

MEMCOMPUTAÇÃO: CARACTERÍSTICAS E APLICAÇÕES EM COMPUTAÇÃO PARALELA

Rodrigo de Sousa Pissardini

Laboratório de Topografia e Geodesia
Departamento de Engenharia de Transportes
Escola Politécnica da Universidade de São Paulo

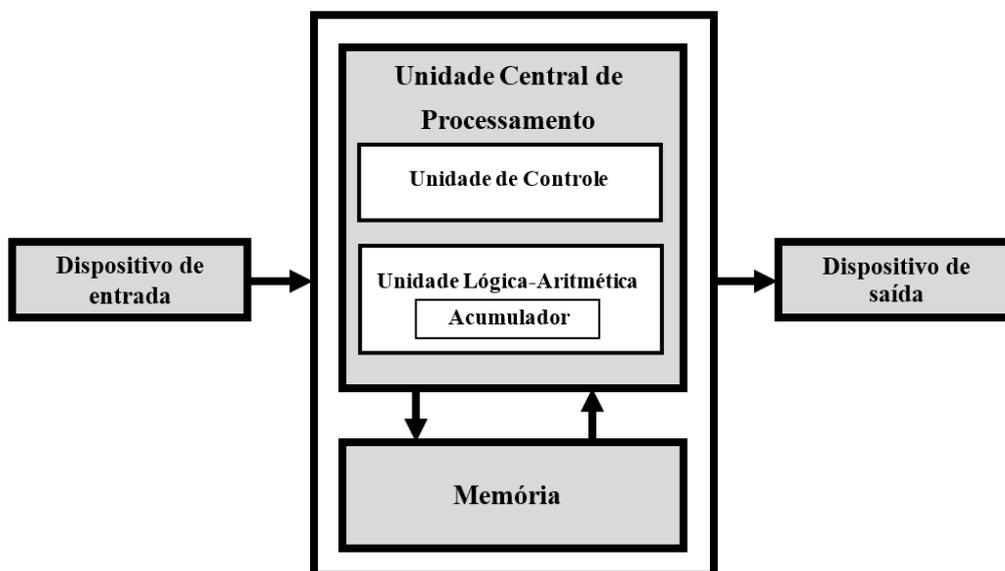
RESUMO

Dispositivos computacionais modernos, independentemente do seu tipo, seguem o Modelo de Von Neumann, no qual o processamento e armazenamento de memória são realizados em unidades separadas, fazendo com que o custo de processamento e armazenamento em termos de energia e tempo sejam dispendiosos e ineficientes. Isto impacta diretamente o desempenho de sistemas computacionais, em especial, daqueles que demandam alto poder computacional tais como sistemas paralelos. Neste artigo, analisa-se o modelo de memcomputação, um novo conceito de computação na qual as unidades físicas de processamento e memória são as mesmas. Esta abordagem permite aumentar a velocidade e eficiência dos dispositivos computacionais, possibilitando o desenvolvimento de novos tipos de arquiteturas de computação.

1 INTRODUÇÃO

A maior parte dos dispositivos computacionais modernos, independentemente do seu tipo, são baseados na arquitetura de John Von Neumann. A arquitetura, proposta em 1945 no artigo *First Draft of a Report on the EDVAC*, é a base do conceito de computador com programa armazenado (VON NEUMANN, 1945; VENTRA; TRAVERSA; PERSHIN, 2014).

Figura 1: Arquitetura de Von Neumann (Fonte: O Autor)



Nesta arquitetura, o processamento de instruções é realizado pela Unidade Central de Processamento (em inglês *Central Processing Unit*- CPU), mas o armazenamento de dados ocorre na Memória. Isto significa que todo processamento realizado nesta arquitetura sempre ocorre em duas fases distintas (processamento e gravação/leitura) e que permanentemente uma grande quantidade de dados está sendo transferida entre a CPU e memória (e vice-versa), o que demanda custo de energia que poderia ser melhor aproveitado. Além disto, a Arquitetura de Von Neumann possui uma limitação de velocidade de transferência de dados entre a CPU e a memória, chamada de “Gargalo de Von Neumann”: isto ocorre porque a transferência de dados entre CPU e memória é menor do que a quantidade de memória disponível, forçando a CPU a esperar continuamente pelos dados a serem recebidos ou enviados para a memória. Diversas técnicas (como uso de cache, *multithreading* e outras) e arquiteturas (e.g. arquitetura de Harvard, máquinas com multiprocessadores e máquinas paralelas), foram propostas para minimizar as deficiências da Arquitetura de Von Neumann, porém, independentemente da performance obtida, o processamento e gravação/leitura de dados são ainda realizadas como duas operações distintas, limitando o desempenho final que pode ser obtido.

No entanto, execução do processamento em dois passos não ocorre por uma limitação lógica, mas por uma limitação física dos componentes usados para criação dos dispositivos computacionais: as atuais memórias de computadores não podem realizar processamento e os processadores não conseguem armazenar memória (VENTRA; TRAVERSA; PERSHIN, 2014; VENTRA; PERSHIN, 2013; 2015). Se as operações de processamento e armazenamento de dados puderem ser coordenadas em uma operação única (processamento e armazenamento ao mesmo tempo), minimiza-se a necessidade de transferência de dados entre a CPU e a memória, permitindo processar dados em menor tempo e reduzir também o custo (computacional e energético) envolvido (VENTRA; PERSHIN, 2013; 2015).

A demanda de energia elétrica nos setores de tecnologia da informação e comunicação será responsável, em 2020, por 14% do consumo global de eletricidade (incluindo demanda de equipamentos tecnológicos de uso geral e de *datacenters*) e, com a popularização de dispositivos móveis e de equipamentos inteligentes (eletrodomésticos, veículos, etc.), a tendência é que este consumo aumente continuamente ao longo dos anos (PICKAVET et al., 2008). Isto torna o investimento em novas tecnologias que consomem menos energia uma necessidade preeminente. Ao mesmo tempo, há uma exigência contínua de aumento de poder computacional em diversos segmentos (e.g. processamento de *big data* para necessidades empresariais e para computação científica – simulações climáticas, análise genômica, etc.), que não pode ser garantida através de abordagens simplistas, como, por exemplo, a redução do tamanho dos transistores ou aumento de quantidade de unidades de processamento em um equipamento. A redução de transistores deve ser limitada até 2016, de acordo com a *International Technology Roadmap for Semiconductors*, já que os materiais dos componentes disponíveis não poderão mais reduzir o seu tamanho e manter a sua capacidade. O aumento de da quantidade de unidades de processamento ainda demanda capacidade de comunicação entre os processadores, que é limitado por diversos fatores, em especial, o barramento computacional (VENTRA; PERSHIN, 2015).

Entre as soluções propostas para solucionar estes problemas e reduzir a ineficiência dos atuais equipamentos computacionais está o conceito de memcomputação. A memcomputação têm aparecido desde 2008 como um segmento promissor na Computação, em especial como uma alternativa a técnicas como Computação Quântica já que esta ainda demanda um grande período de tempo em pesquisas até o

desenvolvimento de equipamentos para o grande público. A memcomputação, no entanto, já possui componentes em produção, pode ser inserido em dispositivos computacionais já existentes sem grandes mudanças e, por sua natureza, possui grande aderência às necessidades de aplicação paralela. Este trabalho faz uma revisão bibliográfica sobre a memcomputação e os componentes associados, desde sua proposta inicial e sistematiza os tópicos envolvidos sobre o tema. O trabalho divide-se em cinco seções, além desta introdução: a seção 2 apresenta os conceitos fundamentais da memcomputação e seus componentes (aspectos físicos e lógicos), a seção 3 desenvolve as possibilidades de *hardware* e *software* que podem ser baseados em sistemas memristivos (com enfoque no uso para computação paralela) e, por fim, a seção 4 apresenta as Considerações Finais sobre a análise realizada.

2 FUNDAMENTOS DE MEMCOMPUTAÇÃO

2.1 CONCEITOS INICIAIS

A memcomputação (o prefixo *mem-* vem da palavra memória) é um modelo de computação que se baseia no pressuposto de que é possível processar e realizar armazenamento no mesmo componente físico, sem que haja interferência entre o processamento e o armazenamento de dados. A proposta inicial foi desenvolvida na Universidade da Califórnia pelo Dr. Leon Ong Chua em 1971, ao criar o conceito de componente chamado de *memristor* (CHUA, 1971; VENTRA, PERSHIN, 2013; 2015). Em 1976, Chua e Kang expandiram o conceito de memristor para desenvolvimento de sistemas memristivos e propuseram a equação geral que descreve componentes e sistemas memristivos (CHUA; KANG, 1976):

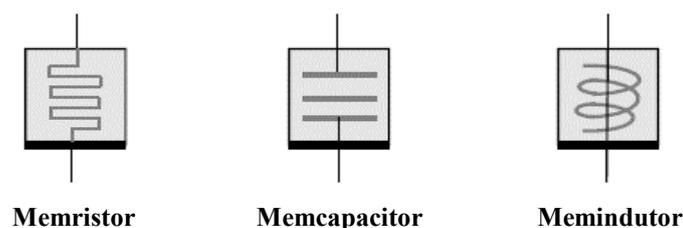
$$y = g(x, u, t)u$$

$$\frac{dx}{dt} = f(x, u, t)$$

No qual x é um vetor de n variável de estado interno, u e y são a entrada e saída (respectivamente) do sistema/ componente e f é uma função contínua n -dimensional e g é a função resposta contínua escalar.

Posteriormente, em Dezembro de 2008, Chua propôs outros hipotéticos componentes chamados de memcondutor e memindutor, baseados na equação geral dos sistemas memristivos (BIOLEK; BIOLEK; BIOLKOVÁ, 2011). Juntos, estes três componentes (memristor, memcapacitor, memindutor) são chamados de memelementos (Figura 2) (VENTRA, PERSHIN, 2013).

Figura 2: Símbolos dos memelementos (Fonte: VENTRA, PERSHIN, CHUA, 2009)



Ao combinar coleções destes três componentes, dispositivos computacionais podem adquirir capacidade de realizar processamento/armazenamento simultâneos, simplificar e reduzir o tamanho do *hardware* desenvolvido e criar uma arquitetura massivamente paralela de alta velocidade (capaz de realizar em segundos cálculos que consumiriam décadas nas máquinas atuais) que pode ser utilizado desde chips de *smartphones* até supercomputadores (VENTRA, PERSHIN, 2015). Por isto são consideradas características dos sistemas memcomputacionais possuir (VENTRA, PERSHIN, 2013; PERSHIN, VENTRA, 2014b):

- Arquitetura massivamente paralela e escalável, combinando processamento de informação e armazenamento;
- Tempo suficientemente longo para armazenamento de dados;
- Habilidade de inicializar estados de memória;
- Habilidade de ler os resultados finais de processamento dos memelementos mais importantes.
- Robustez contra pequenas variações e ruídos.

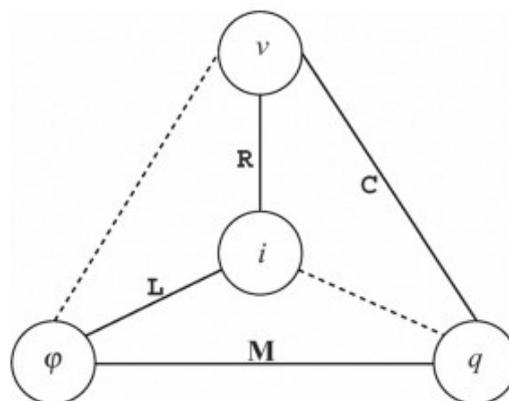
Sistemas memcomputacionais podem compor um dispositivo computacional por completo ou serem utilizados em associação com dispositivos computacionais baseados na Arquitetura de Von Neumann para realizar tarefas específicas.

2.2 ARQUITETURA E COMPONENTES

2.2.1 Memristor

Memristor (ou *memory resistor*) é um tipo de componente eletrônico, tendo sido proposto teoricamente por Chua (1971) ao analisar outras possíveis combinações entre as quatro variáveis fundamentais dos circuitos (corrente elétrica i , voltagem v , carga q e fluxo magnético φ – sendo as duas últimas variáveis descritas como integrais do tempo da corrente e da voltagem) (Figura 3) (KAVEHEI et al., 2009, RADWAN; FOUUDA, 2015). Por isto, o memristor é chamado de “quarto elemento fundamental dos circuitos eletrônicos” (ao lado do resistor, capacitor e indutor).

Figura 3: As quatro variáveis fundamentais dos circuitos e os componentes associados (R- Resistor, C- Capacitor, L-Indutor, M- Memristor) (KAVEHEI et al., 2009).



Um resistor padrão é um componente eletrônico passivo de duplo terminal utilizado em eletrônica para limitar a corrente elétrica em um circuito e dissipar energia térmica. Esta limitação é chamada de resistência ou impedância e trata-se de um valor constante para o componente. O memristor é definido também como um componente eletrônico passivo de duplo terminal, mas que altera o seu estado (resistência) conforme a quantidade da carga elétrica que flui em si (apresentando como comportamento um ciclo de histerese pinçado no plano tensão-corrente). Quando a corrente elétrica flui em uma direção através do circuito, a resistência aumenta. Porém se a corrente elétrica flui em direção contrária, a resistência diminui (CHUA, 2014).

Isto ocorre porque a resistência elétrica do componente não é constante, mas flexível o suficiente para ser alterada para cada variação do fluxo de energia que passa por ele. Desta forma, se o memristor perder energia, ele ainda mantém a última resistência obtida até que uma nova carga elétrica seja aplicada sobre ele. Assim, o componente armazena memória, no que é chamado de propriedade de não-volatilidade. No entanto, memristores são elementos puramente dissipativos e nenhuma energia é armazenada no componente. Pela flexibilidade de variação dos valores de resistência, um memristor permite armazenar dados em multinível e não apenas de forma binária (CHUA, 1971; VENTRA; PERSHIN, 2015). Um tutorial descritivo do uso do memristor pode ser visto em Chua (2014) e Radwan e Fouda. (2015). Um exemplo descritivo teórico do comportamento de um memristor pode ser vista em Yin et al. (2015).

Ainda que a proposta teórica do memristor tenha sido realizada por Chua (1971), sua primeira implementação prática foi realizada apenas em 2008 pelo Dr. Stanley Williams e sua equipe na Hewlett-Packard (HP). Esta demora ocorreu, em especial, porque os materiais existentes para criação de circuitos até então, não retinham memória (VENTRA, PERSHIN, 2015). A pesquisa da HP produziu memórias baseadas em dióxido de titânio que podem mudar a resistência e manter seu estado transformado. Desenvolveu-se uma camada de dióxido de titânio de 50 nm de largura entre dois eletrodos de 5 nm, sendo um de titânio e outro de platina (Figura 4 e 5). São agrupadas duas camadas, sendo que uma delas possui uma ligeira depleção de átomos de oxigênio. Quando um campo elétrico é aplicado, os locais vazios de oxigênio derivam, mudando a resistência (STRUKOV *et al.*, 2008; WILLIAMS, 2008; RADWAN; FOUDA, 2015).

Figura 4: Modelo de memristor desenvolvido pela HP

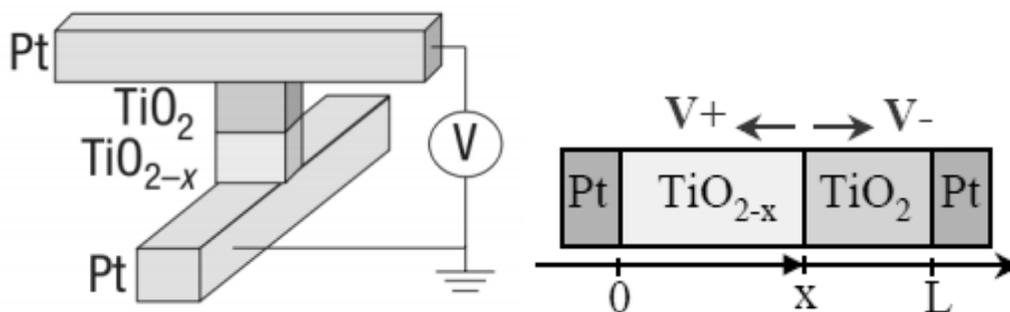
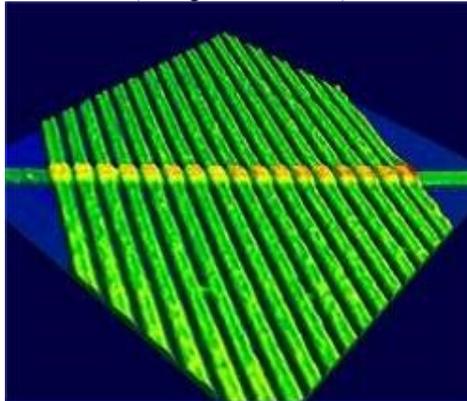


Figura 5: Imagem feita em microscópio de força atômica, mostrando 17 memristores lado a lado.
(Imagem: HP Labs)



O memristor tem sido desenvolvido, no entanto, com outros tipos de materiais, tais como polímero (EROKHIN; FONTANA, 2008), grafeno (JEONG et al., 2010), ferroelétrico (CHANTBOUALA et al., 2012), silício (MEHONIC et al., 2012) e outros. A principal vantagem obtida dos desenvolvimentos realizado é o fato de que o memristor pode ser trabalhado ao nível do nanômetro, o que permite prosseguir com a tendência da Lei de Moore além dos limites dos atuais dispositivos eletrônicos utilizados. Além disto, pelas suas características, é capaz de substituir não apenas resistores normais, mas também transistores como componentes de processamento.

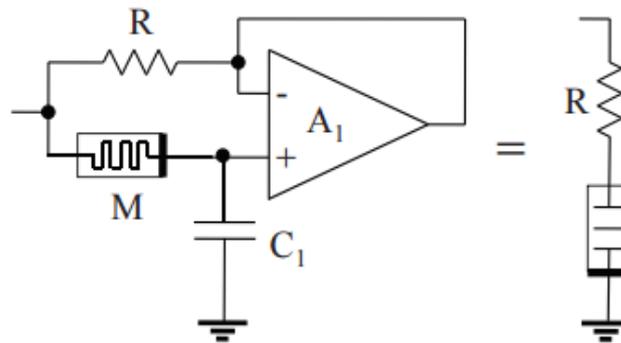
2.2.2 Memcapacitor

Um capacitor (ou condensador) padrão é um dispositivo eletrônico passivo de duplo terminal que armazena energia eletrostaticamente em um campo elétrico, permitindo armazenar energia potencial elétrica. Seu modelo principal utiliza dois condutores metálicos (placas) isolados mas paralelos: carregar um capacitor significa retirar uma quantidade de carga elétrica de uma placa para outra através da diferença de potencial entre elas. Chama-se de capacitância (ou capacidade elétrica) à capacidade do condutor concentrar energia e manter esta carga durante um período de tempo. Capacitores típicos não mudam sua capacitância, independentemente da quantidade de carga que seja aplicada sobre ele.

Um memcapacitor é um componente eletrônico de duplo terminal que permite armazenar carga elétrica e mudar sua capacitância dependendo das últimas tensões aplicadas sobre ele, proporcionando também capacidade de memória e processamento aos capacitores. Diferente, no entanto, dos memristores que consomem toda energia aplicada sobre eles, memcapacitores possibilitam reaproveitar parte da energia aplicada sobre eles, minimizando o consumo total do dispositivo computacional (BIOLEK; BIOLEK; BIOLKOVÁ, 2011; VENTRA; PERSHIN, 2015).

O conceito teórico dos memcapacitores foi desenvolvido por Ventra, Pershin e Chua (2009). Devido às suas características, o memcapacitor pode ser baseado em um memristor (Figura 6) (PERSHIN; VENTRA, 2010b).

Figura 6: Circuito simulando um memcapacitor e seu equivalente lógico (PERSHIN; VENTRA, 2010b)



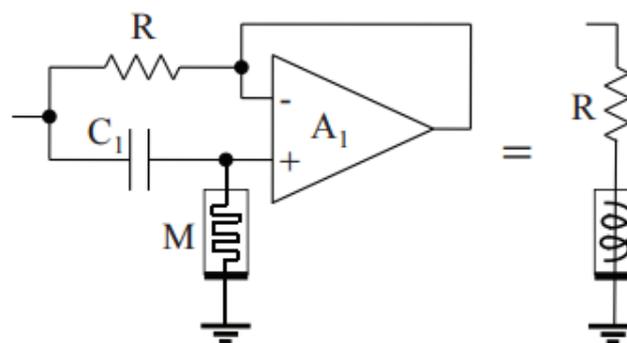
Memcapacitores puros têm sido desenvolvidos com materiais ferroelétricos de custo relativamente elevado. No entanto, grupos de pesquisa têm também desenvolvido memcapacitores em versões em silício de baixo custo (VENTRA; PERSHIN, 2015).

2.2.3 Memindutor

Um indutor padrão (também chamado de solenoide ou bobina) é um componente eletrônico passivo de duplo terminal, cuja principal função é acumular energia através da geração de um campo magnético (lei de Ampere) e impedir variações na corrente elétrica que passa em si. O modelo principal é um fio condutor enrolado de forma helicoidal (de forma autossustentada ou sobre um núcleo). Um memindutor é um componente de dois terminais de duplo terminal não-linear que armazena energia como um memcapacitor ao deixar a corrente elétrica fluir em si como um memristor (VENTRA; PERSHIN, 2015).

As implementações práticas de memindutores, atualmente, ainda são constituídas de grandes bobinas magnéticas de fios que inviabilizam a sua utilização em aplicações práticas em dispositivos computacionais comerciais atuais. A tendência, no entanto, é que com o tempo seu tamanho reduza (assim como ocorreu com os memristores) e seja utilizado em computadores comuns (VENTRA; PERSHIN, 2015). Pershin e Ventra (2010) também propuseram modelos de memindutor utilizando circuitos memristivos (PERSHIN, VENTRA, 2010) (Figura 7).

Figura 7: Circuito simulando um memindutor e seu equivalente lógico (PERSHIN; VENTRA, 2010).



3 APLICAÇÕES DE MEMCOMPUTAÇÃO

3.1 APLICAÇÕES EM *HARDWARE*

Por suas características físicas, *hardware* baseados em memelementos podem permitir diversas vantagens em relação a *hardware* baseado em componentes clássicos. Em especial, pode-se considerar o fato que os memelementos operam no nível do nanômetro, permitindo desenvolver tecnologias que cumpram a Lei de Moore para além do prazo limite estabelecido.

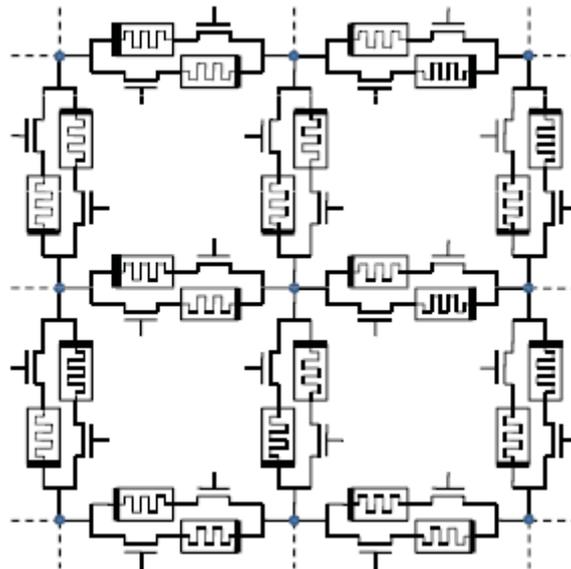
Equipamentos analógicos baseados em memelementos utilizam estes componentes para realizar parte de suas funções, ainda que não haja uma estrutura desenvolvida para tornar estes processos computacionais. As abordagens analógicas incluem osciladores senoidais (TALUKDAR et al., 2011), circuitos analógicos (nos quais as baixas voltagens são utilizada para aplicação utilizar os memcomponentes como componentes analógicos e as altas voltagens são utilizadas para programar os estados dos memcomponentes - PERSHIN; VENTRA, 2010b), filtros adaptativos (como filtro passa-baixa) (DRISCOLL et al., 2010; CHEW; LI, 2012) e outros.

Aplicações digitais incluem, principalmente, desenvolvimento de novos tipos de memórias e processadores. As principais propostas defendem a utilização de dispositivos memcomputacionais em conjunto com dispositivos clássicos, realizando ações específicas. Dispositivos memcomputacionais, ao permitir o armazenamento de dados e processamento simultâneo faz com que as arquiteturas computacionais não demandem o processamento plenamente central em um dispositivo computacional (apenas que o controle seja realizado de forma central). Isto permite minimizar a carga de processamento e comunicação com a CPU ao permitir que operações sejam realizadas ao nível do componente (por exemplo ao nível de memória ou no processamento responsivo a um sensor) e ao permitir que processamento paralelo possa ocorrer já que cada memelemento da memória pode processar seus dados independentemente (desde que os dados não sejam dependentes entre si, o que deve ser gerenciado pelo dispositivo de controle dos memelementos) (VENTRA; PERSHIN, 2015). Entre os principais desenvolvimentos digitais podem ser descritos:

- ***Dynamic Computing Random Access Memory (DCRAM) e outros tipos de memórias (VENTRA et al., 2014):*** memórias atuais baseiam-se em capacitores. Em uma memória padrão, um *bit* específico é representado pela carga armazenada em um único capacitor, o que exige um grande número de capacitores para representar um programa (VENTRA; PERSHIN, 2015). A utilização de memelementos pode aumentar a capacidade e reduzir o custo de energia para operações em memória. Enquanto memórias DRAM possuem cerca de 18 Gbits/cm², memelementos podem ter tamanho de 20 nm e abranger 100 Gbits/cm². O modelo de DCRAM utiliza memcapacitores e memristores, suportando operações como WRITE, READ, REFRESH e outras funções lógicas, além de permitir que operações lógicas e processamento ocorram diretamente em memória, reduzindo a transferência entre memória e CPU. Além disso, múltiplas operações lógicas podem ser representadas por um mesmo componente (polimorfismo), sendo as operações mudadas instantaneamente através de mudança de tensão. Instruções como “calcule x e y”, “calcule x ou y” e “além disso calcule z” podem ser manuseadas por dois memcapacitores ao invés de diversos capacitores e transistores fixos e regulares. O desenvolvimento de novos tipos de memórias tem sido a principal aplicação prática atual de memelementos (como, por exemplo, na The Machine da HP).

- **Processador Memristivo (VENTRA; PERSHIN, 2014):** um processador memristivo (Figura 8) é um conceito de processador formado por uma rede em grade, no qual cada ponto da grade possui unidades memcomputacionais anexadas. Cada unidade possui dois dispositivos memcomputacionais conectados simetricamente, em paralelo, e dois interruptores. Os interruptores fornecem acesso aos dispositivos individualmente, enquanto a ligação paralela simetriza a resposta dos dois componentes (VENTRA; PERSHIN, 2014).

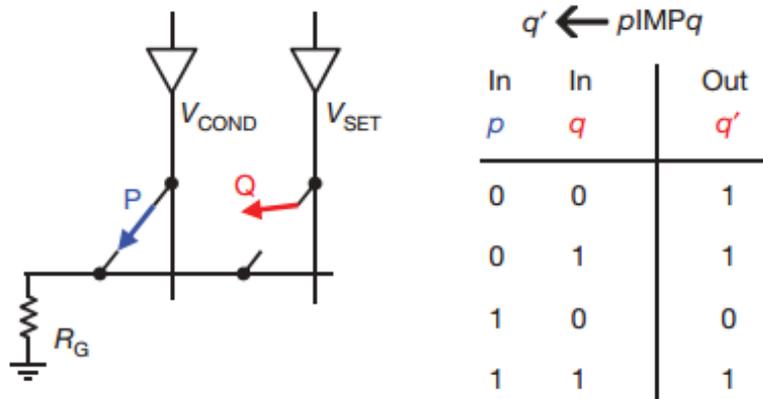
Figura 8: Processador Memristivo (VENTRA;PERSHIN,2014)



Pelo fato de utilizar sistemas não-lineares, processadores memcomputacionais são também adequados para uso em criptografia e geração de números aleatórios. Propostas de controle e versões mais simples de sistemas caóticos podem ser realizados através de memristores (RADWAN; FOUDA, 2015).

- **Porta Lógica “Implica” (BORGHETTI et al., 2010):** o memristor pode ser utilizado para desenvolvimento da porta lógica “IMPLICA” que permite realizar todas as operações binárias possíveis entre duas variáveis (sendo, por isto, chamada de porta universal) (Figura 9). Desta forma, é possível utilizar um mesmo conjunto de memelementos para realizar diferentes funções lógicas, reduzindo a quantidade de componentes necessários para processamento. A porta IMPLICA obtém dois pulsos simultâneos aplicados sobre os interruptores P e Q (que são memristores) respectivamente, de tal forma que a saída de Q é dependente de P. Desta forma, a porta $p \text{ IMP } q \rightarrow q$ é equivalente a $(\text{NOT } p) \text{ OR } q$.

Figura 9: Porta lógica IMPLICA e sua tabela-verdade (BORGHETTI et al., 2010).



3.2 APLICAÇÕES EM SOFTWARE

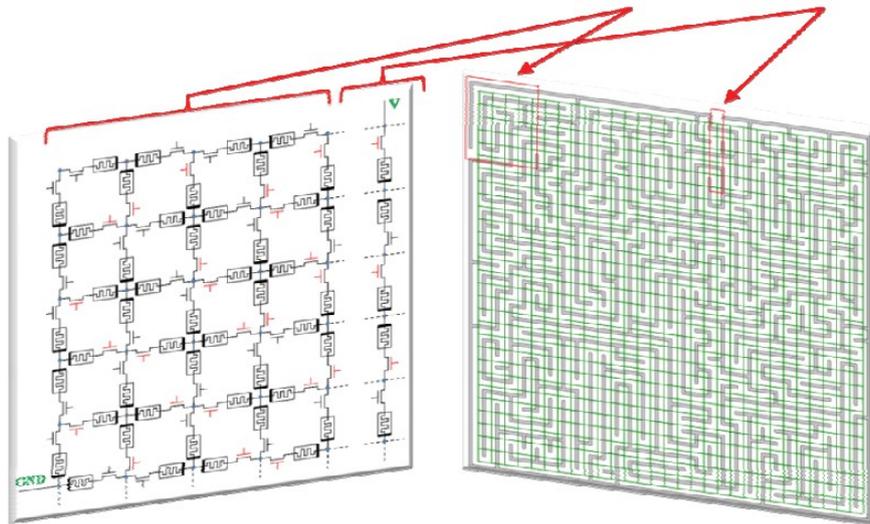
Entre as principais vantagens apresentadas por equipamentos memcomputacionais, em especial, salienta-se a sua capacidade de realização de processamento massivamente paralelo. Em uma aplicação paralela típica, um conjunto de processadores (com suas memórias distintas) calcula as partes de um programa que podem ser realizadas paralelamente e que comunicam-se entre si para chegar a uma resposta final. Independente se esta comunicação ocorre internamente a um dispositivo computacional ou mediante uma rede de computadores, ainda se requer energia e tempo para transferir informações. As principais propostas de paralelismo usando memcomputação envolvem a divisão do problema paralelo de tal forma que cada parte de um problema é modelado como um memelemento, sendo as operações executadas em paralelo. Outros memelementos ou mesmo sistemas baseados na Arquitetura de Von Neumann são responsáveis por obter os resultados finais dos memelementos mais importantes do sistema e unifica-los (VENTRA; PERSHIN. 2015).

Os principais tipos de processamento que podem ser realizados por sistemas memristivos incluem-se:

- **Operações paralelizáveis (VENTRA; PERSHIN. 2015):** partes paralelizáveis de um código, ao invés de serem separadas conforme a quantidade de processadores, podem ser separadas para processamento em um conjunto de memelementos. Por exemplo, se um trecho de código *for* realiza operações sobre três variáveis, podem ser separados grupos de três memristores que realizam operações cíclicas sobre seus dados e mantendo estes dados armazenados em si até que sejam necessários. Desta forma, o processamento centralizado é distribuído entre os diversos memelementos e a combinação com memcapacitores permite que uma mesma energia possa ser reciclada para diferentes operações.
- **Solução da saída de um labirinto (VENTRA; PERSHIN, 2011; 2015):** labirintos e suas soluções servem como modelos na teoria dos grafos, topologia, navegação robótica, otimização de tráfego e outras áreas da ciência. Considerando-se um labirinto clássico, com uma saída e uma entrada, a solução de saída é qualquer caminho que permita interligar entrada e saída. Algoritmos ingênuos testam todas as possibilidades de caminho, enquanto algoritmos mais complexos ou baseados em sistemas paralelos utilizam heurísticas apropriadas ou dividem o labirinto em etapas consecutivas, resolvendo cada parte e unificando o resultado. Em qualquer abordagem utilizada, o processamento se torna mais lento

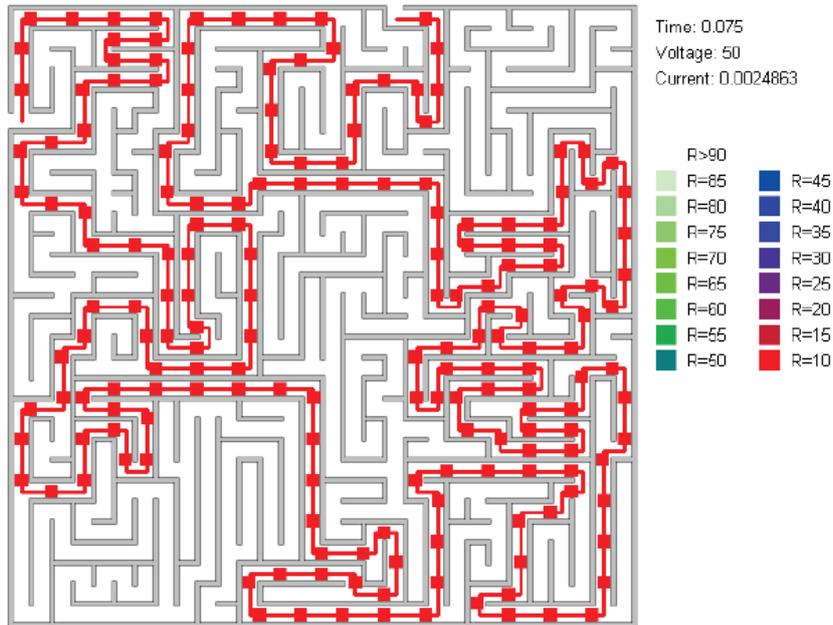
e exige maior poder computacional conforme aumenta o tamanho do labirinto. A proposta memcomputacional de resolução do labirinto de Ventra e Pershin (2011) utiliza um conjunto de memristores, mapeando cada memristor para cada curva do labirinto, todos em alto estado de resistência (Figura 10).

Figura 10: Conjunto de memristores modelando um labirinto (VENTRA; PERSHIN, 2011).



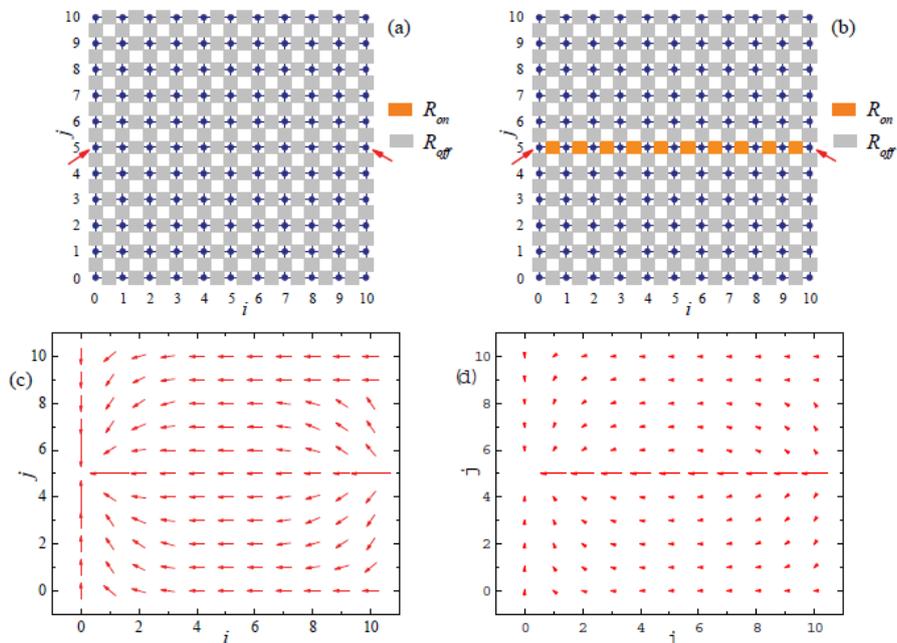
É aplicado um pulso de tensão nos memristores que representam a entrada e saída do labirinto: a tendência é que a corrente fluirá ao longo do caminho da solução fechando um circuito. Ao percorrer o labirinto, o pulso modificará as resistências dos memristores, sendo bloqueada nos caminhos sem saída. Após o pulso desaparecer, a solução do caminho estará armazenada nos memristores sob a forma de baixa resistência (Figura 11) (PERSHIN; VENTRA, 2011). Qualquer que seja o tempo de solução desta técnica, ela sempre será mais rápida do que a versão convencional de resolução.

Figura 11: Solução de um labirinto (PERSHIN; VENTRA, 2011).



- **Menor caminho (VENTRA; PERSHIN, 2014):** a resolução de menor caminho é realizado de forma semelhante à resolução de saída de um labirinto (Figura 12).

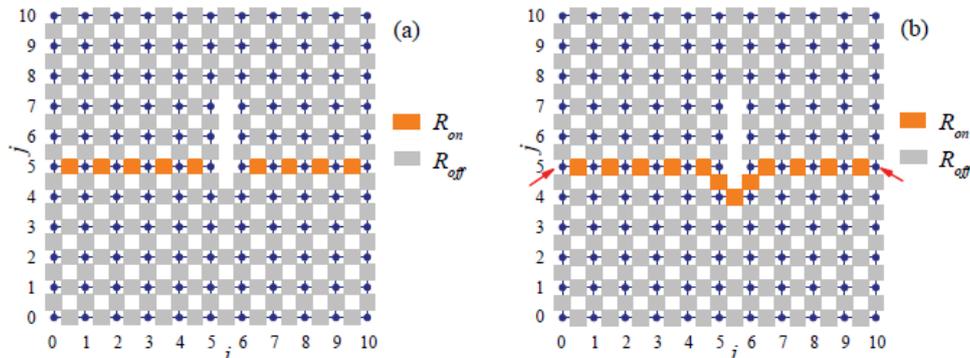
Figura 12: Menor caminho em uma rede de memristores. Considera-se R_{on} como um caminho de baixa resistência e R_{off} um caminho de alta resistência.



Modela-se um conjunto de nós de um grafo (pontos de interesse, estados, etc.) sob a forma de uma rede de memristores, todos com alta resistência. Aplica-se um pulso de tensão nos pontos de entrada e saída, sendo que a corrente fluirá por todos os pontos. Ao final, o menor caminho é dado pelo subconjunto de memristores

com estado com menor resistência. O processo é mantido mesmo em redes de memristores danificada (Figura 13).

Figura 13: Descoberta de menor caminho em uma rede de memristores danificada (PERSHIN; VENTRA, 2014b).



- **Redes neurais artificiais e circuitos neuromórficos (PERSHIN; VENTRA, 2010a):** o cérebro humano, utiliza o mesmo conjunto de componentes (cerca de 10^{11} neurônios, com cerca de 10^{14} sinapses) para processamento/armazenamento, realizando cerca de 10^{16} operações por segundo e consumindo cerca de 10–25 watts diários neste processo (KANDEL;SCHWARTZ; JESSEL, 2000; VENTRA; PERSHIN, 2013; 2015; VENTRA; TRAVERSA; PERSHIN, 2014). Pelas suas características, redes de memristores apresentam comportamento mais próximo das redes neurais biológicas do que aquele apresentado pelas redes neurais artificiais operando em Arquitetura de Von Neumann. Pershin e Ventra (2010a) conseguiram, utilizando emuladores de memristores, imitar todas as propriedades sinápticas do cérebro humano e operar a formação de memória associativa (PERSHIN; VENTRA, 2010a) e utilizar também memcapacitores com sinapses eletrônicas (PERSHIN; VENTRA, 2014b). Wang et al. (2013) propuseram uma arquitetura neuromórfica adaptativa que auto-ajusta seus parâmetros de acordo com a frequência de estímulos. A arquitetura permitiu reproduzir o comportamento da ameba.

4 CONSIDERAÇÕES FINAIS

A memcomputação é, ainda, um segmento incipiente da computação. Não existem, ainda, equipamentos completos com esta tecnologia e não há, também, o domínio pleno sobre quais os tipos mais eficientes de sistemas operacionais e do *software* que deve ser desenvolvido para obter a máxima eficiência de dispositivos memcomputacionais. Em especial, deve-se determinar se o desenvolvedor terá acesso diretamente aos recursos de *hardware* ou se os recursos memcomputacionais serão tratados pelo sistema operacional e o desenvolvedor continuará desenvolvendo da mesma forma como faz hoje.

No entanto, a tecnologia básica memcomputacional já existe e não se demanda grande salto de tecnologia para seu refinamento. Os primeiros dispositivos reais (DCRAM) estão previstos para 2018 e, com a popularização da tecnologia, a tendência é o aumento de investimentos e maturação do segmento.

REFERÊNCIAS BIBLIOGRÁFICAS

BAYAT, F. M.; SHOURAKI, S. B. Mixed analog-digital crossbar-based hardware implementation of sign-sign LMS adaptive filter. **Analog Integr. Circuits Signal Process.**, vol. 66, no. 1, pp. 41- 48, 2011.

BIOLEK, D.; BIOLEK, Z.; BIOLKOVÁ, V. Behavioral Modeling of Memcapacitor. **Radio engineering**, v.20, n.1, pp.228-223, 2011.

BORGHETTI, J. et al. ‘Memristive’ switches enable ‘stateful’ logic operations via material implication. **Nature**, n.464, pp. 873-876, 2010.

CHANTHBOUALA, A. et al. A ferroelectric memristor. **Nature Materials**, v.11, pp. 860-864, 2012.

CHEW, S. Z.; LI, L. Printed circuit board based memristor in adaptive low pass filter. **Electronic Letters**, i.25, pp.1610- 1611, 2012.

CHUA, L.O. Memristor: The Missing Circuit Element. **IEEE Transactions on Circuit Theory**, CT-18, n.5, pp.507-519, 1971.

CHUA, L.O. If It’s Pinched It’s a Memristor. **Semicond. Sci. Technol.**, n.10, v.29, 2014.

CHUA, L.O.; KANG, S.M. Memristive devices and systems. **Proceedings of the IEEE**, v.64, n.2, pp. 209-223, 1976.

DRISCOLL, T. et al. Memristive Adaptive Filter. **Applied Physics Letters**, v.97, i.9, 2010.

EROKHIN, V.; FONTANA, M.P. **Electrochemically controlled polymeric device: a memristor (and more) found two years ago.** Disponível em <<http://arxiv.org/ftp/arxiv/papers/0807/0807.0333.pdf>>. 2008. Último acesso em 10 de Maio de 2015.

JEONG, H.Y. et al. Graphene Oxide Thin Films for Flexible Nonvolatile Memory Applications. **Nano Letters**, v.10, n.11, pp.4381-4386, 2010.

KANDEL, E. R.; SCHWARTZ, J. H.; JESSEL, T. M. **Principles of neural science**, McGraw-Hill, 2000.

KAVEHEI, O. et al. The Fourth Element: Insights into the Memristor. **Communications, Circuits and Systems, 2009. ICCAS 2009. International Conference on**, pp. 921-927, 2009.

MEHONIC, A. et al. Resistive switching in silicon suboxide films. **Journal of Applied Physics**, v.111, n.7, 2012.

PERSHIN, Y.V.; VENTRA, M.. Experimental demonstration of associative memory with memristive neural networks. **Journal Neural Networks**, v. 23, i.7, pp.881-886, 2010a.

PERSHIN, Y.V.; VENTRA, M. Practical approach to programmable analog circuits with memristors. **Circuits and Systems I: Regular Papers, IEEE Transactions on**, v. 57, n.8, pp.1857-1864, 2010b.

PERSHIN, Y.V.; VENTRA, M.. Solving mazes with memristors: a massively-parallel approach. **Phys. Rev**, v. 84, n.4, 2011.

PERSHIN, Y.V.; VENTRA, M. Memcapacitive neural networks. **Electronics Letters**, v. 50, 2014a.

PERSHIN, Y.V.; VENTRA, M.. Memcomputing: a Computing Paradigm to Store and Process Information on the Same Physical Platform. **Computational Electronics (IWCE), 2014 International Workshop on**, pp.1-2, 2014b.

PICKAVET, M. et al. Worldwide energy needs for ICT: the rise of power-aware networking. **2nd International Symposium on Advanced Networks and Telecommunication Systems**, ANTS 2008, pp. 1–3, December 2008.

RADWAN, A.; FOUDA, M. **On the Mathematical Modeling of Memristor, Memcapacitor, and Meminductor**. Springer, 2015.

STRUKOV, D.B. et al. The Missing Memristor Found. **Nature**, v. 453, pp.80-83, 2008..

TALUKDAR, A. et al, Generalized model for Memristor-based Wien family oscillators, **Microelectron. J.**, vol. 42, no. 9, pp. 1032-1038, 2011.

VENTRA, M. et al. Dynamic Computing Random Access Memory: a brain-inspired computing paradigm with memelements. **Circuits and Systems (ISCAS), 2014 IEEE International Symposium on**, pp. 1070-1073, 2014

VENTRA, M.; PERSHIN, Y.V. The parallel approach. **Nature Physics**, v.9, pp. 200-202, 2013.

VENTRA, M.; PERSHIN, Y.V. Memcomputing: a computing paradigm to store and process information on the same physical platform. **Nature Physics**, v.9, 2014.

VENTRA, M.; PERSHIN, Y.V. Just Add Memory. **Scientific American**, n.312, pp. 56-61, 2015.

VENTRA, M.; PERSHIN, Y.V.; CHUA, L.O. Circuit elements with memory: *memristors*, *memcapacitors* and *meminductors*. **Proc. IEEE**, v. 97, i.10, pp.1717-1724, 2009.

VENTRA, M.; TRAVERSA, F.; PERSHIN, Y.V. Memcomputing: computing with and in memory. 2014.

VON NEUMANN, J. **First Draft of a Report on the EDVAC**. Technical Report. 1945.

WANG, F.Z. et al. Adaptive Neuromorphic Architecture (ANA). **Neural Networks**, v. 45, n. 12 , pp. 111–116, 2013.

WILLIAMS, R. How we found the missing memristor. **IEEE Spectr.**,v.45,n.12, pp. 28–35, December 2008.

YIN, Z. et al. What are Memristor, Memcapacitor and Meminductor ? **Circuits and Systems II: Express Briefs, IEEE Transactions on**, n.4, v.62, pp. 402-406, 2015.