



Um Mapeamento da Literatura sobre o uso de Metodologias Ativas na Construção do Pensamento Computacional Paralelo

Flávio Ramon Almeida de Souza, Instituto de Ciências Exatas e Naturais/UFPA, souramon@hotmail.com, <https://orcid.org/0000-0002-8022-2468>

Marcos Rodrigo Mendes Saavedra, Instituto de Ciências Exatas e Naturais/UFPA, marcosaavedraa3@gmail.com, <https://orcid.org/0000-0002-4698-6202>

Josivaldo de Souza Araújo, Instituto de Ciências Exatas e Naturais/UFPA, josivaldo@ufpa.br, <https://orcid.org/0000-0001-6890-6923>

Resumo: Apesar da programação paralela ter se tornado uma realidade consolidada nos grandes centros de pesquisa, ainda se apresenta em formação nas matrizes curriculares dos cursos de graduação em Computação. A necessidade de uma formação básica e comum para os alunos desenvolverem as habilidades e competências específicas dessa área é fundamental, não apenas devido à popularização da computação de alto desempenho, mas também pela demanda, cada vez mais crescente, de um mercado promissor. Apesar dessa necessidade, a construção do Pensamento Computacional Paralelo ainda é realizada na forma de aulas tradicionais dentro das Universidades. O objetivo deste trabalho é identificar, através de um mapeamento da literatura, estudos que utilizem metodologias ativas para o ensino e a construção do Pensamento Computacional Paralelo, abordando recursos, desafios e métodos que estimulem alunos e professores no processo de ensino e aprendizado da construção do Pensamento Paralelo. Dos estudos retornados, 8 relatam experiências que vão das aulas tradicionais à utilização de jogos.

Palavras-chave: Pensamento Paralelo, Metodologias Ativas, Programação Paralela.

A Mapping of Literature on the use of Active Methodologies in the Construction of Parallel Computational Thinking

Abstract: Although parallel programming has become a consolidated reality in large research centers, it still presents itself in training in the curricular matrices of undergraduate courses in Computing. The need for a basic and common training for students to develop the specific skills and competencies of this area is fundamental, not only due to the popularization of high-performance computing, but also due to the increasing demand for a promising market. Despite this need, the construction of Parallel Computational Thinking is still carried out in the form of traditional classes within the Universities. The objective of this work is to identify, through a mapping of the literature, studies that use active methodologies for the teaching and construction of addressing resources, challenges and methods that stimulate students and teachers in the process of teaching and learning the construction of Parallel Thinking. Of the studies returned, 8 report experiences ranging from traditional classes to the use of games.

Keywords: Parallel Thinking, Active Methodologies, Parallel Programming.

1. Introdução

A Computação de Alto Desempenho (CAD) ganhou visibilidade nas últimas décadas devido as diferentes áreas e, conseqüentemente, problemas que consegue solucionar. A

popularização de arquiteturas *multicore* e *manycore* fizeram com que a construção de códigos paralelos saísse dos grandes centros de pesquisa e chegasse ao processo de ensino e aprendizagem nas Universidades, mais precisamente, nas disciplinas dos cursos de graduação em Computação, com o nome de Programação ou Computação Paralela. E esse avanço não está relacionado apenas com o ensino de uma área que está em constante evolução, mas também, na necessidade de se formar uma mão de obra altamente qualificada, através da teoria que envolve a CAD, e do aprendizado na elaboração de códigos paralelos, através de suas várias metodologias e bibliotecas paralelas (FENWICK, NORRIS 2020).

O ensino da Computação Paralela envolve conceitos que resultam no desenvolvimento de aptidões práticas (DURAES, *et.al.* 2022). Há uma preocupação crescente com as habilidades e competência que devem ser desenvolvidas. Essa preocupação se reflete tanto nos eixos de formação dos cursos de graduação em Computação, quanto nas matrizes curriculares da área, de forma nacional, como na SBC (Sociedade Brasileira de Computação) (ZORZO, *et al.* 2017), e internacional, na ACM (*Association for Computing Machinery*) e no IEEE (*Institute of Electrical and Electronics Engineers*) (ACM, IEEE-CS, 2020).

Existe um consenso, entre alguns autores, de que o conteúdo da Programação Paralela deve ser distribuído ao longo da matriz curricular, pois isso possibilitaria ao aluno construir e definir não apenas o chamado Pensamento Paralelo de uma forma contínua, mas também visualizar esses conceitos de uma forma natural e comum do ensino, do mesmo modo como acontece com a programação chamada de sequencial, e não como um conteúdo avançado, de difícil compreensão e distante da realidade (ALMEIDA JR., *et. al.* 2022). Nas fases iniciais de aprendizagem, os alunos naturalmente, conseguem identificar os aspectos paralelos dos problemas, permitindo o desenvolvimento de códigos otimizados e focado no Pensamento Paralelo (DURAES, *et. al.* 2020).

O objetivo deste trabalho é apresentar um mapeamento sobre as metodologias ativas desenvolvidas para a construção do Pensamento Computacional Paralelo, e que podem ser utilizadas no processo de ensino e aprendizagem da Programação Paralela, como também estimular o uso desses e de outros métodos de forma a despertar o interesse de alunos e professores na área da Computação de Alto Desempenho. Para isso, foi realizado um Mapeamento Sistemático da Literatura (MSL) onde são apontadas as principais metodologias, ferramentas, arquiteturas e processos utilizados.

2. Trabalhos Relacionados

Apesar do avanço da Programação Paralela nos cursos de graduação e pós-graduação, ainda se percebe poucos trabalhos relatando experiências sobre metodologias utilizadas para a construção do pensamento paralelo no processo de ensino e aprendizagem desta disciplina. A introdução dos conceitos do chamado Pensamento Computacional Paralelo ainda se mostram confusos e dispersos em trabalhos publicados pela academia.

No estudo desenvolvido por (ALMEIDA JR., *et. al.* 2022) é apresentada uma pesquisa realizada com vários professores, de diferentes instituições de ensino superior, que trabalham ou já trabalharam com a disciplina de Programação Paralela em cursos de graduação ou pós-graduação. O trabalho discute, entre outros assuntos, as metodologias utilizadas, os desafios enfrentados por docentes e discentes durante o processo de ensino e aprendizagem da Programação Paralela. Entre os principais desafios apontados estão a ausência de disciplinas introdutórias do assunto, e o uso de aulas tradicionais, o que acaba desestimulando, e muitas vezes, dificultando a construção o Pensamento Paralelo.



Em outro trabalho de (ALMEIDA JR., *et. al.* 2021) relata um mapeamento da literatura relacionado ao ensino e aprendizado da Programação Paralela utilizando, especificamente, o uso de placas gráficas. O trabalho destaca a carência de estudos utilizando esse tipo de arquitetura e a necessidade de se instituir uma base para o chamado Pensamento Paralelo, assim como destaca as metodologias utilizadas no processo de ensino e aprendizagem utilizando esse tipo de arquitetura. Os autores relatam que de um total de 445 estudos sobre a Programação Paralela, apenas 2 foram selecionados, e ainda não descreviam, de forma clara, as bases para se chegar ao Pensamento Paralelo.

No estudo de (DURAES, *et. al.* 2020) também é realizado um mapeamento da literatura em um período de 20 anos, entre 1989 e 2019. Foram retornados 819 trabalhos, destes apenas 94 foram incluídos na pesquisa pelos critérios estabelecidos. Os autores destacam que a metodologia de ensino tradicional ainda é a mais utilizada, e isso acaba desmotivando o interesse dos alunos pela área. Isso pode ser percebido pelo baixo número de alunos matriculados em disciplinas e cursos de Programação Paralela.

3. O Pensamento Computacional e o Pensamento Paralelo

Nos últimos anos, muito se tem discutido sobre as dificuldades da construção do chamado Pensamento Paralelo. Professores e pesquisadores da área da CAD veem com preocupação o baixo interesse de alunos pela área da Programação Paralela. Talvez isso possa ser explicado pelas dificuldades encontradas por esses discentes em disciplinas iniciais, como Algoritmos e Programação, que acabam sendo responsáveis por grandes índices de evasão dos cursos de graduação em Computação (ALMEIDA JR., *et. al.* 2022).

Com o objetivo de reduzir essa evasão e auxiliar o processo de aprendizagem, reduzindo as dificuldades, uma habilidade vem ganhando a atenção dos pesquisadores, é o chamado Pensamento Computacional (PC). O PC se baseia em Lógica de Programação e outros conceitos da Computação como abstração, decomposição e avaliação, visando não apenas a resolução de problemas, mas também a formulação destes (WING, 2006). Nas referências de formação para os cursos de graduação em Computação da Sociedade Brasileira de Computação (SBC) (ZORZO, *et. al.*, 2017), o Pensamento Computacional aparece como uma competência esperada (SILVA, FALCÃO, 2020).

Os fundamentos do Pensamento Computacional já estão sendo utilizados, não apenas em disciplinas da educação básica ou do ensino fundamental, mas também nas disciplinas do ensino superior. Em um estudo apresentado por (SILVA, FALCÃO, 2020) são discutidos os impactos na aprendizagem do uso do PC na Programação em cursos de Licenciatura em Computação. Dessa forma, pode-se utilizar o PC como uma abordagem que venha contribuir para a superação de dificuldades no ensino e aprendizagem da Programação Paralela, e conseqüentemente, na formação do Pensamento Computacional Paralelo.

3.1. Metodologias Ativas na Construção do Pensamento Computacional Paralelo.

Assim como o Pensamento Computacional pode ser utilizado como uma abordagem para o ensino da Programação dita sequencial, o Pensamento Paralelo também, pode ser construído através das bases do PC, empregando metodologias que possam ir além das aulas tradicionais, e que estimulem alunos e professores no processo de ensino e aprendizagem da Programação Paralela (MOTA, NEVES, 2020).

Existem metodologias que podem ter abordagens teóricas, como por exemplo, a sala de aula invertida (BACHIEGA, *et. al.* 2017), a computação desplugada (Lopes, *et. al.*, 2020), a gamificação, aprendizagem baseada em problemas (SANTOS JR., *et. al.* 2020), entre outros; como também, as abordagens mais práticas, como o aprendizado colaborativo na prática e a aprendizagem baseada em projetos práticos.



4. Mapeamento Sistemático da Literatura

Um Mapeamento Sistemático da Literatura (MSL) é um meio de identificar, avaliar e interpretar pesquisas relevantes para uma determinada questão de pesquisa, área ou fenômeno de interesse. Também ajuda a identificar lacunas na área em que foi realizado, possibilitando assim, uma melhor abordagem para novas atividades de pesquisa (ALMEIDA JR., *et. al.* 2022).

Para isso, são realizados levantamentos de estudos primários, seguidos por uma seleção dos trabalhos retornados através de critérios de inclusão e exclusão previamente definidos. As questões de pesquisas, os critérios de seleção, bem como outras etapas necessárias são definidos no chamado protocolo de pesquisa. O objetivo deste MSL é apresentar um mapeamento sobre as metodologias que podem ser utilizadas no ensino da programação paralela na construção do chamado Pensamento Computacional Paralelo, apontando métodos, arquiteturas e dificuldades apontadas (ALMEIDA JR. *et al.* 2021).

4.1. Questões de Pesquisa

A questão de pesquisa (QP) tem por objetivo auxiliar no processo de pesquisa e na extração dos dados. Para este trabalho foi definida a seguinte questão de pesquisa:

- **(QP): Quais foram as metodologias ativas utilizadas no processo da construção do Pensamento Paralelo?**

Para auxiliar na resposta da questão principal, foram definidas as seguintes questões secundárias (QS):

- **(QS.1) Quais foram as ferramentas (bibliotecas, linguagens, frameworks) utilizadas para dar suporte à construção do Pensamento Paralelo no processo de ensino-aprendizagem?**
- **(QS.2) Quais foram as principais arquiteturas (CPU, GPU, Híbridas) utilizadas na construção do Pensamento Paralelo?**
- **(QS.3) Quais foram as percepções identificadas sobre o Pensamento Paralelo?**
- **(QS.4) Quais foram as principais dificuldades relatadas por professores e alunos na utilização do Pensamento Paralelo durante o uso das metodologias ativas?**

Essas questões foram definidas visando identificar o cenário em que o Pensamento Paralelo é utilizado durante o ensino da Programação Paralela.

4.2. Processo de Busca e Seleção dos Estudos Primários

O processo de busca dos trabalhos foi realizado utilizando a *string* de busca definida no Quadro 1. Foram considerados apenas estudos completos publicados em periódicos ou conferências, escritos em inglês ou português.

Após a finalização do processo de busca, foram aplicados os Critérios de Inclusão e Exclusão para realizar a seleção dos estudos retornados. Os critérios de Inclusão utilizados no trabalho podem ser visualizados no Quadro 2, enquanto os critérios de Exclusão podem ser visualizados no Quadro 3. Dos 28 trabalhos retornados, após a aplicação dos critérios de exclusão e inclusão, foram incluídos e lidos na sua integralidade apenas 8 estudos.

Para auxiliar na execução e seleção do trabalho, foi utilizada a ferramenta Parsifal. Os artigos foram analisados por dois alunos, um de doutorado e outro da graduação, sendo que os desempates eram realizados pelo professor orientador.

**Quadro 1 – String de Busca Utilizada.**

Idioma	String
Inglês	(“parallel thinking” AND “active methodologies”) AND (“approach” OR “tool” OR “methodology” OR “model” OR “technique”) AND (“learning” OR “teaching”) AND (“parallel” AND (“programming” OR “codes”))

Na Tabela 1, são apresentadas as fontes utilizadas como base para a aplicação da *string* de busca, a quantidade de artigos retornados por base consultada, assim como a quantidade de trabalhos excluídos por cada critério e a quantidade de trabalhos adicionados, também, por cada critério.

Quadro 2 – Critérios de Inclusão

Critérios de Inclusão (CI)
CI.1. Estudos primários que abordem técnicas, ferramentas, e/ou tecnologias educacionais sobre o pensamento paralelo no ensino da programação paralela.
CI.2. Estudos que descrevam metodologias ativas empregadas ao ensino e aprendizagem da programação paralela que sejam apoiadas no contexto do pensamento paralelo.

Quadro 3 – Critérios de Exclusão

Critérios de Exclusão (CE)
CE.1. Estudos que não estejam disponíveis livremente para consulta ou download (em versão completa).
CE.2. Estudos que claramente não atendam as questões de pesquisa;
CE.3. Estudos que não sejam artigos completos, enquadrados como resumos, <i>keynote speeches</i> , cursos, tutoriais, <i>workshops</i> e afins;
CE.4. Estudos que estejam fora do período estipulado para a pesquisa.
CE.5. Estudos que não estiverem escritos na língua inglesa ou portuguesa.
CE.6. Estudos Secundários.

A pesquisa foi realizada levando em consideração os últimos 10 anos, pois foi considerado como ponto de partida, a base para a formação da matriz curricular definida pela ACM/IEEE em 2013 que, inicialmente, era apenas para os cursos de graduação em Ciência da Computação, mas atualmente, considera todos os cursos de Computação. As diretrizes propostas destacam as áreas de conhecimento, e entre elas, está a Programação Paralela. Dessa forma, foram classificados os trabalhos publicados de janeiro de 2013 a dezembro de 2022.

O ensino da Programação Paralela e Distribuída aparece, também, em diferentes disciplinas nas diretrizes elaboradas pela Sociedade Brasileira em Computação (SBC), que através das Referências de Formação para os Cursos de Graduação em Computação (ZORZO *et. al.* 2017) define, além dos principais eixos de formação, também, as competências e habilidades esperadas, bem como os conteúdos específicos a serem desenvolvidos nos cursos de Computação no Brasil.

5. Resultados e Discussões

Nesta seção, serão apresentadas as respostas para as questões de pesquisa e questões secundárias definidas na elaboração do protocolo deste mapeamento sistemático. As respostas



serão baseadas no número de identificação (ID) dos estudos primários (EP), que estão listados e podem ser visualizados no Quadro 4.

Tabela 1 – Fontes e Quantidade de Trabalhos por Critério Utilizado.

Fontes	Estudos retornado	Remoção Duplicado	Excluídos 1ª Seleção	2ª Seleção							
				Excluídos						Incluídos	
				CE1	CE2	CE3	CE4	CE5	CE6	CI1	CI2
<i>ACM</i>	5	3	2	0	1	1	0	0	0	0	1
<i>Compendex</i>	2	2	0	0	0	0	0	0	0	0	0
<i>IEEE</i>	0	0	0	0	0	0	0	0	0	0	0
<i>Science Direct</i>	0	0	0	0	0	0	0	0	0	0	0
<i>Scopus</i>	21	11	2	0	0	0	0	0	2	4	3
<i>Web of Science</i>	0	0	0	0	0	0	0	0	0	0	0
TOTAL	28	16	4	0	1	1	0	0	2	4	4
				4						8	

(QP): Quais foram as metodologias ativas utilizadas no processo da construção do Pensamento Paralelo?

Foram apontadas diferentes metodologias para a construção do Pensamento Paralelo, entre elas estão: **Uso de Jogo [EP.1]**, onde um jogo foi desenvolvido com o objetivo de facilitar o ensino das abstrações que envolvem a programação paralela, e isso era realizado através de blocos de montagens. Com os desafios propostos era possível apresentar várias soluções para o mesmo problema, visualizando as vantagens e desvantagens de cada solução de forma animada. **Módulos de Ensino [EP.2]**, foram criados três módulos com diferentes conteúdos de forma a introduzir os conceitos de Programação Paralela por meio de pequenos experimentos práticos de laboratórios. Os módulos se baseavam em: Lógica Digital, Introdução à Programação e Sistemas Digitais. **Aulas Tradicionais Expositivas [EP.3]**, foram ministradas aulas de microprocessadores de alto desempenho, projeto de rede de interconexão de alto desempenho e desenvolvimento de software de sistemas de alto desempenho. **Aprendizagem Baseada em Problemas [EP.4]**, os conceitos eram estudados à medida que problemas eram apresentados como desafios. Os desafios eram propostos de forma similar a uma maratona de programação. Dessa forma, o Pensamento Paralelo era construído de forma teórica e prática. **Aprendizagem Baseada em Projetos**, em [EP.5] foram definidos projetos onde, a partir da formação de grupos, os assuntos eram discutidos e as soluções propostas para cada caso. Os projetos eram baseados em *Deep Learning*, *Big Data*, Reconhecimento de Imagens, Redes Neurais, entre outros. Em [EP.6], foi desenvolvido um cluster com Raspberry Pi para desenvolver os conceitos do Pensamento Paralelo, de forma que fosse possível verificar a execução de forma interativa e delegar para determinados nós do cluster tarefas distintas, de forma síncrona e assíncrona. Foi utilizado como um dos exemplos o *MapReduction*. Em [EP.7], os alunos usaram a interface *Jupyter* para acessar o *Palmetto* e o *CloudLab*, com o objetivo de se usar a infraestrutura de um supercomputador para praticar a Programação Paralela em um ambiente real. Com isso, os alunos puderam implementar atributos como acoplamento, escalabilidade, disponibilidade e confiabilidade na construção do Pensamento Paralelo. Em [EP.8], foram planejados cenários de implementação como o problema do coletor do cupom em *Assembly*, que é apropriado para



demonstrar princípios *multicore* no hardware de baixo nível, mas também ideal para os alunos apreciarem a eficiência e simplicidade que as instruções lógicas oferecem quando executadas a nível de bits.

Quadro 4 – Estudos Incluídos no Mapeamento Sistemático da Literatura.

ID	Base	Referência
EP.1	Scopus	Solorzano, A.L.V. and Charao, A.S. <i>BlocklyPar: From Sequential to Parallel with Block-based Visual Programming</i> . In: Proceedings - Frontiers in Education Conference, FIE . October. 2021.
EP.2	Scopus	Zhu, Z. and Bhowmik, U.K. and Wang, Y. and Cheng, Z. and Chen, Y.W. <i>Having it All: Infusing Parallel Computational Thinking in the Lower-level Computer Engineering Curriculum Using Extended Learning Modules</i> . In: ASEE Annual Conference and Exposition, Conference Proceedings 2021 .
EP.3	Scopus	Chen, J. and Impagliazzo, J. and Shen, L. <i>High-Performance Computing and Engineering Educational Development and Practice</i> . In: Proceedings - Frontiers in Education Conference, FIE . October. 2020.
EP.4	Scopus	<i>Learning Parallel Programming through Programming Challenges</i> . In: Proceedings - Frontiers in Education Conference, FIE . October. 2020.
EP.5	ACM	<i>Parallel programming course development based on parallel computational thinking</i> . In: ACM International Conference Proceeding , p.103-109. 2018.
EP.6	Scopus	Li, B. and Mooring, J. and Blanchard, S. and Johri, A. and Leko, M. and Cameron, K.W. <i>SeeMore: A kinetic parallel computer sculpture for educating broad audiences on parallel computation</i> . In: Journal of Parallel and Distributed Computing . vol.105. p.183-199. 2017.
EP.7	Scopus	Ngo, L.B. and Srinath, A.T. and Denton, J. and Ziolkowski, M. <i>Unifying Computing Resources and Access Interface to Support Parallel and Distributed Computing Education</i> . In: Journal of Parallel and Distributed Computing . vol.118. p.201-212. 2018.
EP.8	Scopus	Levandowski, B. and Perouli, D. and Brylow, D. <i>Using Embedded xinu and the Raspberry Pi 3 to Teach Parallel Computing in Assembly Programming</i> . In: IEEE 33rd International Parallel and Distributed Processing Symposium Workshops , p.334-341. 2019.

(QS.1) Quais foram as ferramentas (bibliotecas, linguagens, frameworks) utilizadas para dar suporte à construção do Pensamento Paralelo no processo de ensino-aprendizagem?

São apontadas várias ferramentas (linguagens, APIs, bibliotecas) utilizadas como suporte à construção do Pensamento Paralelo, durante o ensino da Programação Paralela. Percebe-se que a linguagem C/C++ ainda é predominante, mas também são utilizadas *Python, Assembly, Java e Scala*. Entre as bibliotecas de comunicação, são citadas as OpenMP, MPI e CUDA.

Em [EP.1], o jogo *BlocklyPar* foi baseado na ferramenta *Block-based tools MIT e Scratch*. Em [EP.2], foram utilizadas as linguagens *Assembly e C*. No [EP.3] e [EP.6], destacam-se as bibliotecas OpenMP e MPI. Já no [EP.4] e [EP.5], também se destacam as bibliotecas OpenMP, MPI e o módulo de programação em CUDA, sendo que no [EP.4], também, foi utilizado o sistema *Mooshak* como programa na maratona de programação instalado sobre uma máquina virtual na *Google Cloud* acessado via Web. No [EP.7], foram utilizadas as linguagens *Python, Java, C++ e Scala*, juntamente com as bibliotecas *MPI, mpi4py e Spark API*. Também foram utilizadas as ferramentas *Palmetto, CloudLab, XSEDE, Hadoop e Jupyter*. Em [EP.8] foi usado o *Raspberry Pi3*, com o sistema operacional *Embedded Xinu*, e a linguagem *Assembly* para ARM.



(QS.2) Quais foram as principais arquiteturas (CPU, GPU, Híbridas) utilizadas na construção do Pensamento Paralelo?

Em [EP.1], utiliza um cenário de aprendizado na Web, através da aplicação do jogo. Já nos estudos [EP.2], [EP.4], [EP.5] e [EP.7] são identificados o uso de arquiteturas híbridas, ou seja, faz-se presente programação utilizando tanto as CPUs, quanto as GPUs. O [EP.3] utilizou as CPUs do supercomputador *Tianhe-2*. Nos estudos [EP.6] e [EP.8] foram utilizados os processadores ARM, talvez por serem uma arquitetura de baixo custo, quando comparado a processadores de supercomputadores, sendo que o [EP.8] utilizou a estrutura Raspberry Pi 3.

(QS.3) Quais foram as percepções identificadas sobre o Pensamento Paralelo?

Em [EP.1] os alunos que utilizaram o jogo e que não tinham cursado programação paralela, e que conheciam apenas a programação sequencial, relataram que aprenderam na prática conceitos como: divisão de tarefas, alocação e compartilhamento de recursos, acesso a dados simultâneos e concorrentes. Em [EP.2], após avaliar os resultados, verificou-se que a aplicação dos módulos apresentou melhorias em relação a taxa de aprendizado, quando comparado com o método tradicional, onde os alunos só teriam aulas sobre os conteúdos nos períodos finais da graduação. Os módulos ajudaram a construir o pensamento paralelo à medida que eram ensinados. No [EP.3] apesar de utilizar o método tradicional, a eficácia do curso de Computação de Alto Desempenho focou em dois aspectos: criar um currículo próprio de CAD e um ambiente prático que beneficiasse os alunos de Computação. Em [EP.4] os ambientes de maratonas se mostraram eficientes para a construção do pensamento paralelo, promovendo um ambiente saudável que estimula os alunos a criarem soluções corretas e criativas. No [EP.5] o mecanismo de *ranking de corretude* foi importante para incentivar os alunos a participarem dentro e fora de sala de aula, e na discussão dos problemas. Também ajudou na solução dos problemas em curtos intervalos de tempo, dando retornos imediatos e estimulando a colaboração em sala de aula. Em [EP.6] a ferramenta *SeeMore* se mostrou promissora na tarefa de incentivar o aprendizado da programação paralela e, conseqüentemente, do pensamento paralelo, através da demonstração de problemas clássicos da área de CAD. No [EP.7] os alunos responderam de forma positiva ao uso dos ambientes *Palmetto* e *CloudLab*, citando, principalmente, a capacidade de trabalhar em ambientes reais de computação. E em [EP.8] se percebeu que, através dos experimentos da programação de baixo nível, conceitos como memória compartilhada e distribuída pareceu ser menos abstrato, principalmente utilizando ferramentas como OpenMP e MPI.

(QS.4) Quais foram as principais dificuldades relatadas por professores e alunos na utilização do Pensamento Paralelo durante o uso das metodologias ativas?

Em [EP.1] alguns conceitos, necessários à construção do pensamento paralelo, não são representados de forma clara pelas tarefas em bloco, no entanto, os autores não descrevem quais conceitos são esses. No [EP.2] as maiores dificuldades foram detectadas nos alunos dos períodos iniciais, pois apresentaram menores desempenhos quando comparados aos alunos de períodos mais adiantados. Em [EP.3] a principal dificuldade apontada foi a multidisciplinariedade necessária para formar os alunos durante o curso, pois vários conceitos, de diferentes assuntos, eram necessários. Em [EP.4] a principal dificuldade de aprendizado foi a construção do Pensamento Paralelo utilizando o MPI. Neste item, os alunos alcançaram as menores pontuações na maratona de programação. No [EP.5] os professores identificaram alguns exemplos muito complexos para que um iniciante conseguisse compreender e implementar uma solução. Contudo, com a construção do pensamento paralelo as soluções eram definidas com mais facilidade. Um outro ponto apresentado pelo autor, foi ausência de discussão dos problemas por alguns alunos, muitas vezes por timidez, acabavam por



implementar as soluções de forma isolada, o que dificultava um retorno de aprendizado por parte destes. Em [EP.6] o público era bastante diversificado, o que ocasionava definições superficiais dos problemas. Em [EP.7] alguns projetos não ficaram claros para alguns alunos e por conta disso, tinham dificuldades em compreender quais os objetivos a serem alcançados. Somava-se a isso, a plataforma *Jupyter* apresentou alguns erros em relação a leitura de *log*. Já no [EP.8] como foi utilizada uma plataforma mais simples, não houveram relatos de dificuldades da metodologia utilizada, apenas no uso da plataforma utilizada pelo professor para que os trabalhos pudessem ser enviados.

6. Ameaça à Validade da Pesquisa

A forma de seleção dos estudos primários e a falta de experiência dos agentes envolvidos são uma das principais ameaças, quando se trata de uma revisão sistemática da literatura. Por conta disso, visando fornecer uma maior credibilidade para a pesquisa, foi realizada uma busca automática nas bases utilizadas, após comprovação de eficácia da *string*, por meio de vários testes. Além disso, foi elaborado o planejamento de um protocolo onde foram definidos os objetivos, escopo, restrições, critérios de seleção e avaliação.

Vale ressaltar, que o objetivo do mapeamento era identificar metodologias ativas que auxiliasse na construção do chamado Pensamento Paralelo. Verifica-se que existem vários estudos sobre o ensino da Programação Paralela, mas sobre a construção efetiva do Pensamento Paralelo, os estudos ainda parecem estar limitados e os conceitos parecem estar confusos e dispersos na literatura. Talvez, encontrar esse limite, pode fazer com que esse trabalho termine por ser classificado como sendo mais um sobre o ensino da Programação Paralela. No entanto, o objetivo é apontar argumentos, para que sejam identificados modelos, que possam auxiliar na construção de elementos onde se possa implementar, de fato, o Pensamento Paralelo.

7. Conclusão e Trabalhos Futuros

A Computação de Alto Desempenho está procurando se consolidar nos cursos de graduação e pós-graduação em Computação das Universidade do Brasil e do mundo, porém ainda não se definiu uma forma conceitual de se formar/construir o chamado Pensamento Paralelo. E uma das habilidades que pode contribuir nessa construção é o Pensamento Computacional, pois já vem sendo utilizado para reduzir a evasão nos cursos de Computação, principalmente em disciplinas ministradas nos períodos iniciais, como Algoritmos e Programação [Silva, 2020].

Percebe-se que os conceitos do chamado Pensamento Computacional Paralelo ainda estão dispersos na literatura, e também, são pouco explorados. A grande maioria dos trabalhos encontrados tratam apenas do ensino da Programação Paralela de uma forma tradicional e expositiva. A construção do pensar em paralelo requer formas e metodologias diferentes do habitual, pois precisa, além de motivar e incentivar os alunos pela área, precisa também desenvolver as habilidades e competências que são próprias e específicas da construção de códigos paralelos. E é nesse contexto que as metodologias ativas podem ser utilizadas.

As metodologias ativas são um forte instrumentos de motivação no processo de ensino e aprendizagem, pois utilizam técnicas que despertam o interesse dos alunos através de jogos ou outros processos interativos e práticos. Renova a forma tradicional de aulas expositivas, o que incentiva e motiva os alunos. Utilizar essas metodologias no processo da construção do Pensamento Paralelo é algo que precisa ser tratado de forma a motivar, também os professores no processo de ensino da Programação Paralela.

Como trabalho futuros, pretende-se estabelecer a construção de teorias que, partindo-se de metodologias ativas, possam despertar o interesse do aluno, bem como desenvolver as



habilidade e competência necessárias para a construção do Pensamento Computacional Paralelo nos alunos de graduação dos cursos de Computação.

Referências Bibliográficas

ACM; IEEE-CS. *Computing Curricula: Paradigms for Global Computing Education*, 2020.

ALMEIDA JR., R.; ARAÚJO, J. S., Uma Avaliação do uso das Placas Gráficas no Ensino da Programação Paralela nos Cursos de Computação no Brasil. In: XXX WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2022, Niterói – RJ. Anais [...] WEI. p. 357–368.

ALMEIDA JR., R. B.; FARIAS, H., SAAVEDRA, M.; de ARAÚJO, J. S. Metodologias de ensino na programação paralela com placas gráficas: Uma revisão sistemática da literatura. In: X CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2021, Evento Online. Anais [...] CBIE. p. 1243-1252.

BACHIEGA, N. G.; SOUZA, P. S. L.; BRUSCHI, S. M.; SOUZA, S. R. S. Mapeamento Sistemático do Ensino Teórico e Prático de Programação Paralela In: WORKHOPS DO CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 2017, Recife – PE. Anais [...] CBIE. p. 1089-1098.

DURAES, T. de J. O., SOUZA, P. S. L., MARTINS, G., CONTE, D. J., BACHIEGA, N. G., BRUSCHI, S. M., Research on Parallel Computing Teaching: State of the Art and Future Directions. IEEE FRONTIERS IN EDUCATION CONFERENCE, 2020, Uppsala, Sweden, 2020, Proceedings [...] FIE. 1-9. Doi: 10.1109/FIE44824.2020.9273914.

FENWICK, G.; Norris, C.; GPGPU Programming for CS Undergraduates: Which one is Superman? ACM Southeast Conference, 2020. Tampa – FL. Proceeding ACMSE. p. 2-9.

LOPES, A. F., SANTANA, T. S., BRAGA, A. H., O Ensino de Pensamento Computacional por meio de Jogos Desplugados e Olimpíadas Científicas: Um Relato de Experiências nos anos Finais do Ensino Fundamental. In: XXVII WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2020. Cuiabá – MT. Anais [...] WEI. p. 96-100.

MOTA, L. P., NEVES, I., Robótica como ferramenta para o Desenvolvimento do Pensamento Computacional e Introdução à Lógica de Programação. In: XXVII WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2020. Cuiabá – MT. Anais [...] WEI. p. 141-145.

SANTOS JR., M. F., SILVA, S. A., Mostra de Ideias: Aplicação de uma Metodologia Gamificada e Baseada em Problemas no curso de Sistemas de Informação da Universidade do Estado de Mato Grosso In: XXVII WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2020. Cuiabá – MT. Anais [...] WEI. p. 151-155.

SILVA, E. O., FALCÃO, T. P., O Pensamento Computacional no Ensino Superior e seu Impacto na Aprendizagem de Programação. In: XXVIII WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 2020. Cuiabá – MT. Anais [...] WEI. p. 171-175.

WING, J. M., Computational Thinking. Communications of the ACM. Vol. 49, nº 3, p. 33-35, 2006.

ZORZO, A. F.; NUNES, D.; MATOS, E.; STEINMACHER, I.; ARAUJO, R. M. de; CORREIA, R.; MARTINS, S. Referenciais de Formação para os Cursos de Graduação em Computação. In *Sociedade Brasileira de Computação (SBC)*, 2017.