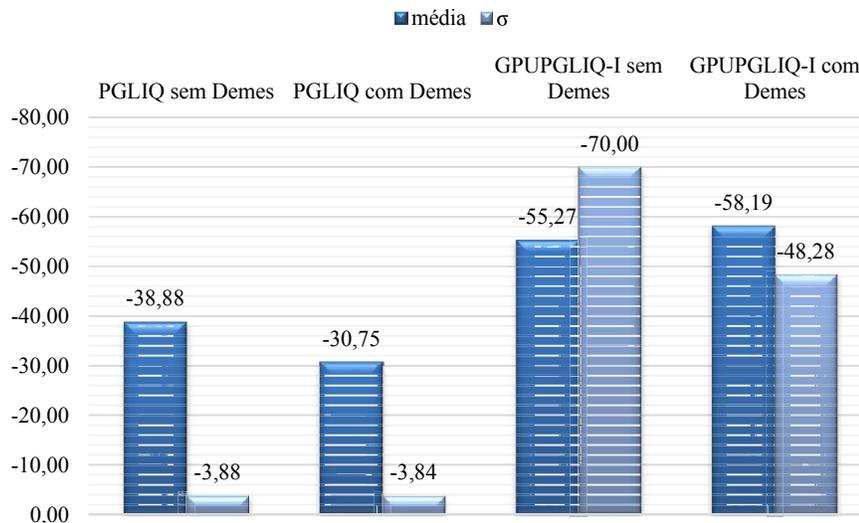


Figura 32. Variação em relação à aptidão AIMGP em porcentagem”.

A Tabela 4 mostra, para os três modelos, o número de indivíduos que, dentre as 100 execuções, representam a solução exata do problema ($erro < 10^{-6}$). O GPU-PGLIQ-I mostra uma quantidade menor de programas que são a solução exata para resolver o caso, mas na utilização de *demes* melhora consideravelmente a eficácia, além de tornar o algoritmo mais paralelizável, reduzindo assim o tempo execução no *cluster*.

| Demes | Número de acertos | | |
|-------|-------------------|-------|-------------|
| | AIMGP | PGLIQ | GPU-PGLIQ-I |
| Não | 2 | 17 | 1 |
| Sim | 3 | 17 | 10 |

Tabela 4. Comparação do número de acertos obtidos no estudo de caso “Distância Euclideana”.

A Tabela 5 mostra a quantidade de acertos, a aptidão e o tempo de execução médio em 100 repetições.

| Demes | GPU-PGLIQ-I | | | | |
|---------|-------------|--------|--------|--------|--------|
| | Não | 2 | 4 | 6 | 8 |
| Tempo | 78.3s | 39.36s | 19.94s | 33.33s | 25.09s |
| Média | 0.0497 | 0.0468 | 0.0416 | 0.0452 | 0.0430 |
| # | 0.0147 | 0.0112 | 0.0256 | 0.0391 | 0.0404 |
| acertos | 1 | 1 | 10 | 20 | 36 |

Tabela 5. Comparação da análise com *demes* obtidos no estudo de caso “Distância Euclideana”.

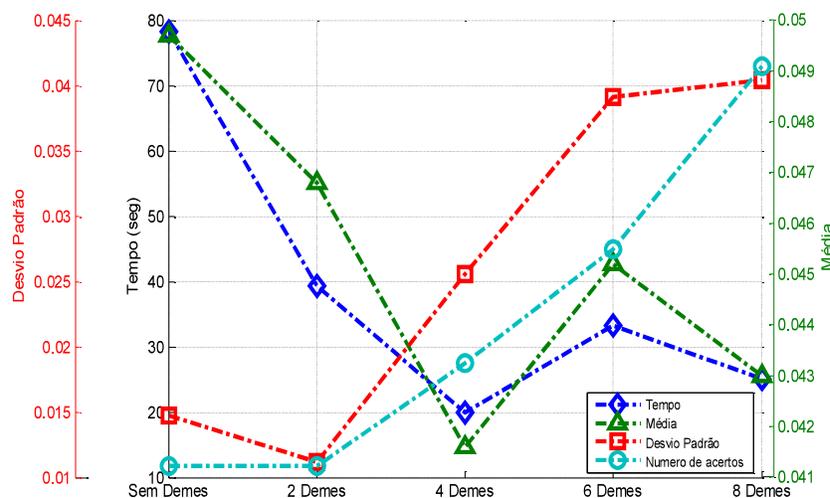


Figura 33. Comparação Gráfica de desempenhos obtidos pelos *demes* no estudo de caso “Distância Euclideana”

O número de acertos aumenta diretamente conforme a quantidade de *demes* aumenta. As análises com 6 e 8 *demes* mostram altos números de acertos embora o valor do erro médio seja maior do que os outros casos e o desvio padrão aumente conforme aumenta o número de *demes*. Os resultados serão comparados com os resultados do caso de análise na seção 4.1.3. O melhor resultado foi apresentado pela GPU-PGLIQ-I com 4 *demes*, considerando o tempo de execução, o valor final médio de aptidão e um desvio padrão relativamente baixo.

4.2.1.

Avaliação de Desempenho:

Tendo como ponto de referência os tempos de execução de, são comparados os tempos de processamento na Tabela 6:

| Avaliações | PGLIQ | PGLIQ (Cluster) | GPU-PGLIQ-I | | |
|------------------|--------|--------------------|------------------------|------------------------|-------------------------|
| | | | Sem demes / SpeedUp | Com demes / SpeedUp | Eficiencia Demes (%) |
| 1.910.000 | 369min | 1560seg | 71.2seg / 21.9 | 18.2seg / 85.7 | 98.12 |
| 2.100.000 | 402min | 1704seg | 78.3seg / 21.7 | 19.9seg / 85.63 | 98.16 |

Tabela 6. Comparação de Tempos para o caso de Estudo de Distância Euclidiana

Os tempos de execução da GPU-PGLIQ-I com *demes* é quase inversamente proporcional ao número de *demes* (para 2 e 4 *demes*). A transmissão de indivíduos entre *demes* e a transmissão de informação com o computador local para a visualização gráfica são processos que diminuem a eficiência. Para o caso de estudo, a eficiência de paralelizar com *demes*, a GPU-PGLIQ-I é sobre 98%. Para mais de 4 *demes*, a eficiência diminui principalmente pelas seguintes razões:

- O número de *demes* excede à quantidade de placas gráficas em um computador. Desta forma, em algumas (ou todas as) placas tem que se rodar dois ou mais algoritmos de PG e, ao se aumentar o número de computadores, o MPI precisa de comunicação LAN para a transmissão de indivíduos.
- O aumento de *demes* demanda mais tempo de sincronização entre os processadores e de espera até que todos os *demes* tenham recebido o indivíduo do *deme* adjacente.

4.3. Chapéu Mexicano

O segundo caso apresenta um problema com duas variáveis de entrada, o qual tem como tarefa a reconstrução de uma superfície a partir de um dado conjunto de pontos. A superfície é dada pela função bidimensional definida pela equação:

$$f_{chapmex}(x, y) = \left(1 - \frac{x^2}{4} - \frac{y^2}{4}\right) \times e^{\left(-\frac{x^2}{8} - \frac{y^2}{8}\right)} \quad (3.3)$$

As variáveis x e y são amostradas uniformemente no intervalo $[-4,4]$ para gerar o conjunto de treinamento e validação. A Figura 25 mostra as curvas evolutivas referentes à média do melhor indivíduo por geração, mas considerando como referência para a comparação o número de indivíduos avaliados pelos modelos.

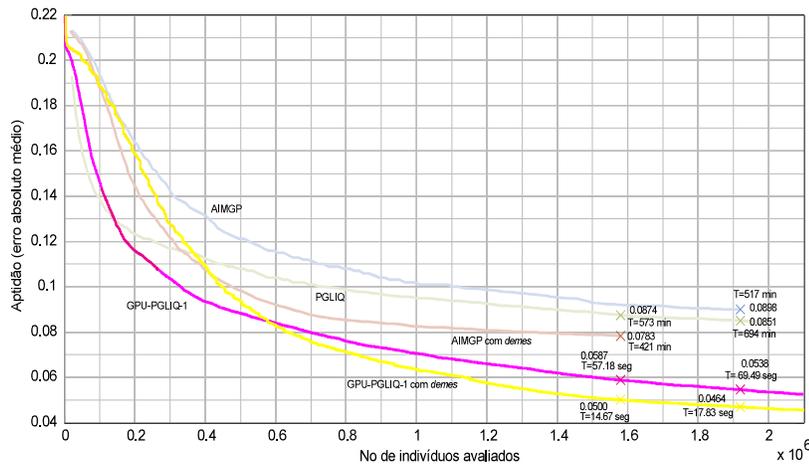


Figura 34. Gráfico evolutivo do estudo de caso “Chapéu Mexicano” em função do tempo médio de cada execução dos modelos.

A Tabela 7 concentra os resultados obtidos a partir dos dados ilustrados pela Figura 34, apresentando também o desvio padrão σ relativo às aptidões dos melhores indivíduos por geração. Quanto às aptidões dos melhores indivíduos encontrados pelos modelos, e considerando-se os dois critérios de comparação

(com e sem *demes*), o desempenho do GPU-PGLIQ-1 é superior à PGLIQ padrão e a AIMGP.

| Demes | Aptidão | | | | | |
|-------|---------|----------|--------|----------|------------|----------|
| | AIMGP | | PGLIQ | | GPUPGLIQ-I | |
| | Média | σ | Média | Σ | Média | σ |
| Não | 0.0898 | 0.049 | 0.0848 | 0.0248 | 0.0538 | 0.0267 |
| Sim | 0.0783 | 0.0495 | 0.0874 | 0.0246 | 0.05 | 0.023 |

Tabela 7. Comparação de média e desvio padrão das aptidões dos melhores indivíduos por geração para o estudo de caso “Chapéu Mexicano” e Variação em Relação.

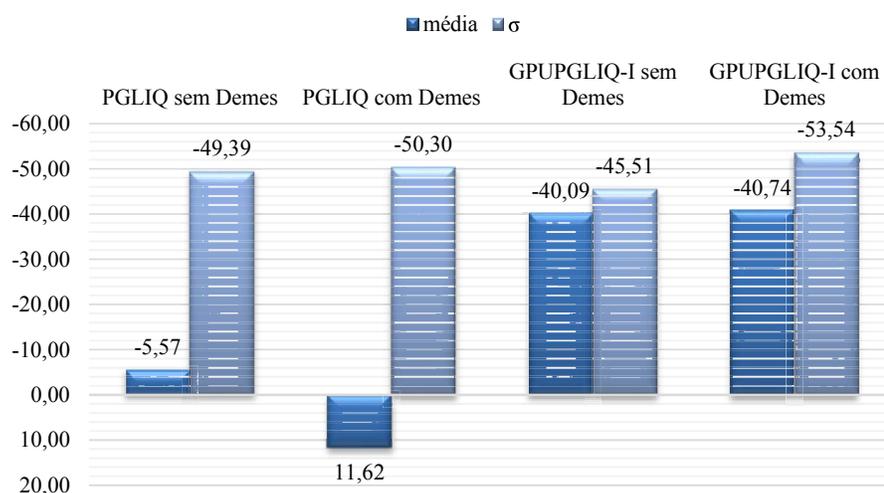


Figura 35. Variação em relação à aptidão AIMGP em porcentagem.

A Figura 35 mostra o desempenho da PGLIQ e a técnica proposta em relação à aptidão do algoritmo AIMGP. A aptidão da GPU-PGLIQ-I tem uma porcentagem baixa, sendo o erro muito menor em comparação a AIMGP e PGLIQ, além de ter uma variância pequena, de forma que em várias repetições as soluções são parecidas. A Figura 36 mostra as comparações do tempo, média e variância.

| Demes | GPU-PGLIQ-I | | | | |
|----------|-------------|--------|--------|---------|---------|
| | Não | 2 | 4 | 6 | 8 |
| Tempo | 69.49s | 34.42s | 17.83s | 29.963s | 22.571s |
| Média | 0.0538 | 0.0682 | 0.0464 | 0.0500 | 0.0573 |
| Σ | 0.0267 | 0.0277 | 0.023 | 0.0391 | 0.0210 |

Tabela 8. Comparação do número de acertos obtidos no estudo de caso “Chapéu Mexicano”.

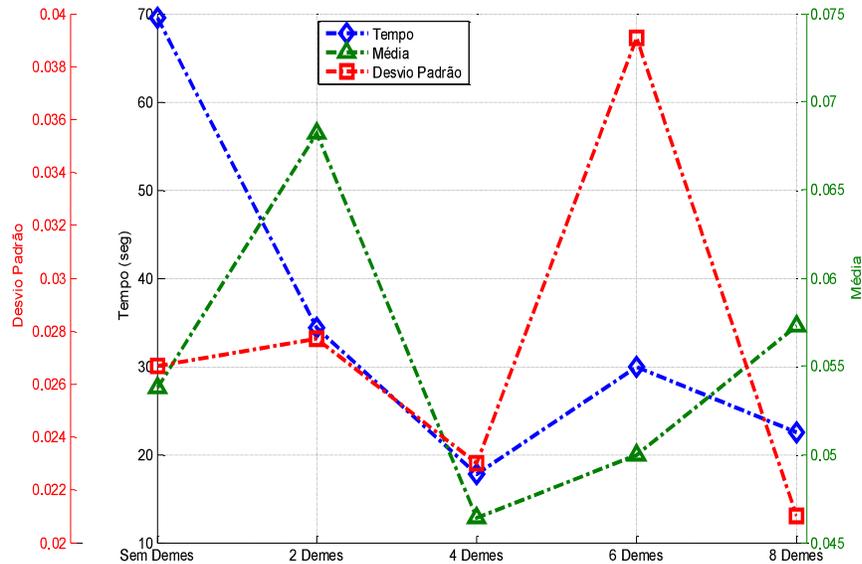


Figura 36. Comparação Gráfica de desempenho obtido pelos *demes* no estudo de caso “Chapéu Mexicano”.

Igual ao estudo de caso de distância euclidiana, o algoritmo é mais eficaz quando trabalha com 4 *demes* considerando o intervalo de análise de 2 a 8 *demes*.

4.3.1. Avaliação de Desempenho

Os tempos de execução em [2] são comparados com os resultados desta pesquisa na Tabela 9:

| Avaliações | PGLIQ | PGLIQ (Cluster) | GPU-PGLIQ-I | | |
|------------|--------|-----------------|---------------------|---------------------|----------------------|
| | | | Sem demes / SpeedUp | Com demes / SpeedUp | Eficiência demes (%) |
| 1.590.000 | 573min | 1299seg | 57.18seg / 22.71 | 14.67seg / 88.54 | 97.44 |
| 1.920.000 | 694min | 1582seg | 69.49seg / 22.77 | 17.83seg / 88.72 | 97.43 |

Tabela 9. Comparação de Tempos para o caso de Estudo de Chapéu Mexicano

A GPU-PGLIQ-I tem um *SpeedUp* 88.54 vezes mais rápido que o algoritmo padrão para a mesma quantidade de avaliações. A eficiência do algoritmo é alta de

modo que, em geral, para problemas de regressão simbólica, a utilização de *demes* é muito eficiente. É importante indicar que os tempos de execução foram calculados a partir da média de 100 experimentos e os processadores do *cluster* são dedicados exclusivamente a um *deme* e uma placa gráfica.

4.4. Estudo de Caso: Robótica Evolutiva

Neste estudo de caso, tem-se como finalidade a análise da GPU-PGLIQ-I em aplicações em robótica evolutiva. Para seu desenvolvimento, utilizou-se a biblioteca *Bullet* SDK para simular o comportamento do hexápode em um ambiente no qual o robô tem como finalidade a aprendizagem do caminhar. Esta análise foi escolhida para avaliar a topologia de paralelismo com multiníveis apresentada na seção 3.1.4, que é utilizada no caso em que o cálculo da aptidão do indivíduo tem que ser processado fora do GPU. Para a etapa de avaliação, o hexápode está em um ambiente simulado onde aprenderá a caminhar.

4.4.1. Configuração

Neste estudo de caso, tem-se um robô hexápode, criado em um ambiente de simulação dinâmica (BULLET). A Figura 37 mostra o robô no ambiente do simulador.

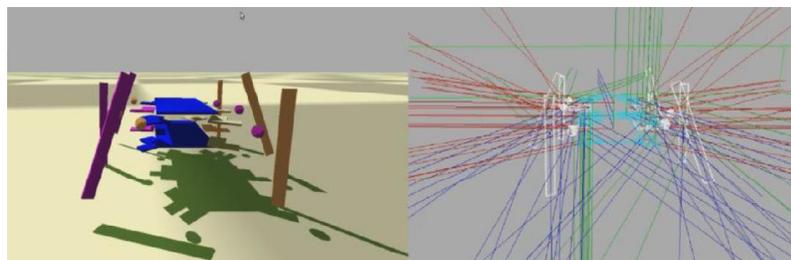
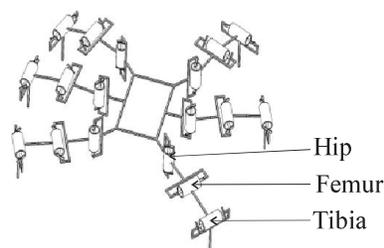


Figura 37. Robô Hexápode no ambiente de simulação BULLET

As configurações do Robô hexápode no ambiente de simulado BULLET são as seguintes:

- O robô tem 18 graus de liberdade.
- Todas as juntas são rotativas.
- A faixa de rotação para cada junta é de $[-0.3, +0.3]$ rad.
- As colisões são definidas só entre cada peça do hexápode e o chão.
- O cálculo da função aptidão é definido como a distância entre o ponto inicial (P_o) e o ponto final (P_f) do robô na simulação.

$$fitness = - \frac{\left| \left\| \overrightarrow{P_o P_f} \right\|^{3/2} \cdot \left\| \overrightarrow{N_{HMax}} \right\| \right|}{\left\| \overrightarrow{P_o P_f} \times \overrightarrow{u_F} \right\| \cdot \left\| \overrightarrow{N_{HMax}} \times \overrightarrow{u_{Np}} \right\|}$$

Where:

- P_o : Ponto Inicial do Robô Hexapode.
- P_f : Ponto Final do Robô Hexapode.
- u_F : Vetor Unitário para a frente.
- u_{Np} : Vetor Unitário Normal (chão de referencia).
- N_{HMax} : Máxima inclinação do Vetor Normal Unitário do Robô Hexápode.

Os parâmetros de entrada a GPU-PGLIQ-I são:

- **Id da perna:**

Cada perna do hexápode tem 3 juntas rotativas. O Id faz referência a uma das 6 pernas do hexápode.

- **Tempo:**

Dado que o tempo de simulação é discreto, uma das entradas é um sinal tipo rampa em intervalos de 0.01.

- **Direção da perna:**

Com valor +1 se for direita e -1 se for esquerda.

- Os parâmetros de saída da GPU-PGLIQ-I são 18 trajetórias que são as referências de cada junta.
- O tempo de simulação é de 10 segundos, os quais são realizados em *batch* pelo simulador.
- A criação do trajeto será em malha aberta.

A programação genética evolui uma sequência de rotações para cada articulação. Assim, as séries geradas são as referências que cada junta rotativa tem para virar ao longo do tempo.

Para o desenvolvimento deste estudo de caso, foram utilizados os *clusters* do Laboratório de Inteligência Computacional (ICA). Entre os testes realizados, a melhor configuração de parâmetros da programação genética foi utilizando 4 *demes* com uma população de 6 indivíduos. De acordo com a topologia de paralelismo multinível proposta na seção 3.1.4, foram utilizados 25 processadores, 24 CPU com MPI e 2 processadores com memória compartilhada utilizando openMP para transmissão de dados da visualização, além das 4 placas gráficas NVIDIA, nas quais são executados os operadores da programação genética. Ao mesmo tempo, os dados são transferidos por protocolo TCP/IP para um computador local onde é possível visualizar os resultados em tempo real. A Figura 38 mostra um exemplo desta topologia, no qual utiliza 5 *demes* com 9 indivíduos, cada deme é processado por uma placa GPU, no gráfico é explicado os *kernels* que são executados na GPU.

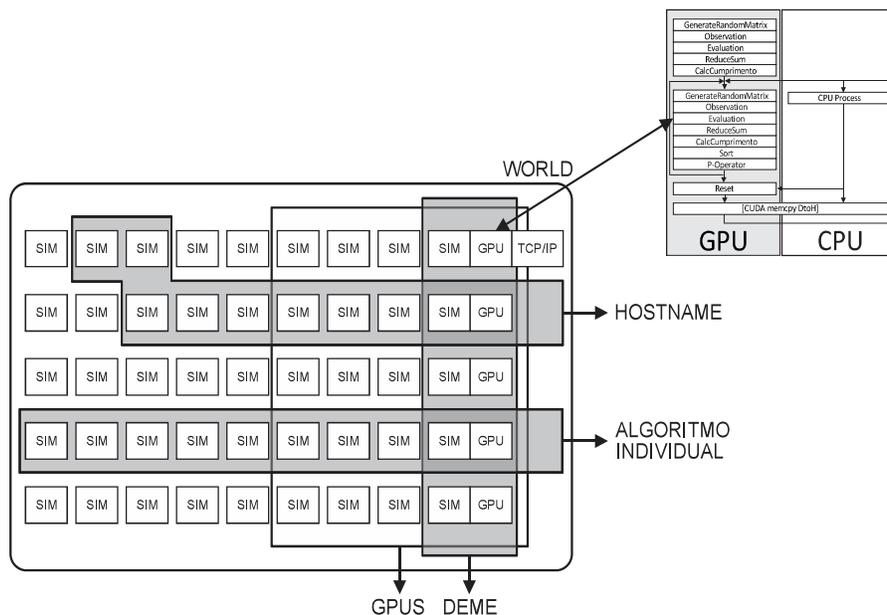


Figura 38. Paralelismo Heterogêneo da PGLIQ para evolução do Robô Hexápode.

A Figura 39 mostra a estrutura que foi implementada para a evolução do robô hexápode.

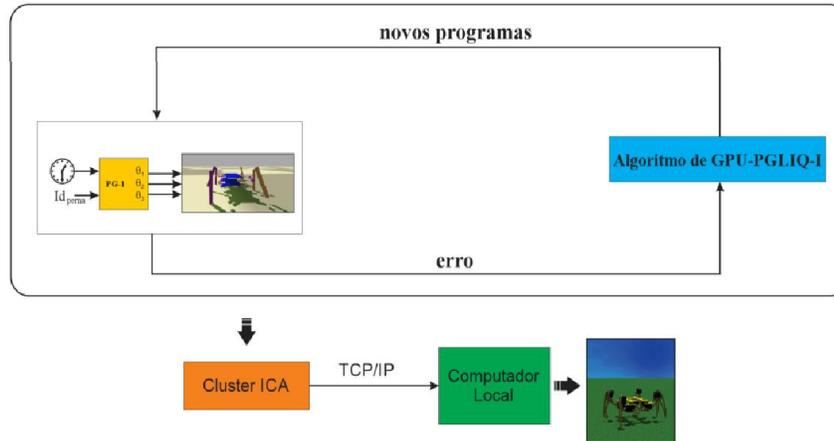
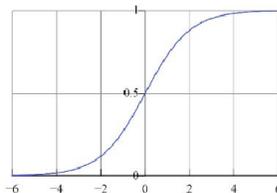


Figura 39. Sistema de evolução do hexápode

Dado que o valor de saída da programação genética são todos os reais, é necessária uma conversão dos valores de saída da programação genética antes de entrar nas juntas do hexápode. Uma função sigmóide foi utilizada como transdutor entre a saída do registro V0 na execução do programa criado pela programação genética e o ângulo de entrada em cada junta do hexápode no simulador local.



$$Out (Simulador) = \frac{1}{1 + e^{-Input(PGLIQ)}}$$

Figura 40. Função Sigmóide

4.4.2. Evolução da caminhada do robô.

A Figura 41 apresenta o resultado da evolução do hexápode em função do número de avaliações. Dos testes realizados, é possível diferenciar dois casos: o primeiro, quando o hexápode chegou a coordenar o movimento das articulações e avançar, atingindo nesta etapa uma distância máxima de 29 cm. No segundo caso, o robô hexápode evoluiu até dar pequenos pulos dando a aparência de correr; nesta etapa o robô começa atingir distâncias maiores.

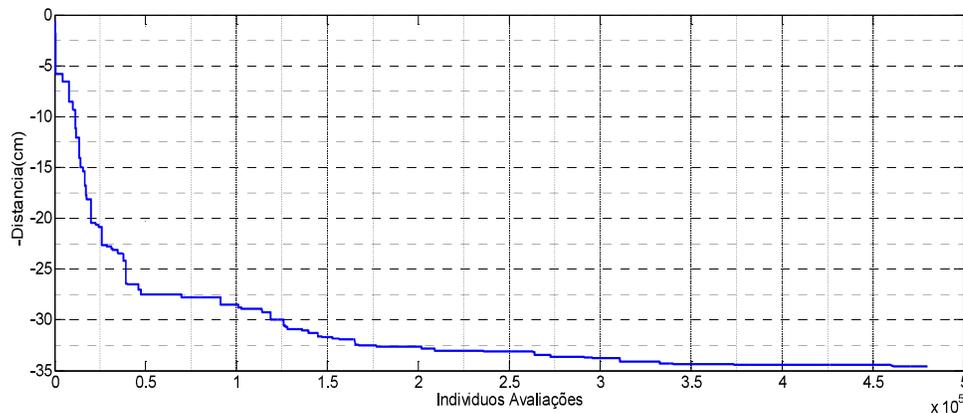


Figura 41. Gráfico evolutivo do estudo de caso “Robótica Evolutiva” em função do número médio de indivíduos avaliados por execução (caso 2).

Então, no primeiro caso, pode-se dizer que ele aprende a caminhar, e no segundo caso o hexápode aprender a “correr”. Na Figura 42, é apresentado o salto que realiza o hexápode na última etapa da evolução. A diferença na configuração do primeiro e segundo caso é que no primeiro é utilizado restrição na velocidade das juntas. É importante indicar que as faixas de rotações do motor são limitadas. Conforme sejam colocadas mais restrições e com o aumento da frequência de simulação, BULLET melhorar a precisão dos resultados, mas requer um maior tempo para sua simulação.

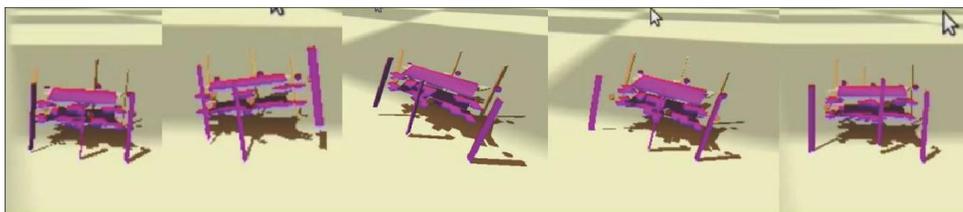


Figura 42. Salto realizado pelo hexápode

Neste estudo de caso, cada simulação do robô demora em média 0.35 segundos. Foram realizadas várias análises utilizando até 84 processadores distribuídos entre a quantidade de *Demes* e Indivíduos. Estes processadores utilizam a topologia híbrida OpenMP/MPI + GPU proposta na Figura 26, na qual como exemplo apresenta uma matriz de 8 *demes* x 10 indivíduos, utilizando um total de 80 processadores.

Na Tabela 10 e Tabela 11, são apresentados os resultados para os dois melhores resultados. O primeiro com a maior distância atingida de 32 cm utilizando 24

processadores e 52500 avaliações; neste caso o robô hexápode não tem restrição na velocidade das juntas e tem uma eficiência do uso de *demes* de 97%. O segundo caso utiliza 84 processadores, um tempo de processamento de 10 min (o menor de todas as análises), atingindo uma eficiência do paralelismo do uso de *demes* de 98.71% e é o melhor resultado quando é utilizado a restrição de velocidade nas juntas do robô hexápode. Na Tabela 10 e Tabela 11 η_{Demes} representa a eficiência do uso de *demes*.

| | <i>Sem Demes</i> | | <i>Demes</i> | |
|----------------------------------|------------------|--------------|--------------|--------------|
| | <i>Demes</i> | <i>Indv.</i> | <i>Demes</i> | <i>Indv.</i> |
| # CPUs | 1 | 6 | 4 | 6 |
| Distância | 26 cm | | 27 cm | 32 cm |
| Tempo | 35 min | | 8.5 min | 34 min |
| η_{Demes} | --- | | 97% | |
| # Avaliação | 52500 | | | 210000 |

Tabela 10. Tabela de Resultados de experimentos utilizando 24 CPUs.

| | <i>Sem Demes</i> | | <i>Demes</i> | |
|----------------------------------|------------------|--------------|--------------|--------------|
| | <i>Demes</i> | <i>Indv.</i> | <i>Demes</i> | <i>Indv.</i> |
| # CPUs | 1 | 6 | 14 | 6 |
| Distância | 28 cm | | 29 cm | |
| Tempo | 138.2 min | | 10 min | |
| η_{Demes} | --- | | 98.71% | |
| #Avaliação | 210000 | | | |

Tabela 11. Resultados de experimentos utilizando o número máximo de CPUs (84).

Um renderizado foi utilizado para visualizar o comportamento do melhor indivíduo ao longo das gerações. A Figura 43 mostra os *frames* da caminhada do robô em 5 segundos de simulação. A sequência está ordenada de cima para baixo e da esquerda para a direita.

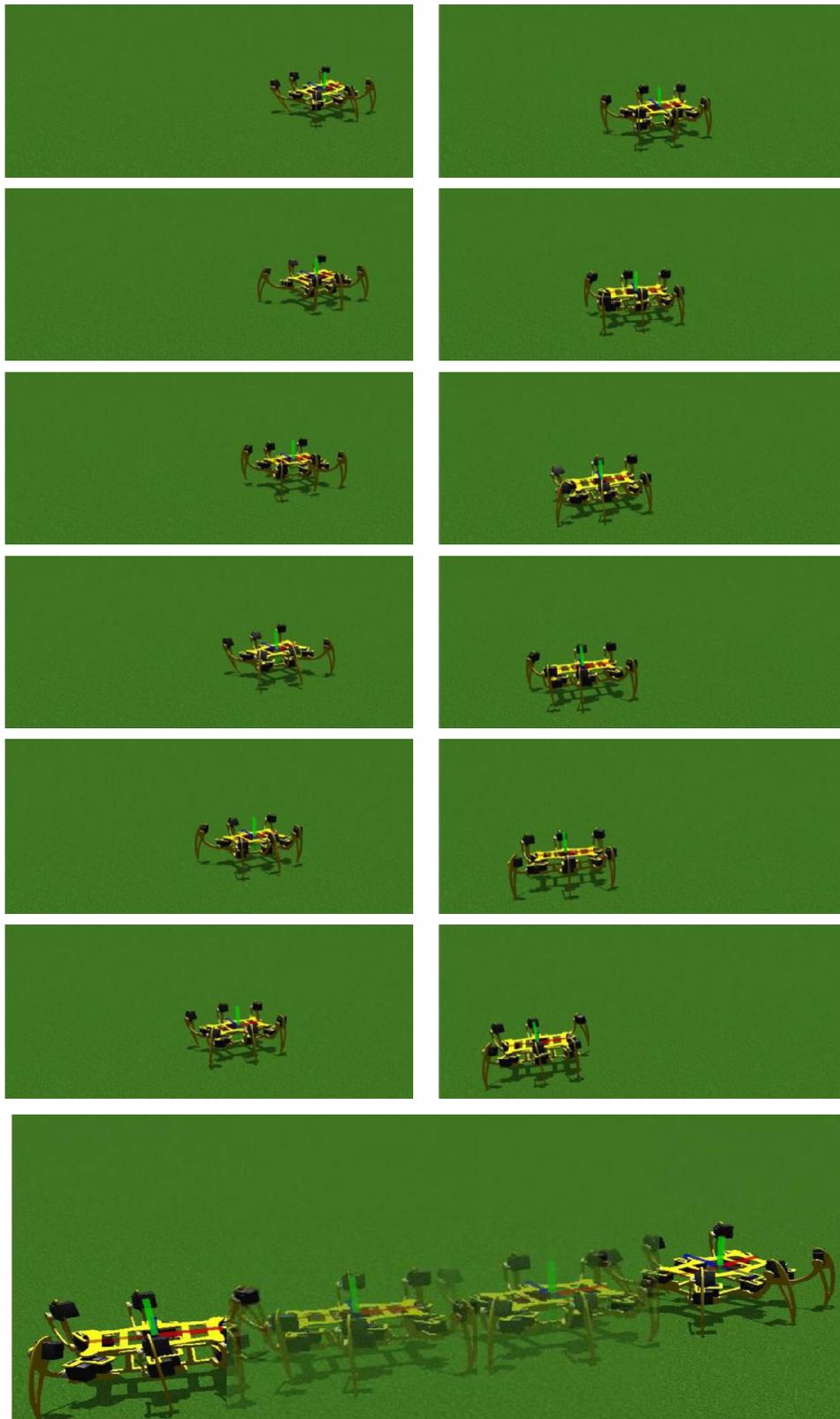


Figura 43. Caminhar robô hexápode.

A Figura 44 mostra as 18 trajetórias das juntas do hexápode com o melhor (caso 2) resultado. Assim, cada coluna é uma das 6 pernas do hexápode e cada fila a posição do motor na perna: Hip, Femur e Tibia.