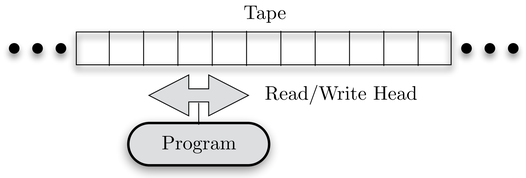
REAL NUMBER SIGNAL PROCESSING CAN DETECT DENIAL-OF-SERVICE ATTACKS

Holger Boche[1], Rafael F. Schaefer[2], H. Vincent Poor[3]
[1] Technical University of Munich
[2] University of Siegen
[3] Princeton University

Turing Machine



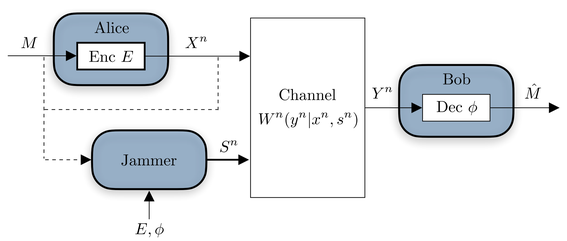
Mathematical model of an abstract machine that manipulates symbols on a strip of tape according to certain given rules

- Turing machines can simulate any given algorithm and therewith provide a simple but very powerful model of computation
- **No** limitations on computational complexity, unlimited computing capacity and storage, and execute programs completely error-free

⇒ **Fundamental performance limits for today's digital computers**

A. M. Turing, "On computable numbers, with an application to the Entscheidungsproblem," *Proc. London Math. Soc.*, vol. 2, no. 42, pp. 230–265, 1936

Communication System



- Let $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{S}$ be finite input, output, state (jamming) alphabets
- For fixed $s^n \in \mathcal{S}^n$, the DMC is $W^n(y^n|x^n,s^n) = \prod_{i=1}^n W(y_1|x_i,s_i)$

Definition: The *arbitrarily varying channel (AVC)* $\mathcal{W}$ is given by

$$\mathcal{W} = \left\{W(\cdot|\cdot,s)\right\}_{s\in\mathcal{S}}.$$

$$F(\mathcal{W}) = \min_{U\in\mathcal{CH}(\mathcal{X};\mathcal{S})} \max_{x\neq\hat{x}} \sum_{y\in\mathcal{Y}} \left| \sum_{s\in\mathcal{S}} W(y|\hat{x},s)U(s|x) - \sum_{s\in\mathcal{S}} W(y|x,s)U(s|\hat{x}) \right|$$

⇒ $\mathcal{W}$ is *symmetrizable* if and only if $F(\mathcal{W}) = 0$

Theorem: The capacity $C(\mathcal{W})$ of an AVC $\mathcal{W}$ is

$$C(\mathcal{W}) = \begin{cases} \min_{q\in\mathcal{P}(\mathcal{S})} C(W_q) & \text{if } F(\mathcal{W}) > 0 \\ \mathbf{0} & \text{if } \boldsymbol{F(\mathcal{W}) = 0} \end{cases}$$

with $W_q(y|x) = \sum_{s\in\mathcal{S}} W(y|x,s)q(s)$.

R. Ahlswede, "Elimination of correlation in random codes for arbitrarily varying channels," *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, vol. 44, no. 2, pp. 159–175, Jun. 1978

I. Csiszár and P. Narayan, "The capacity of the arbitrarily varying channel revisited: Positivity, constraints," *IEEE Trans. Inf. Theory*, vol. 34, no. 2, pp. 181–193, Mar. 1988

Blum-Shub-Smale (BSS) Machine

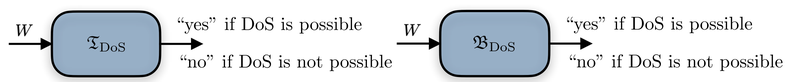
- It can store arbitrary real numbers, can compute all field operations on $\mathbb{R}$, i.e., "+" and "·", and can compare real numbers according to the relations "<", ">", and "="
- A BSS machine is similar to a Turing machine in the sense that it operates on an infinite strip of tape according to a so-called program. This is a finite directed graph with five types of nodes associated with different operations: input node, computation node, branch node, shift node, and output node

BSS-computable functions are input-output maps Φ of the BSS machine $\mathfrak{B}$, i.e., for every input $\vec{x}$, the output $\Phi_{\mathfrak{B}}(\vec{x})$ is defined if the ouput is reachable by the program of $\mathfrak{B}$.

A set $\mathcal{A} \subset \mathbb{R}^N$ is *BSS-decidable* if there is a BSS machine $\mathfrak{B}_{\mathcal{A}}$ such that for all $\vec{x} \in \mathbb{R}^N$ we have $\mathfrak{B}_{\mathcal{A}}(\vec{x}) = \chi_{\mathcal{A}}(\vec{x})$, i.e., the characteristic function $\chi_{\mathcal{A}}$ of the set $\mathcal{A}$ is BSS-computable.

L. Blum, M. Shub, and S. Smale, "On a theory of computation and complexity over the real numbers: *NP*-completeness, recursive functions and universal machines," *Bull. Amer. Math. Soc.*, vol. 21, no. 1, pp. 1–46, Jul. 1989

Detection Framework



- Task of a Turing machine $\mathfrak{T}$ or of a BSS machine $\mathfrak{B}$ is to **detect denial-of-service attacks**

This is an Entscheidungsproblem, since for a given $\mathcal{W}$, the algorithm should answer the question whether or not a denial-of-service attack is possible

- A hypothetical algorithm takes all channels and partitions this set into two disjoint subsets
  - $\mathcal{M}_{\text{DoS}}^{\text{c}}$ are those $\mathcal{W}$ for which $C(\mathcal{W}) > 0$
  - $\mathcal{M}_{\text{DoS}}$ are those $\mathcal{W}$ for which a denial-of-service attack is possible, i.e., $\mathcal{W}$ with $C(\mathcal{W}) = 0$

$$\mathcal{M}_{\text{DoS}} = \left\{\mathcal{W} : F(\mathcal{W}) = 0\right\}$$

- Since $\mathcal{M}_{\text{DoS}}$ is characterized by the continuous function $F(\cdot)$, the set is well defined

⇒ **Analytically, this is easy to answer! And algorithmically...?**

Question: Is there an algorithm (or Turing machine) $\mathfrak{T}$ that takes $\mathcal{W}$ as an input and **outputs $\mathfrak{T}(\mathcal{W}) = 1$ if the Jammer is able to perform a denial-of-service attack** and **otherwise outputs $\mathfrak{T}(\mathcal{W}) = 0$**?

- Framework also important for system evaluation and verification

H. Boche, R. F. Schaefer, and H. V. Poor, "Performance evaluation of secure communication systems on Turing machines," in *Proc. 10th IEEE Int. Workshop Inf. Forensics Security*, Hong Kong, Dec. 2018, pp. 1–7

Turing Detectability

Theorem: For all $|\mathcal{X}| \geqslant 2$, $|\mathcal{S}| \geqslant 2$, and $|\mathcal{Y}| \geqslant 2$, there is **no** Turing machine $\mathfrak{T} : \mathcal{CH}_c(\mathcal{X},\mathcal{S};\mathcal{Y}) \to \{0,1\}$ with $\mathfrak{T}(\mathcal{W}) = 1$ if and only if $\mathcal{W} \in \mathcal{M}_{\text{DoS}}$.

- We look for a Turing machine that stops for every channel $\mathcal{W} \in \mathcal{CH}_c(\mathcal{X},\mathcal{S};\mathcal{Y})$ and further $\mathfrak{T}(\mathcal{W}) = 1$ if and only if $\mathcal{W} \in \mathcal{M}_{\text{DoS}}$

⇒ Such a Turing machine does **not** exist
⇒ This question is **algorithmically undecidable**!
⇒ This provides a negative answer to Question 1

H. Boche, R. F. Schaefer, and H. V. Poor, "Denial-of-service attacks on communication systems: Detectability and jammer knowledge," *IEEE Trans. Signal Process.*, vol. 68, pp. 3754–3768, 2020

- Feedback does **not** help – detection problem remains undecidable on Turing machines

H. Boche, R. F. Schaefer, and H. V. Poor, "On the algorithmic solvability of channel dependent classification problems in communication systems," *IEEE/ACM Trans. Netw.*, 2021

BSS Detectability

Theorem: Let $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{S}$ be arbitrary finite alphabets. Then there exists a BSS machine $\mathfrak{B}$ that outputs $\mathfrak{B}(W) =$ "yes" if and only if $W \in \mathcal{M}_{\text{DoS}}$, i.e., the DoS detection problem is BSS-decidable.

Main proof ingredient:

- Exploit connections to the theory of semialgebraic sets
- Show that both sets $\mathcal{M}_{\text{DoS}}$ and $\mathcal{M}_{\text{DoS}}^{\text{c}}$ are semialgebraic
- The result remains **true** also in case where the Jammer also knows the transmitted message, i.e., the most powerful jammer

Conclusion

- *Detection framework based on Turing machines*
  ⇒ Turing machines provide fundamental performance limits for today's digital computers and therewith of traditional signal processing
  ⇒ Turing machines are **not** capable of detecting DoS attacks!
  ⇒ Feedback does **not** help – detection problem remains undecidable
- *Detection framework based on BSS machines*
  ⇒ Allows the processing and storage of arbitrary reals
  ⇒ BSS machines are capable of detecting DoS attacks!
  ⇒ Real number signal processing enables the detection of DoS attacks

⇒ Solution to the DoS detectability problem: **Computing model** is very important!

Supported by the German Federal Ministry of Education and Research (BMBF) under Grants 16KIS1003K and 16KIS1004, by the German Research Foundation (DFG) under Grants BO 1734/20-1, EXC-2092 – 390781972, EXC-2111 – 390814868, and SCHA 1944/6-1, as well as by the U. S. National Science Foundation under Grants CCF-0939370 and CCF-1908308.