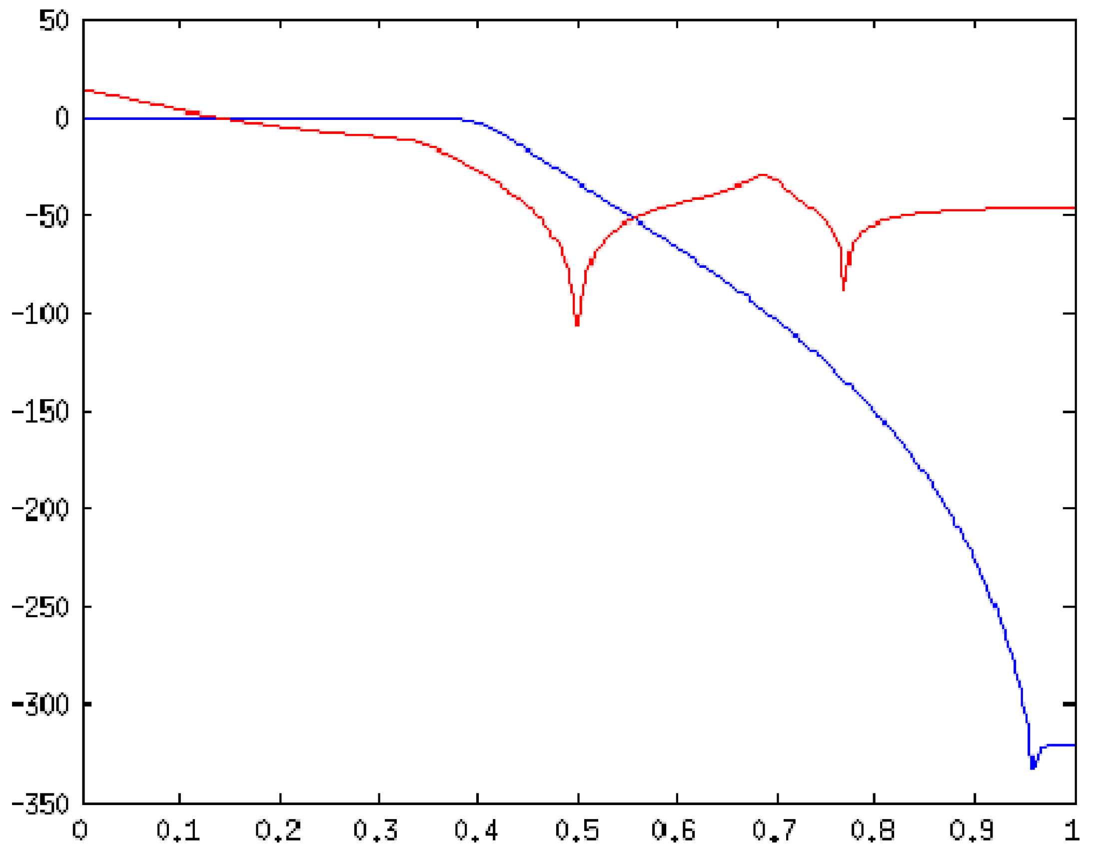
Verificação de Propriedades de Filtros Digitais Implementados com Aritmética de Ponto Fixo

Mauro Freitas, **Mikhail Ramalho**, Lucas Cordeiro, Waldir Júnior e Eddie Filho
maurokenny@gmail.com, **mikhail.ramalho@gmail.com**, lucascordeiro@ufam.edu.br,
waldirsabino@ufam.edu.br, eddie@ctpim.org.br
UFAM

1. Introdução

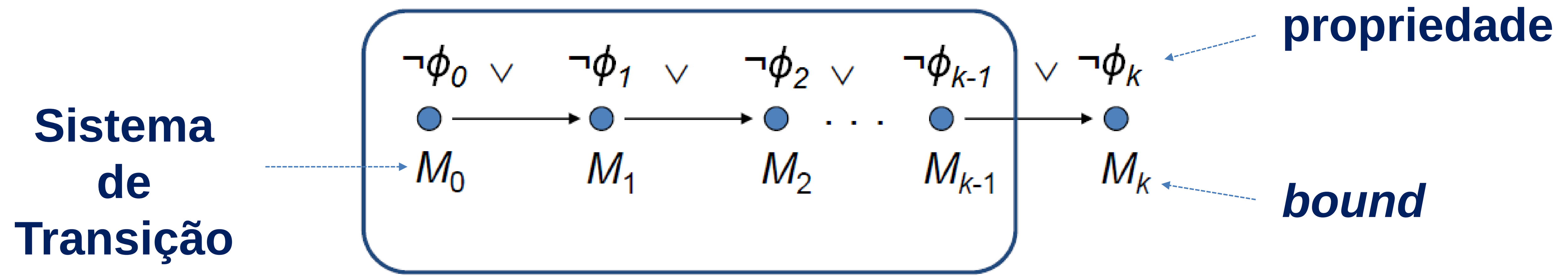
- Implementação em ponto flutuante de filtros digitais pode gerar efeitos indesejados na resposta em frequência do filtro, tanto em magnitude quanto em fase, além de problemas como *overflow* e instabilidade.
- Utilizando a técnica BMC é possível saber se a quantidade de bits utilizada no projeto do filtro apresenta precisão suficiente.



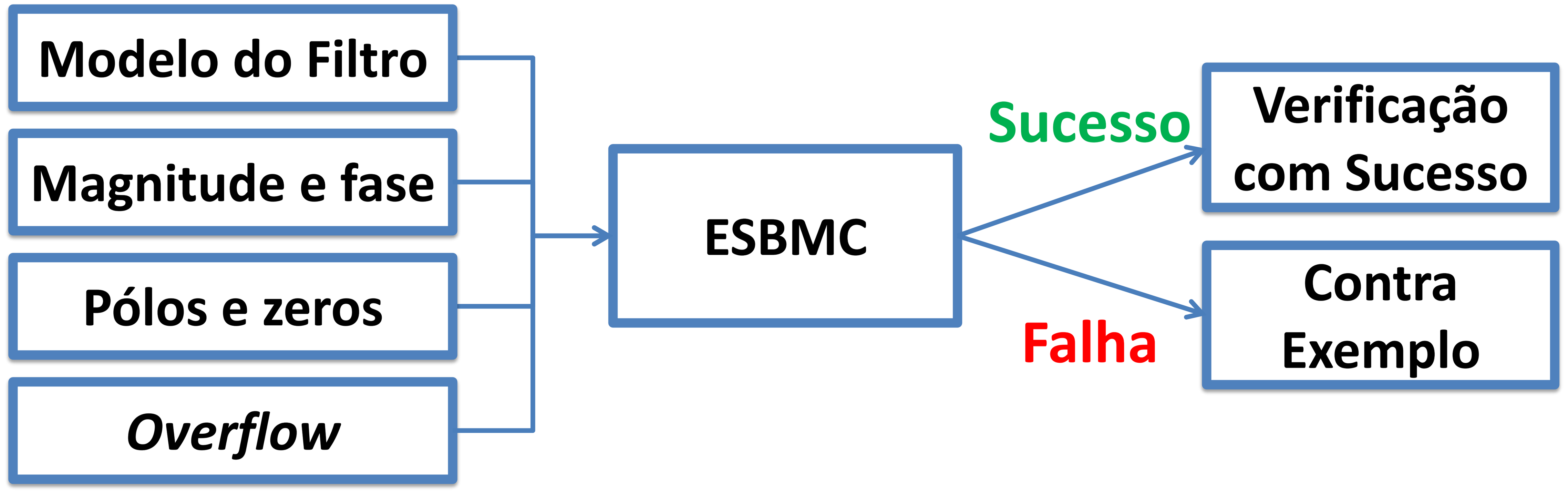
Resposta em magnitude para um filtro IIR Chebyshev, de ordem 12.
Projetado
Implementado

2. A Técnica Bounded Model Checking (BMC)

- Checa a negação de uma propriedade em uma dada profundidade.



3. Metodologia de Verificação



4. Configuração dos Experimentos

- Ubuntu 64 bits, Intel i7-2600, 3.40GHz, 24GB RAM, ESBMC v1.21, SMT Z3 v3.2.
- Filtros testados:
  - Passa-baixas e Passa-altas;
  - IIR: Butterworth, Chebyshev e Elíptico;
  - FIR: Equiripple, Janela de Hann e Maximally Flat.

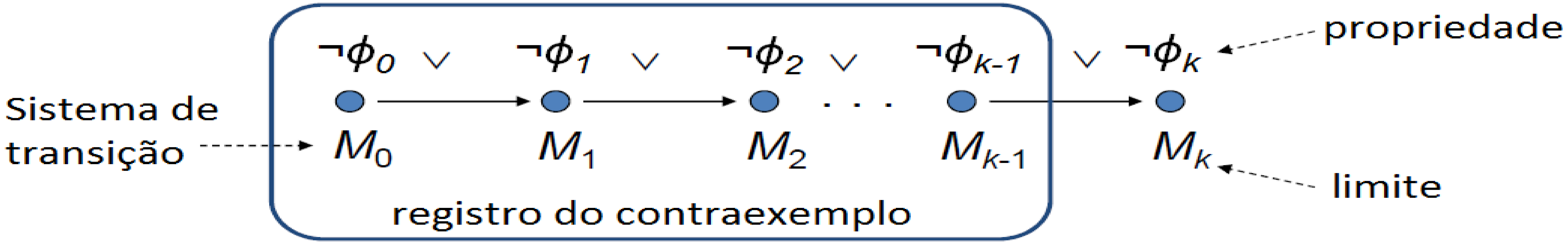
5. Conclusões

- O método pode detectar *overflow*, instabilidade e respostas em magnitude e fase, divergentes do projeto do filtro.
- É possível encontrar e detectar falhas em filtros de baixa e média ordem, com um tempo de verificação moderado.
- A técnica gera contra exemplo mostrando os estados que levaram o filtro a falhar.

* Ferramenta e *benchmark* disponíveis em www.esbmc.org

A Técnica *Bounded Model Checking*

- Checa a negação de uma propriedade em uma dada profundidade.



- Sistema de transição M com profundidade k:
  - Estado: contador de programa e valor de variáveis.
- Traduzido em uma condição de verificação ψ tal que:

> ψ é satisfatível se, e se somente se, houver um contra exemplo de profundidade máxima k.

Magnitude e fase

- Armazenamento ao longo dos somadores, multiplicadores e atrasos.

$$I_{pb\_fase}, I_{pa\_fase} \Leftrightarrow |\sphericalangle H(k) - \sphericalangle H_{fixed}(k)| > limiar$$

$$\begin{aligned}
I_{pb_mag} \Leftrightarrow & ((|H(k)| < A_p) \wedge (0 \leq \frac{2\pi k}{N} \leq \omega_p)) \\
& \vee ((|H(k)| > A_c) \wedge (\frac{2\pi k}{N} = \omega_c)) \\
& \vee ((|H(k)| > A_r) \wedge (\omega_r \leq \frac{2\pi k}{N} \leq \pi)),
\end{aligned}$$

$$\begin{aligned}
I_{pa_mag} \Leftrightarrow & ((|H(k)| > A_r) \wedge (0 \leq \frac{2\pi k}{N} \leq \omega_r)) \\
& \vee ((|H(k)| < A_c) \wedge (\frac{2\pi k}{N} = \omega_c)) \\
& \vee ((|H(k)| < A_p) \wedge (\omega_p \leq \frac{2\pi k}{N} \leq \pi))
\end{aligned}$$

Overflow

- Overflow e Underflow
- Contraexemplo é gerado em caso de falha
- Adição, subtração, multiplicação e divisão podem ser realizadas com uma representação em ponto fixo.

$$
\begin{aligned}
I_{overflow} \Leftrightarrow\ & ((\tilde{x}[n]h[0] > V_{max}) \vee \\
& (\tilde{x}[n-1]h[1] > V_{max}) \vee \\
& \dots\ (\tilde{x}[n-N-1]h[N-1] > V_{max}))
\end{aligned}
$$

$$
\begin{aligned}
I_{underflow} \Leftrightarrow\ & ((\tilde{x}[n]h[0] < V_{min}) \vee \\
& (\tilde{x}[n-1]h[1] < V_{min}) \vee \\
& \dots\ (\tilde{x}[n-N-1]h[N-1] < V_{min}))
\end{aligned}
$$

Pólos e zeros

- Algoritmo de decomposição QR
- Utilizado para decompor a função de transferência em pólos e zeros
- Criação da propriedade e verificação se cada módulo dos pólos é menor que 1 para garantir a estabilidade dos filtros.

Experimentos

- Objetivo: Verificar se filtros digitais mantinham suas características de projeto após quantização.

- Setup:
  - ESBMC v1.20 com SMT Solver Z3 3.2
  - Intel Core i7-2600, 3.40 GHz com 24 GB of RAM rodando Ubuntu 64-bits
  - Fp = 0,3; Fc = 0,4; Fr=0,5 rad. Ganho mínimo fixado em -0,9dB, atenuação mínima na banda corte e de rejeição em -3dB e -6dB
  - 18 IIR: Butterworth, Chebyshev, Elíptico (equiripple na banda de passagem e rejeição)
  - 18 FIR: Equiripple, Janela de Hann e Maximally Flat

Resultados

- Nenhum caso de teste excedeu 1200 segundos

- Não ocorreram casos de estouro de memória

- Todos os filtros foram verificados manualmente para comprovar a eficácia da ferramenta quanto a verificação das propriedades.

- Os resultados mostram que é possível se detectar falhas em filtros de baixa e média ordem, com um tempo de verificação moderado