Moore-Machine Filtering for Timed and Untimed Pattern Matching

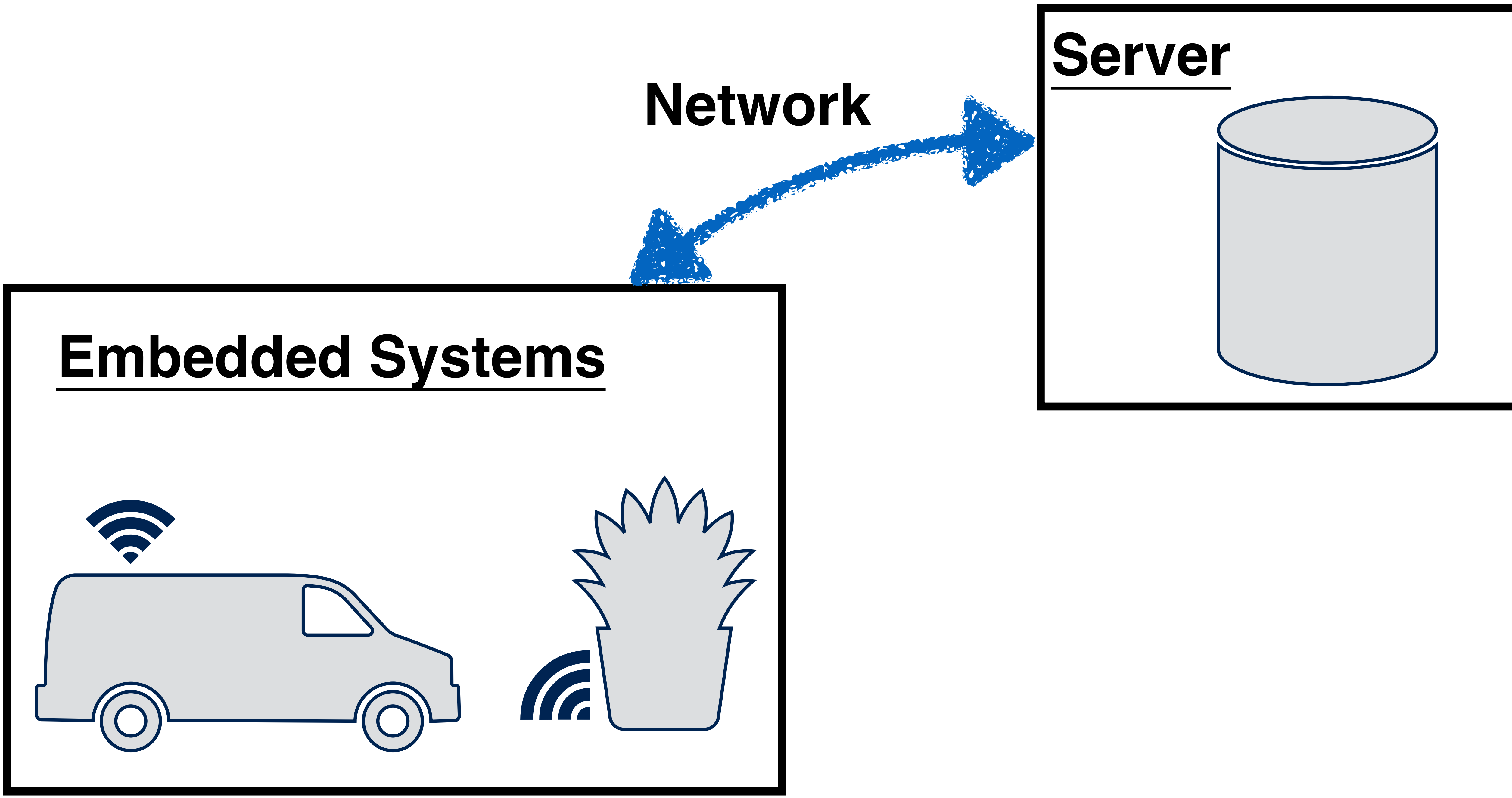
Masaki Waga[1] and Ichiro Hasuo[1]

National Institute of Informatics[1]

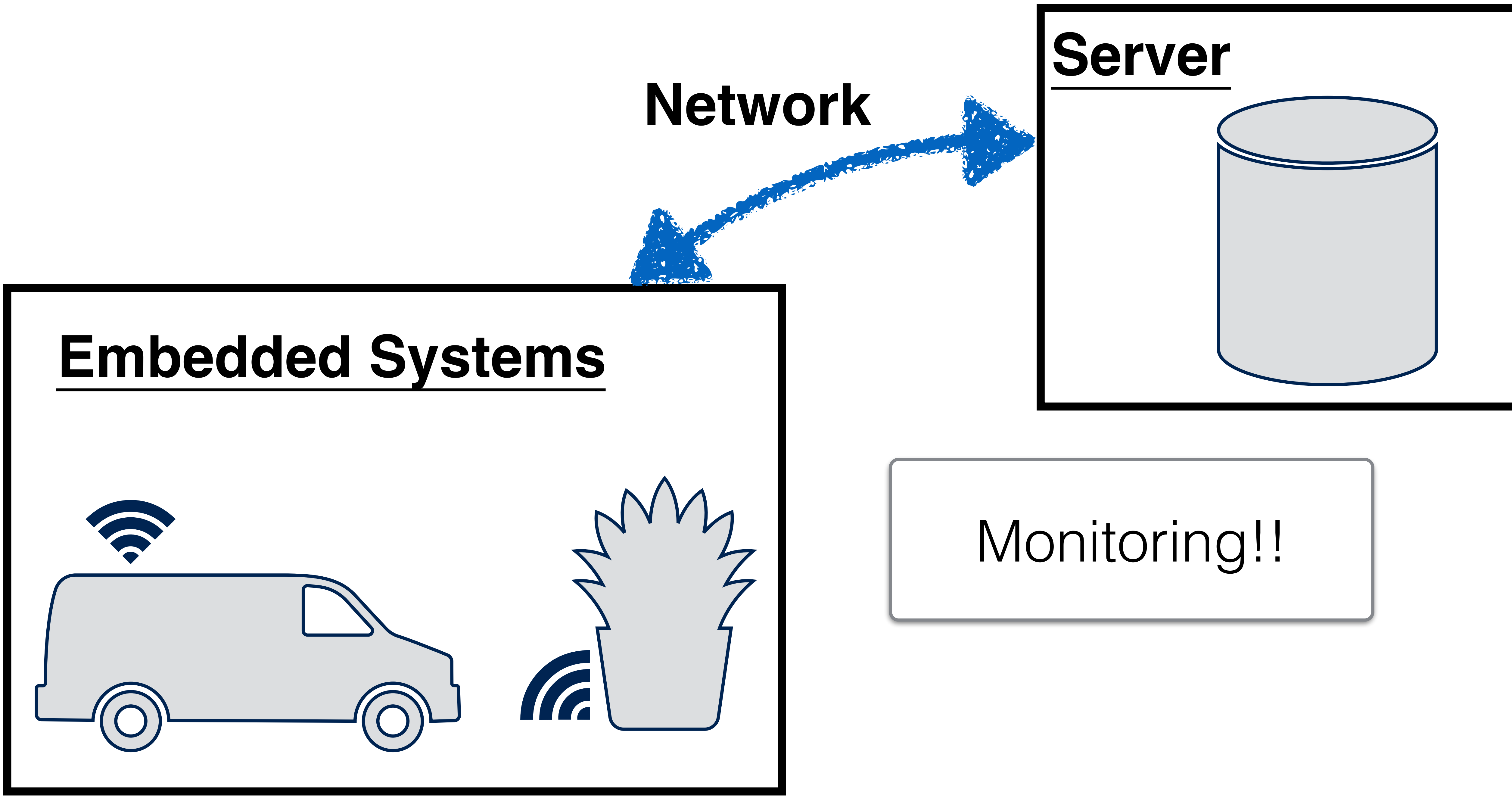
2 Oct. 2018, EMSOFT 2018

The authors are supported by JST ERATO HASUO Metamathematics for Systems Design Project (No. JPMJER1603), and JSPS Grants-in-Aid No. 15KT0012 & 18J22498.

IoT / Embedded Systems Connected via Network



IoT / Embedded Systems Connected via Network



Monitoring!!

Timed Pattern Matching

[Ulus et al., FORMATS'14]

Input

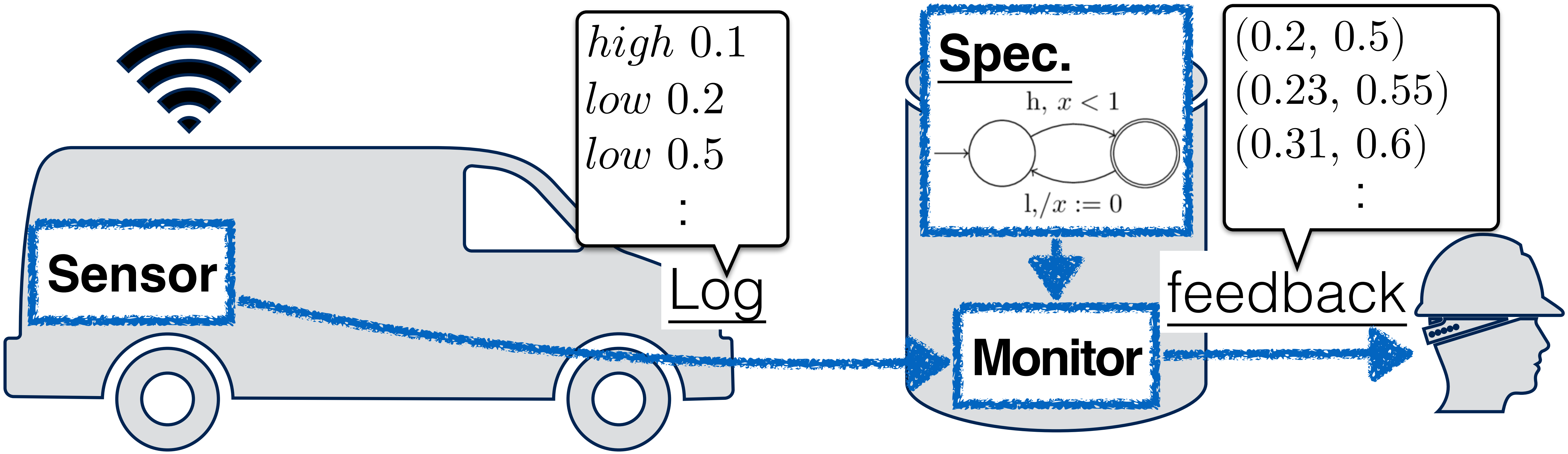
- **Time-series data** (***Logs*** of a car/ a robot)
  e.g., The gear of a car: ***high*** at 0.1s, ***low*** at 0.2s, …
- **Real-time spec.** (***Spec.*** useful for debugging)
  e.g., Frequent gear change of a car

Output

- The intervals where the ***spec.*** is satisfied in the ***log***
  e.g., Frequent gear change in 0.2s-0.7s

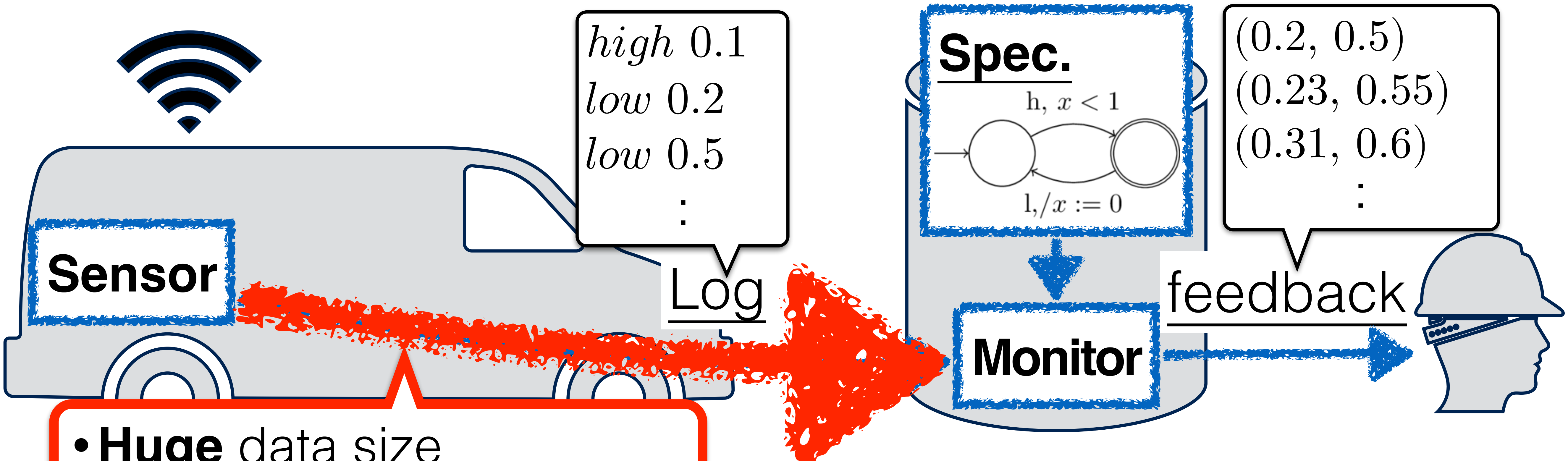
M. Waga (NII)

Remote Monitoring



M. Waga (NII)

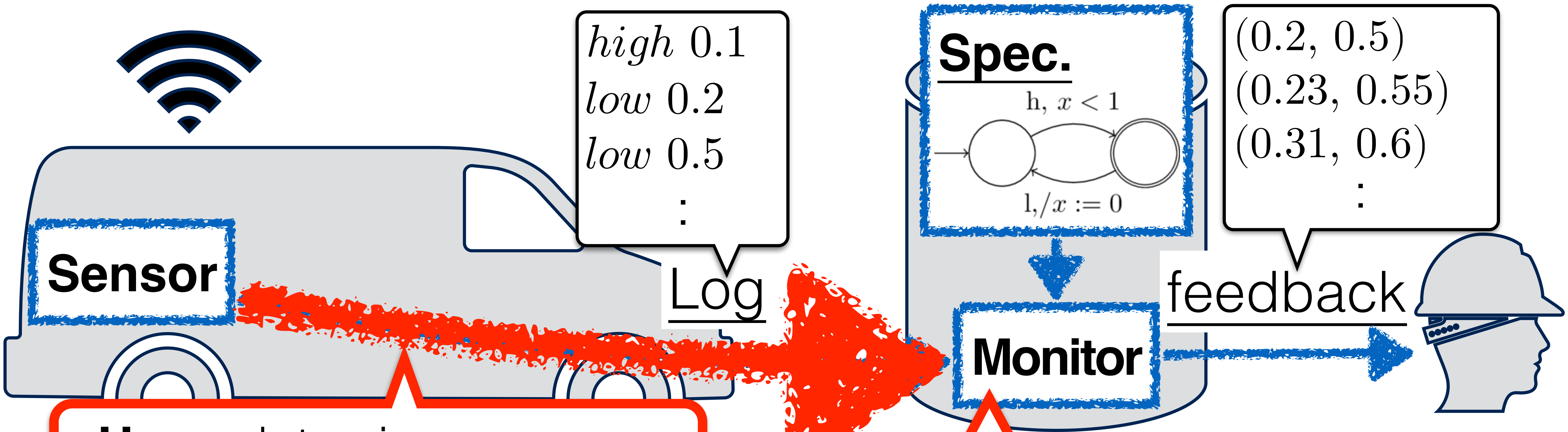
Remote Monitoring



- **Huge** data size
- **Limited** network capacity

⇒ Reduce the size!!

Remote Monitoring

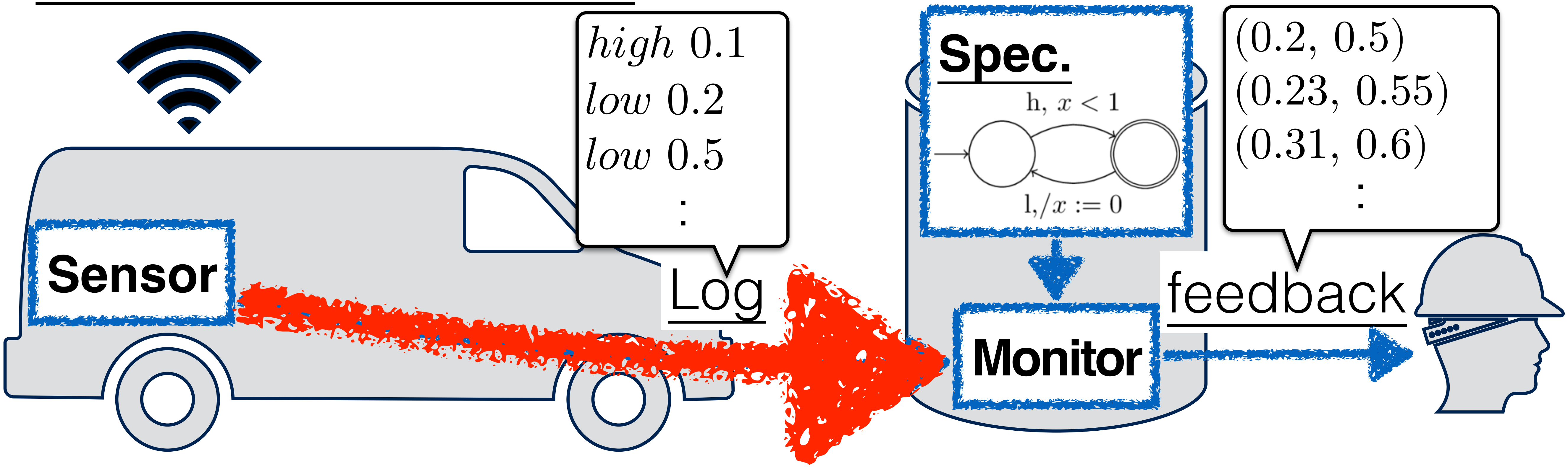


- **Huge** data size
- **Limited** network capacity
  ⇒ Reduce the size!!

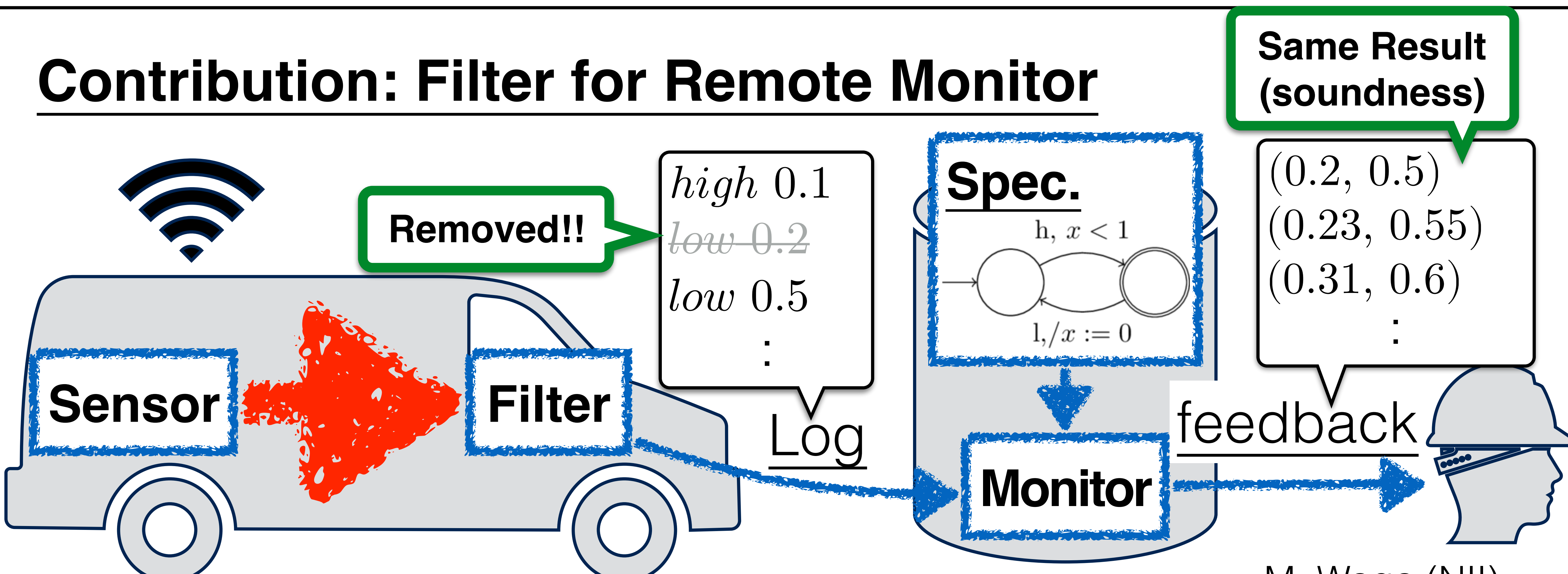
- Timed Pattern Matching
- **Assumption**: **Too Heavy** for embedded systems

M. Waga (NII)

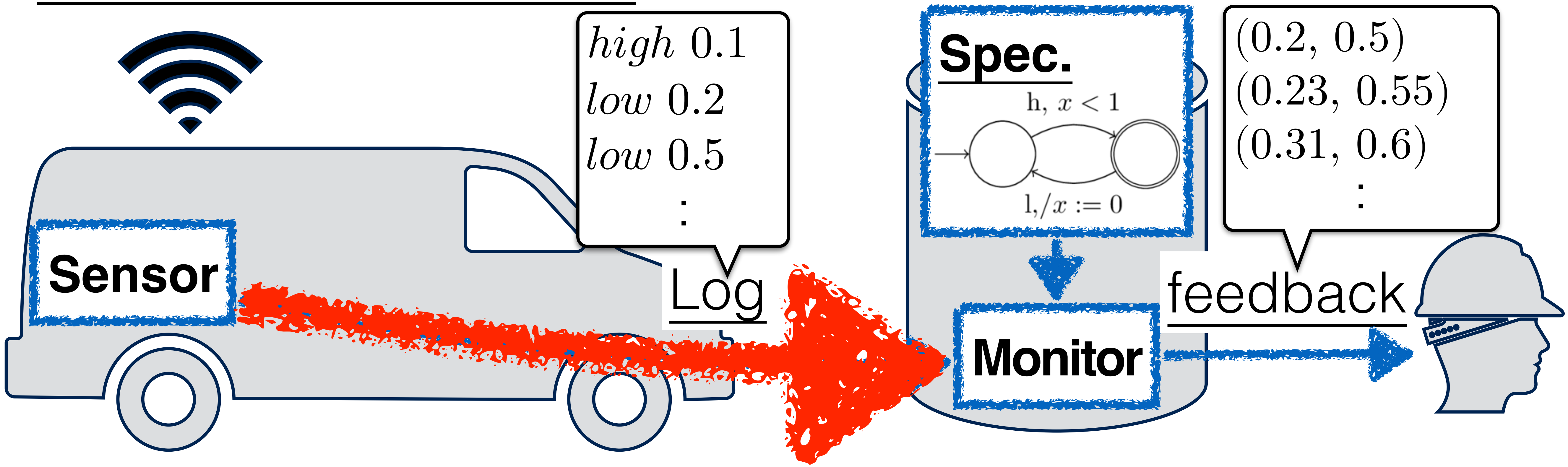
Naive Remote Monitor



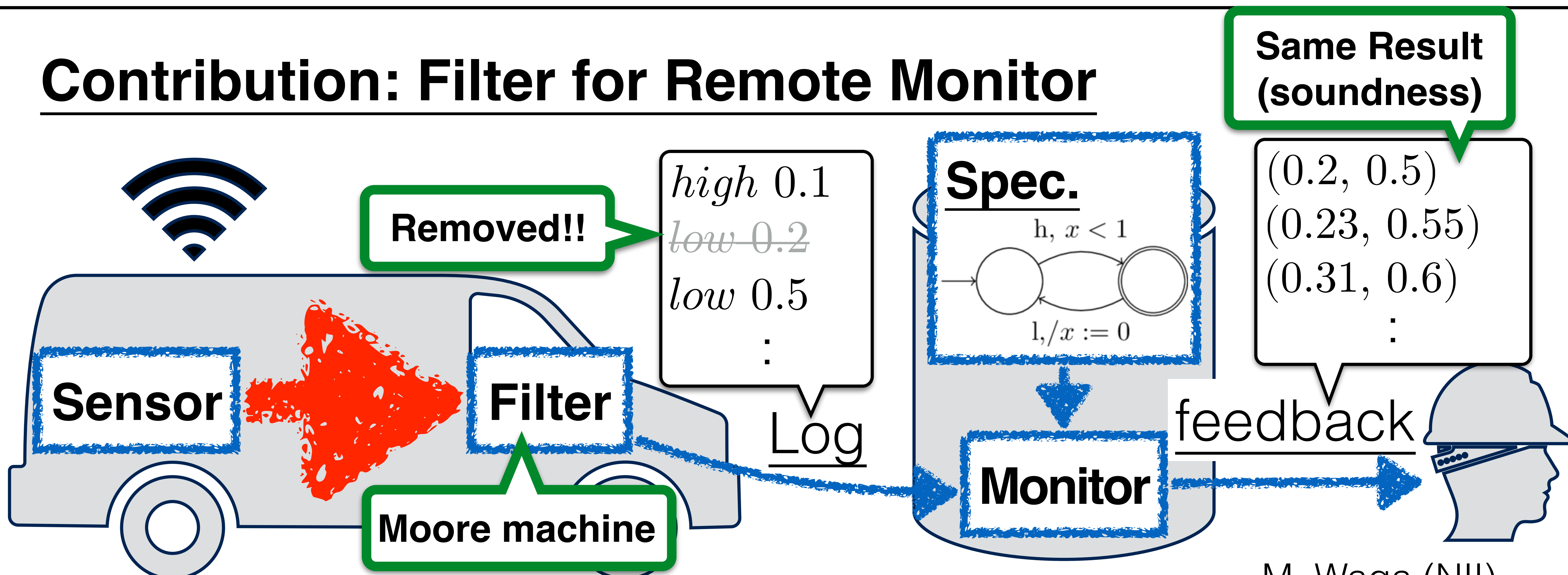
Contribution: Filter for Remote Monitor



Naive Remote Monitor

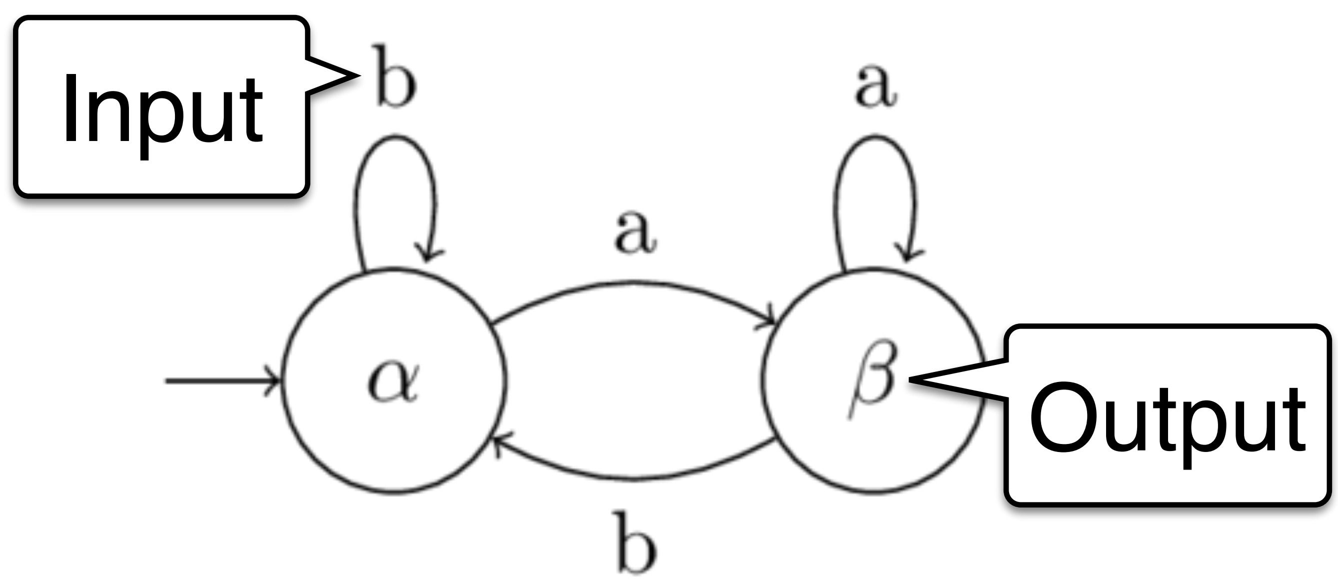


Contribution: Filter for Remote Monitor



Moore Machine

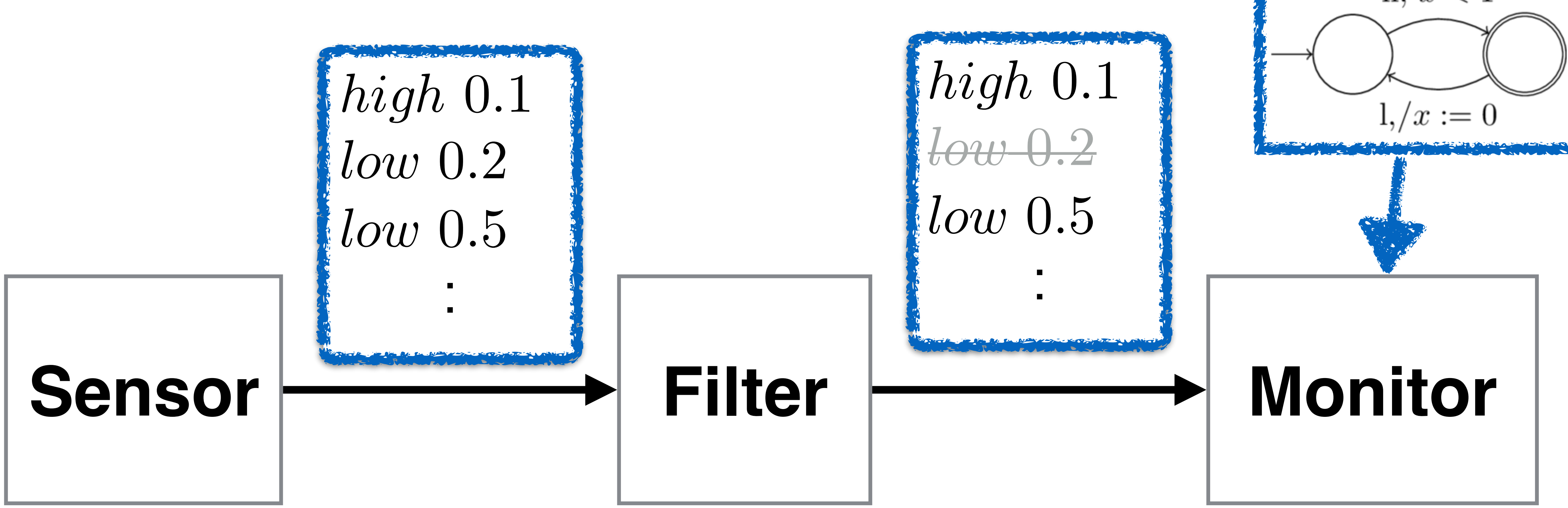
DFA + state-dependent output



- Efficient (linear time)
- Hardware implementable

M. Waga (NII)

Contribution



- Moore machine filter from NFA / TA
- Soundness (the result is the same)
- Implementation & experiment
  - Data size ≈ 1/100 — 1/2

Related Works

Timed Pattern Matching

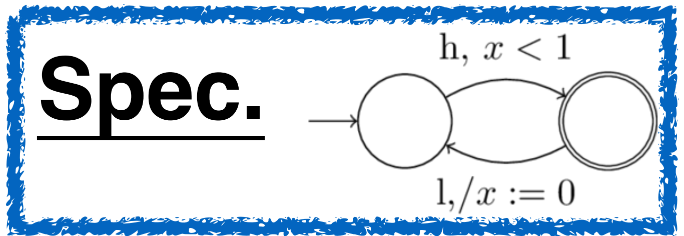
- Offline Algorithm
  - [Ulus+, FORMATS'14]
- Online Algorithm
  - [Ulus+, TACAS'16]
- Optimization (skipping)
  - [**Waga**+, FORMATS'16]
  - [**Waga**+, FORMATS'17]

Filtering for Pattern Matching

- Multiple string
  - [Salmela+, JEA'06]
  - [Kandhan+, PVLDB'10]
- regular expression
  - (different problem)
  - [Liu+, ACNS'12]

:

This Work



Outline

Sensor → Filter → Monitor

Theoretical preparation

- Moore Machine Filter for **Untimed** Pattern Matching
  - Spec.: NFA $\mathcal{A}$
  - Log: word ($w \in \Sigma^*$)

Main Problem

- Moore Machine Filter for **Timed** Pattern Matching
  - Spec.: TA $\mathcal{A}$
  - Log: timed word ($w \in (\Sigma \times \mathbb{R}_{>0})^*$)
- Experiments (**timed**)

(Untimed) Pattern Matching

Input

Log

- Word $w \in \Sigma^*$
- NFA $\mathcal{A}$

Spec.

Output

indices

$$\mathrm{Match}(w,\,\mathcal{A}) = \{(i,j) \mid w(i,j) \in L(\mathcal{A})\}$$

Example

$w = \mathrm{dbadcdc}$

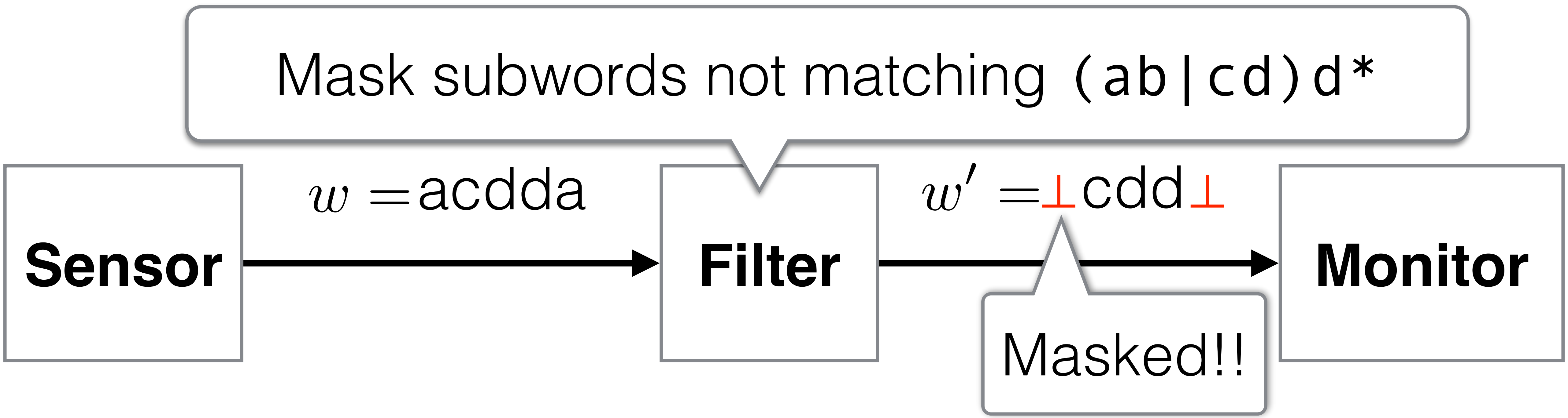
$L(\mathcal{A}) = $ dc*{ba|dc}

$\mathrm{Match}(w,\,\mathcal{A}) = \{(1,3),\,(4,7)\}$

$w(1,3) = \mathrm{dba} \in L(\mathcal{A})$

$w(4,7) = \mathrm{dcdc} \in L(\mathcal{A})$

Idea: Filtering by Masking

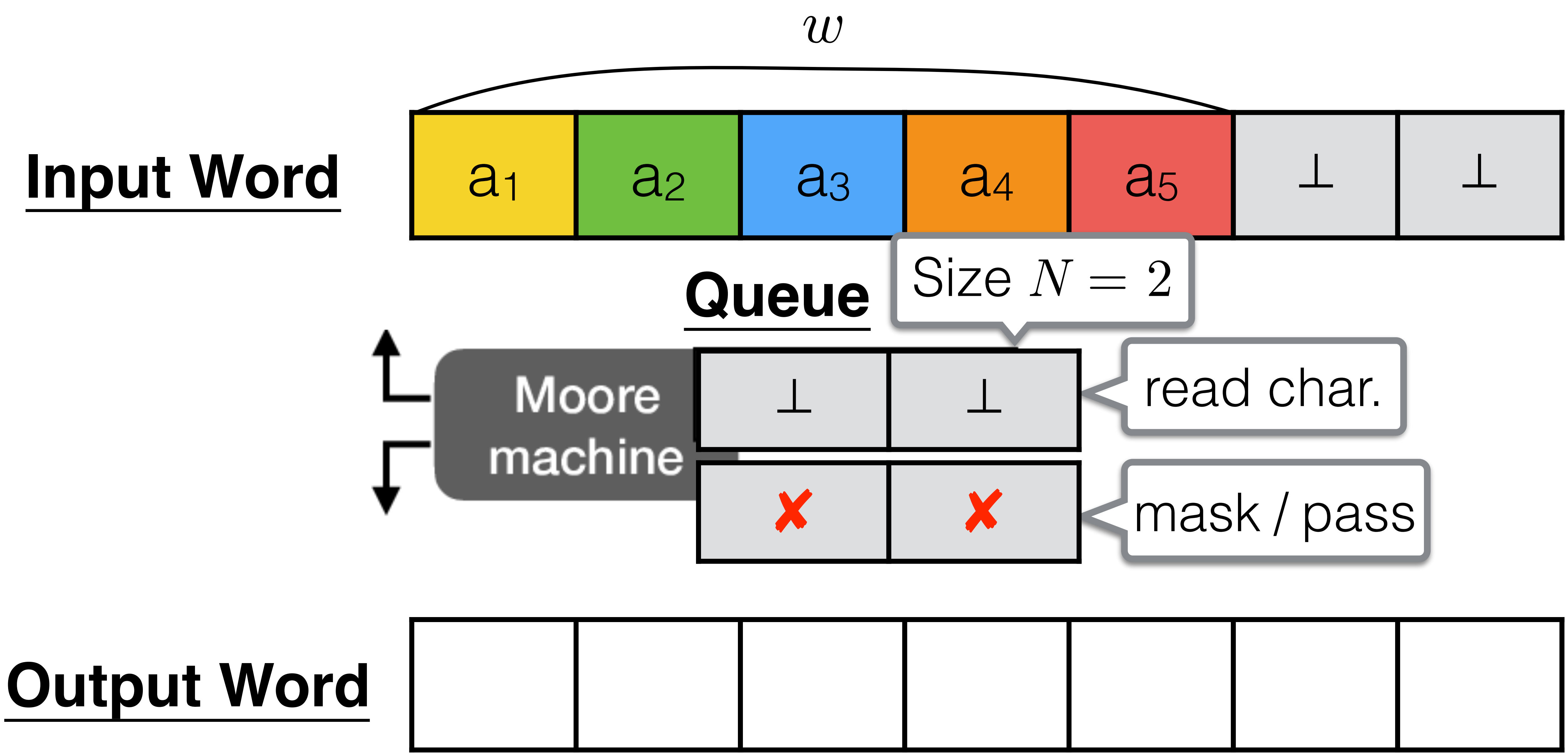


Goal

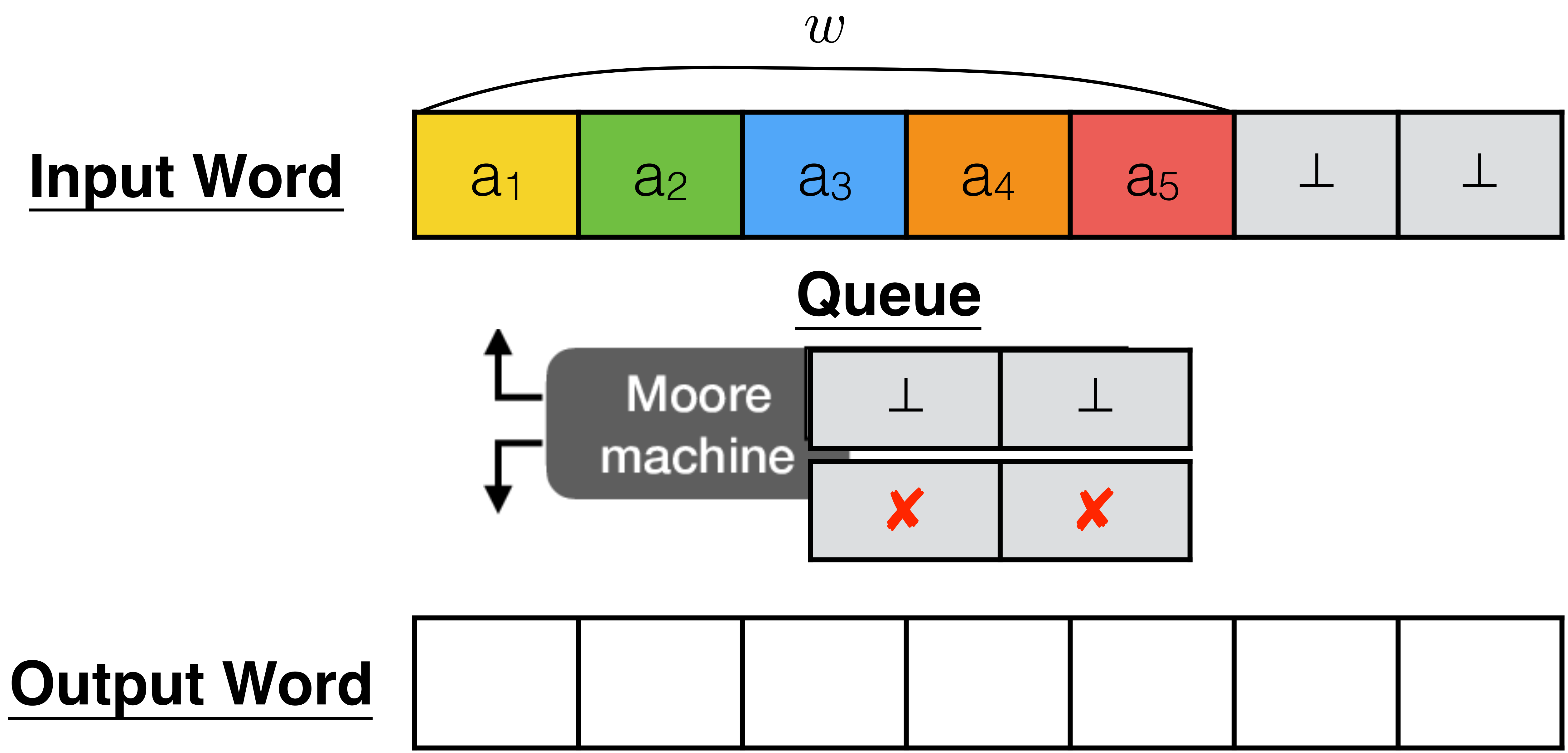
Construct a sound and efficient filter
⇒ by a Moore machine

DFA + state-dependent output

Overview of Filtering

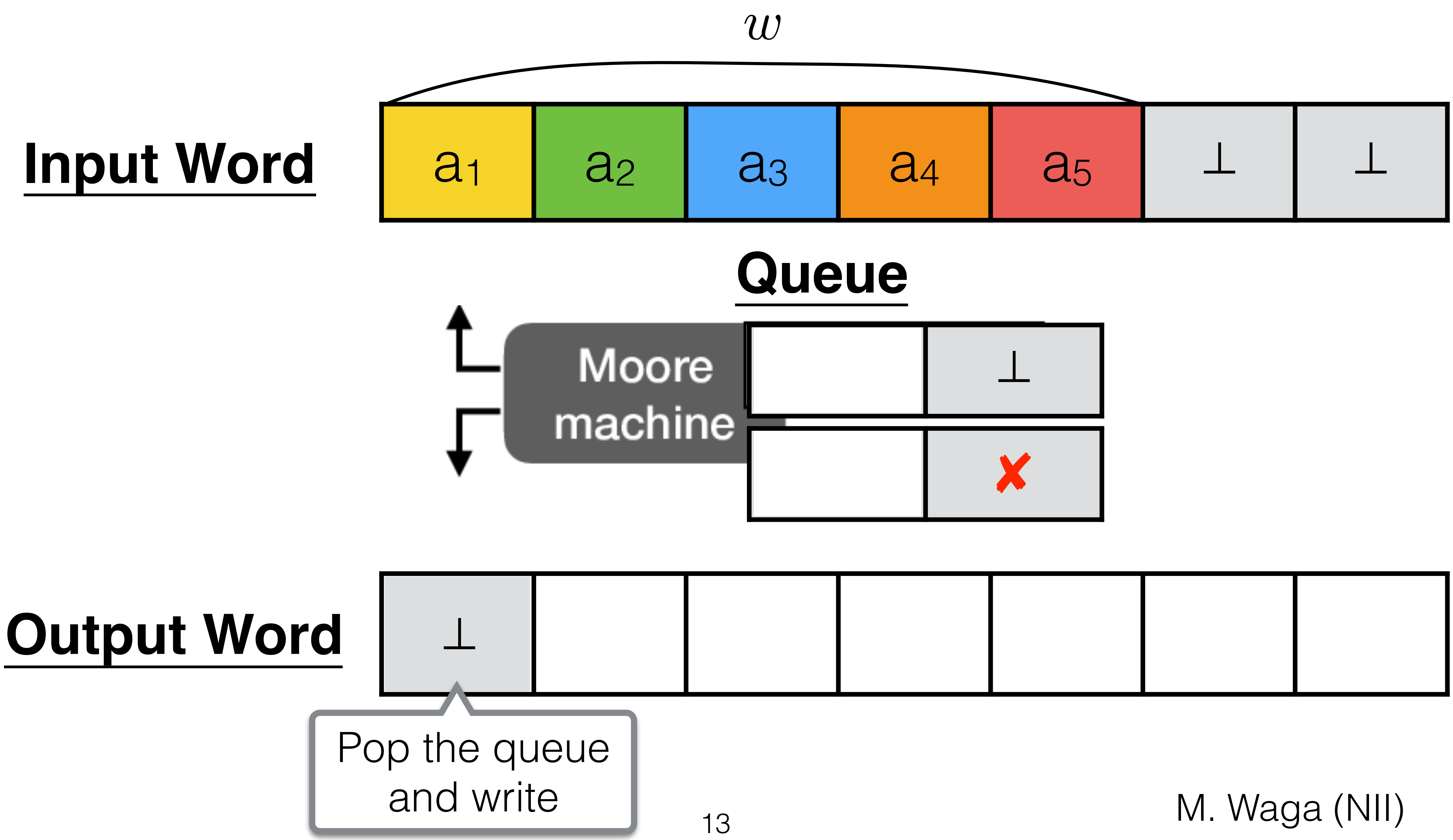


Overview of Filtering

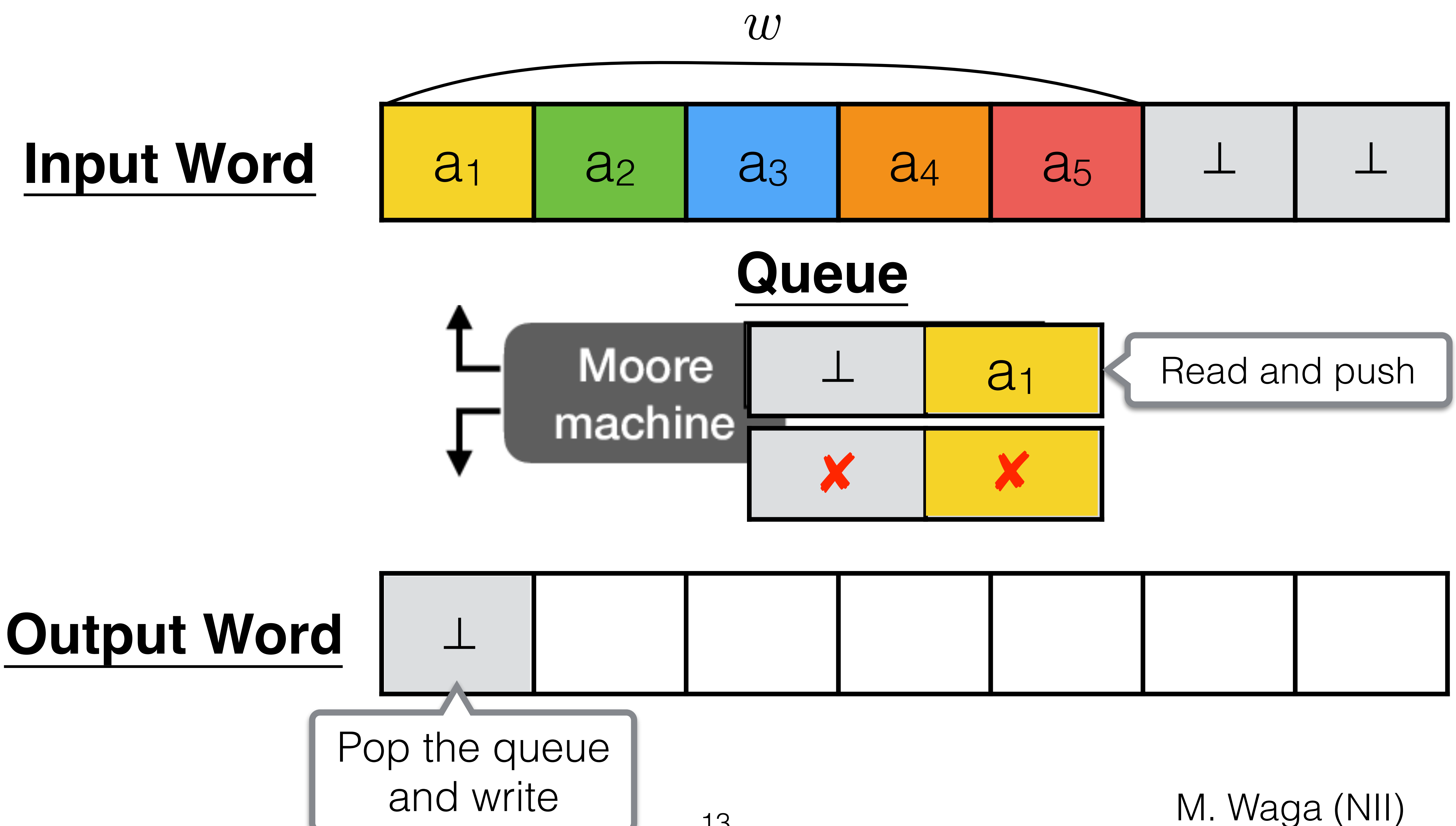


M. Waga (NII)

Overview of Filtering

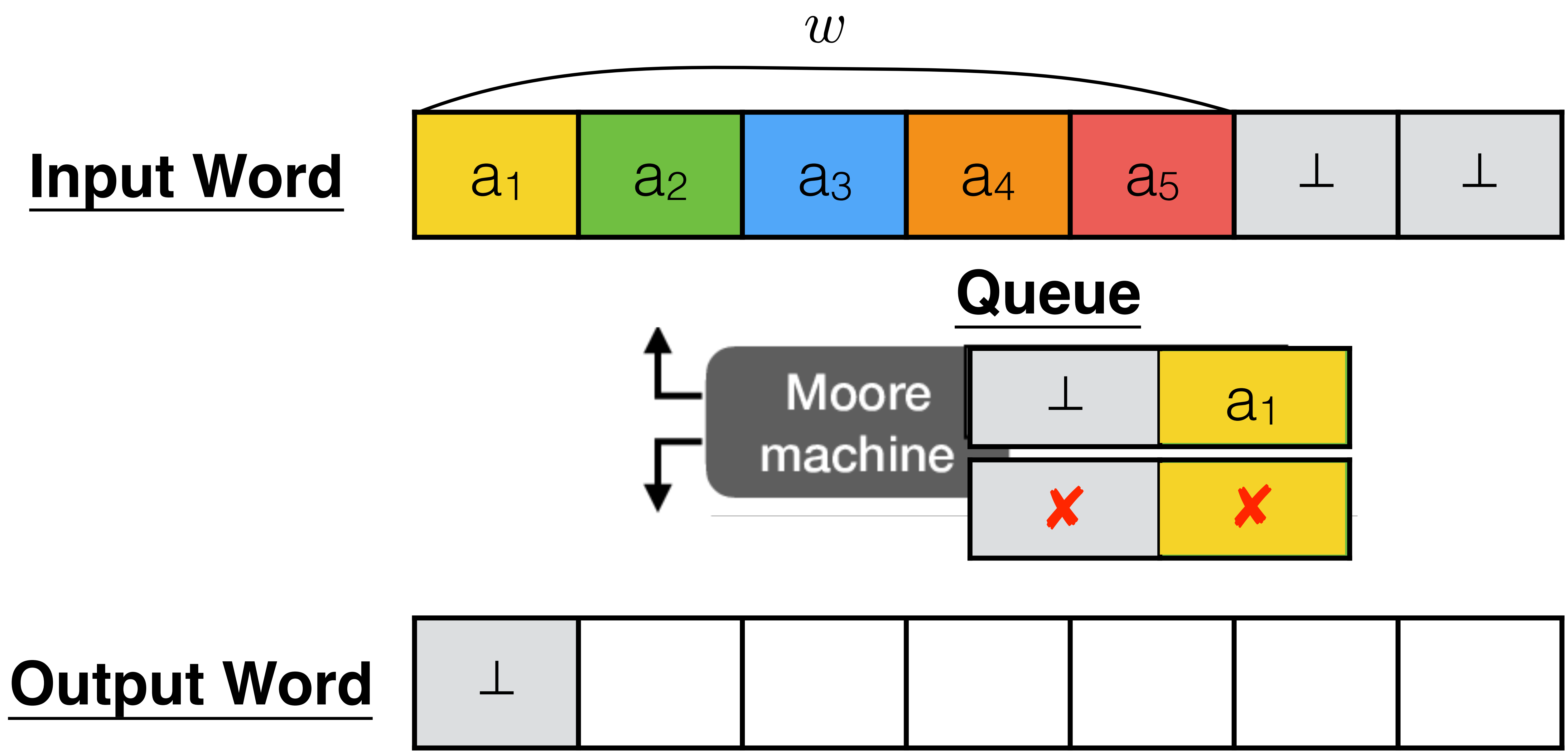


Overview of Filtering



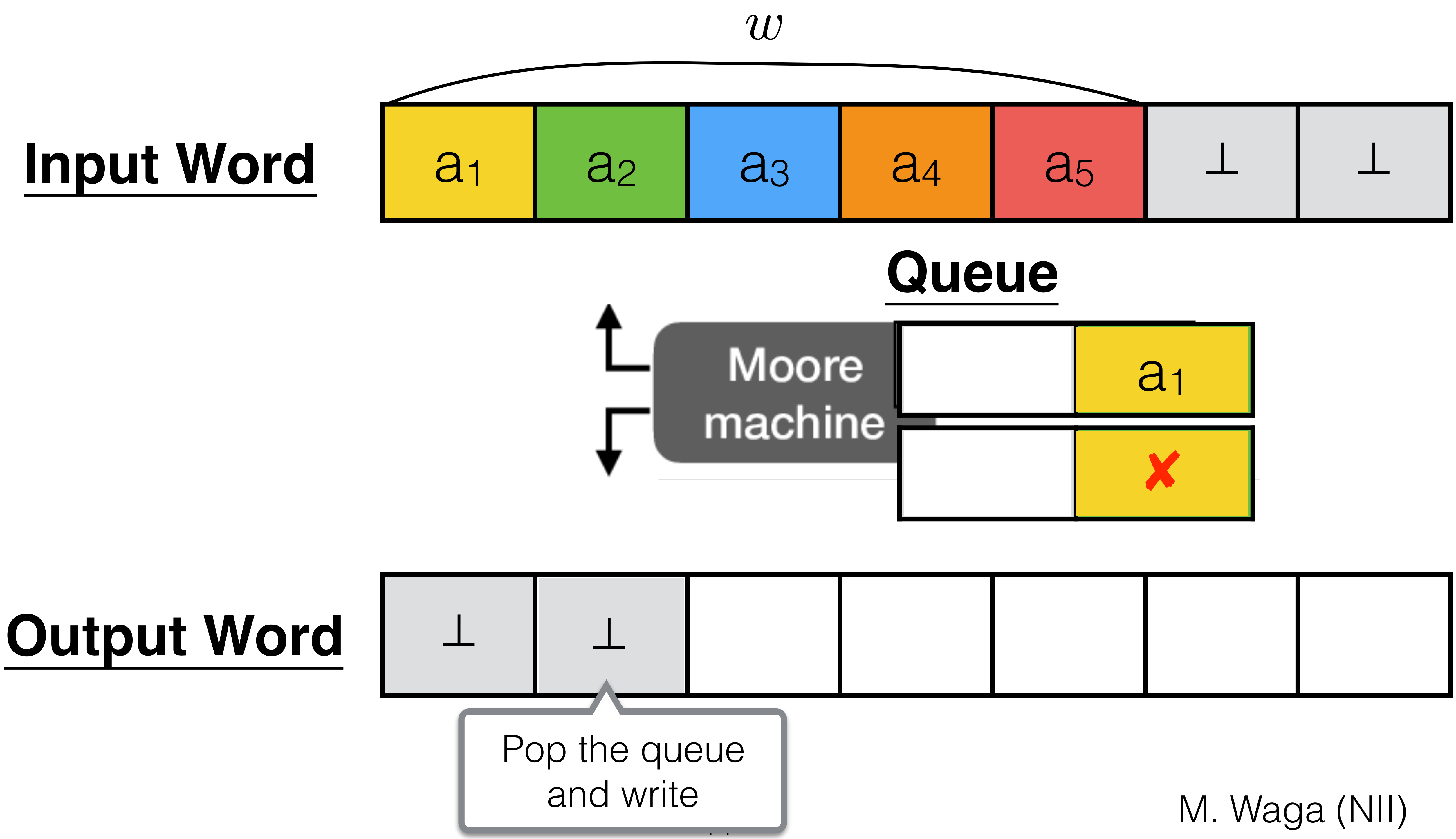
M. Waga (NII)

Overview of Filtering



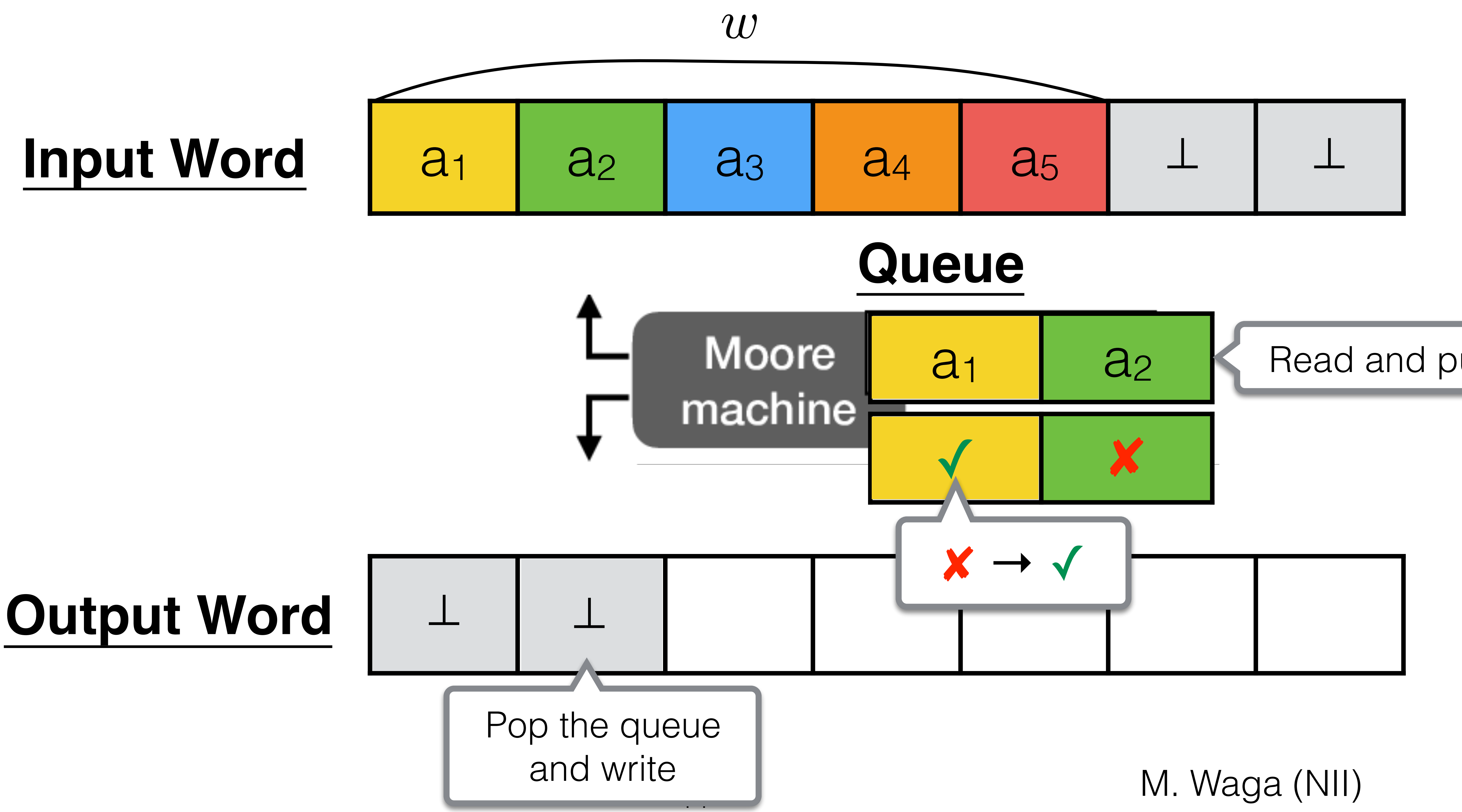
M. Waga (NII)

Overview of Filtering

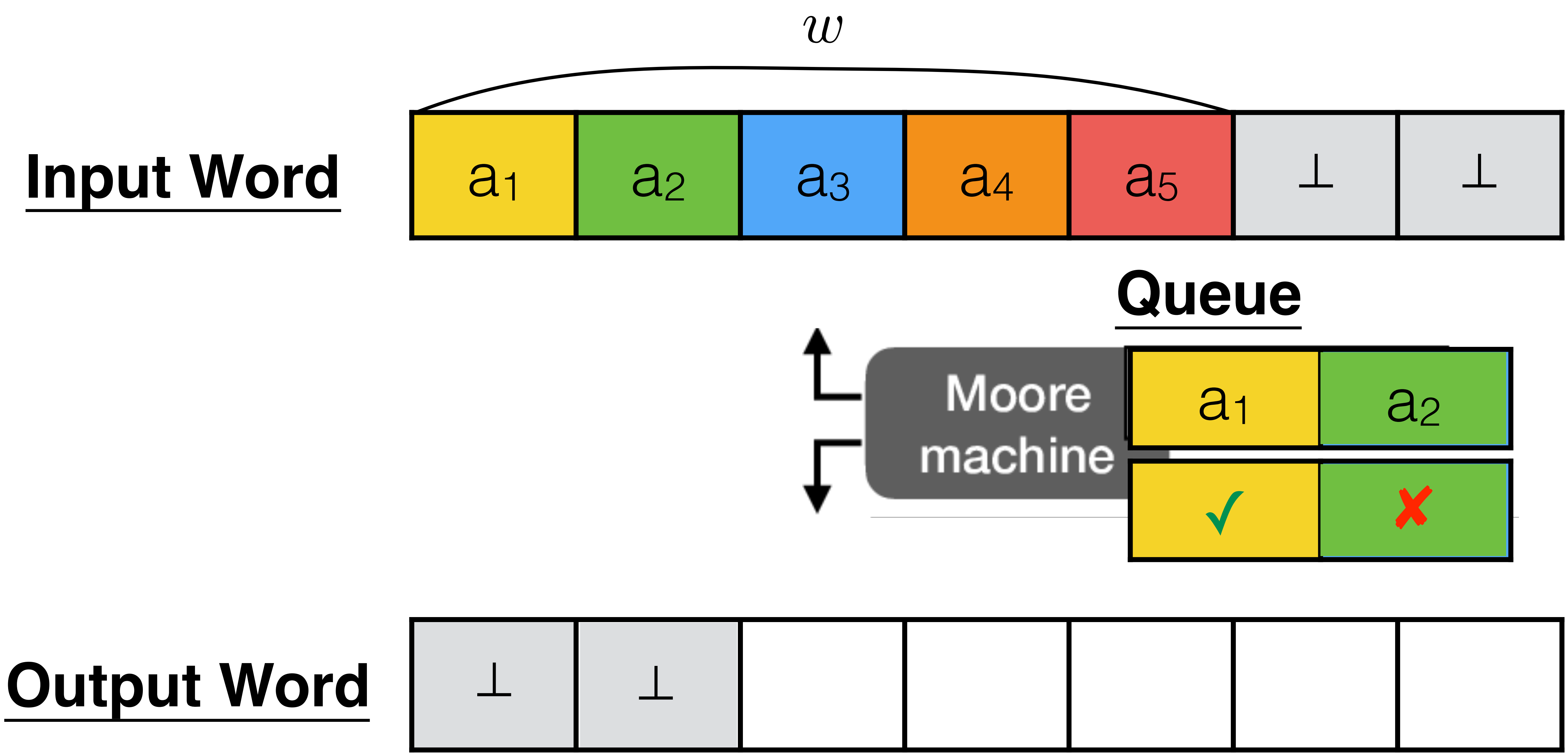


M. Waga (NII)

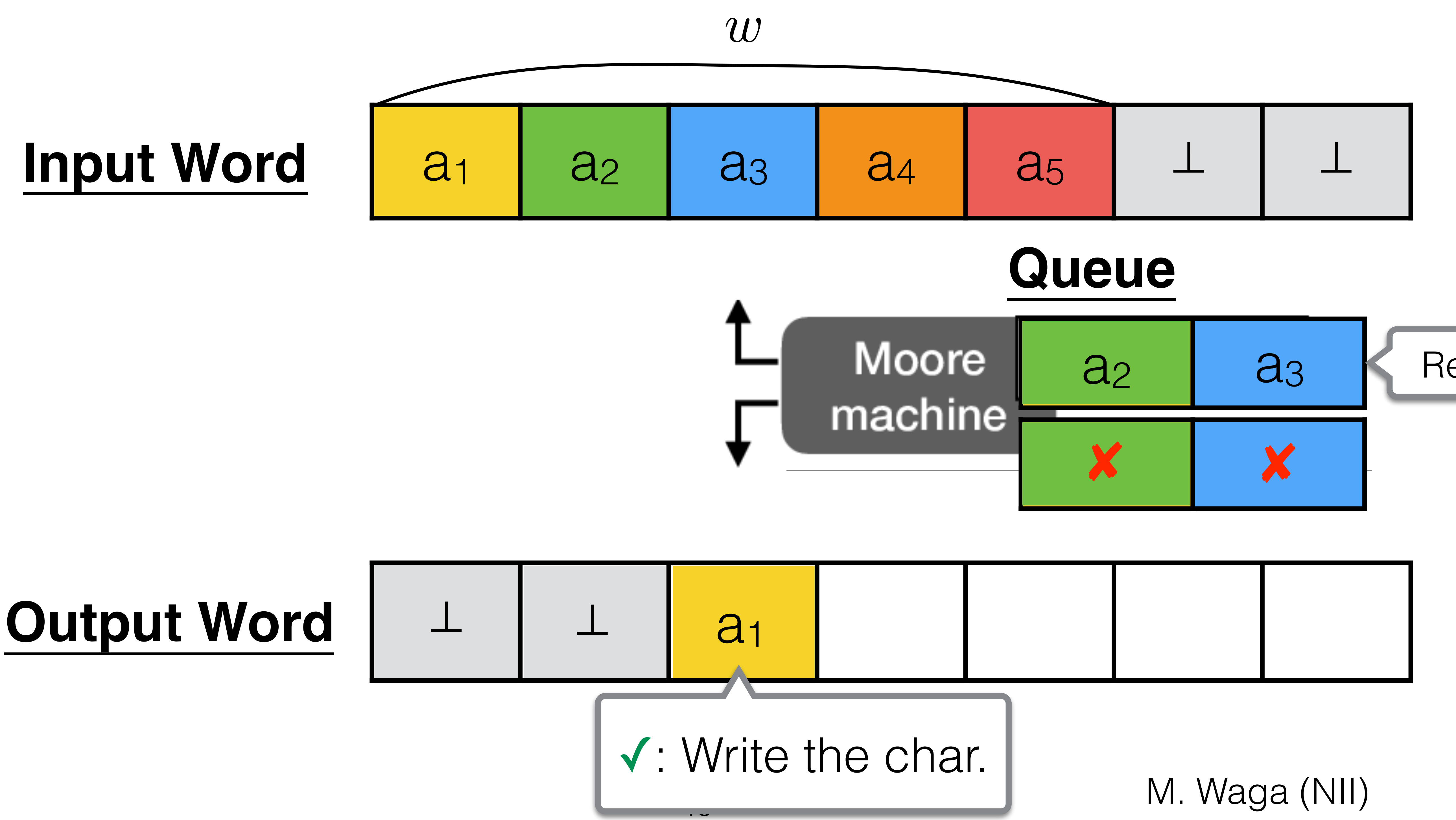
Overview of Filtering



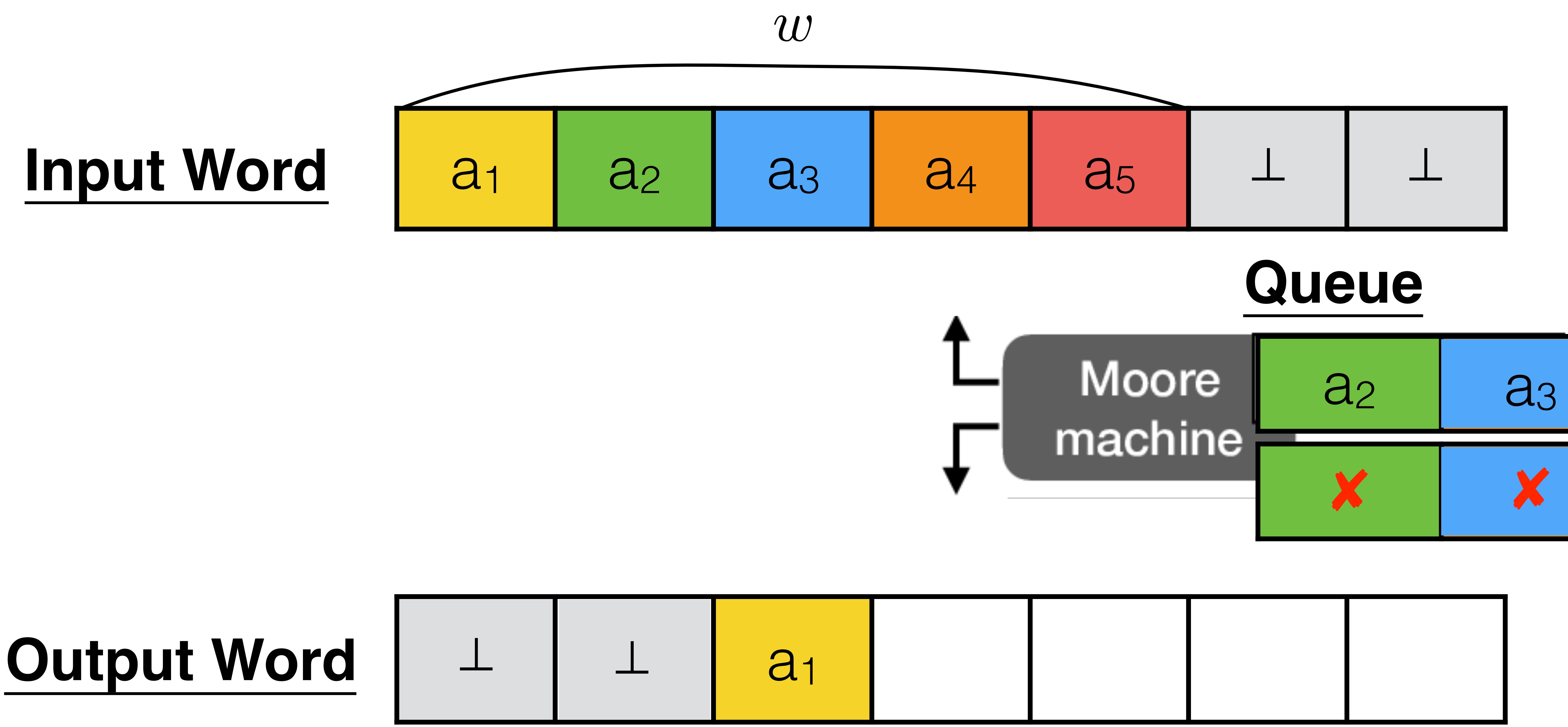
Overview of Filtering



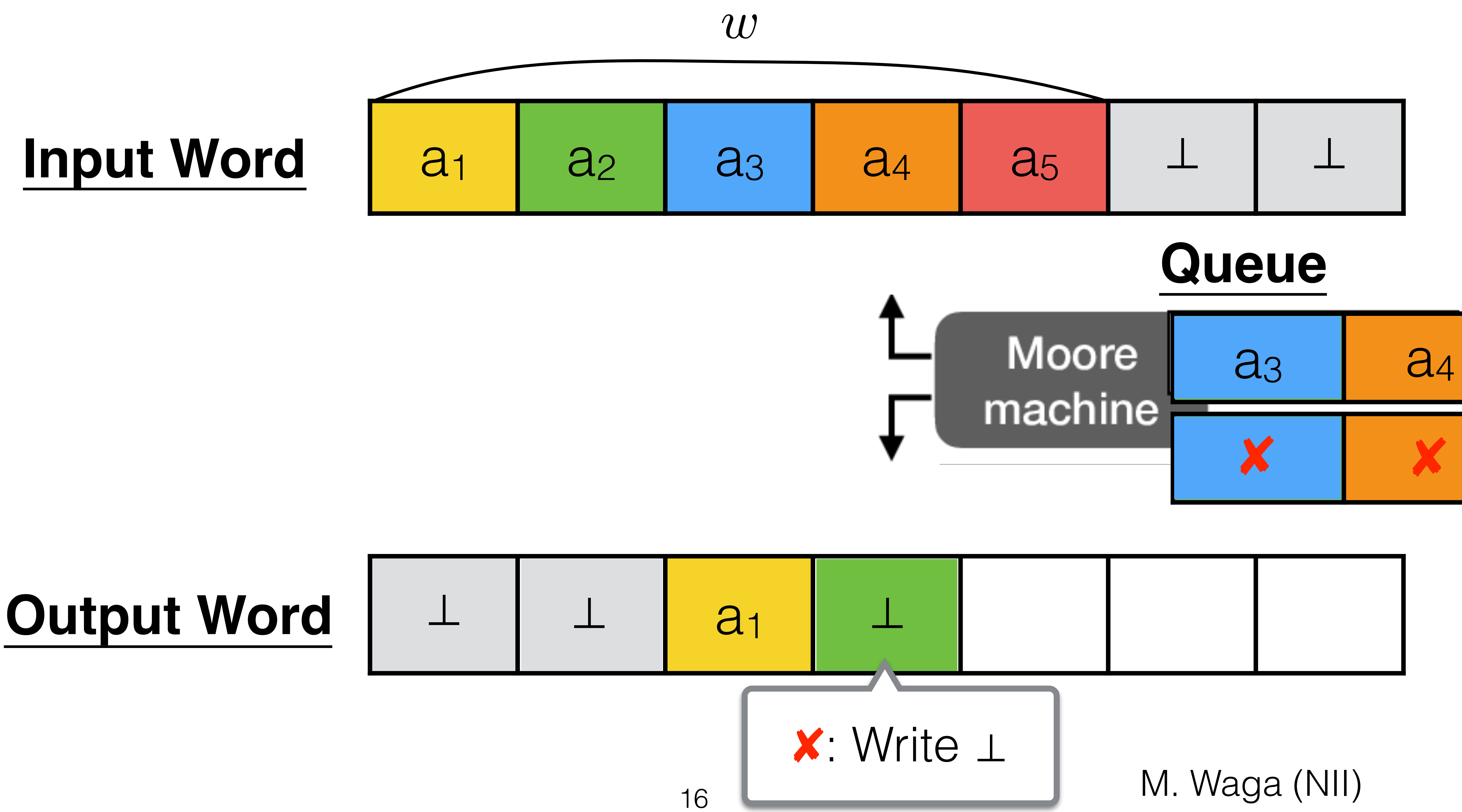
Overview of Filtering



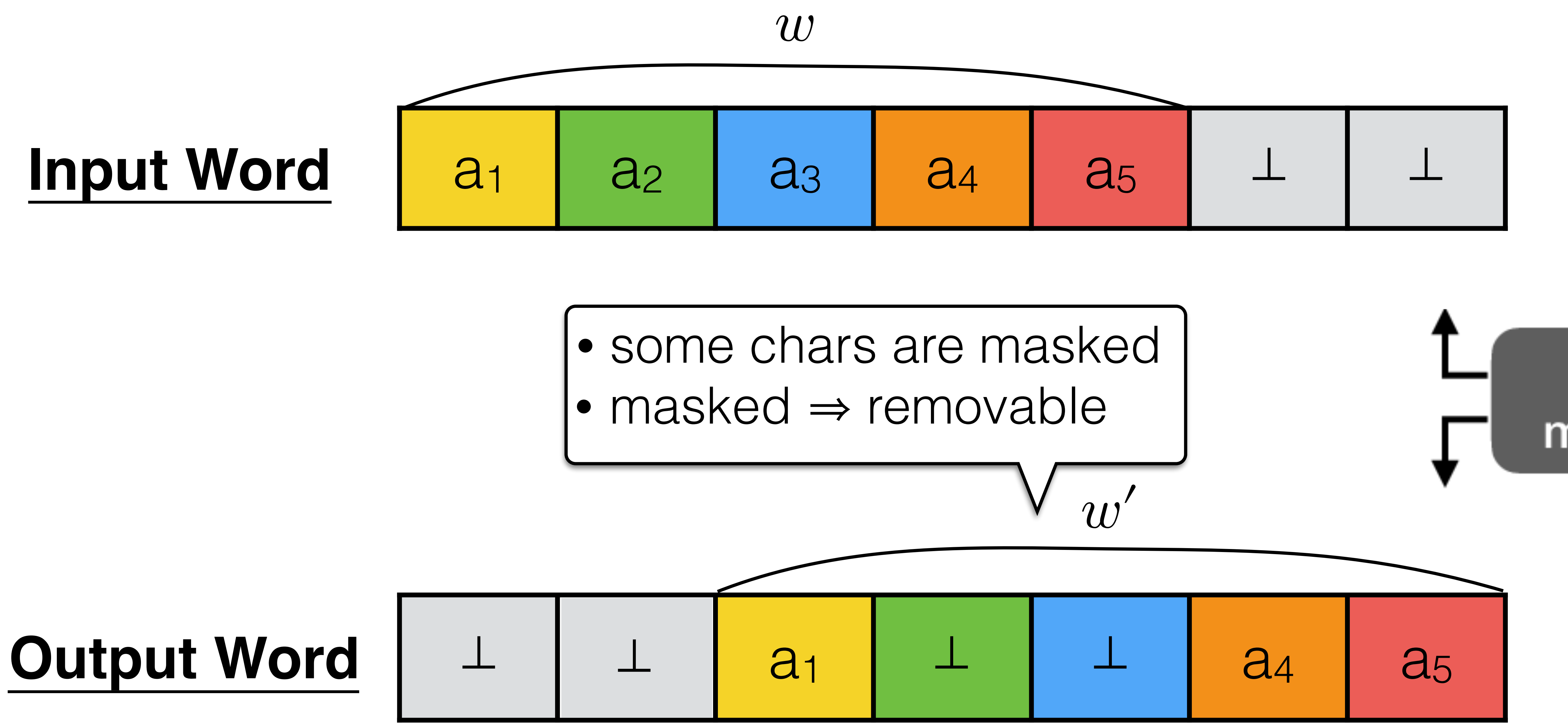
Overview of Filtering



Overview of Filtering



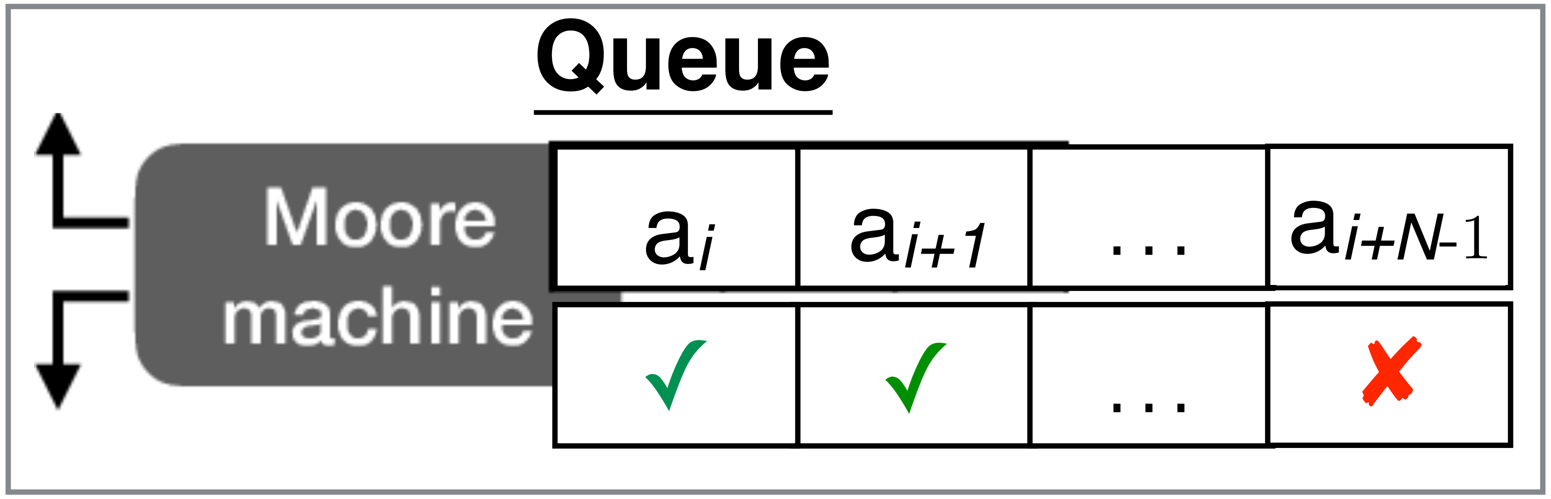
Overview of Filtering



Untimed Filter Construction

1. add a **counter** to each state
   ← Count: distance from the init. state mod N
2. **powerset** construction
   ← Moore machine (deterministic)
3. add **queue**

Idea of Mask / Unmask



- Surely Unused ⇒ **Mask** (✘)
- Surely Used ⇒ **Pass** (✓)
- Not Sure ⇒ **Pass** (✓)

M. Waga (NII)

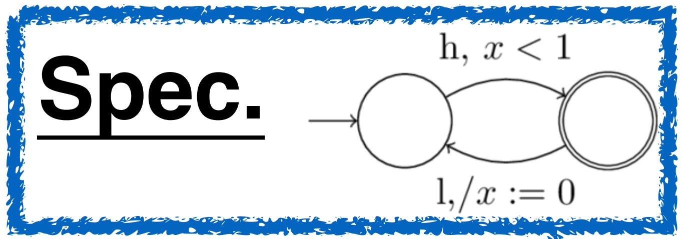
Properties of Untimed Moore Machine Filter

Log: $w = a_1 a_2 \dots a_n$ Pattern: $\mathcal{A}$ Queue size: N

Thm. (soundness)
Our filter Moore machine $\mathcal{M}$ does not change the result of the untimed pattern matching problem.

Thm. (finite completeness)
Our filter Moore machine $\mathcal{M}$ masks any unused char. if we have $\max\{|w| \mid w \in \mathrm{L}(\mathcal{A})\} \leq N < \infty$.

Thm. (monotonicity)
The Moore machine filter with N-length queue does not mask more events than the filter with with mN-length queue.



Outline

Sensor → **Filter** → **Monitor**

Theoretical preparation

- Moore Machine Filter for **Untimed** Pattern Matching
  - Spec.: NFA $\mathcal{A}$
  - Log: word ($w \in \Sigma^*$)

Main Problem

- Moore Machine Filter for **Timed** Pattern Matching
  - Spec.: TA $\mathcal{A}$
  - Log: timed word ($w \in (\Sigma \times \mathbb{R}_{>0})^*$)
- Experiments (**timed**)

M. Waga (NII)

Timed Pattern Matching

[Ulus et al., FORMATS'14]

Input

Log

- Timed Word $w \in (\Sigma \times \mathbb{R}_{>0})^*$
- TA $\mathcal{A}$

Spec.

Output

$\mathrm{Match}(w, \mathcal{A}) =$

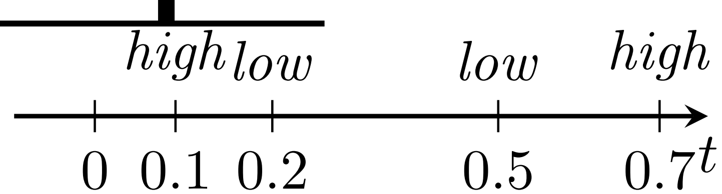
$\{(t,t') \mid w|_{(t,t')} \in L(\mathcal{A})\}$

Intervals where spec. is satisfied

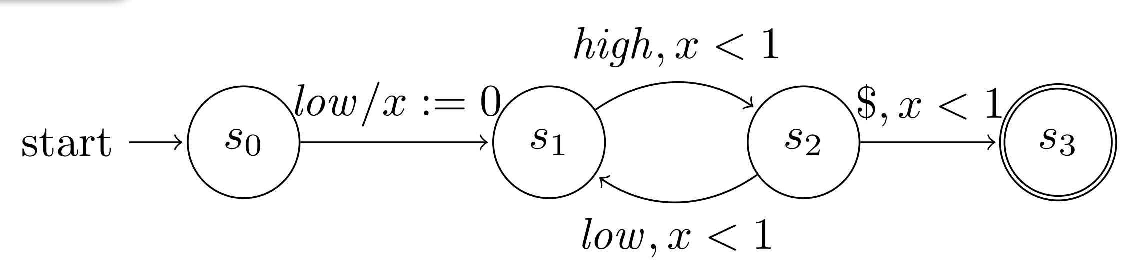
Find frequent gear change

Example

$w =$



$\mathcal{A} =$



$\mathrm{Match}(w, \mathcal{A}) = \{(t,t') \mid 0.2 \leq t < 0.5,\ 0.7 < t'\}$

M. Waga (NII)

Timed Filter Construction

1. add a **counter** to each state
   ← Count: distance from the init. state mod N
2. **powerset** construction
   ← impossible for timed automata
   (state space will be infinite)
   ⇒ Overapproximation!!
3. add **queue**

One-Clock Determinization

[Krichen & Tripakis, FMSD'09]

Thm.

For any TA $\mathcal{A}$, there exists a DRTA $\mathcal{A}^{\mathrm{rt}}$ s.t. $\mathrm{L}(\mathcal{A}) \subseteq \mathrm{L}(\mathcal{A}^{\mathrm{rt}})$

DRTA : (deterministic real-time autom.)

- Restricted form of TA
- Guard: the duration at the source state
- Transition function $\Delta : Q \times (\Sigma \times \mathbb{R}_{\geq 0}) \to Q$

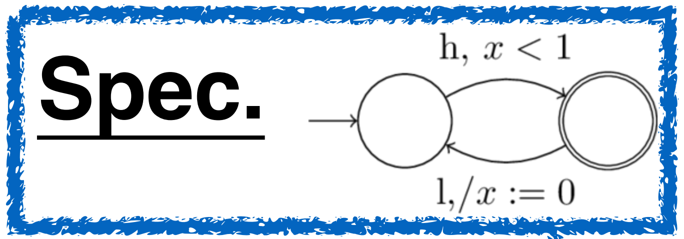
Timed Filter Construction

1. add a **counter** to each state
   ← Count: distance from the init. state mod N
2. **One-clock determinization**
   ← DRTA: similar to DFA
3. add **queue**

Properties of Timed Moore Machine Filter

Thm. (soundness)
Our filter Moore machine $\mathcal{M}$ does not change the result of the timed pattern matching problem.

Completeness does not hold ⇒ experimental results



Outline

Sensor → **Filter** → **Monitor**

Theoretical preparation

- Moore Machine Filter for **Untimed** Pattern Matching
  - Spec.: NFA $\mathcal{A}$
  - Log: word ($w \in \Sigma^*$)

Main Problem

- Moore Machine Filter for **Timed** Pattern Matching
  - Spec.: TA $\mathcal{A}$
  - Log: timed word ($w \in (\Sigma \times \mathbb{R}_{>0})^*$)
- Experiments (**timed**)

M. Waga (NII)

Research Questions

Sensor → Filter → Monitor

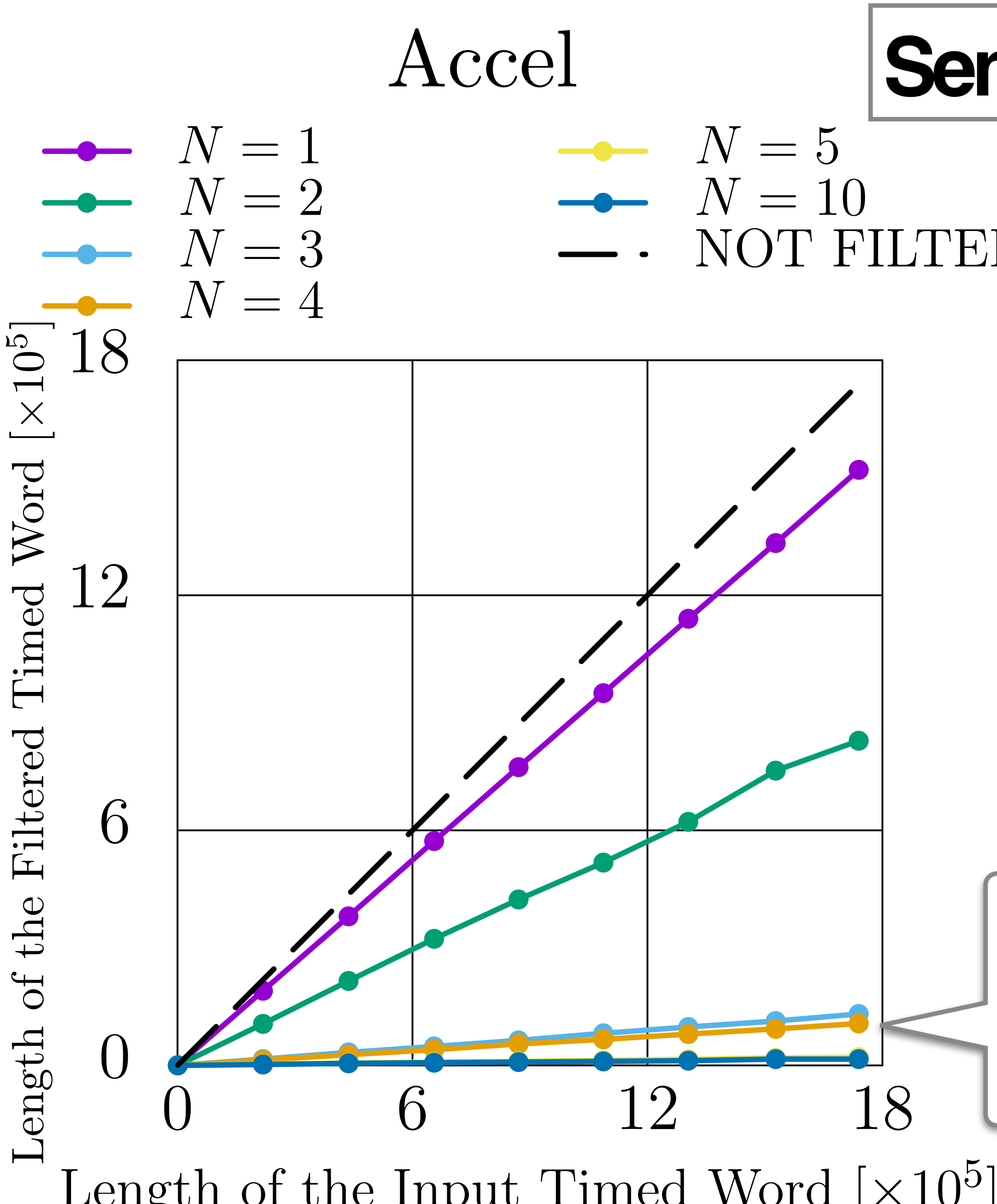
- RQ1: Does our filter Moore machine mask many events?
  - Theoretically, only soundness
  - Queue size vs. filtering rate
- RQ2: Is our filter Moore machine online capable?
  - Linear time? Constant memory usage?

Environment of Experiment

- MacBook Pro Early 2013
  - Intel Core i5 2.6 GHz and DDR3 1600MHz 8 GB
- Mac OS 10.13.4
- clang-900.0.39.2 with optimization flag -O3
- Used 3 benchmarks (Accel, Gear, Torque)

Automotive

RQ1: Filtering Rate



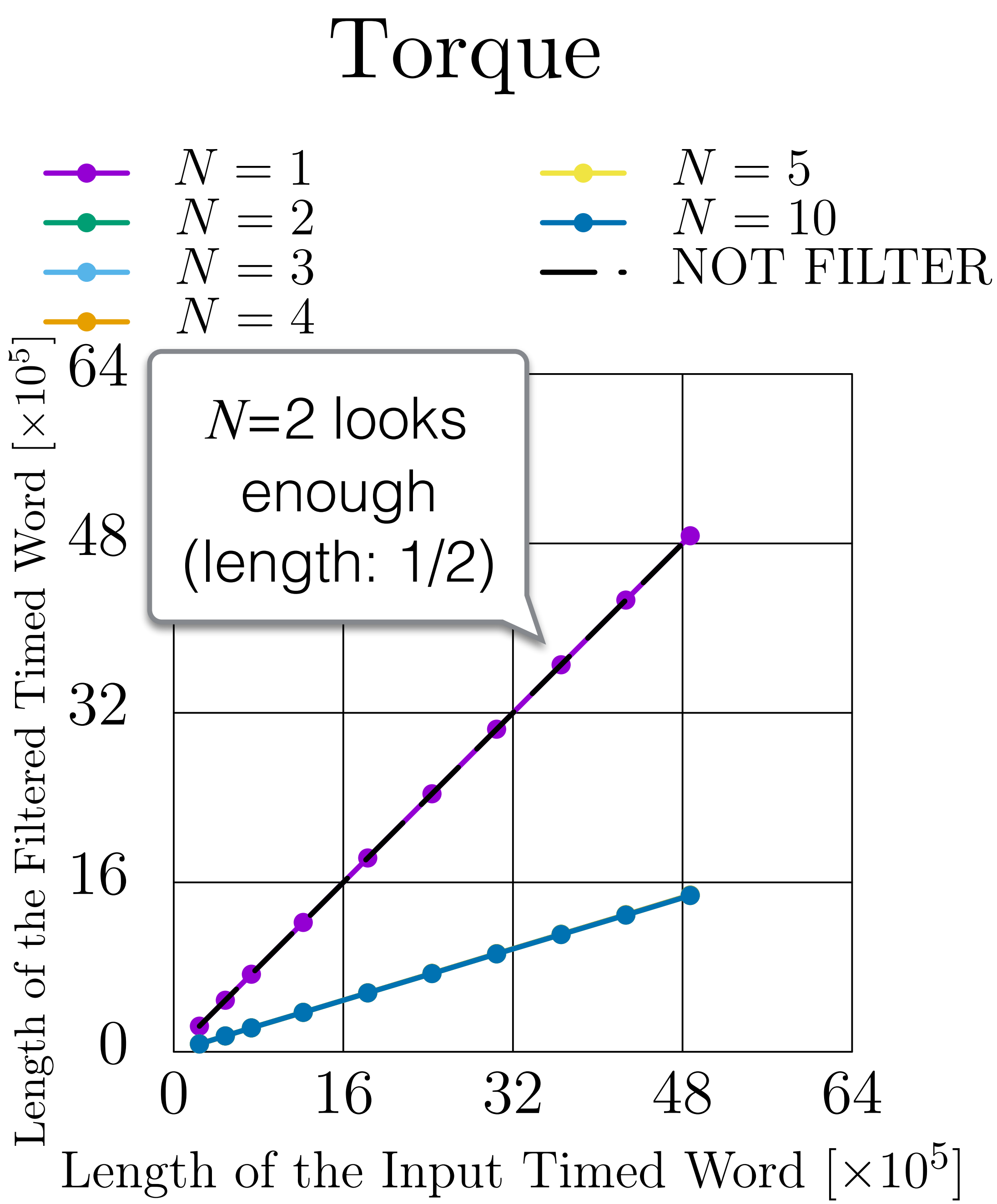
N: size of queue

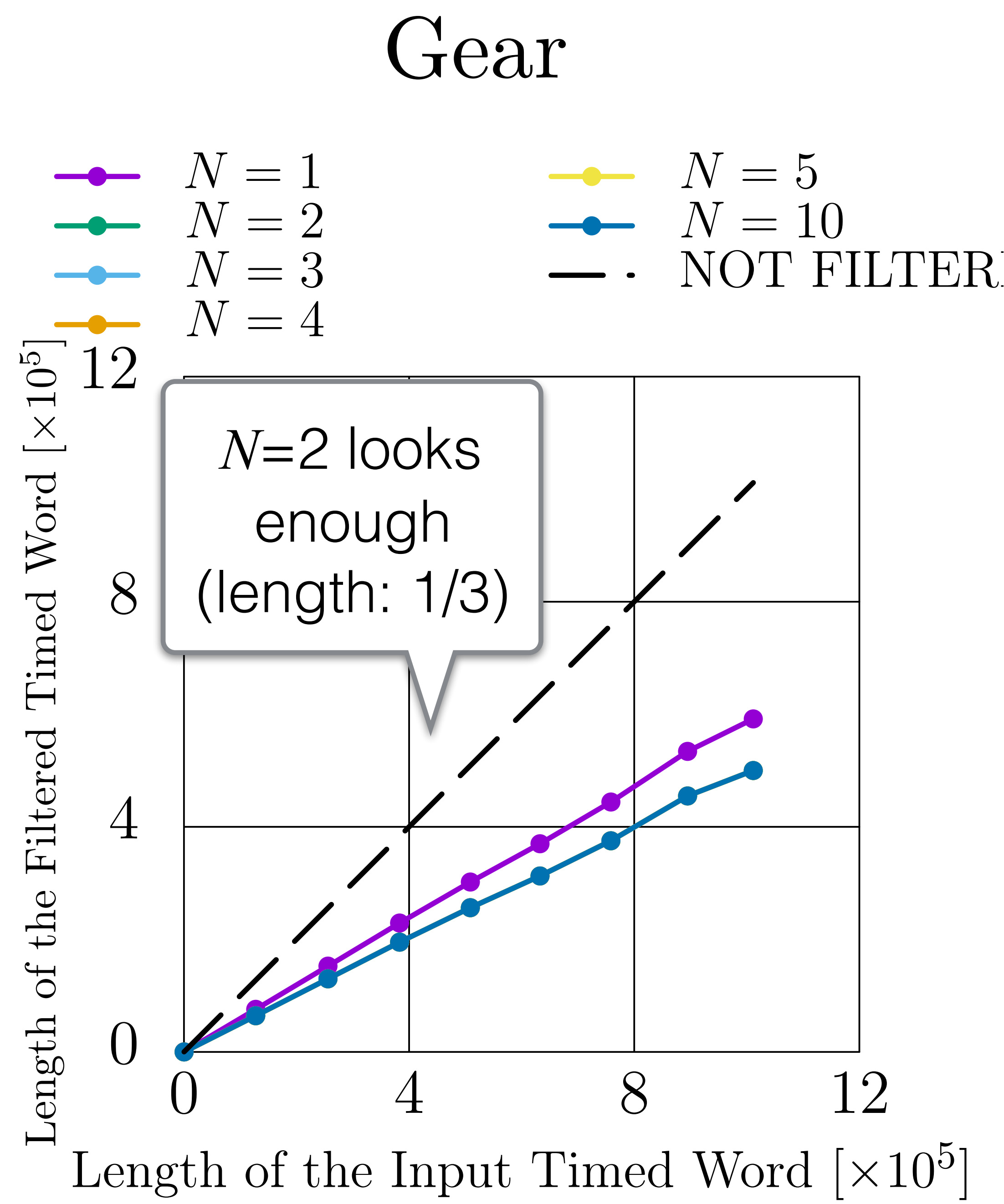
N=10 looks enough (length: 1/100)

M. Waga (NII)

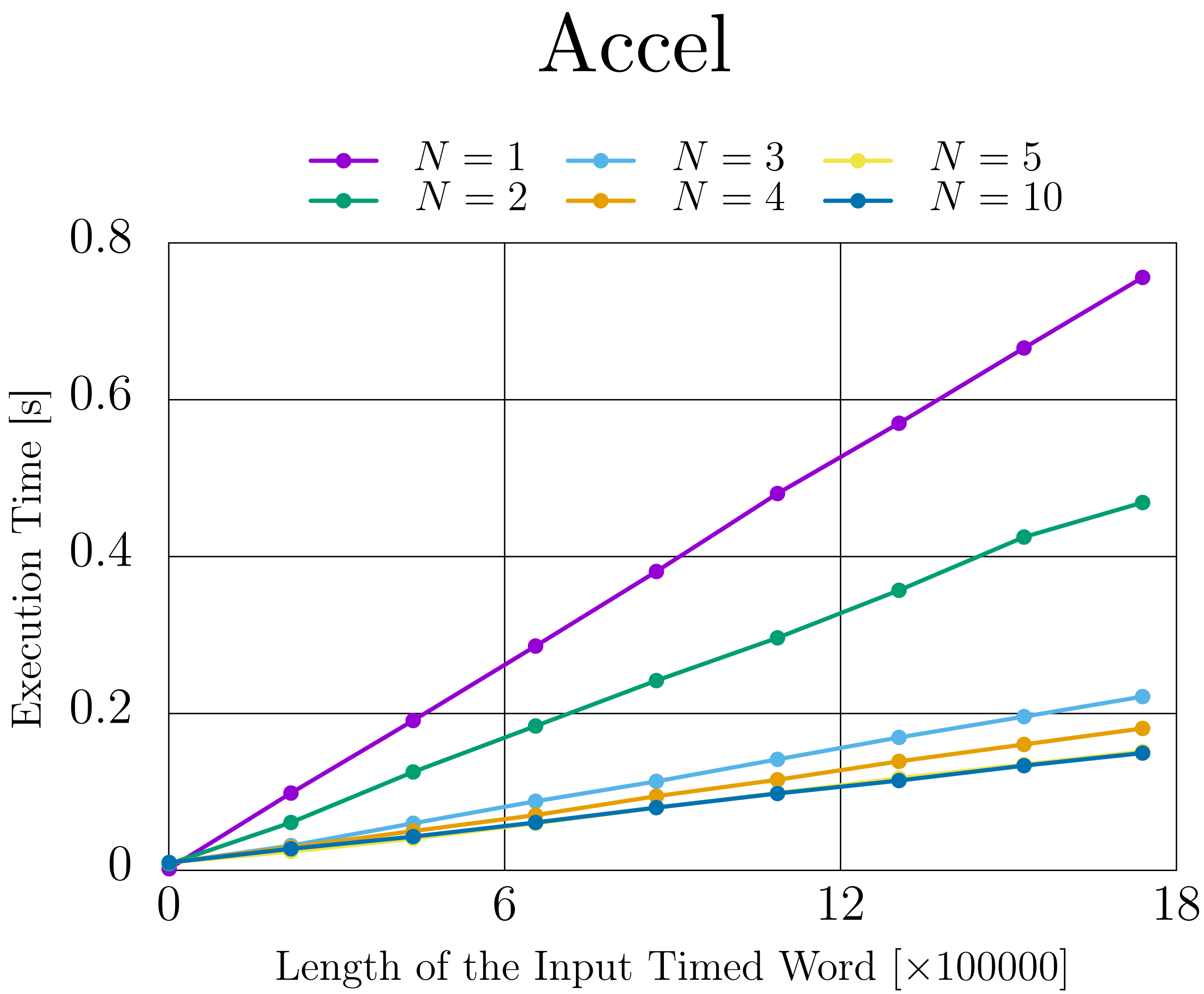
RQ1: Filtering Rate (Cont'd)

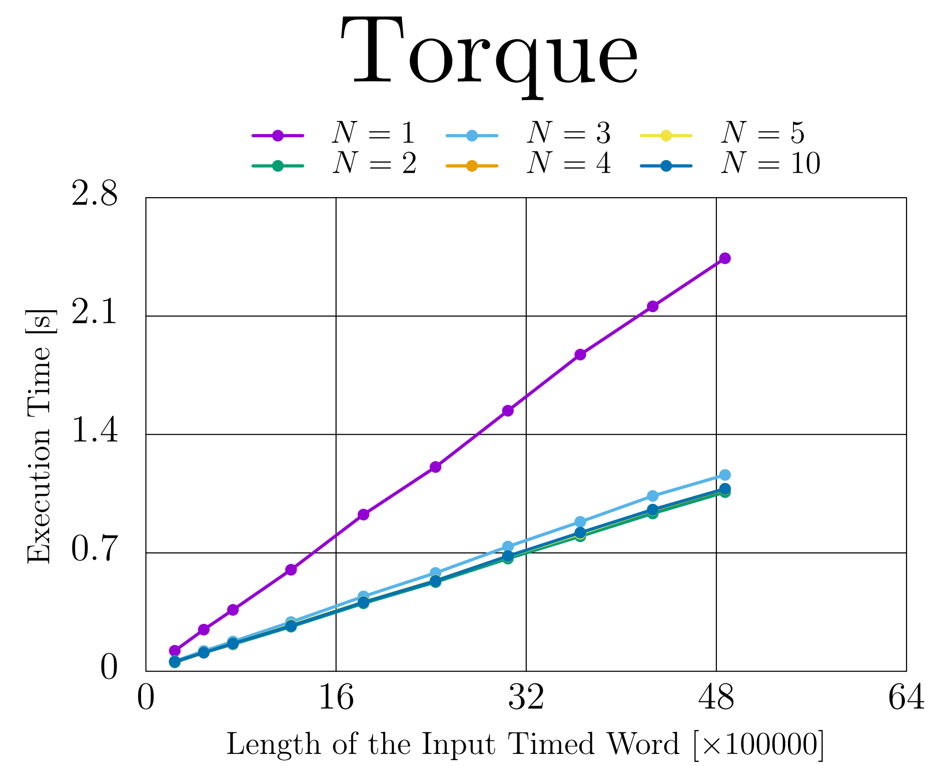
Sensor → Filter → Monitor

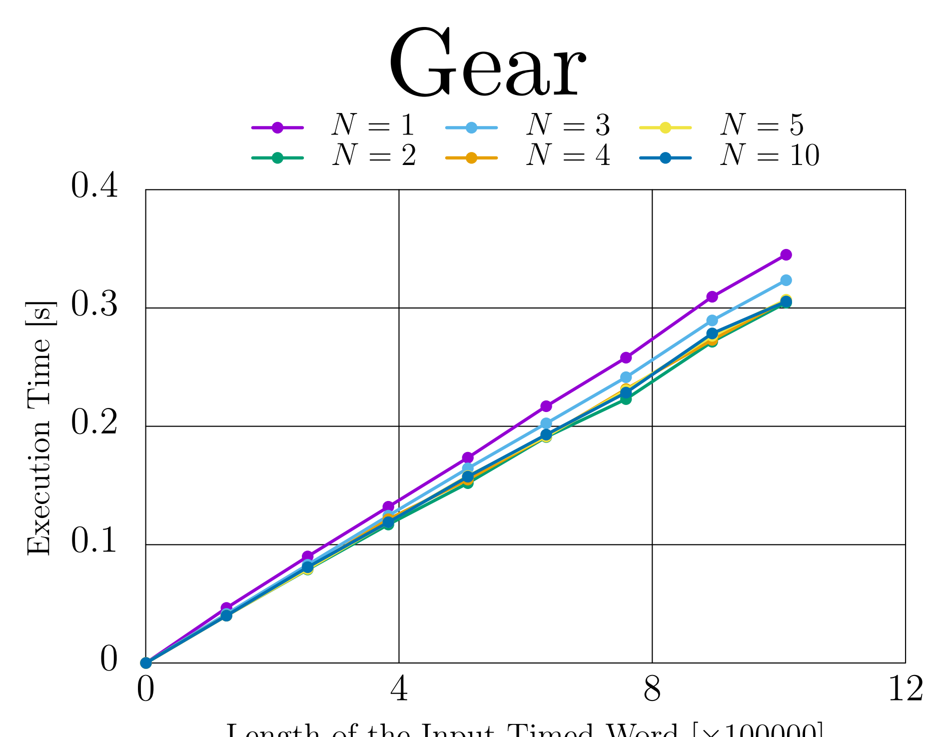




RQ2: Execution Time

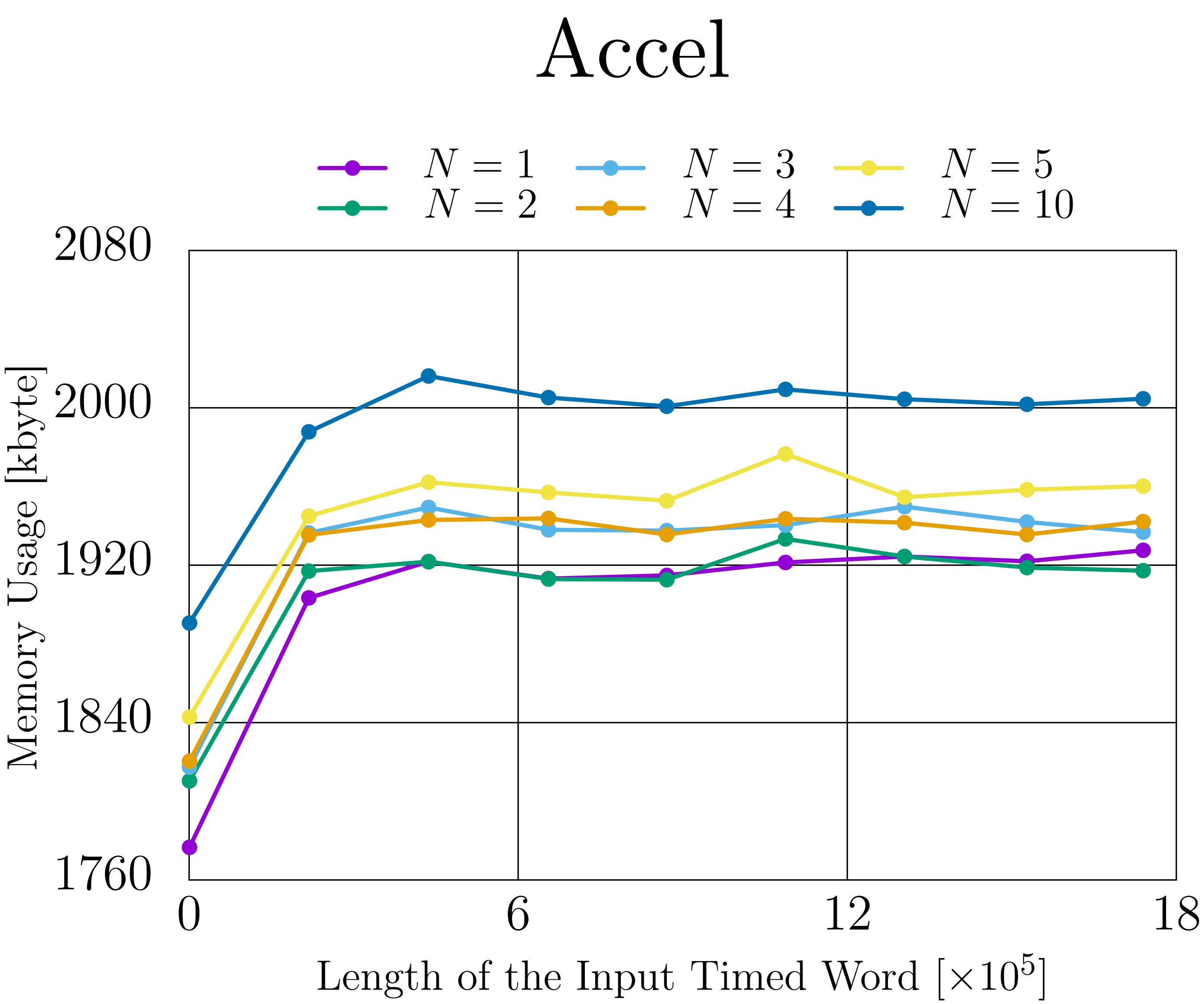


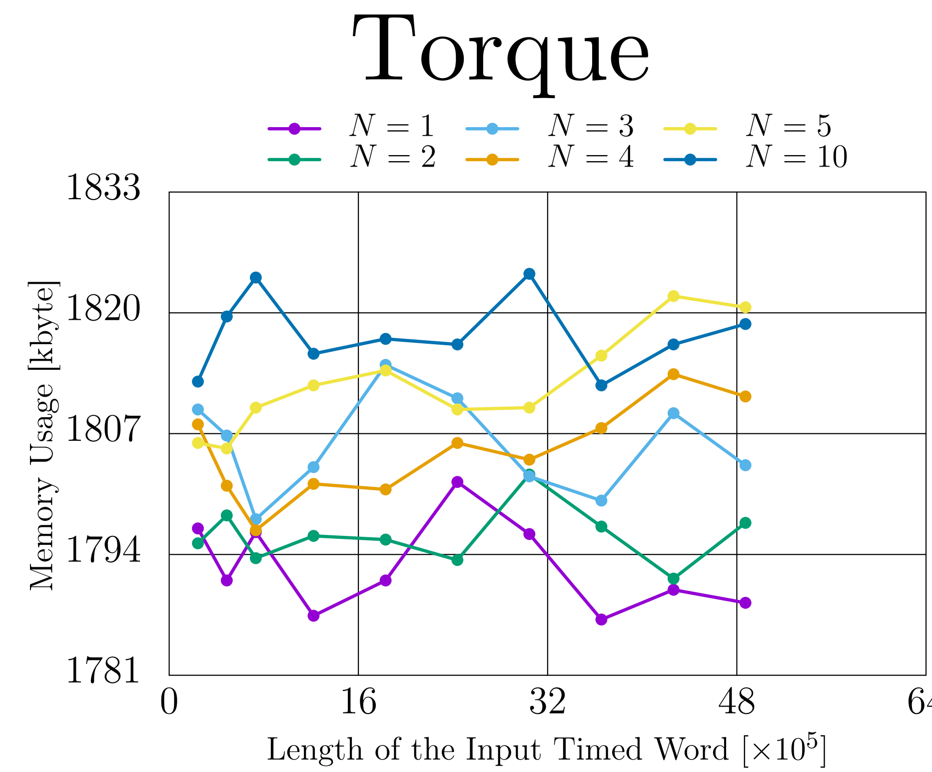


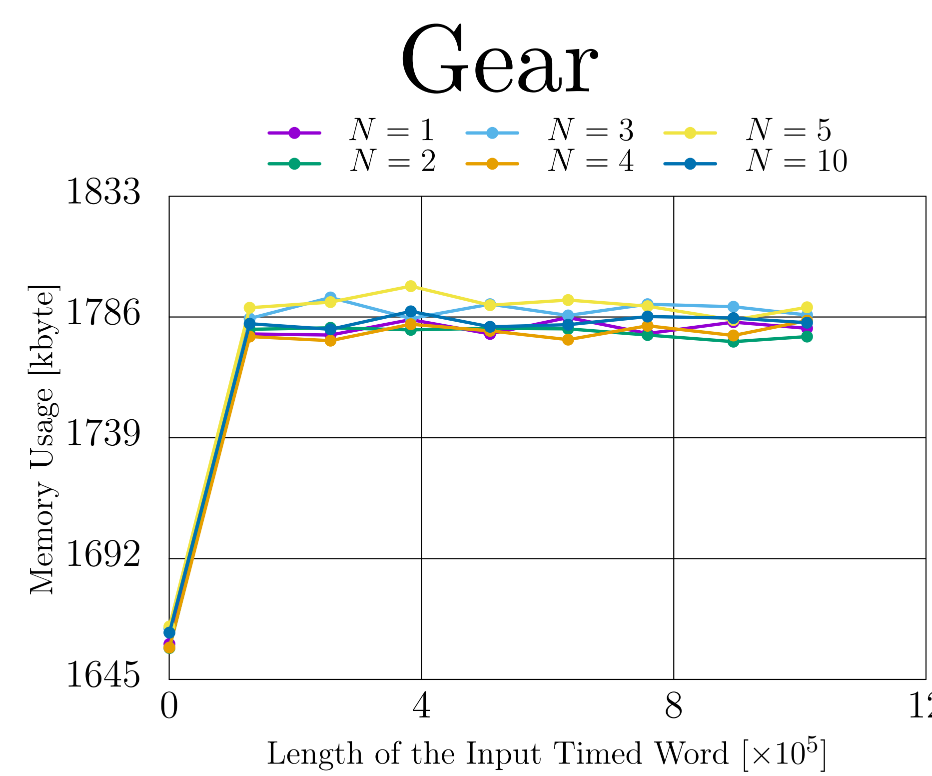


Linear Execution Time

RQ2: Memory Usage







Constant Memory Usage

M. Waga (NII)

Conclusion

Sensor → Filter → Monitor

- We proposed Moore machine filter for untimed/timed pattern matching
- Reduces the data size to send
- online capable
- (omitted in this talk) it also accelerates timed pattern matching.

Future Works

- Case study in an embedded system
- Extension to distributed systems
  - Many sensor node vs one central server
- Hardware Implementation (e.g., FPGA, ASIC)