

5. Conclusões e Trabalhos Futuros

5.1. Conclusões

O paralelismo com GPU atingiu speedup em torno de 22 sem *demes* e 85 utilizando 4 *demes* comparados com os tempos de referência calculados na PGLIQ. Assim, melhora-se a eficiência da PGLIQ ao longo de seu processo. Conseguiu-se transferir quase 100% a PGLIQ dentro do GPU para casos de regressão simbólica, permitindo uma aceleração considerável.

A eficiência da paralelização com *demes* é de quase 98%. Esta eficiência se apresenta para pouca quantidade de *demes* (máximo 4), na qual o tempo de sincronização entre as diferentes populações é pequeno.

A extensão da PGLIQ proposta apresenta a média de resultados melhores com uma variância baixa no erro final. Embora no estudo de caso de distância euclidiana não se consiga atingir muitas vezes a solução exata, com ajuda dos *demes* isto é solucionado. Isto se deve ao fato de que a técnica proposta armazena o melhor resultado que tem até a última geração, a qual já tem um erro e tenta resolver o problema com esse erro presente. Apesar de maior dificuldade para encontrar uma solução exata, ajuda na aproximação da solução. Contudo, pode-se concluir que a extensão proposta melhora a eficácia da PGLIQ padrão com ajuda de *demes* e consegue em média soluções muito mais próximas e similares.

As técnicas de visualização gráfica permitem entender o que acontece dentro da PGLIQ na etapa de *reset*, além de mostrar a dinâmica da evolução dentro do Indivíduo Quântico. Também permitem visualizar as novas estruturas que são criadas pela extensão proposta.

A Topologia com paralelismo multinível proposta (MPI-OpenMP-GPUs) para processamento dentro do *cluster*, além de trabalhar com a PGLIQ, também é compatível com qualquer tipo de algoritmo evolutivo que utilize os seguintes processos: *demes*; avaliação de indivíduos em paralelo e processamento do núcleo central em GPU; e transferência de resultados em tempo real a um computador

remoto para o *cluster*. Esta topologia pode ajudar a muitos pesquisadores a paralelizar seus algoritmos e executá-los no *cluster* de uma forma mais eficiente.

O caso de robótica evolutiva ajudou a representar os benefícios de utilizar sistemas híbridos quando o sistema tem diferentes processos, alguns assíncronos e alguns agrupados apenas para trabalhar entre eles (como um *deme*), reduzindo o tempo de processamento consideravelmente até 15 minutos. A PGLIQ resolveu satisfatoriamente o estudo de caso de robótica aplicada. O algoritmo agora, junto com as plataformas de simulação e as topologias paralelas multiníveis, formam um ponto de partida para novas pesquisas e desafios mais complexos, com base na utilização eficiente dos recursos de alto desempenho.

5.2.

Trabalhos Futuros

No desenvolvimento desta dissertação, encontram-se alguns pontos os quais podem fornecer início de novas pesquisas.

Podem-se aproveitar as características de Programação Linear da PGLIQ para criar uma PG com entidades matriciais, a qual utiliza matrizes de registros em lugar de um só registro, permitindo, trabalhar com múltiplas saídas e com operadores matriciais, incrementando a dinâmica da evolução e tornando-o mais eficiente. Esta abordagem se encaixa perfeitamente no conceito de programação dinâmica dentro da arquitetura Kepler, a qual fornece vários níveis de paralelismo dentro dos operadores da PGLIQ.

Pode-se sintetizar ainda o algoritmo de GPU-PGLIQ para placas gráficas AMD e NVIDIA, além de paralelizar utilizando *Cloud Computing*. Além disso, pode-se desenvolver uma biblioteca que enrole a topologia proposta na seção 3.1.4, na qual varios algoritmos evolutivos possam explorar os recursos de alto desempenho aproveitando o potencial dos sistemas distribuídos em futuras pesquisas.