

# Uma estrutura para análise de confiabilidade de circuitos combinacionais usando inferência bayesiana aproximada

Shivani Bathla, estudante de pós-graduação, IEEE, e Vinita Vasudevan, membro, IEEE

**Resumo**— Uma abordagem comumente usada para calcular a taxa de erro nas saídas primárias (POs) de um circuito é comparar as cópias livres de falhas e defeituosas do circuito usando portas XOR. Este modelo resulta em baixa precisão com métodos não baseados em amostragem para estimativa de confiabilidade. Uma alternativa é usar uma única cópia do circuito com uma representação de quatro valores para cada rede correspondente aos sinais corretos e incorretos. Um problema nesta formulação é a propagação precisa das probabilidades associadas. Usamos a estrutura da inferência bayesiana (BI) para resolver esse problema. Derivamos a distribuição de probabilidade condicional (CPD) correspondente aos sinais de quatro valores e encontramos a taxa de erro de saída usando várias técnicas aproximadas de BI. Com nossa formulação, demonstramos que a taxa de erro de saída é dimensionada com as probabilidades de erro da porta. É garantido que seja zero quando a probabilidade de erro da porta for zero, desde que sejam usados algoritmos de BI aproximados baseados na propagação de crenças do produto da soma (BP). Embora as imprecisões aumentem com probabilidades de erro de porta muito baixas, ele é capaz de capturar a confiabilidade relativa das saídas entre si. Também propomos um novo método para encontrar a taxa geral de erro do circuito como a função de partição para um estado fixo de POs. Este método fornece uma melhoria significativa na precisão quando comparado com o método existente que utiliza portas OR.

**Termos de Indexação**— Inferência Bayesiana (BI), Redes Bayesianas (BNs), taxa de erro, confiabilidade, probabilidade de sinal.

## I. INTRODUÇÃO

O escalonamento AGRESSIVO da tecnologia CMOS levou a um aumento acentuado em defeitos de fabricação e falhas transitórias em portas devido a baixas tensões de limiar, variações de processo, eletromigração e diafonia. Isto resultou em portas lógicas não confiáveis, cujas saídas não são completamente determinadas pelas entradas da porta. Em vez disso, dadas as entradas, só podemos especificar a probabilidade de que a saída de uma porta seja zero ou um. Este comportamento probabilístico também pode ocorrer devido ao envelhecimento dos circuitos. É explorado por circuitos imprecisos para aplicações resistentes a erros para obter economia de energia.

O modelo mais comumente usado para uma porta não confiável é o modelo de erro de Von Neumann [1]. É essencialmente um modelo de canal binário simétrico no qual a probabilidade de obter um

Artigo recebido em 12 de dezembro de 2022; revisado em 9 de janeiro de 2023; aceito em 14 de janeiro de 2023. Data de publicação 26 de janeiro de 2023; data da versão atual 22 de março de 2023. (Autor correspondente: Shivani Bathla.)

Os autores pertencem ao Departamento de Engenharia Elétrica, IIT Madras, Chennai 600036, Índia (e-mail: ee13s064@ee.iitm.ac.in; vinita@ee.iitm.ac.in).

Versões coloridas de uma ou mais figuras deste artigo estão disponíveis em <https://doi.org/10.1109/TVLSI.2023.3237885>.

Identificador de Objeto Digital 10.1109/TVLSI.2023.3237885

1063-8210 © 2023 IEEE. O uso pessoal é permitido, mas a republicação/redistribuição requer permissão do IEEE.

Consulte <https://www.ieee.org/publications/rights/index.html> para obter mais informações.

Uso licenciado autorizado limitado a: UNIVERSIDADE DE SÃO PAULO. Baixado em 23 de março de 2024 às 18:21:46 UTC do IEEE Xplore. Aplicam-se restrições.

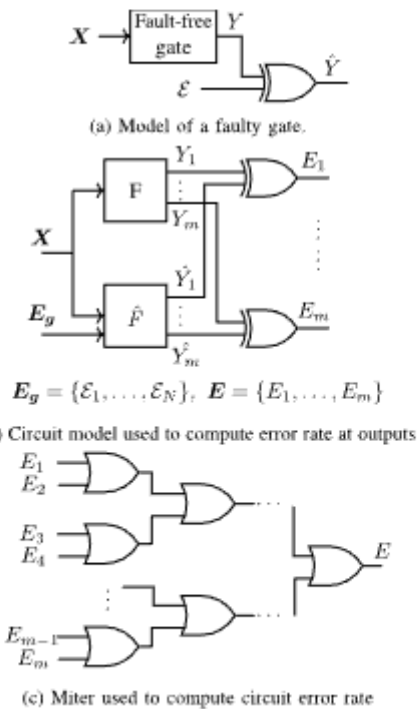


Figura 1. Modelos para análise de confiabilidade de circuitos.  $F$  denota o circuito sem erros e  $\hat{F}$  denota o circuito com defeito.  $E_i$  denota a fonte independente de erro para a  $i$ -ésima porta no circuito defeituoso, e  $E_g$  é o conjunto de todas as fontes de erro.  $E_i$  denota o sinal de erro correspondente ao  $i$ -ésimo PO, e  $E$  denota erro em pelo menos um dos POs. (a) Modelo de uma porta defeituosa. (b) Modelo de circuito usado para calcular a taxa de erro nas saídas. (c) Miter usada para calcular a taxa de erro do circuito.

um em vez de zero é o mesmo que obter um zero em vez de um. Os erros em cada porta são considerados independentes um do outro. No nível da porta, este modelo é equivalente a representar uma porta defeituosa como uma porta livre de falhas seguida por uma porta XOR, com a outra entrada da porta XOR conectada a uma fonte independente de erro ( $E$ ), como mostrado na Fig. 1(a).

As saídas dos circuitos projetados usando essas portas não confiáveis têm uma probabilidade de erro diferente de zero. O projeto, síntese e otimização desses circuitos requerem métodos eficientes para calcular essa probabilidade de erro. Este cálculo envolve uma comparação das saídas do sistema livre de erros ( $F$ ) e com defeito.

circuito (F) usando portas XOR, conforme mostrado na Fig. A métrica mais comumente utilizada para avaliação desses circuitos é a taxa de erro em cada saída primária (PO) do circuito [2], [3], [4], [5]. É definido como a probabilidade de erro em uma saída específica, calculada a média de todos os valores possíveis das entradas primárias (PIs). É calculado como a probabilidade do sinal das saídas das portas XOR  $\{E1, E2, \dots, Em\}$  na Fig. 1 (b), com as probabilidades de sinal dos PIs (X) definidas como 0,5. A outra métrica utilizada é a taxa de erro do circuito, que é a probabilidade de ocorrer um erro em pelo menos uma das saídas. Isso normalmente é calculado após conectar uma árvore de portas OR às portas XOR, como mostrado na Figura 1 (c).

O problema da análise de confiabilidade é, portanto, um problema de cálculo de probabilidades de sinais em um sistema que possui circuitos livres de falhas e circuitos defeituosos. O cálculo das probabilidades de sinais em um circuito é conhecido por ser #P completo [6] e é difícil mesmo em circuitos livres de erros. A análise de confiabilidade é significativamente mais difícil devido aos sinais e portas adicionais necessários para modelar circuitos errôneos, bem como devido ao loop reconvergente adicional criado pela conexão de portas XOR na saída. Métodos exatos para cálculo de probabilidade de sinal incluem métodos baseados em matrizes de transferência probabilísticas (PTMs) [7], [8], modelos de portas probabilísticas (PGMs) [9], diagramas de decisão binários (BDDs) [3], [10], contagem de modelos ponderados (VMC) [3] e redes bayesianas (BNs) [11], [12]. Contudo, a complexidade exponencial de tempo/espaco associada limita a aplicabilidade destes métodos a circuitos relativamente pequenos.

Vários métodos aproximados foram propostos na literatura. Os métodos baseados em simulação incluem simulação lógica usando a estrutura de Monte Carlo (MC), modelo de computação estocástica (SCM) [13] e métodos de inferência bayesiana (BI) baseados em amostragem [5], [14]. Nestes métodos, cada porta errada requer a geração de um fluxo adicional de números aleatórios. Como a precisão depende do número de amostras utilizadas, a complexidade do tempo aumenta à medida que as probabilidades de erro da porta diminuem. Portanto, com baixas probabilidades de erro de porta, as técnicas baseadas em amostragem tornam-se inadequadas para uso dentro de uma estrutura de otimização que requer cálculos de taxa de erro em cada iteração. Outra desvantagem é que estes métodos são inflexíveis no sentido de que qualquer alteração no circuito requer uma reavaliação completa.

Uma alternativa é usar métodos aproximados determinísticos baseados no cálculo da probabilidade do sinal. Os métodos propostos em [9] e [15] assumem que as entradas de uma porta são independentes e, portanto, têm precisão limitada. Para melhorar a precisão, os coeficientes de correlação de sinal propostos em [16] e [17] são utilizados nos métodos propostos em [4], [18], [19] e [20].

O principal desafio nas técnicas aproximadas determinísticas é contabilizar as correlações entre as redes. Isso inclui as correlações entre as redes livres de erros e as redes errôneas correspondentes, bem como as correlações devido a fanouts reconvergentes. Os métodos de análise aproximada propostos em [4], [15], [21], [22], [23], [24], [25] e [20] apresentam uma característica interessante. Em vez de usar duas cópias do circuito, eles usam uma única cópia na qual cada rede está associada a probabilidades adicionais que levam em conta as probabilidades livres de erros e errôneas.

valores do sinal. Esses métodos são atraentes, pois não apenas removem o loop reconvergente adicional criado pelas portas XOR, mas também evitam a necessidade de manter correlações entre as redes livres de erros e as redes errôneas correspondentes. Em [4], toda rede está associada a duas probabilidades condicionais que dão a probabilidade de ocorrência de erro em uma rede, dado o seu valor correto. Nos métodos propostos em [15], [20], [21], [22], [23], [24] e [25], cada rede é modelada como uma variável aleatória de quatro valores, correspondente ao seu erro e valores livres de erros. As probabilidades associadas são as probabilidades conjuntas da rede livre de erros e da rede errônea correspondente. O problema aqui é a propagação das quatro probabilidades. Os métodos existentes assumem independência entre as entradas da porta [15], [21] ou calculam coeficientes de correlação de sinal [4], [20] ou coeficientes de correlação de confiabilidade [23], [24], [25]. O principal problema com estes métodos são, portanto, as imprecisões que surgem devido à estimativa e propagação destas correlações. Normalmente, são usados coeficientes de correlação de sinais pareados, limitando ainda mais a precisão.

Neste artigo, propomos um novo algoritmo para análise de confiabilidade baseado em BI aproximado. Como em [15], usamos uma única cópia do circuito, com cada rede modelada como uma variável aleatória de quatro valores com as probabilidades associadas. No entanto, em vez de propagar coeficientes de correlação, classificamos o problema como um problema de BI. Para cada porta do circuito, derivamos a distribuição de probabilidade condicional (CPD) que determina a probabilidade de cada um dos quatro estados da saída, dados todos os estados possíveis das entradas. Métodos de inferência aproximados (e sempre que possível, exatos) são então usados para determinar a taxa de erro nos POs. Embora métodos baseados em amostragem também possam ser usados, o foco deste artigo são os métodos determinísticos aproximados.

Com nossa formulação, demonstramos que a taxa de erro de saída é dimensionada com as probabilidades de erro da porta. Também é garantido que seja zero quando a probabilidade de erro da porta for zero, desde que sejam usados algoritmos de BI aproximados baseados na propagação de crenças do produto da soma (BP). Esta propriedade não é válida se o modelo da Fig. 1 (b) for usado para cálculo de confiabilidade. Neste caso, muitas vezes a taxa de erro calculada na saída é de 0,5, o que é apenas ruído. Em contraste, embora as imprecisões aumentem com probabilidades de erro de porta baixas, nosso método é capaz de capturar a confiabilidade relativa das saídas entre si para probabilidades de erro de porta tão baixas quanto 10<sup>-6</sup>. Portanto, pode ser usado para identificar OPs mais suscetíveis a erros.

Propomos também uma nova formulação para encontrar a taxa de erro do circuito como a função de partição correspondente a um estado fixo de POs. Este método elimina as portas OR adicionais conectadas às saídas XOR na Fig. 1 (c) e resulta em uma melhoria significativa na precisão. Ele também possui tempos de execução muito bons e é adequado para uso em uma rotina de otimização.

O restante deste artigo está organizado da seguinte forma. A Seção II apresenta a notação utilizada e os antecedentes do BI. Discutimos a formulação proposta na Seção III, os resultados obtidos na Seção IV e uma comparação mais detalhada com trabalhos relacionados na Seção V. Por fim, apresentamos nossas conclusões.

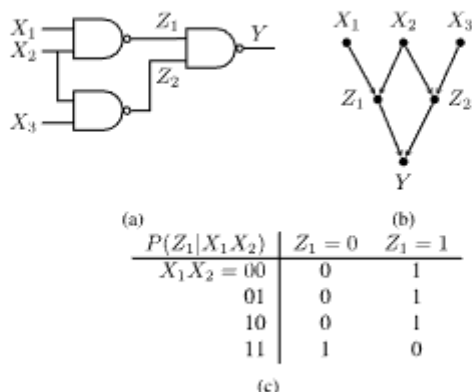


Figura 2. (a) Exemplo de circuito (b) modelo BN do circuito e (c) CPD de uma porta NAND livre de erros.

## II. FUNDO

### A. Notação

Usamos a seguinte notação. Usamos letras maiúsculas para denotar variáveis aleatórias, letras minúsculas para denotar os estados ou valores que uma variável aleatória pode assumir e letras em negrito para denotar um conjunto de variáveis aleatórias. Portanto,  $P(Z = z)$  denota a probabilidade de a variável aleatória  $Z$  assumir o valor  $z$ .  $X$  e  $\bar{X}$  são usados para denotar o conjunto de PIs livres de erros e errôneos. Da mesma forma,  $Y$  e  $\bar{Y}$  denotam o conjunto de POs livres de erros e errôneos do circuito, respectivamente.

### B. Redes Bayesianas

Um BN é um modelo gráfico probabilístico que captura a distribuição conjunta de probabilidade (JPD) sobre um conjunto de variáveis ( $Z$ ) usando um gráfico acíclico direcionado (DAG). Cada nó no DAG é uma variável aleatória e as arestas capturam a relação causal entre as variáveis. O BN correspondente a um circuito é um DAG que possui arestas direcionadas de cada entrada para a saída de uma porta. As variáveis no BN correspondem, portanto, a redes no circuito que são PI ou saídas de portas no circuito. Variáveis que são conectadas por uma aresta de entrada a uma variável  $Z_i$  são chamadas de pais de  $Z_i$  e denotadas como  $\text{Pa}Z_i$ . Cada variável está associada a um CPD que especifica a probabilidade da variável, dado o estado dos seus pais. Na linguagem dos circuitos, o CPD especifica a probabilidade de a saída de uma porta assumir um valor específico, dados os valores da entrada. A Figura 2 mostra um exemplo de BN para um circuito e CPD para uma porta NAND livre de erros.

Uma propriedade fundamental satisfeita por uma BN é a seguinte. Propriedade 1: Dado o estado de suas variáveis pai, uma variável

capaz no BN é condicionalmente independente de todos os não sucessores.

Para a rede da Figura 2(b), esta propriedade significaria, por exemplo,  $P(Y | Z_1, Z_2, X_1) = P(Y | Z_1, Z_2)$  e  $P(Z_1, Z_2 | X_1, X_2) = P(Z_1 | X_1, X_2)P(Z_2 | X_1, X_2)$ . Como resultado desta propriedade, o JPD de todas as variáveis da rede ( $Z$ ) pode ser escrito de forma fatorada da seguinte forma:

$$P(Z) = \prod_{\text{Dia} \in ZP(\text{Dia} | \text{PaDia})} \quad (1)$$

Uso licenciado autorizado limitado a: UNIVERSIDADE DE SÃO PAULO. Baixado em 23 de março de 2024 às 18:21:46 UTC do IEEE Xplore. Aplicam-se restrições.

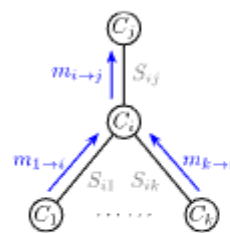


Figura 3. Representação da passagem de mensagens do cluster  $C_i$  para o cluster  $C_j$ .

onde  $\text{Pa}Z_i$  são os pais de  $Z_i$ ,  $P(Z_i | \text{Pa}Z_i)$  é o CPD associado a  $Z_i$  e  $Z$  é o conjunto de todas as variáveis do BN. Cada CPD é considerado um fator sobre o conjunto de variáveis  $\{Z_i \cup \text{Pa}Z_i\}$ , e o produto dos CPDs é calculado como um produto fatorial (definido no Apêndice). A distribuição de probabilidade de uma variável  $Z_i$ ,  $P(Z_i)$ , é referida como a probabilidade marginal de  $Z_i$ . A probabilidade do sinal de  $Z_i$  é definida como  $P(Z_i = 1)$ .

A tarefa de calcular as probabilidades marginais de todas as redes do circuito é uma das tarefas de inferência possíveis em um BN. Os métodos determinísticos para BI aproximado são técnicas variacionais que otimizam um funcional energético utilizando algoritmos BP. Na maioria desses métodos, BP é realizado usando o algoritmo BP de produto de soma [26, cap. 11], que realiza a passagem de mensagens em um gráfico de cluster não direcionado. Cada nó neste gráfico está associado a um cluster ou conjunto de variáveis no BN e é denotado como  $C_i$ . A aresta entre dois clusters  $C_i$  e  $C_j$  está associada a um subconjunto de variáveis contidas em ambos os clusters, ou seja, cada aresta está associada a um peso  $S_{ij}$ ,  $j \in C_i \cap C_j$ . Os pesos  $S_{ij}$  são chamados de conjuntos separados. Cada CPD do BN, chamado de fatores iniciais, é atribuído a um único cluster no gráfico de cluster não direcionado que contém a variável correspondente e seus pais. O algoritmo BP é um algoritmo de passagem de mensagens no qual um cluster  $C_i$  transmite uma mensagem para  $C_j$  com base nas mensagens de todos os seus outros vizinhos. 3. A mensagem  $m_{i \rightarrow j}$  do cluster  $C_i$  para o cluster  $C_j$  é calculada usando a seguinte operação de soma-produto

$$m_{i \rightarrow j}(S_{ij}) = \sum_{j \in \psi_i} \prod_{k \in \text{Vizinhos}(i) \setminus j} m_{k \rightarrow i}(\text{Ciclo}) \quad (2)$$

onde  $\psi_i$  é o produto dos fatores iniciais atribuídos ao cluster  $C_i$ . O produto fatorial de  $\psi_i$  e mensagens recebidas dos vizinhos de  $C_i$  é marginalizado sobre variáveis que não estão presentes no conjunto conjunto  $S_{ij}$ .

Em técnicas de inferência exata, o grafo de cluster é restringido a ser uma árvore com variáveis de conjuntos separados  $S_{ij}$ ,  $j \in C_i \cap C_j$ . Essa árvore é chamada de árvore de junção ou árvore de junção (JT). A inferência exata requer duas rodadas de passagem de mensagens – uma passagem descendente de um nó escolhido aleatoriamente para as folhas da árvore e uma passagem ascendente de volta das folhas para a raiz. A complexidade de tempo e espaço da inferência exata usando BP é exponencial no tamanho máximo do cluster. Circuitos com loops reconvergentes longos e aninhados tendem a ter clusters maiores, tornando inviável a inferência exata. Os métodos aproximados limitam o tamanho máximo do cluster, mas permitem ciclos

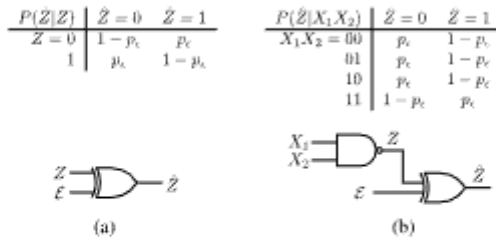


Figura 4. Modelo de erro e CPD para um sinal errado e uma porta NAND com defeito. (a) Sinal errado. (b) Porta NAND com defeito.

no gráfico de cluster. Isso inclui BP loopy (LBP) [27] e suas variantes [28], [29], BP generalizado (GBP) [30] e propagação de gráfico de junção iterativa (IJGP) [31]. Nestes métodos, o algoritmo de passagem de mensagens é executado iterativamente em um gráfico loopy até a convergência. As seguintes propriedades são satisfeitas após a convergência.

1) Cada cluster está associado a um JPD ( $Q_i(C_i)$ ) válido, mas possivelmente aproximado, que é calculado da seguinte forma:

$$Q_i(C_i) = \psi \cdot \prod_{j \in \text{Vizinhos}(i)} m_{j \rightarrow i}(S_j) \quad (3)$$

2) A probabilidade marginal de uma variável é a mesma em todos os clusters em que está presente. Observe que nem todos os algoritmos BP usam a operação soma-produto exata para calcular mensagens. Um exemplo é o método do mini-balde ponderado (WMB) proposto em [32]. Outro algoritmo para inferência aproximada é a técnica incremental build-infer-proximal (IBIA) [33] que utiliza uma sequência de árvores de junção. O tamanho máximo do cluster em cada uma das árvores de junção é definido como um valor especificado pelo usuário. A soma do produto BP é usada em cada árvore de junção para inferir probabilidades. Na maioria desses algoritmos, é possível trocar o tempo de execução por maior precisão, tendo um número maior de variáveis em um cluster.

As outras técnicas utilizadas para inferência são métodos baseados em WMC [34]. Eles são destinados à inferência exata e podem ser usados apenas para circuitos pequenos. Existem também várias técnicas baseadas em amostragem, como amostragem lógica probabilística (PLS) [14], amostragem de importância pré-propagada de evidências [35] e amostragem de Gibbs [36]. A precisão obtida com esses métodos depende do método utilizado para geração de amostras e do número de amostras utilizadas.

### III. FORMULAÇÃO PROPOSTA

#### A. Motivação

Conforme discutido na Seção I, o modelo de erro usado para análise de confiabilidade é o modelo de erro de Von Neumann. Se  $Z$  e  $\hat{Z}$  denotam os valores corretos e incorretos da saída de uma porta e  $p_e$  denota a probabilidade de erro da porta, o CPD  $P(\hat{Z}|Z)$  especificado por este modelo é mostrado na Fig. A figura também tem a representação correspondente em nível de porta de um sinal errado, que é uma porta XOR que possui um sinal de erro independente  $E$ .  $p_e$  é, portanto, a probabilidade do sinal de erro, ou seja,  $p_e = P(E = 1)$ .

Com base nisso, o modelo de erro de porta que consiste em uma porta livre de erros seguida por uma porta XOR [mostrada na Fig. 1 (a)] pode ser equivalentemente representado como um CPD das saídas, dados os valores de entrada, denotados como  $P(\hat{Z}|X)$ . Isso pode ser derivado da seguinte forma:

$$\begin{aligned} P(\hat{Z}|X) &= \sum_{E} P(Z|X)P(\hat{Z}|Z, E)P(E) \\ &= \sum_{E} P(Z|X)P(\hat{Z}|Z). \end{aligned} \quad (4)$$

Com

Por exemplo, o CPD para uma porta NAND de duas entradas com defeito obtido usando (4) é mostrado na Fig. Observe que todos esses produtos são obtidos usando o produto fatorial conforme definido no Apêndice.

Nós motivar nossa formulação usando um exemplo.

A Figura 5 (a) e (b) mostra o modelo de erro e o BN correspondente para análise de confiabilidade do circuito mostrado na Figura 2 (a). Como todas as variáveis neste BN têm valor binário, denotamos este modelo BN como o BN de dois valores. Conforme explicado na Seção II-B, o JPD de todas as variáveis,  $Z$ , no BN pode ser escrito como um produto dos CPDs de todas as variáveis [usando (1)]. Para este exemplo, portanto,

$$P(Z) = P(X_1)P(X_2)P(X_3)f_1f_2f_3P(E|Y, \hat{Y})$$

onde

$$\begin{aligned} f_1 &= P(Z_1|X_1, X_2)P(\hat{Z}_1|X_1, X_2) = P(Z_1, \hat{Z}_1|X_1, X_2) \\ f_2 &= P(Z_2|X_2, X_3)P(\hat{Z}_2|X_2, X_3) = P(Z_2, \hat{Z}_2|X_2, X_3) \\ f_3 &= P(Y|Z_1, Z_2)P(\hat{Y}|Z_1, \hat{Z}_1, Z_2, \hat{Z}_2) = P(Y, \hat{Y}|Z_1, \hat{Z}_1, Z_2, \hat{Z}_2). \end{aligned}$$

(5)

Este resultado segue da Propriedade 1 para BNs. Aqui, para facilitar a explicação, os Pls ( $X = \{X_1, X_2, X_3\}$ ) são considerados livres de erros. A probabilidade de erro do circuito,  $P(E)$ , pode ser obtida somando o JPD,  $P(Z)$ , sobre os estados de todas as outras variáveis do circuito, em um processo denominado eliminação de variáveis [26], [37].

Cada um dos fatores  $f_1$ ,  $f_2$  e  $f_3$  em (5) pode ser considerado um CPD, representando o JPD das saídas livres de erros e errôneas, dados os valores das entradas livres de erros e errôneas. CPDs  $P(Y|Z_1, Z_2)$  e  $P(\hat{Y}|Z_1, \hat{Z}_1, Z_2, \hat{Z}_2)$  são matrizes  $4 \times 2$  e, portanto, o produto do fator dará  $P(Y, \hat{Y}|Z_1, Z_2, \hat{Z}_1, \hat{Z}_2)$  como uma matriz  $16 \times 4$ .

Com base neste agrupamento de CPDs, pode-se facilmente perceber que o sistema para análise de confiabilidade pode ser representado como um único BN, com cada variável  $\{Z\}$  assumindo quatro valores  $\{00, 01, 10, 11\}$  que denotam os valores de o sinal correspondente nos circuitos sem falhas e com defeito. Isso é mostrado na Figura 5 (c). Referimo-nos ao modelo da Fig. 5 (c) como o BN de quatro valores. No exemplo, as variáveis  $Z_1$ ,  $Z_2$  e  $Y$  têm quatro valores. Como os Pls são considerados precisos para o cálculo da taxa de erro, eles têm dois valores. Em geral, os Pls também podem ter quatro valores. O CPD de cada porta representa o JPD das saídas livres de erros e erradas, dados os valores das entradas livres de erros e erradas. Portanto,

$$P(Y|Z_1, Z_2) = P(Y, \hat{Y}|Z_1, \hat{Z}_1, Z_2, \hat{Z}_2). \quad (6)$$

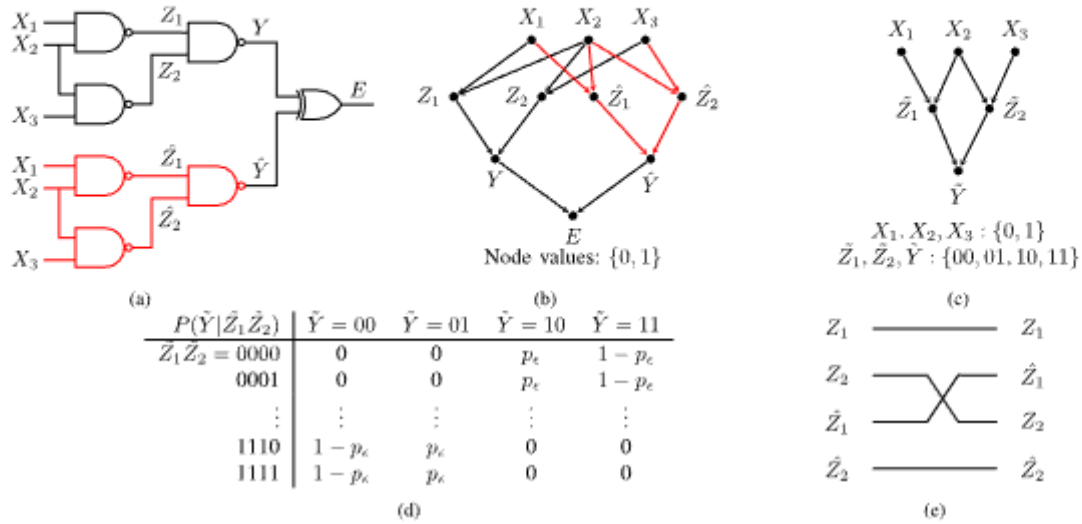


Figura 5. (a) Modelo de erro para cálculo da taxa de erro para o circuito mostrado na Figura 2 (a). (b) BN de dois valores para o modelo de erro. (c) BN de quatro valores para o modelo de erro. (d) CPD de uma porta NAND de duas entradas no BN de quatro valores. (e) Mudança na ordem das variáveis implementada usando a matriz de permutação.

#### B. CPDs no BN de quatro valores

Praticamente, o CPD de uma porta de duas entradas no BN de quatro valores,  $P(Y | Z_1, Z_2)$ , pode ser obtido como segue. Dados  $P(Y | Z_1, Z_2)$  e  $P(\hat{Y} | \hat{Z}_1, \hat{Z}_2)$ , temos

$$P(Y, \hat{Y} | Z_1, Z_2, \hat{Z}_1, \hat{Z}_2) = P(Y | Z_1, Z_2) P(\hat{Y} | \hat{Z}_1, \hat{Z}_2). \quad (7)$$

Vê-se que os CPDs em (6) e (7) diferem apenas na ordem das variáveis de entrada. Especificamente,  $\hat{Z}_1$  e  $Z_2$  devem ser trocados como mostrado na Fig. 5(e). Uma mudança na ordem das variáveis implica uma permutação das linhas do CPD. Isso pode ser feito pré-multiplicando por uma matriz de permutação da seguinte forma:

$$P(\hat{Y} | \hat{Z}_1, \hat{Z}_2) = P_m \times P(Y, \hat{Y} | Z_1, Z_2, \hat{Z}_1, \hat{Z}_2). \quad (8)$$

Aqui,  $P_m$  é uma matriz de permutação e  $\times$  é usado para representar a multiplicação de matrizes. Utilizando o procedimento descrito em [8], a matriz de permutação pode ser escrita da seguinte forma:

$$P_m = P(Z_1, \hat{Z}_1, Z_2, \hat{Z}_2 | Z_1, Z_2, \hat{Z}_1, \hat{Z}_2) = P(Z_1 | Z_1) P(\hat{Z}_1 | \hat{Z}_1, Z_2, \hat{Z}_2) P(Z_2 | Z_2, \hat{Z}_1, \hat{Z}_2) P(\hat{Z}_2 | \hat{Z}_2) =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Aqui,  $\otimes$  denota o produto tensorial ou produto de Kronecker das duas matrizes. As probabilidades condicionais mostradas acima são PTMs obtidas usando a abordagem descrita em [8]. O procedimento para uma porta com um número arbitrário de entradas é semelhante.

#### C. Cálculo da Taxa de Erros e Confiabilidade dos POs

A taxa de erro e a confiabilidade dos POs de um circuito são definidas a seguir.

**Definição 1 (Taxa de erro ( $p_{Ei}$ )):** É a probabilidade de erro em um PO  $Y_i$ , calculada a média de todos os valores de entrada possíveis. Supõe-se que todas as entradas são igualmente prováveis. É calculado da seguinte forma:

$$p_{Ei} = P(Y = 01) + P(Y = 10). \quad (9)$$

**Definição 2 (Confiabilidade ( $r_{Yi}$ )):** A confiabilidade de uma saída  $Y_i$  é definida como  $r_{Yi} = 1 - p_{Ei}$ .

O processo de calcular a média de entradas igualmente prováveis é equivalente a calcular a probabilidade de erro na saída após definir a probabilidade do sinal dos PIs em 0,5.

Como pode ser visto em (7) e (8), o CPD do líquido  $Y$  nas formulações de quatro valores é simplesmente uma permutação do produto fatorial dos CPDs da rede livre de erros ( $Y$ ) e da rede errônea ( $\hat{Y}$ ) em uma formulação de dois valores. Portanto, o produto fatorial de todos os CPDs na formulação de quatro valores dará o mesmo JPD geral que a formulação de dois valores. O cálculo exato das probabilidades marginais  $P(Y)$  e  $P(\hat{Y})$  na formulação de quatro e dois valores envolve a soma do JPD sobre o estado de todas as outras variáveis. Portanto,  $P(Y) = P(Y, \hat{Y})$ . Assim, a taxa de erro [(9)] estimada utilizando ambas as formulações é a mesma se forem utilizados métodos de inferência exatos.

As proposições a seguir mostram que para inferência aproximada usando produto de soma BP, é garantido que o BN de quatro valores forneça uma taxa de erro zero quando os PIs e as portas estão livres de erros. O significado deste teorema é discutido após as provas.

**Definição 3:** Uma rede  $Z_i$  é considerada livre de erros em relação a uma distribuição de probabilidade  $P$  se ela satisfaz  $P(Z_i = 01) = P(Z_i = 10) = 0$ .

**Definição 4:** O CPD de uma rede  $Y$  com pais  $Pa Y$  é considerado livre de erros em relação a uma distribuição  $P$  se satisfizer o seguinte:

$$P(Y = 01 | Pa Y = s) = P(Y = 10 | Pa Y = s) = 0$$

onde

$$s \in S = \{Z_i \in \{00, 11\} \quad \forall Z_i \in Pa Y\}.$$

Proposição 1: Dado um conjunto de redes livres de erros  $Z = \{Z_1, \dots, Z_n\}$ , a probabilidade conjunta  $P(Z)$  é avaliada como zero se o estado de qualquer  $Z_i \in Z$  for 01 ou 10.

Prova: A probabilidade marginal de um  $Z_i$  líquido pode ser obtida a partir da probabilidade conjunta  $P(Z)$  como segue:  $P(Z_i) = \sum Z \cdot Z_i P(Z)$ . Usando a Definição 3,  $P(Z_i)$  é avaliado como zero para os estados 01 e 10. Como as probabilidades são não negativas,  $P(Z)$  deve ser zero para todos os estados tais que  $Z_i \in \{01, 10\}$ .

Proposição 2: Se tanto o conjunto de pais  $Pa Y$  quanto o CPD  $P(Y|Pa Y)$  de uma rede  $Y$  estão livres de erros,  $Y$  também está livre de erros.

Prova: Seja  $S$  o conjunto de todos os estados possíveis de  $Pa Y$ .  $S$  pode ser dividido em dois conjuntos disjuntos  $S_1$  e  $S_2$ , onde  $S_1$  compreende estados onde todas as redes são 00 ou 11 (estados precisos da rede) e  $S_2 = S \setminus S_1$  (estados onde pelo menos uma rede é imprecisa). As operações de soma-produto para encontrar  $P(Y = 01)$  podem ser escritas da seguinte forma:

$$P(Y = 01) = \sum_{s \in S_1} P(Y = 01|Pa Y = s)P(Pa Y = s) + \sum_{s \in S_2} P(Y = 01|Pa Y = s)P(Pa Y = s).$$

Como o CPD não contém erros, o primeiro termo da soma é zero. No segundo termo, a soma é sobre estados de  $Pa Y$  tal que pelo menos um  $Z_i \in Pa Y$  é 01 ou 10. Usando a Proposição 1, a probabilidade conjunta  $P(Pa Y) = 0$  para esses estados. Portanto,  $P(Y = 01) = 0$ . Da mesma forma,  $P(Y = 10) = 0$ .

Teorema 1: O saída erro avaliar estimado usando

O produto da soma BP no BN de quatro valores é zero se todos os Pls e portas estiverem livres de erros.

Prova: Nesta prova, usamos  $P$  para denotar os CPDs exatos presentes como fatores iniciais e  $Q(C)$  para denotar o JPD aproximado de um cluster  $C$  obtido após a convergência do produto soma BP. Seja  $Z$  uma rede no circuito,  $C_k = \{Z, Pa Z, R\}$  seja o cluster ao qual é atribuído o CPD da rede  $Z$  ( $P(Z|Pa Z)$ ) e  $Q_k(C_k)$  seja o JPD das variáveis em  $C_k$ . Usando (3), a distribuição marginal  $Q_k(Z)$  pode ser escrita da seguinte forma:

$$\begin{aligned} Q_k(Z) &= \sum_{C_k \setminus Z} Q_k(C_k) = \sum_{C_k \setminus Z} \prod_{m_j \rightarrow i \in N \text{ oito } (i)} P(Z_i | Pa Z_i, R_i) \\ &= \sum_{Pa Z, R} P(Z | Pa Z) Q_k(Pa Z) Q_k(R | Z, Pa Z) \\ &= \sum_{Pa Z} P(Z | Pa Z) Q_k(Pa Z). \end{aligned} \quad (10)$$

O segundo passo na equação segue a regra da cadeia de probabilidade e porque um dos fatores iniciais atribuídos a  $C_k$  é o CPD de  $Z$ .

Seja  $X \in C_0$  um Pl, de modo que  $Pa X = \emptyset$  e o fator inicial correspondente seja  $P(X)$ . Portanto, de (10), temos  $Q_0(C_0) = P(X)$ . Considere uma rede  $Z \in C_1$  que é impulsionada pelos Pls. Conforme descrito na Seção II, após a convergência no produto soma BP, a probabilidade marginal de uma variável é a mesma em todos os clusters em que ela está presente. Portanto, todos os

TABELA I

TAXA DE ERRO DE SAÍDA ESTIMADA USANDO LBP NO BN DE DOIS VALORES [FIG. 5(B)] E BN DE QUATRO VALOR [FIG. 5(C)] PARA DIFERENTES PROBABILIDADES DE ERRO DE PORTÃO

$p_e$	2-valued BN	4-valued BN
$10^{-1}$	0.501	0.213
$10^{-3}$	0.492	$2.5 \times 10^{-3}$
$10^{-6}$	0.492	$2.5 \times 10^{-6}$
0	0.492	0

os pais de  $Z$  também estão livres de erros em relação a  $Q_1$ . Como tanto o CPD quanto os insumos em (10) estão livres de erros, usando a Proposição 2, a probabilidade marginal obtida após o produto da soma também está livre de erros. Assim, indo para cada cluster na ordem topológica das redes, todas as redes estão livres de erros em relação à distribuição aproximada obtida após o produto soma BP.

Em particular, isto também se aplica às OP. Portanto, a taxa de erro de saída em todos os POs é avaliada como zero.

Existem diversas variantes do produto soma BP que diferem na construção do gráfico de cluster utilizado. No LBP e suas variantes, cada nó no grafo de cluster contém uma variável e seus pais, e todos os conjuntos de conjuntos contém uma única variável. Em GBP e IJGP, clusters e conjuntos de conjuntos podem ter tamanhos maiores. A IBIA utiliza a soma do produto BP numa sequência de JTs aproximados. Dado que o algoritmo subjacente de passagem de mensagens é o mesmo, as estimativas obtidas com todos estes métodos obedecem ao Teorema 1. Em contraste, isto não é garantido para a formulação de dois valores. Como resultado, a taxa de erro estimada usando-o não se adapta bem às probabilidades de erro de porta, mesmo para o pequeno circuito da Fig. 5 (a), conforme mostrado na Tabela I.

Corolário 1: A taxa de erro de saída estimada após assumir a independência entre as entradas de uma porta na formulação de quatro valores é zero se todos os Pls e portas estiverem livres de erros.

Prova: No LBP, cada set-set contém uma única variável e as mensagens são em termos de marginais de variáveis. Cada CPD é multiplicado pelos marginais dos insumos para obter os marginais do produto. Isto é o mesmo que a estimativa dos marginais assumindo a independência entre os insumos. Portanto, usando o Teorema 1, a taxa de erro estimada é zero quando os Pls e as portas estão livres de erros.

Observe que isso também pode ser mostrado percorrendo as redes na ordem topológica e aplicando a Proposição 2.

#### D. Cálculo da Taxa de Erro do Circuito

Definição 5: Taxa de erro do circuito:  $(pE)$  É a probabilidade de ocorrer erro em pelo menos uma das saídas. Equivalentemente, a confiabilidade do circuito é  $1 - pE$ .  $pE$  é calculado após conectar uma árvore de portas OR aos POs, como mostrado na Figura 1 (c). Nós sabemos disso

$$\begin{aligned} pE &= P(E_1 \vee E_2 \vee \dots \vee E_m = 1) \\ &= 1 - P(E_1 = 0, E_2 = 0, \dots, E_m = 0). \end{aligned} \quad (11)$$

Aqui,  $\vee$  representa o operador OR. Portanto, a probabilidade de ocorrer um erro em qualquer uma das saídas pode ser calculada a partir da probabilidade de não haver erro em nenhuma das saídas. Na estrutura do BI, a probabilidade conjunta de não ocorrer nenhum erro em nenhuma das saídas pode ser calculada

definindo  $E1 = E2 = \dots = Em = 0$  como afirma a evidência e calculando a probabilidade da evidência. A probabilidade de evidência é referida como função de partição na literatura BN. O cálculo da função de partição pode ser feito usando algoritmos de BI exatos e também aproximados. Observe que o LBP para cálculo da função de partição normalmente exigirá um número maior de iterações para convergência e não é o mesmo que assumir independência entre as entradas da porta.

Para calcular a função de partição necessária usando a formulação de quatro valores, conectamos uma porta fictícia adicional a cada saída com um CPD associado  $P(E_i | Y_i)$  dado pela seguinte equação:

$P(\text{Não}   \text{Sim})$	Não = 0	Não = 1
$\neg Y_i = 00$	1	0
01	0	1
10	0	1
11	1	0

As saídas das portas fictícias podem ser definidas como estados de evidência.

4. RESULTADOS

Avaliamos a formulação proposta para análise de confiabilidade calculando taxas de erro de saída individuais e a taxa de erro geral do circuito para vários valores de probabilidades de erro de porta. Utilizamos circuitos pertencentes a dois conjuntos de benchmarks combinacionais, nomeadamente, ISCAS'85 [38] e os recentes benchmarks EPFL'15 [39]. Os benchmarks foram sintetizados usando a ferramenta Cadence Genus v15.2 usando a biblioteca de tecnologia Faraday 55 nm. Todos os experimentos foram realizados em um sistema Linux Intel i7-8700 de 3,7 GHz com 64 GB de memória.

Para calcular as taxas de erro das saídas, os PIs são considerados livres de erros, com probabilidade de sinal de 0.5. No entanto, o próprio modelo suporta qualquer probabilidade/confiabilidade de sinal para os PIs. Assumimos que as probabilidades de erro de todas as portas são iguais. Mais uma vez, o modelo suporta quaisquer probabilidades arbitrárias de erro de porta. Como em todos os trabalhos anteriores [4], [16], [40], portas com fan-in maior que dois são substituídas por combinações equivalentes de portas de duas entradas. Na combinação equivalente, todas as portas, exceto a porta que aciona a saída final, são consideradas livres de erros. Por exemplo, em uma porta AND de quatro entradas substituída por uma combinação de três portas AND de duas entradas, apenas a porta AND final está errada.

A. Linha de base

Para validar o modelo proposto, utilizamos taxas de erro obtidas na simulação lógica como linha de base. A simulação é realizada na configuração do circuito mostrada na Fig. 1(b), onde os PIs alimentam tanto o bloco lógico ideal (F) quanto o bloco lógico errôneo ( $\neg F$ ). Cada porta em  $\neg F$  possui um sinal de erro de entrada independente com probabilidade estática  $p_e$  como entrada. As saídas correspondentes de F e  $\neg F$  são submetidas a XOR para determinar as taxas de erro (ou equivalentemente as confiabilidades) das saídas do circuito. O número total de entradas do modelo é  $NPI + Ng$  onde NPI é o número de PIs e Ng é o número de portas no circuito. O número de amostras (Ns) utilizadas para a simulação deve ser

TABELA II

ERRO RELATIVO MÉDIO (EM %) NA TAXA DE ERRO DE SAÍDA E TEMPO DE EXECUÇÃO REQUERIDO PARA BENCHMARKS ISCAS'85 USANDO DIFERENTES TÉCNICAS DETERMINÍSTICAS DE INFERÊNCIA APROXIMADA NA FORMULAÇÃO BN DE QUATRO VALORES. O TAMANHO MÁXIMO DO CLUSTER PARA IBIA, WMB E IJGP FOI DEFINIDO PARA 10, E A PROFUNDIDADE DO LOOP PARA HAK FOI DEFINIDA PARA 3

Methods	Avg. Relative Error (%)		Runtime (s)	
	$p_e = 0.1$	$p_e = 10^{-3}$	$p_e = 0.1$	$p_e = 10^{-3}$
IBIA (10)	1.3	11.6	4	4
LBP*	3.8	25.9	0.006	0.006
HAK (LOOP3)*	4.1	44.4	259	468
IJGP (10)+	27.1	46.5	386	385
WMB (10)+	8.6	1750.4	620	619

\*: Methods implemented in tool libDAI [41]  
+: Methods implemented in tool Merlin [42]

grande o suficiente para garantir que as probabilidades de erro de porta e saída sejam bem estimadas. O tempo de execução necessário para as simulações MC é  $O(N_g N_s)$ .

Para muitos dos grandes benchmarks EPFL (multiplicador, div e assim por diante), a simulação de Monte Carlo (MC) com 106 não foi possível porque ficou sem memória. Portanto, escolhemos os resultados da simulação obtidos com 105 vetores como linha de base em todas as nossas avaliações. Isto implica que para taxas de erro de saída da ordem de  $10^{-3}$ , o desvio padrão  $((p(1 - p))/105)^{1/2}$  no valor estimado é uma ordem de grandeza menor. Contudo, taxas de erro mais baixas não podem ser estimadas de forma muito fiável. Também para uma probabilidade de erro de porta de  $10^{-6}$ , esta comparação não foi possível, uma vez que seriam necessárias pelo menos 107 amostras.

B. Escolha de Algoritmos de Inferência Determinística

Avaliamos técnicas determinísticas aproximadas de BI baseadas no produto soma BP que são implementadas em duas ferramentas publicamente disponíveis, libDAI [41] e Merlin [42], e um método mais recente IBIA [33] na formulação BN proposta. A Tabela II mostra a comparação do tempo de execução necessário e do erro relativo na taxa de erro média em todos os POs e todos os benchmarks ISCAS'85 para diferentes métodos de BI. O tamanho máximo do cluster foi definido como 10 para os métodos IBIA [33], WMB [32] e IJGP [31], e a profundidade do loop de 3 foi definida para HAK [43], que é a variante de loop duplo do GBP. Os parâmetros foram escolhidos com base nas restrições de tempo de execução e memória. Nestes métodos, o tempo de execução para inferência é exponencial no tamanho máximo do cluster. Para IBIA também depende do número de árvores de junção construídas, enquanto para HAK e IJGP depende do número de iterações até a convergência.

Para uma probabilidade de erro de porta de 0.1, o erro relativo médio é inferior a 5% com IBIA, LBP e HAK. Por outro lado, para probabilidade de erro de porta de  $10^{-3}$ , o IBIA fornece o menor erro seguido pelo LBP. Para WMB, a operação soma-produto é aproximada durante o cálculo das mensagens e não é garantido que satisfaça o Teorema 1. Como pode ser visto na tabela, ela não se adapta bem às probabilidades de erro de porta.

Para ambas as probabilidades de erro de porta, o tempo de execução obtido com LBP é o menor, seguido pelo IBIA. Pode-se observar na tabela que os tempos de execução são relativamente independentes das probabilidades de erro da porta. Escolhemos LBP e IBIA

TABELA III

COMPARAÇÃO DE ERRO MÁXIMO E RMSE OBTIDOS USANDO IBIA NO BNS DE DOIS E QUATRO VALORES PARA DIFERENTES PROBABILIDADES DE ERRO DE PORTA ( $p_e$ ). PARA  $p_e = 0$ , O BN DE QUATRO VALORES DÁ UMA TAXA DE ERRO DE ZERO PARA TODAS AS SAÍDAS. (A) ERRO MÁXIMO. (B) REQM

	$p_e = 0.1$		$p_e = 0.001$		$p_e = 0$
	2-valued	4-valued	2-valued	4-valued	
c432	0.038	0.026	0.102	4.06-4	0.408
c499	0.007	0.007	0.124	0.001	0.5
c880	0.070	0.007	0.142	$9.81 \times 10^{-4}$	0.23
c1355	0.002	0.006	0.007	$7.71 \times 10^{-4}$	0.014
c1908	0.293	0.005	0.441	$7.18 \times 10^{-4}$	0.5
c2670	0.060	0.003	0.489	0.003	0.5
c3540	0.076	0.037	0.489	0.057	0.5
c5315	0.086	0.032	0.491	0.004	0.5
c6288	0.022	0.004	0.453	0.177	0.5
c7552	0.084	0.033	0.490	0.014	0.5
Mean	0.07	0.02	0.365	0.026	0.415

	$p_e = 0.1$		$p_e = 0.001$		$p_e = 0$
	2-valued	4-valued	2-valued	4-valued	
c432	0.019	0.017	0.044	$2.21 \times 10^{-4}$	0.172
c499	0.004	0.005	0.071	0.001	0.126
c880	0.015	0.002	0.044	$3.72 \times 10^{-4}$	0.073
c1355	0.001	0.003	0.007	$4.15 \times 10^{-4}$	0.008
c1908	0.055	0.002	0.261	$2.82 \times 10^{-4}$	0.27
c2670	0.008	$5.77 \times 10^{-4}$	0.075	$2.89 \times 10^{-4}$	0.078
c3540	0.029	0.020	0.377	0.017	0.4
c5315	0.024	0.006	0.153	0.001	0.163
c6288	0.004	0.002	0.375	0.083	0.461
c7552	0.022	0.005	0.196	0.002	0.214
Mean	0.024	0.006	0.166	0.01	0.196

para avaliação neste trabalho, uma vez que esses métodos fornecem boas precisões com tempos de execução razoáveis.

C. Comparação com a formulação BN de dois valores

Conforme mencionado anteriormente, nos referimos ao modelo da Figura 5 (b) como o modelo de dois valores e ao modelo da Figura 5 (c) como o modelo de quatro valores. A Tabela III apresenta o erro máximo e a raiz quadrada média (rms) na estimativa das taxas de erro de produto para os benchmarks ISCAS'85 usando as formulações de dois valores e de quatro valores. Como os erros obtidos usando a formulação de dois valores foram muito grandes, relatamos o valor absoluto dos erros em vez dos erros relativos. Tabulamos os resultados obtidos com o IBIA. As estimativas obtidas com o LBP também seguem uma tendência semelhante. A tabela apresenta os resultados para probabilidades de erro de porta,  $p_e = 0.1, 10^{-3}$  e 0. Vê-se que o cálculo usando a formulação de dois valores não se adapta bem à probabilidade de erro de porta ( $p_e$ ). Para  $p_e$  de  $10^{-3}$ , o erro máximo para muitas saídas é próximo de 0.5, o que é apenas ruído. Isto também é verdade para  $p_e = 0$ . Tanto o erro máximo quanto o RMSE são significativamente mais baixos quando a formulação de quatro valores é usada. As taxas de erro de saída são zero quando  $p_e$  é 0, conforme garantido pelo Teorema 1. A Fig. 6 mostra as taxas de erro em várias saídas no benchmark c5315 para diferentes probabilidades de erro de porta e vê-se que ela se adapta bem às probabilidades de erro de porta.

À medida que a probabilidade de erro da porta diminui, a correlação entre o sinal livre de erros e o sinal errôneo correspondente aumenta. Para variáveis aleatórias discretas, a informação mútua (MI) entre os dois sinais é uma medida desta

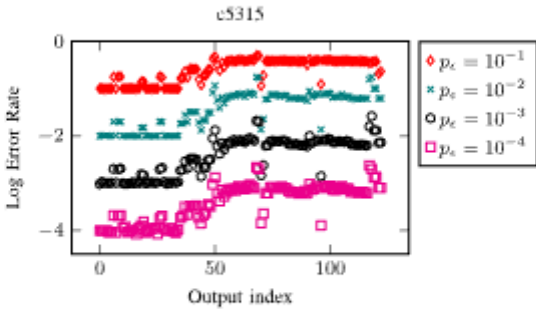


Figura 6. Taxa de erro de saída inferida usando IBIA no BN de quatro valores formulação para diferentes probabilidades de erro de porta.

TABELA IV

INFORMAÇÃO MÉDIA MÚTUA ENTRE AS REDES LIVRES DE ERROS E AS REDES ERRADAS CORRESPONDENTES PARA VÁRIAS PROBABILIDADES DE ERRO DE PORTÃO ( $p_e$ )

	$p_e = 0.1$	$p_e = 10^{-3}$	$p_e = 10^{-4}$
c17	0.055	0.148	0.151
adder	0.088	0.205	0.21
alu_ctrl	0.046	0.154	0.159
dec	0.012	0.043	0.046
im256cnt	0.044	0.153	0.159
priority	0.064	0.133	0.136
router	0.049	0.089	0.091

correlação. O MI entre um sinal S e o sinal errado correspondente  $\hat{S}$  é definido como segue:

$$MI = \sum_{\hat{S}} \sum_S P(S, \hat{S}) \log \frac{P(S, \hat{S})}{P(S)P(\hat{S})}. \quad (12)$$

A Tabela IV mostra o MI médio entre os sinais correspondentes para alguns benchmarks pequenos, calculados usando o IBIA. Para estes benchmarks, a IBIA realiza uma inferência exata ou o erro na estimativa é muito pequeno (da ordem de  $10^{-4}$ ). É visto claramente que o MI aumenta à medida que a probabilidade de erro da porta diminui. Este aumento do MI é resolvido na formulação de quatro valores, mas a formulação de dois valores luta (e não consegue) capturá-lo. Como pode ser visto na Tabela III, o erro de estimativa obtido com o modelo de dois valores aumenta drasticamente à medida que a probabilidade de erro da porta é reduzida.

D. Taxa de erro de saída com o modelo BN de quatro valores

A Tabela V apresenta o erro relativo na taxa de erro calculada nos POs e no tempo de execução para vários benchmarks. O erro relativo médio (máximo) é a média (máximo) do erro relativo absoluto sobre todas as saídas. O erro percentual na confiabilidade correspondente ( $r_Y$ ) é mostrado entre parênteses. Para o IBIA, usamos tamanhos de cluster (ou seja, o número máximo de variáveis em um cluster) de 7 e 10. Os resultados são relatados para duas probabilidades de erro de porta diferentes ( $p_e = 0.1, 10^{-3}$ ). A tabela também mostra o número de POs e o número total de portas obtidas após a substituição de portas com fan-in maior que dois com combinações equivalentes de portas de duas entradas.

Tempo de execução: Para ambos os valores de  $p_e$ , o mesmo número de amostras foi utilizado para simulação lógica. Conforme discutido, o tempo de execução do IBIA e do LBP também é relativamente independente das probabilidades de erro de porta. Portanto, a tabela possui apenas um conjunto de tempos de execução.



TABELA V

ESTATÍSTICAS DE BENCHMARKS UTILIZADOS PARA AVALIAÇÃO E TRADE-OFF ENTRE TEMPO(S) DE EXECUÇÃO E PRECISÃO EM TERMOS DE ERRO RELATIVO MÉDIO E MÁXIMO (%) NA TAXA DE ERRO (ER) USANDO IBIA E LBP. O ERRO RELATIVO DE CONFIABILIDADE (R) ESTÁ INDICADO ENTRE PARÊSES. OS RESULTADOS OBTIDOS COM LBP, IBIA USANDO MAX-CLUSTER SIZE DE 7 E 10, E SIMULAÇÕES MC SÃO MOSTRADOS NAS COLUNAS MARCADAS COMO "L", "I7", "I10" E "MC", RESPECTIVAMENTE. OS CIRCUITOS ISCAS'85 E EPFL'15 SÃO SEPARADOS POR UMA LINHA

Circuit	#Gates	#POs	Average Relative Error - ER (R) (%)						Maximum Relative Error - ER (R) (%)						Runtime (s)			
			p <sub>e</sub> = 0.1			p <sub>e</sub> = 10 <sup>-8</sup>			p <sub>e</sub> = 0.1			p <sub>e</sub> = 10 <sup>-8</sup>			L*	I7*	I10*	MC†
			L	I7	I10	L	I7	I10	L	I7	I10	L	I7	I10				
c432	182	7	13	4	4	30 (0.4)	3 (0.04)	1 (0.02)	40	8	7	61 (0.9)	7 (0.1)	3 (0.04)	0.03	0.4	0.9	16
c499	189	32	2	2	2	34 (0.1)	24 (0.08)	26 (0.08)	3	3	3	41 (0.1)	31 (0.1)	34 (0.1)	0.03	0.6	1.1	20
c1355	209	32	2	2	1.2	25 (0.1)	22 (0.07)	10 (0.03)	4	3	2	44 (0.1)	31 (0.1)	27 (0.08)	0.03	0.7	1.3	20
c1908	285	25	2	0.4	0.4	4 (0.05)	4 (0.03)	2 (0.02)	11	1.3	1.2	36 (0.6)	10 (0.2)	6 (0.07)	0.04	0.9	2	23
c880	308	26	2	0.9	0.5	8 (0.1)	4 (0.02)	4 (0.02)	8	4	2	27 (0.2)	23 (0.2)	18 (0.1)	0.05	0.8	2	24
c2670	613	139	1	0.8	0.1	2 (0.02)	1.5 (0.02)	0.7 (0.01)	20	22	0.8	27 (0.5)	25 (0.4)	15 (0.3)	0.1	1.2	2	52
c3540	930	22	6	5	4	25 (1)	24 (1)	23 (0.9)	35	38	35	78 (9)	75 (8)	74 (6)	0.1	4	8	53
c7552	1203	108	2	0.8	1	7 (0.1)	5 (0.1)	4 (0.08)	11	7	8	52 (3)	45 (2)	33 (2)	0.2	5	7	97
c5315	1232	123	2	1.4	1.2	8 (0.1)	6 (0.05)	4 (0.04)	14	21	19	42 (0.5)	50 (0.5)	48 (0.41)	0.2	4	8	79
c6288	2166	32	8	0.7	0.3	138 (12)	89 (11)	52 (7)	39	14	1.1	466 (28)	178 (30)	117 (21)	0.3	5	10	227
alu_ctrl	110	25	1.0	1.1	0.9	1 (0.01)	3 (0.01)	4 (0.01)	4	4	4	9 (0.04)	9 (0.04)	9 (0.04)	0.02	0.2	0.5	10
int2float	263	7	2	1.0	0.7	1 (0.01)	2 (0.01)	1 (0.01)	6	3	2	3 (0.01)	4 (0.03)	3 (0.03)	0.04	0.9	3	16
cas1c	825	11	9	9	9	14 (0.1)	13 (0.1)	14 (0.1)	30	30	30	49 (0.4)	49 (0.4)	49 (0.4)	0.1	7	17	41
i2c	1018	141	2	1.2	1.1	4 (0.02)	3 (0.01)	3 (0.01)	10	7	5	24 (0.1)	15 (0.08)	15 (0.07)	0.2	4	8	62
bar	2933	128	22	22	20	43 (0.7)	43 (0.7)	41 (0.7)	23	23	23	46 (0.8)	46 (0.8)	45 (0.8)	0.5	53	74	168
max	3413	130	8	2	2	100 (2)	31 (0.5)	15 (0.2)	11	7	5	150 (2)	74 (1)	33 (0.4)	0.5	40	24	303
arbiter	4938	129	2	2	2	8 (0.03)	7 (0.02)	6 (0.02)	7	7	6	19 (0.1)	18 (0.05)	17 (0.05)	1	86	188	288
sin	5179	25	0.3	0.4	0.4	59 (19)	59 (20)	54 (19)	2	2	2	128 (35)	114 (33)	107 (33)	1	88	111	520
voter	8369	1	2	2	2	107 (33)	74 (23)	53 (17)	2	2	2	107 (33)	74 (23)	53 (17)	1	25	30	749
square	17886	127	0.4	0.3	0.3	102 (21)	75 (16)	73 (15)	7	6	6	172 (34)	129 (30)	133 (29)	3	5528	1925	1616
multiplicar	21875	128	1	0.3	0.3	100 (24)	100 (25)	89 (23)	5	5	5	239 (38)	192 (38)	195 (38)	4	1162	2404	2263
div	36218	128	2	1.1	1.1	37 (7)	22 (4)	21 (4)	31	13	10	266 (22)	89 (13)	117 (13)	6	447	683	3377
Mean			4.2	2.7	2.4	39.0 (5.5)	27.9 (4.7)	22.2 (4.0)	14.7	10.3	8.2	94.8 (9.4)	58.2 (8.3)	52.4 (7.3)				

\*Runtimes with Python implementations for LBP, IBIA.  
†Runtimes with Cadence Incisive v15.2; Number of vectors=10<sup>6</sup>

Como pode ser visto na tabela, o tempo de execução para simulações LBP e MC é aproximadamente linear no número de portas, o que é esperado. Existem alguns desvios nas simulações MC, uma vez que usamos o Cadence Incisive, que é um simulador orientado a eventos. O tempo de execução do IBIA também aumenta com o número de portas. Mas a dependência exata é mais difícil de prever. Conforme mencionado anteriormente, o IBIA constrói uma sequência de árvores de junção. À medida que Ng aumenta, o número de árvores de junção também aumenta, mas o número exato depende da estrutura do gráfico e dos loops reconvergentes nele. Devido a isso, existem algumas flutuações nos tempos de execução.

Para o mesmo circuito, o tempo de execução do IBIA aumenta com o tamanho máximo do cluster em quase todos os casos, conforme esperado. Para esses tamanhos de cluster, o tempo necessário é dominado pelo tempo necessário para construir as árvores de junção, e não pelo tempo de inferência. As exceções ocorrem porque o IBIA leva mais tempo para construir uma árvore de junção com o tamanho máximo de cluster especificado.

O tempo de execução do IBIA é muito melhor do que as simulações MC para a maioria dos circuitos e é comparável para alguns benchmarks grandes (quadrado e multiplicador). No entanto, se a probabilidade de erro de porta for reduzida ainda mais, o tempo de execução do IBIA e do LBP não mudará muito, mas as simulações MC levarão tempos significativamente maiores, uma vez que o número de amostras necessárias seria maior. Por exemplo, o tempo de execução médio em todos os benchmarks para probabilidades de erro de porta de 10<sup>-1</sup>, 10<sup>-3</sup> e 10<sup>-6</sup> com IBIA é 276, 240 e 303 s, respectivamente. Mas a simulação lógica completa não é possível para p<sub>e</sub> = 10<sup>-6</sup>.

Precisão: Para p<sub>e</sub> = 0,1, o erro relativo máximo médio reduz de 15% com LBP para 8% com IBIA usando um tamanho de cluster de 10. No entanto, o erro relativo médio médio

obtido com todos os três métodos é pequeno (2%–4%). Portanto, para esta e maiores probabilidades de erro de porta, estimativas razoavelmente precisas podem ser obtidas em tempos de execução muito pequenos se assumirmos independência entre as entradas de uma porta.

Quando as probabilidades de erro de porta são reduzidas para 10<sup>-3</sup>, o erro relativo nas estimativas das taxas de erro é maior, com o IBIA fornecendo estimativas significativamente melhores do que o LBP, com um erro relativo máximo médio de 52% e 95%, respectivamente. Para alguns benchmarks grandes em circuitos aritméticos EPFL, o erro relativo máximo é superior a 100%. Contudo, o erro relativo nas fiabilidades correspondentes (mostradas entre parênteses) é muito menor. Isso ocorre porque para probabilidades de erro de porta pequenas, a taxa de erro de saída é pequena e a confiabilidade correspondente é grande. Portanto, embora o erro absoluto tanto na confiabilidade quanto na taxa de erro seja o mesmo, o erro percentual em relação à confiabilidade é muito menor. Relatamos ambos porque alguns dos trabalhos anteriores relatam resultados com taxas de erro, enquanto outros relatam confiabilidade. Em média, o erro relativo máximo e médio na confiabilidade obtido com o IBIA é de cerca de 7% e 4%, respectivamente. Como esperado, para ambas as probabilidades de erro de porta, observamos que a precisão melhora à medida que o tamanho do cluster aumenta.

Entre os métodos existentes [4], [23], [24], [25], [44], os resultados foram relatados para benchmarks ISCAS'85 com probabilidades de erro de porta de 0,01 ou maiores. Não vimos nenhum resultado para grandes benchmarks EPFL para quaisquer probabilidades de erro de porta. Uma comparação justa dos resultados obtidos com os métodos existentes é difícil, uma vez que as netlists sintetizadas podem variar em cada caso. No entanto, apenas para comparar a tendência geral, observamos que o erro relativo médio na fiabilidade das OP reportadas com estes métodos está entre 1,2% e 3,8%.

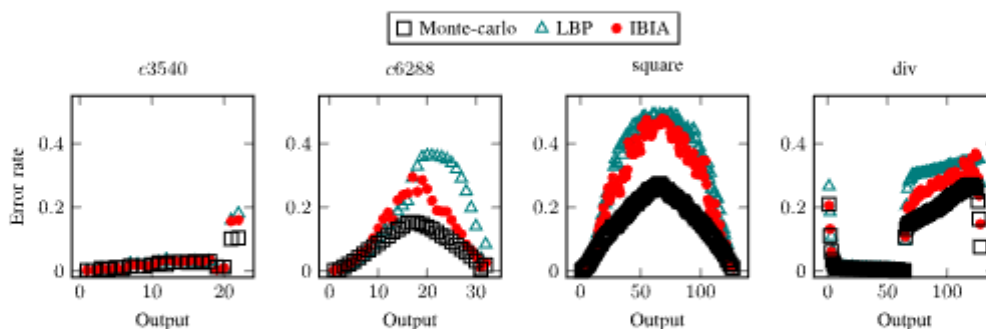


Figura 7. Taxa de erro nos POs obtidos usando simulações MC, IBIA e LBP para  $p_e = 10^{-3}$ .

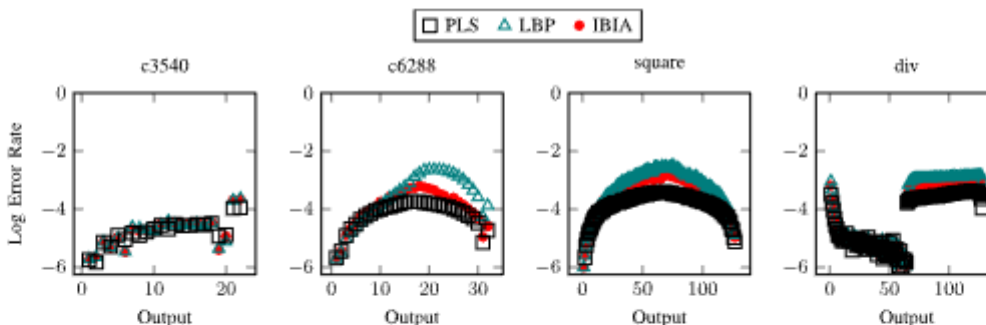


Figura 8. Taxa de erro nos POs obtidos usando PLS, IBIA e LBP para  $p_e = 10^{-6}$ .

para probabilidades de erro de porta de 0,1 e 0,01. Em contraste, o erro médio de confiabilidade com a formulação proposta é de 0,7% com IBIA (tamanho do cluster = 10) e 2,5% se assumirmos independência entre as entradas de uma porta. Muito poucos resultados estão disponíveis para probabilidades de erro de porta  $p_e \leq 0,001$ . Tanto [4] quanto [20] relatam erros calculados em média em várias probabilidades de erro de porta. Assim, uma comparação direta não é possível. Em [20], com probabilidade de erro de porta zero, o erro relativo na taxa de erro é de 45% para um circuito pequeno como c2670. Em contraste, mostramos que nosso método garante taxas de erro zero se algoritmos baseados na soma do produto BP (como LBP, GBP e IBIA) forem usados para inferência.

é ainda reduzido para  $p_e = 10^{-6}$ . As simulações MC não foram possíveis para esta probabilidade de erro de porta devido a erros de memória. Portanto, utilizamos PLS [14], uma técnica de BI aproximada baseada em amostragem. A implementação do PLS foi retirada do kit de ferramentas SMILE [45], e o número de amostras foi definido como 107. Observa-se que a taxa de erro estimada aumenta bem quando a probabilidade de erro da porta é reduzida. Mais uma vez, a relativa fiabilidade entre os resultados é bem captada tanto pelo LBP como pela IBIA. As taxas de erro obtidas com o LBP são consistentemente maiores do que com o IBIA. Embora o PLS seja possivelmente mais preciso, o tempo de execução é cerca de uma ordem de magnitude maior que o IBIA. Por exemplo, o benchmark div requer 440 minutos com PLS, enquanto o tempo de execução com IBIA é de apenas 11 minutos.

A Figura 7 mostra as taxas de erro calculadas em cada PO usando MC, IBIA (com tamanho de cluster = 10) e LBP para alguns dos benchmarks nos quais o erro percentual na taxa de erro estimada é grande. Verifica-se que, embora a IBIA e o LBP sobrestimem as taxas de erro, são capazes de captar muito bem as taxas de erro dos resultados relativamente uns aos outros na maioria dos casos. Em alguns casos de teste como div e c6288, a precisão melhora quando clusters maiores são usados. A precisão obtida com IBIA é comparável ao LBP para quadrado e c3540. Como a taxa de erro relativa entre os resultados é aproximadamente preservada por ambos os métodos de inferência (com maior precisão pelo IBIA do que pelo LBP), estes podem ser usados para identificar os resultados que são mais suscetíveis a erros. Isso permitirá a aplicação direcionada de técnicas para melhorar a confiabilidade do circuito.

A Figura 8 mostra o logaritmo das taxas de erro nos POs obtidos usando LBP e IBIA quando a probabilidade de erro da porta

E. Taxa de erro de circuito com o modelo BN de quatro valores

A Tabela VI mostra o erro relativo na taxa de erro do circuito (Definição 5) para  $p_e = 10^{-3}$ . Também mostra as estimativas obtidas após as simulações MC na configuração do circuito mostrada na Fig. Para  $p_e = 0,1$ , a taxa de erro do circuito

torna-se um e ambos os métodos fornecem estimativas quase precisas. A tabela apresenta uma comparação dos dois métodos utilizados, a saber, (a) conectar uma árvore de portas OR e (b) usar o cálculo da função de partição [ver (11)]. Verifica-se que o erro relativo na estimativa é significativamente menor quando a formulação PR é utilizada para ambos os métodos, especialmente para os benchmarks ISCAS'85 mais pequenos. Embora a média de todos os benchmarks seja comparável, verifica-se que o LBP

TABELA VI  
TAXA DE ERRO DE CIRCUITO (pE) OBTIDA USANDO SIMULAÇÕES MC, O(S) TEMPO(S) DE EXECUÇÃO NECESSÁRIO(S) E O ERRO RELATIVO (EM%) NA TAXA DE ERRO DE CIRCUITO OBTIDA USANDO DOIS MÉTODOS DE INFERÊNCIA. IBIA/LBP-OR REFERE-SE A COMPUTAÇÃO APÓS CONECTAR UMA ÁRVORE DE PORTÕES E IBIA/LBP-PR REFERE-SE A COMPUTAÇÃO USANDO A FUNÇÃO DE PARTIÇÃO

	pE	Relative Error (%)				Runtime (s)	
		MC	IBIA-OR	IBIA-PR	LBP-OR	LBP-PR	LBP-PR <sup>1</sup> IBIA-PR <sup>2</sup>
c432	0.041	32	4	169	23	0.02	1
c499	0.087	21	18	47	5	0.02	1
c880	0.094	11	5	25	3	0.04	2
c1355	0.085	6	5	40	4	0.02	1
c1908	0.101	35	15	62	4	0.04	2
c2670	0.210	21	0.3	46	2	0.1	3
c3540	0.186	53	27	179	2	0.2	10
c5315	0.329	40	7	45	1	0.2	8
c6288	0.778	23	21	28	34	0.8	11
c7552	0.339	43	23	77	3	0.2	7
alu_cnf	0.059	23	1	46	2	0.01	1
arbiter	0.352	7	7	9	5	0.5	172
cavlc	0.164	6	7	7	5	0.1	18
i2c	0.317	33	12	44	1	0.1	7
int2float	0.046	8	4	9	1	0.03	3
voter	0.238	53	53	107	104	12	31
bar	0.679	0	9	2	31	1	71
nux	0.263	175	97	275	5	1	35
sin	0.864	16	16	16	0.3	3	110
div	0.978	2	2	2	9	224	775
square	1.000	0	0	0	0	6	3526
multiplier	1.000	0	0	0	0	14	5516
Mean		27.6	15.2	56.1	11.1	12	469

<sup>1</sup> Runtime with C++ implementation of LBP from libDAI  
<sup>2</sup> Runtime with Python implementation of IBIA

fornece erros relativos significativamente mais baixos para um grande número de benchmarks. Para casos de teste quadrados e multiplicadores, a taxa de erro do circuito é muito próxima de um e ambos os métodos funcionam bem. Embora o LBP exija mais iterações para convergir quando a formulação PR é usada, ele é bastante rápido, tornando-o adequado para uso em uma estrutura de otimização. Espera-se que o tempo de execução do IBIA seja maior, pois ele usa clusters maiores. Dito isto, os tempos de execução relatados para LBP e IBIA não são diretamente comparáveis, uma vez que ambos são implementados em linguagens de programação diferentes. O tempo médio de execução para LBP e IBIA para pE = 0.1 é de 17 e 377 s, respectivamente, o que é semelhante aos tempos de execução para pE = 0,001.

V. COMPARAÇÃO COM TRABALHOS RELACIONADOS  
Comparamos nossa abordagem com os métodos existentes que

use (a) uma formulação de quatro valores ou (b) técnicas de BI para estimativa de confiabilidade.

Várias abordagens existentes [4], [15], [20], [21], [23], [24], [25] usam uma única cópia do circuito com probabilidades adicionais para cada rede. O problema aqui é a propagação precisa dessas probabilidades na presença de fanouts reconvergentes. Estimativas rápidas podem ser obtidas se as entradas da porta forem consideradas independentes [15], [21]. No entanto, como pode ser visto nos resultados, a precisão deste método diminui à medida que a probabilidade de erro da porta diminui. Os métodos em [4] e [20] usam uma extensão do método do coeficiente de correlação (CCM) para calcular coeficientes de correlação de sinal. Em [23], [24] e [25], os coeficientes de correlação são calculados em relação à confiabilidade do sinal. Essas correlações são estimadas usando métodos analíticos [4], [16], [23] ou métodos baseados em simulação [20], ou métodos híbridos que combinam essas duas abordagens [46]. Normalmente, correlações entre pares são calculadas, o que limita a precisão dessas

métodos. Em [20] e [46], a precisão é limitada pelo comprimento dos fluxos de bits usados para estimar os coeficientes de correlação. Os métodos propostos em [4], [23], [24], [25] e [44] requerem o cálculo de probabilidades de sinal em circuitos livres de erros para estimar correlações de confiabilidade. Este em si é um problema #P-completo. Estimativas precisas obtidas usando BDDs foram usadas em [4]. No entanto, isto limita a escalabilidade do método a circuitos relativamente pequenos. Por outro lado, nosso método não requer essas probabilidades. Evitamos imprecisões na estimativa dos coeficientes de correlação derivando CPDs correspondentes aos sinais de quatro valores e usando técnicas de BI que fornecem distribuição conjunta aproximada em grupos maiores de variáveis. Ao contrário das abordagens existentes, em nossa abordagem é possível equilibrar tempo de execução e precisão aumentando o tamanho dos clusters.

As técnicas de BI para análise de confiabilidade foram utilizadas em [5], [11], [12], [47] e [48]. Em [5], a formulação de dois valores foi usada juntamente com as técnicas de inferência aproximada baseadas em amostragem. Tal como acontece com todos os métodos de amostragem, esta abordagem é inflexível no sentido de que qualquer alteração no circuito requer uma reavaliação completa e torna-se dispendiosa à medida que a probabilidade de erro da porta diminui. Como a complexidade de tempo para técnicas de amostragem aumenta linearmente com o número de nós na rede, ela pode ser reduzida à metade usando o BN de quatro valores proposto em vez da formulação de dois valores. Métodos exatos de BI também foram usados para a estimativa de probabilidades de erro de porta com base em parâmetros de nível de dispositivo [11], [47] e para o cálculo de limites de confiabilidade, identificando o vetor de entrada do pior caso [12], [48]. No entanto, a inferência exata só é possível para circuitos pequenos.

VI. CONCLUSÃO

Propomos um novo algoritmo para a estimativa da taxa de erro/confiabilidade em circuitos probabilísticos e não confiáveis. Nosso método é bem dimensionado com probabilidades de erro de porta e preserva a confiabilidade relativa das saídas. Também propomos um novo método para calcular a taxa geral de erro do circuito, lançando-o como um problema de estimativa da função de partição em BNs. Esta formulação fornece boas precisões dentro de tempos de execução razoáveis, tornando-a adequada para uso em uma estrutura de otimização. Embora tenhamos demonstrado resultados para circuitos CMOS, os métodos propostos são gerais e podem ser usados para circuitos construídos com dispositivos pós-CMOS. Eles também podem ser usados para análise e projeto de circuitos aproximados. Para estes circuitos, a formulação de quatro valores pode ser aplicada diretamente às tabelas verdade precisas e imprecisas, uma vez que a taxa de erro é independente da implementação.

APÊNDICE

Uma distribuição condicional P(Y|X) é um fator  $\phi$  sobre o conjunto de variáveis  $Y \cup X$ . Sejam  $X, Y, Z$  conjuntos disjuntos de variáveis e  $\phi_1(X, Y), \phi_2(Y, Z)$  sejam dois fatores. O produto fatorial [26, cap. 4]  $\phi_1\phi_2$  fornece um fator  $\psi$  que é obtido da seguinte forma:  $\forall x, y, z \in \text{Domínio}(X, Y, Z)$

$$\psi(X, Y, Z = x, y, z) = \phi_1(x, y)\phi_2(y, z).$$
 Se os fatores  $\phi_1$  e  $\phi_2$  contêm conjuntos disjuntos de variáveis, então

o produto fatorial é igual ao produto tensorial.

## REFERÊNCIAS

- [1] J. Von Neumann, "Lógica probabilística e a síntese de organismos confiáveis a partir de componentes não confiáveis", *Automata Stud.*, vol. 34, pp. 43–98, dezembro de 1956.
- [2] L. Tan, Z. Li, G. Su e D. Wang, "Análise assintoticamente linear e esquemas de alocação de probabilidade de porta em circuitos probabilísticos", *IEEE Trans. Integridade em escala muito grande. (VLSI) Sistema*, vol. 28, não. 2, pp. 596–606, fevereiro de 2020.
- [3] N.-Z. Lee e J.-H.-R. Jiang, "Rumo à avaliação formal e verificação do projeto probabilístico", *IEEE Trans. Computação*, vol. 67, não. 8, pp. 1202–1216, agosto de 2018.
- [4] MR Choudhury e K. Mohanram, "Análise de confiabilidade de circuitos lógicos", *IEEE Trans. Integração de Design Auxiliado por Computação. Sistema de Circuitos*, vol. 28, não. 3, pp. 392–405, março de 2009.
- [5] T. Rejimon, K. Lingasubramanian e S. Bhanja, "Modelagem de erro probabilístico para circuitos lógicos de nanodomínio", *IEEE Trans. Integridade em escala muito grande. (VLSI) Sistema*, vol. 17, não. 1, pp. 55–65, janeiro de 2009.
- [6] B. Krishnamurthy e IG Tollis, "Técnicas aprimoradas para estimar probabilidades de sinal", *IEEE Trans. Computação*, vol. 38, não. 7, pp. 1041–1045, julho de 1989.
- [7] S. Krishnaswamy, GF Viamontes, IL Markov e JP Hayes, "Avaliação e aprimoramento de confiabilidade precisa por meio de matrizes de transferência probabilística", em *Proc. Projeto, Autom. Teste Eur.*, março de 2005, pp.
- [8] S. Krishnaswamy, GF Viamontes, IL Markov e JP Hayes, "Matrizes de transferência probabilística na análise de confiabilidade simbólica de circuitos lógicos", *ACM Trans. Des. Automático. Elétron. Sistema*, vol. 13, não. 1, pp. 1–35, 2008.
- [9] J. Han, H. Chen, E. Boykin e J. Fortes, "Avaliação de confiabilidade de circuitos lógicos usando modelos de portas probabilísticas", *Microelectron. Rel.*, vol. 51, não. 2, pp.
- [10] O. Keszocze, "Análise, computação e otimização de métricas de erro baseadas em BDD", *IEEE Access*, vol. 10, pp.
- [11] W. Ibrahim, V. Beiu e A. Beg, "GREDA: Uma ferramenta EDA de confiabilidade de porta rápida e mais precisa", *IEEE Trans. Integração de Design Auxiliado por Computação. Sistema de Circuitos*, vol. 31, não. 4, pp. 509–521, abril de 2012.
- [12] W. Ibrahim, M. Shousha e JW Chinnneck, "Estimativa precisa e eficiente dos limites de confiabilidade de circuitos lógicos", *IEEE Trans. Computação*, vol. 64, não. 5, pp. 1217–1229, maio de 2015.
- [13] J. Han, H. Chen, J. Liang, P. Zhu, Z. Yang e F. Lombardi, "Uma abordagem computacional estocástica para avaliação de confiabilidade precisa e eficiente", *IEEE Trans. Computação*, vol. 63, não. 6, pp. 1336–1350, junho de 2014.
- [14] M. Henrion, "Propagando incerteza em redes bayesianas por amostragem lógica probabilística", em *Incerteza em Inteligência Artificial (Machine Intelligence & Pattern Recognition)*, vol. 5, JF Lemmer e LN Kanal, Eds. Amsterdã, Holanda: Holanda do Norte, 1988, pp.
- [15] DT Franco, MC Vasconcelos, L. Naviner e J.-F. Naviner, "Análise de confiabilidade de circuitos lógicos com base na probabilidade de sinal", em *Proc. 15º Int. IEEE. Conf. Electron., Circuits Syst.*, agosto de 2008, pp.
- [16] S. Ercolani, M. Favalli, M. Damiani, P. Olivo e B. Ricco, "Estimativa de probabilidade de sinal em redes lógicas combinacionais", em *Proc. 1st Eur Test Conf.*, janeiro de 1989, pp. 132–133.
- [17] S. Ercolani, M. Favalli, M. Damiani, P. Olivo e B. Ricco, "Medidas de testabilidade em testes pseudoaleatórios", *IEEE Trans. Integração de Design Auxiliado por Computação*, vol. 11, não. 6, pp. 794–800, junho de 1992.
- [18] S. Sivaswamy, K. Bazargan e M. Riedel, "Estimativa e otimização da confiabilidade de circuitos digitais ruidosos", em *Proc. 10º Int. Simp. Elétron de qualidade. Design*, março de 2009, pp.
- [19] N. Mohyuddin, E. Pakbaznia e M. Pedram, "Propagação de erro probabilístico em um circuito lógico usando o cálculo de diferença booleana", em *Técnicas Avançadas em Síntese Lógica, Otimizações e Aplicações*. Cham, Suíça: Springer, 2011, pp.
- [20] H. Jahanirad, "CC-SPRA: Abordagem de coeficientes de correlação para análise de confiabilidade baseada em probabilidade de sinal", *IEEE Trans. Integridade em escala muito grande. (VLSI) Sistema*, vol. 27, não. 4, pp. 927–939, abril de 2019.
- [21] DT Franco, MC Vasconcelos, L. Naviner e J.-F. Naviner, "Confiabilidade de circuitos lógicos sob múltiplas falhas simultâneas", em *Proc. 51º Simpósio Centro-Oeste. Circuits Syst.*, agosto de 2008, pp.
- [22] JT Flaquer, JM Daveau, L. Naviner e P. Roche, "Análise rápida de confiabilidade de circuitos lógicos combinatórios usando probabilidades condicionais", *Microelectron. Rel.*, vol. 50, pp. 1215–1218, setembro/novembro. 2010.
- [23] C. Chen e R. Xiao, "Um modelo rápido para análise e melhoria da confiabilidade do circuito em nível de porta", *Integração*, vol. 50, pp. 107–115, junho de 2015.
- [24] J. Cai e C. Chen, "Análise de confiabilidade de circuito usando correlações de confiabilidade de sinal", em *Proc. Internacional IEEE. Conf. Suave. Qual., Rel. Seguro. Companheiro (QRS-C)*, julho de 2017, pp.
- [25] K. Sikander, S. Zhan e C. Chen, "Um modelo analítico para estimativa de confiabilidade de circuito", em *Proc. Internacional IEEE. Simpósio Centro-Oeste. Sistema de Circuitos (MWSCAS)*, agosto de 2021, pp.
- [26] D. Koller e N. Friedman, *Modelos Gráficos Probabilísticos: Princípios e Técnicas*. Cambridge, MA, EUA: MIT Press, 2009.
- [27] KP Murphy, Y. Weiss e MI Jordan, "Propagação de crença loopy para inferência aproximada: um estudo empírico", em *Incerteza em Inteligência Artificial*. Amsterdã, Holanda: Elsevier, 1999, pp.
- [28] W. Wiegnerinck e T. Heskes, "Propagação de crenças fracionárias", em *Advances in Neural Information Processing Systems*, vol. 15. Cambridge, MA, EUA: MIT Press, 2003.
- [29] MJ Wainwright, TS Jaakkola e AS Willsky, "Algoritmos de propagação de crenças reponderadas em árvore e estimativa aproximada de ML por correspondência de pseudo-momento", em *Proc. Internacional Oficina Artif. Intel. Estatista.*, 2003, pp.
- [30] JS Yedidia, WT Freeman e Y. Weiss, "Propagação de crenças generalizadas", em *Proc. NIPS*, vol. 13, 2000, pp.
- [31] R. Mateescu, K. Kask, V. Gogate e R. Dechter, "Algoritmos de propagação de gráfico de junção", *J. Artif. Intel. Res.*, vol. 37, pp. 279–328, março. 2010.
- [32] Q. Liu e A. Ihler, "Limitando a função de partição usando a desigualdade de titular", em *Proc. Internacional Conf. Mach. Aprenda.*, 2011, pp. [33] S. Bathla e V. Vasudevan, "IBIA: inferência bayesiana via incremento operações mentais de construção e inferência aproximadas em árvores de cliques", 2022, *arXiv:2202.12003*.
- [34] M. Chavira e A. Darwiche, "Sobre inferência probabilística por contagem de modelos ponderados", *Artif. Intel.*, vol. 172, n.ºs. 6–7, pp. 772–799, abril de 2008.
- [35] C. Yuan e MJ Druzdzel, "Um algoritmo de amostragem de importância baseado na pré-propagação de evidências", em *Uncertainty in Artificial Intelligence*. Amsterdã, Holanda: Elsevier, 2002, pp.
- [36] AE Gelfand, "amostragem de Gibbs", *J. Amer. Estado. Associado*, vol. 95, não. 452, pp. 1300–1304, dezembro de 2000.
- [37] NL Zhang e D. Poole, "Explorando a independência causal na inferência de rede bayesiana", *J. Artif. Intel. Res.*, vol. 5, pp. 301–328, dezembro de 1996.
- [38] F. Brglez, "Uma netlist neural de 10 circuitos de benchmark combinacionais", *Proc. Simulação de falha ATPG da sessão especial do IEEE*, junho de 1985, pp.
- [39] L. Amarú, P.-E. Gaillardon e G. De Micheli, "O conjunto de benchmark combinacional EPFL", em *Proc. Internacional Registro da oficina. Sintetizador. (WLS)*, 2015, pp.
- [40] S. Bhanja e N. Ranganathan, "Inferência bayesiana em cascata para estimativa de atividade de comutação com entradas correlacionadas", *IEEE Trans. Integridade em escala muito grande. (VLSI) Sistema*, vol. 12, não. 12, pp. 1360–1370, dezembro de 2004.
- [41] JM Mooij, "libDAI: Uma biblioteca C++ gratuita e de código aberto para inferência aproximada discreta em modelos gráficos", *J. Mach. Aprender. Res.*, vol. 11, pp. 2169–2173, agosto de 2010.
- [42] R. Marinescu, (2016). Merlim. Acesso em: 15 out. 2021. [Online]. Disponível: <https://github.com/radum2275/merlin/>
- [43] T. Heskes, K. Albers e B. Kappen, "Inferência aproximada e otimização restrita", em *Incerteza em Inteligência Artificial*. Amsterdã, Holanda: Elsevier, 2003, pp.
- [44] C. Chen, J. Cai e S. Zhan, "Um modelo de ponto triplo para análise de confiabilidade em nível de circuito", em *Proc. Internacional IEEE. Simp. Sistema de Circuitos (ISCAS)*, maio de 2018, pp.
- [45] MJ Druzdzel, "Smile: Modelagem estrutural, inferência e mecanismo e gênio de aprendizagem: Um ambiente de desenvolvimento para modelos teóricos de decisão gráfica", em *Proc. Conf. AAAI. Artefato. Intell.*, 1999, pp.
- [46] S. Zhan e C. Chen, "Um método híbrido para probabilidade de sinal e estimativa de confiabilidade com circuitos combinacionais", *Integração*, vol. 87, pp. 275–283, novembro de 2022.
- [47] W. Ibrahim e V. Beiu, "Usando redes bayesianas para calcular com precisão a confiabilidade de portas semicondutoras de óxido metálico complementar", *IEEE Trans. Rel.*, vol. 60, não. 3, pp. 538–549, setembro de 2011. [48] W. Ibrahim e H. Ibrahim, "Multithreaded e reconvergente consciente algoritmos para estimativa precisa de confiabilidade de circuitos digitais", *IEEE Trans. Rel.*, vol. 68, não. 2, pp. 514–525, junho de 2019.