

When is CAN the weakest link?

A bound on Failures-In-Time in CAN-Based Distributed Real-Time Systems

Arpan Gujarati

Björn B. Brandenburg



Max
Planck
Institute
for
Software Systems

Failures due to Transient Faults

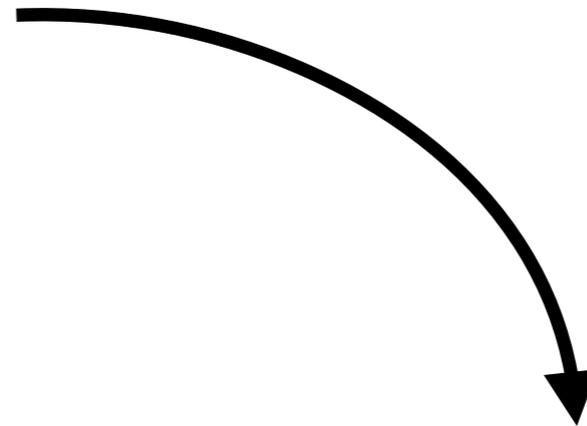
Harsh environments

- ➔ Spark plugs
- ➔ Hard radiation
- ➔ High-power machinery

Failures due to Transient Faults

Harsh environments

- ➔ Spark plugs
- ➔ Hard radiation
- ➔ High-power machinery



Electromagnetic Interference (EMI)

- ➔ **Bit-flips** in the hosts
- ➔ ... and in the network

Failures due to Transient Faults

Harsh environments

- ➔ Spark plugs
- ➔ Hard radiation
- ➔ High-power machinery

Electromagnetic Interference (EMI)

- ➔ **Bit-flips** in the hosts
- ➔ ... and in the network

EMI-induced transient faults

- ➔ Manifest as **program-visible failures**

Failures due to Transient Faults

Transmission failures
(faults on the wire)

Commission failures
(bit-flips in the memory buffers)

Crash failures
(due to fault-induced exceptions)

Failures due to Transient Faults

Transmission failures
(faults on the wire)

Commission failures
(bit-flips in the memory buffers)

Crash failures
(due to fault-induced exceptions)



Tolerated by **error detection**
and **retransmissions**

Failures due to Transient Faults

Transmission failures
(faults on the wire)



Tolerated by **error detection**
and **retransmissions**

Commission failures
(bit-flips in the memory buffers)



Tolerated by active
replication of tasks on
independent hosts

Crash failures
(due to fault-induced exceptions)

Failures due to Transient Faults

Transmission failures
(faults on the wire)

Tolerated by **error detection**
and **retransmissions**

Commission failures
(bit-flips in the memory buffers)

Tolerated by active
replication of tasks on
independent hosts

Crash failures
(due to fault-induced exceptions)

- How to decide the **best replication strategy**?
 - ➔ Is Triple Modular Redundancy (TMR) enough? or is Quadruple Modular Redundancy (QMR) required?
 - ➔ Would you replicate only the high-frequency tasks? or only the high-criticality tasks?

Retransmissions vs. Replication Tradeoff

Retransmissions vs. Replication Tradeoff

For tolerating retransmissions-induced delays

- (Ensure no deadline violations)
- **The more slack, the better!**

Retransmissions vs. Replication Tradeoff

For tolerating retransmissions-induced delays

- (Ensure no deadline violations)
- **The more slack, the better!**

versus

Active replication of tasks

- **Reduced slack** in the schedule

Retransmissions vs. Replication Tradeoff

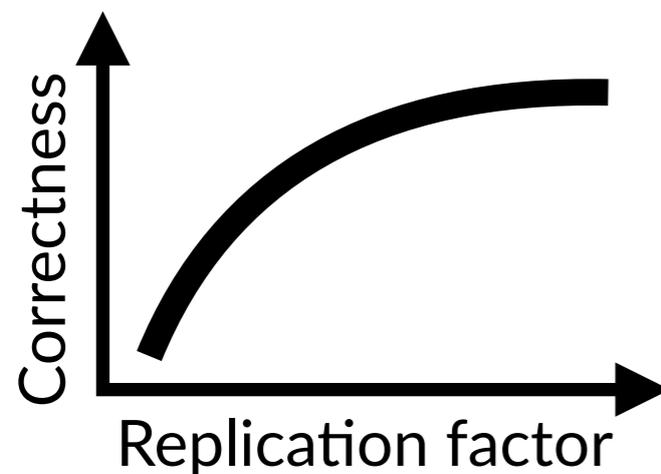
For tolerating retransmissions-induced delays

- (Ensure no deadline violations)
- **The more slack, the better!**

versus

Active replication of tasks

- **Reduced slack** in the schedule



Retransmissions vs. Replication Tradeoff

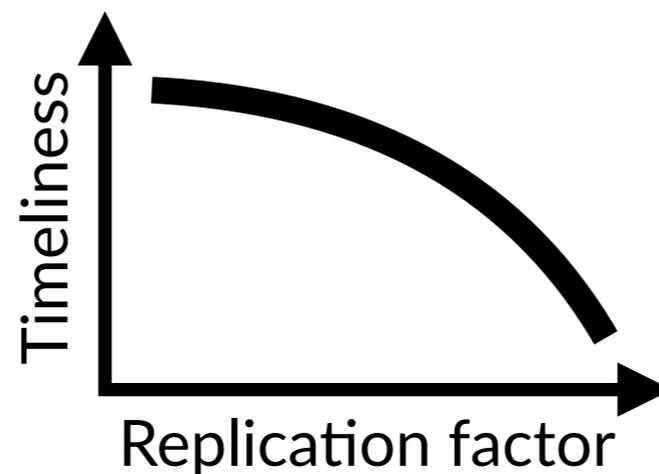
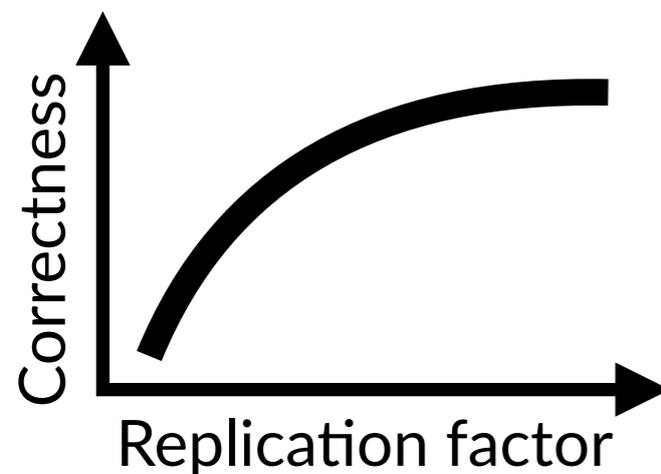
For tolerating retransmissions-induced delays

- (Ensure no deadline violations)
- **The more slack, the better!**

versus

Active replication of tasks

- **Reduced slack** in the schedule



Retransmissions vs. Replication Tradeoff

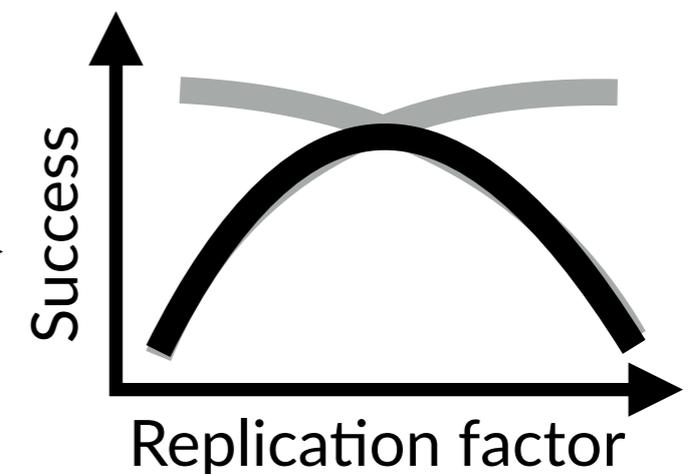
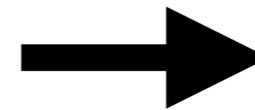
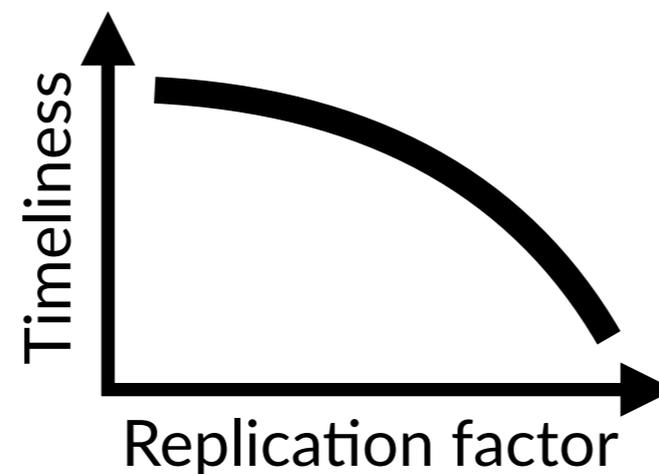
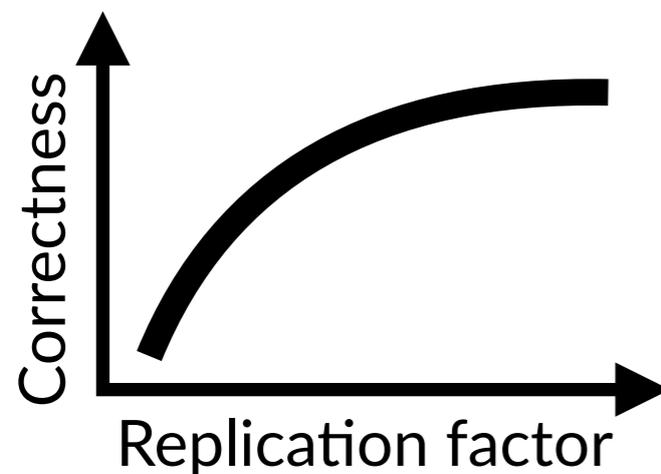
For tolerating retransmissions-induced delays

- (Ensure no deadline violations)
- **The more slack, the better!**

versus

Active replication of tasks

- **Reduced slack** in the schedule



Retransmissions vs. Replication Tradeoff

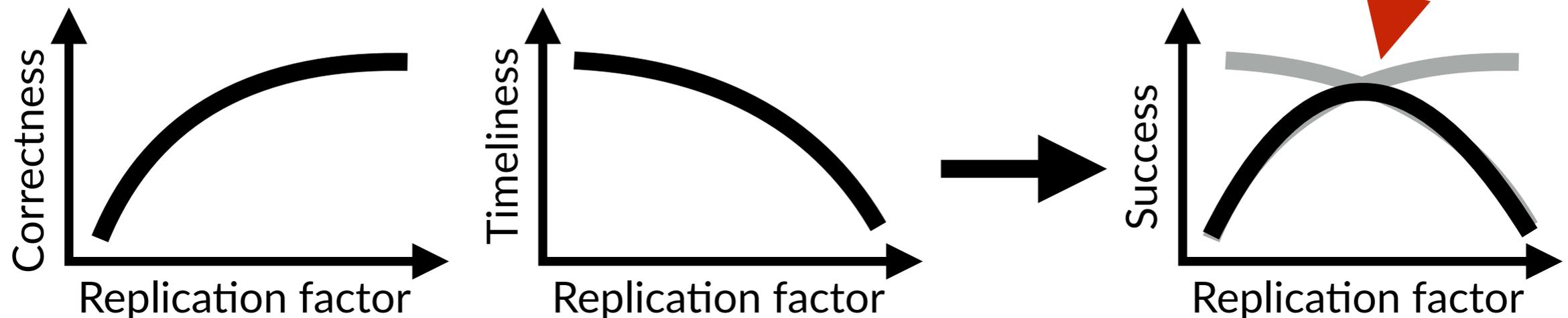
For tolerating retransmissions-induced delays

- (Ensure no deadline violations)
- **The more slack, the better!**

versus

Active replication of tasks

- **Reduced slack** in the schedule



How to statically determine the **optimal replication factor**?

This Work

For CAN-based distributed real-time systems...

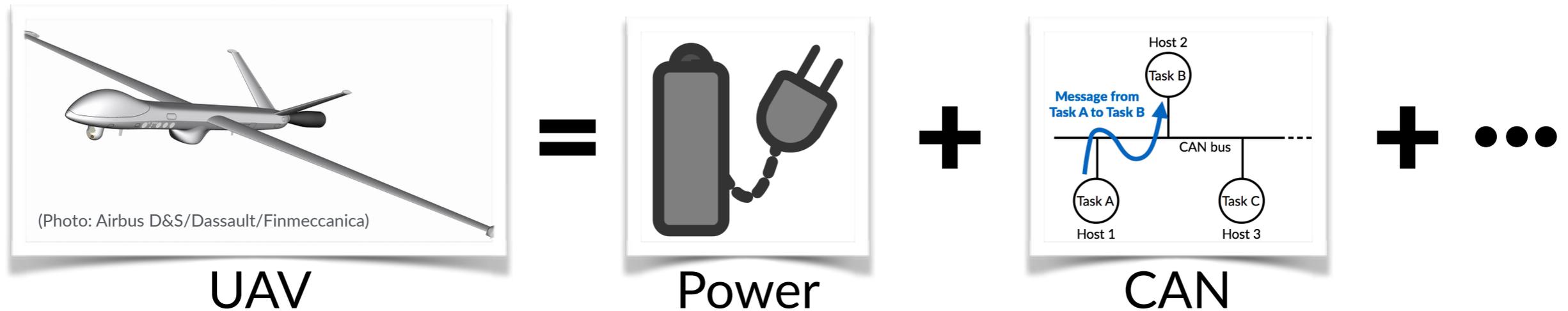
- **Probabilistic analysis**
 - Quantify the replication vs. retransmissions tradeoff

The Larger Picture

The CAN-based system is just one component in a safety-critical system...

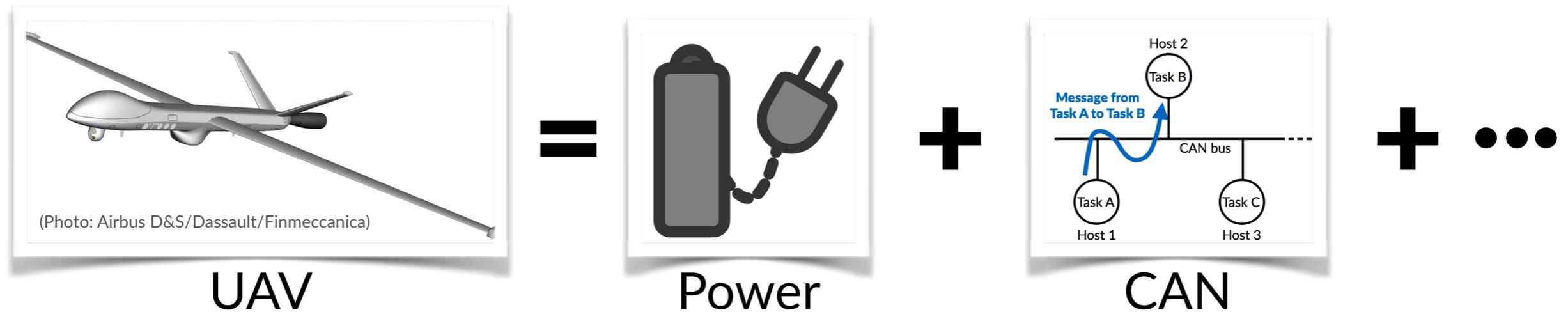
The Larger Picture

The CAN-based system is just one component in a safety-critical system...



The Larger Picture

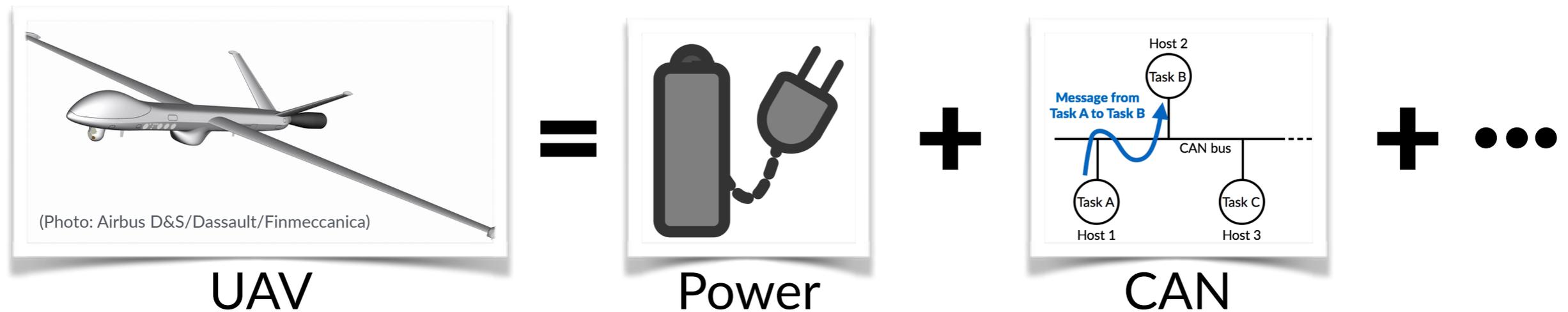
The CAN-based system is just one component in a safety-critical system...



Replicate tasks, add more ECUs to the CAN subsystem

The Larger Picture

The CAN-based system is just one component in a safety-critical system...

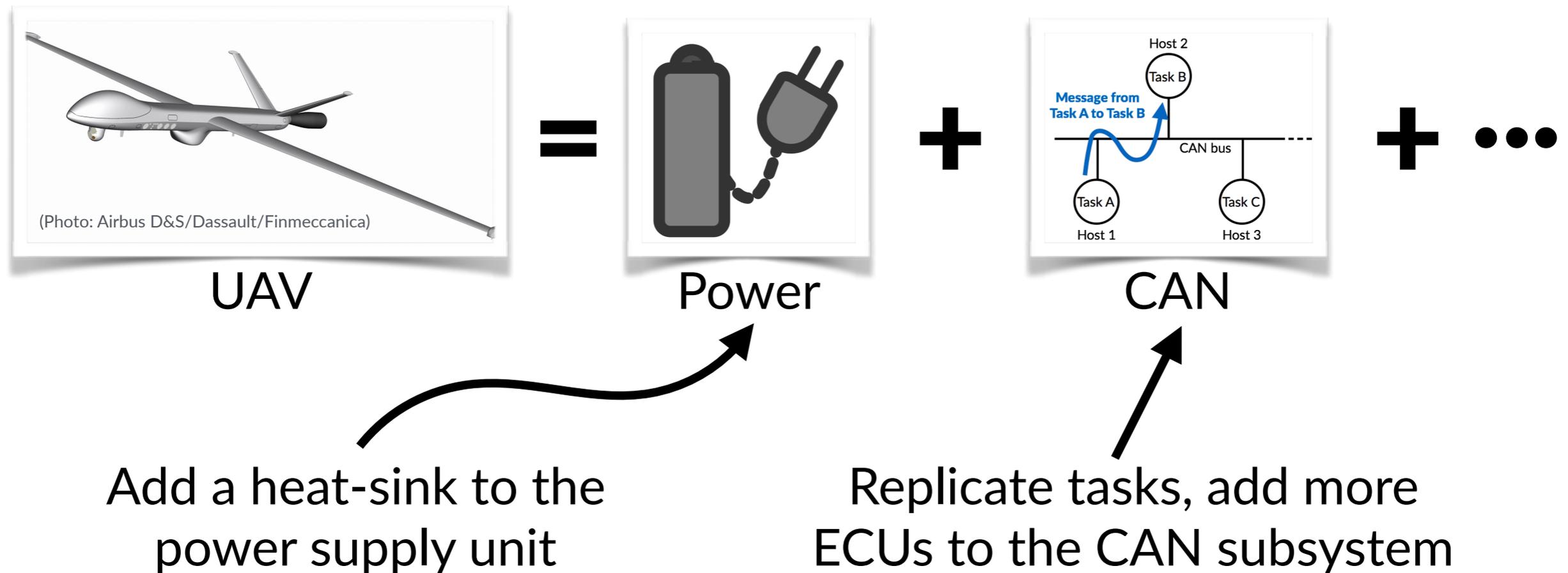


Add a heat-sink to the power supply unit

Replicate tasks, add more ECUs to the CAN subsystem

The Larger Picture

The CAN-based system is just one component in a safety-critical system...



What if the UAV has **strict weight constraints?**

- ➔ and you can either add the heat sink or the additional ECUs
- ➔ How do you decide the **best choice?**

Failures-In-Time (FIT) Rate

Expected #failures in one billion operating hours

→ e.g., 1M UAVs flying for 1K hours each

HIGH TEMPERATURE GATE BIAS (HTGB)

Table 1. FIT Rate Calculations for AFCT-5717ATPZ

Component	Telecordia Information/ Data Source	Quantity	Component Base Rate (FITs)	Quality Factor	Total Component Failure Rate (FITs)
DFB Laser	Avago Data @ 40 °C	1	20.0	0.8	16.0
Monitor PIN	Photodiode	1	7.7	0.8	6.2
10G PIN	Photodiode	1	7.7	0.8	6.2
Capacitors	Fixed Ceramic	27	0.2	1	5.4
Resistor	Thick Film	21	0.51	1	10.7
Thermistor	Thermistor	1	2.10	1	2.1
Ferrite Chip (Inductor)	Power Filter	14	2.30	1	32.2
MOSFET	Supplier Info: On Semiconductor	1	4.00	1	4.0
EEPROM	2 Kbit CMOS	1	6.40	1	6.4
DAC	Supplier Info: National Semiconductor	1	6.00	1	6.0
Post-Amp IC, Gennum 16QFN	Assume: (91 - 170 Transistor)	1	23.00	1	23.0
Laser Driver IC	Supplier Info: Vitesse	1	6.4	1	6.4
µProcessor	Supplier Info: Atmel	1	28.0	1	28.0
Connector	PCB, Edge / Multi-Pin	20	0.130	1	2.6
Total Module Failure Rate @ 40 °C (FITs)					157.70
MTBF @ 40 °C (Hours)					6.34E+06

Temperature Factor @ 40°C: 1
Stress Factor at 50%: 1
Environmental Factor: 1

Reliability Prediction Based On
Telecordia SR-332 Issue 2 - Parts Count Method

FITs at other temperatures can be derived following the procedure of Telcordia SR-332, assuming activation energy, E_a , of 0.35 eV to determine the component temperature factor π . Table 2 shows FITs at different temperatures for the transceiver.

Table 2. FIT rates at different operating case temperatures, following the Telcordia Parts Count Method

T _{case} (°C)	FITs	MTBF (Hours)
25	82	1.22E+07
40	158	6.34E+06
50	236	4.24E+06
60	344	2.91E+06
70	490	2.04E+06

The limitations of the FIT prediction based on the Parts Count Method include the fact that the piece part failure rates are mostly obtained from Telcordia database, which may not be exhaustive for state-of-the-art piece parts, and that the results are independent of true module environmental stress tests. Nevertheless, the information obtained from the Parts Count Method is a useful reference during design-in and evaluation. Whenever possible, Avago substitutes internal data for the FIT rates of individual components, and predictions will be updated as more current data becomes available.

Two other commonly used terms for mean time between failures, the inverse of failure rate) and MTTF (mean time to failure), defined as 1/λ. MTBF is useful for equipment that will be repaired and then returned to service, but despite the commonplace assumption, it does not guarantee a minimum time between failures, only a mean. MTTF is technically more correct mathematically, but the two terms are [except for a few situations] equivalent and MTBF is the more commonly used.

FAILURE RATE @ 90°C & 60% UCL

HRS	0°C	3724
10E+05		3579
10E+06		1825

FAILURE RATE @ 90°C & 60% UCL

ALENT	HRS	0°C	3722
10E+05		3722	
10E+06		3377	
10E+07		3665	
10E+08		904	

that stem operation. Intermittent operation.

Concern. A supply surely and cause essential.

and how it can

related term that needs is the amount of time ate in its intended does not necessarily some applications a short service life.

Figure 1: The bathtub curve, failure rate plotted against time with the three life-cycle phases: infant mortality, useful life and wear-out.

FIT rates are widely used in the industry

Failures-In-Time (FIT) Rate

Expected #failures in one billion operating hours

→ e.g., 1M UAVs flying for 1K hours each

Table 2. FIT rates at different operating case temperatures, following the Telcordia Parts Count Method

T _{case} (°C)	FITs	MTBF (Hours)
25	82	1.22E+07
40	158	6.34E+06
50	236	4.24E+06
60	344	2.91E+06
70	490	2.04E+06

HIGH TEMPERATURE GATE BIAS (HTGB)

Table 1. FIT Rate Calculations for AFCT-5717ATPZ

Component	Telcordia Information/ Data Source	Quantity	Component Base Rate (FITs)	Quality Factor	Total Component Failure Rate (FITs)
DFB Laser	Avago Data @ 40 °C	1	20.0	0.8	16.0
Monitor PIN	Photodiode	1	7.7	0.8	6.2
10G PIN	Photodiode	1	7.7	0.8	6.2
Capacitors	Fixed Ceramic	27	0.2	1	5.4
Resistor	Thick Film	21	0.51	1	10.7
Thermistor	Thermistor	1	2.10	1	2.1
Ferrite Chip (Inductor)	Power Filter	14	2.30	1	32.2
MOSFET	Supplier Info: On Semiconductor	1	4.00	1	4.0
EEPROM	2 Kbit CMOS	1	6.40	1	6.4
DAC	Supplier Info: National Semiconductor	1	6.00	1	6.0
Post-Amp IC, Gennum 16QFN	Assume: (91 - 170 Transistor)	1	23.00	1	23.0
Laser Driver IC	Supplier Info: Vitesse	1	6.4	1	6.4
µProcessor	Supplier Info: Atmel	1	28.0	1	28.0
Connector	PCB, Edge / Multi-Pin	20	0.130	1	2.6
Total Module Failure Rate @ 40 °C (FITs)					157.70
MTBF @ 40 °C (Hours)					6.34E+06

Temperature Factor @ 40°C: 1
Stress Factor at 50%: 1
Environmental Factor: 1

Reliability Prediction Based On Telcordia SR-332 Issue 2 - Parts Count Method

Failure Rate @ 90°C & 60% UCL: 3724, 3579, 1825

Failure Rate @ 90°C & 60% UCL: 3722, 3722, 3377, 3665, 904

Failure Rate vs Time graph showing phases: Early (Infant Mortality) Failures, Constant (random) Failures, and Wear Out Failures.

Figure 1: The bathtub curve, failure rate plotted against time with the three life-cycle phases: infant mortality, useful life and wear-out.

Table 2. FIT rates at different operating case temperatures, following the Telcordia Parts Count Method

T _{case} (°C)	FITs	MTBF (Hours)
25	82	1.22E+07
40	158	6.34E+06
50	236	4.24E+06
60	344	2.91E+06
70	490	2.04E+06

The limitations of the FIT prediction using the Parts Count Method include... The Telcordia database, which... results are independent of true module... Nevertheless, the information... is a useful... Whenever... Avago substitutes internal data for the FIT rates... current data becomes available.

Failure rate is the inverse of MTBF. MTBF is useful for equipment that will be repaired and then returned to service, but despite the commonplace assumption, it does not guarantee a minimum time between failures, only a mean. MTTF is technically more correct mathematically, but the two terms are [except for a few situations] equivalent and MTBF is the more commonly used.

FIT rates are widely used in the industry

Failures-In-Time (FIT) Rate

Expected #failures in one billion operating hours

→ e.g., 1M UAVs flying for 1K hours each

Table 2. FIT rates at different operating case temperatures, following the Telcordia Parts Count Method

T _{case} (°C)	FITs	MTBF (Hours)
25	82	1.22E+07
40	158	6.34E+06
50	236	4.24E+06
60	344	2.91E+06
70	490	2.04E+06

HIGH TEMPERATURE GATE BIAS (HTGB)

Table 1. FIT Rate Calculations for AFCT-5717ATPZ

Component	Quantity	Component Base Rate (FITs)	Quality Factor	Total Component Failure Rate (FITs)
DFB Laser	1	20.0	0.8	16.0
Monitor PIN	1	7.7	0.8	6.2
10G PIN	1	7.7	0.8	6.2
Capacitors	27	0.2	1	5.4
Resistor	21	0.51	1	10.7
Thermistor	1	2.10	1	2.1
Ferrite Chip (Inductor)	14	2.30	1	32.2
MOSFET	1	4.00	1	4.0
EEPROM	1	6.40	1	6.4
DAC	1	6.00	1	6.0
Post-Amp IC, Gennum 16QFN	1	23.00	1	23.0
Laser Driver IC	1	6.4	1	6.4
µProcessor	1	28.0	1	28.0
Connector	20	0.130	1	2.6
Total Module Failure Rate @ 40 °C (FITs)				157.70
MTBF @ 40 °C (Hours)				6.34E+06

Table 2. FIT rates at different operating case temperatures, following the Telcordia Parts Count Method

T _{case} (°C)	FITs	MTBF (Hours)
25	82	1.22E+07
40	158	6.34E+06
50	236	4.24E+06
60	344	2.91E+06
70	490	2.04E+06

The limitations of the FIT prediction using the Parts Count Method include... (text partially obscured)

Figure 1: The bathtub curve, failure rate plotted against time with the three life-cycle phases: infant mortality, useful life and wear-out.

FIT rates are widely used in the industry

When is the CAN-based distributed real-time system the **weakest link in the system?**

This Work

For CAN-based distributed real-time systems...

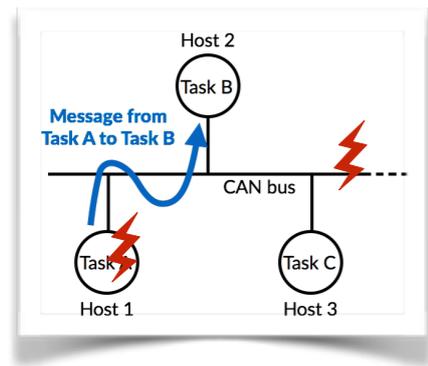
- **Probabilistic analysis**

- Quantify the replication vs. retransmissions tradeoff

- **FIT rate analysis**

- Builds upon the proposed probabilistic analysis

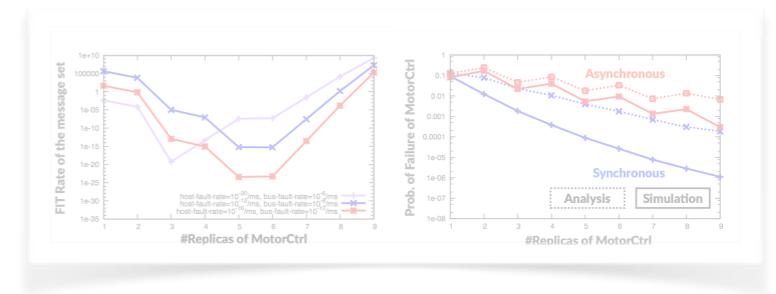
Overview



Model

$$\sum_{H' \subseteq H} \Phi_{crash}^{H'} \cdot \sum_{M'_1 \subseteq M_1} \left(\Phi_{timely}^{H', M'_1} \cdot \Phi_{correct}^{H', M'_1} \right)$$

Analysis



Evaluation

Fault Abstraction & Modeling

Transmission failures
(faults on the wire)

Commission failures
(bit-flips in the memory buffers)

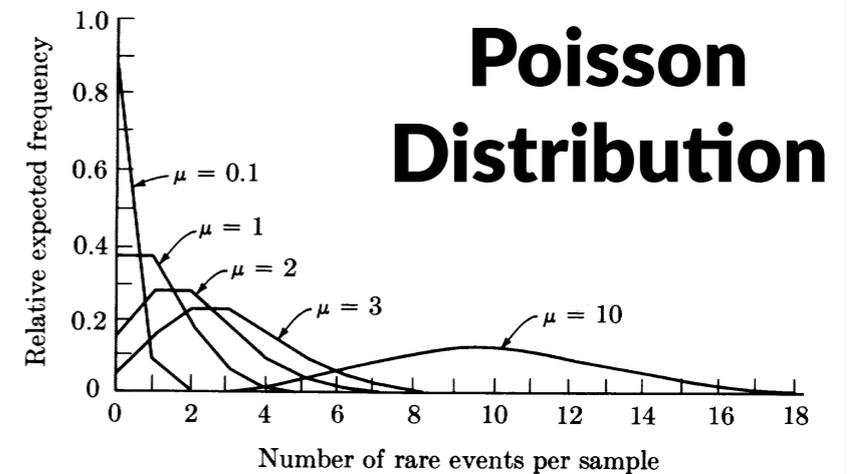
Crash failures
(due to fault-induced exceptions)

Fault Abstraction & Modeling

Transmission failures
(faults on the wire)

Commission failures
(bit-flips in the memory buffers)

Crash failures
(due to fault-induced exceptions)

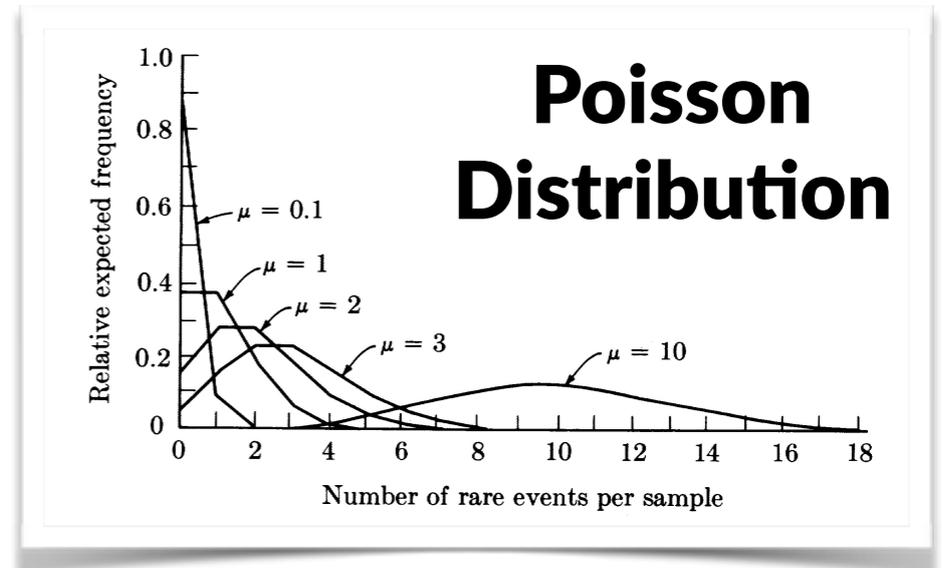


Fault Abstraction & Modeling

Transmission failures
(faults on the wire)

Commission failures
(bit-flips in the memory buffers)

Crash failures
(due to fault-induced exceptions)



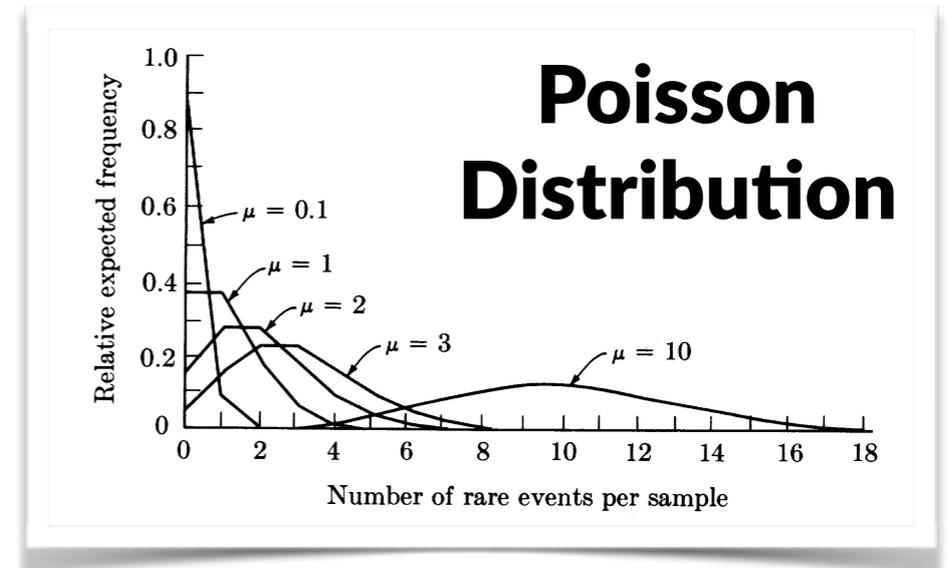
Probability that each message is omitted / corrupted / retransmitted

Fault Abstraction & Modeling

Transmission failures
(faults on the wire)

Commission failures
(bit-flips in the memory buffers)

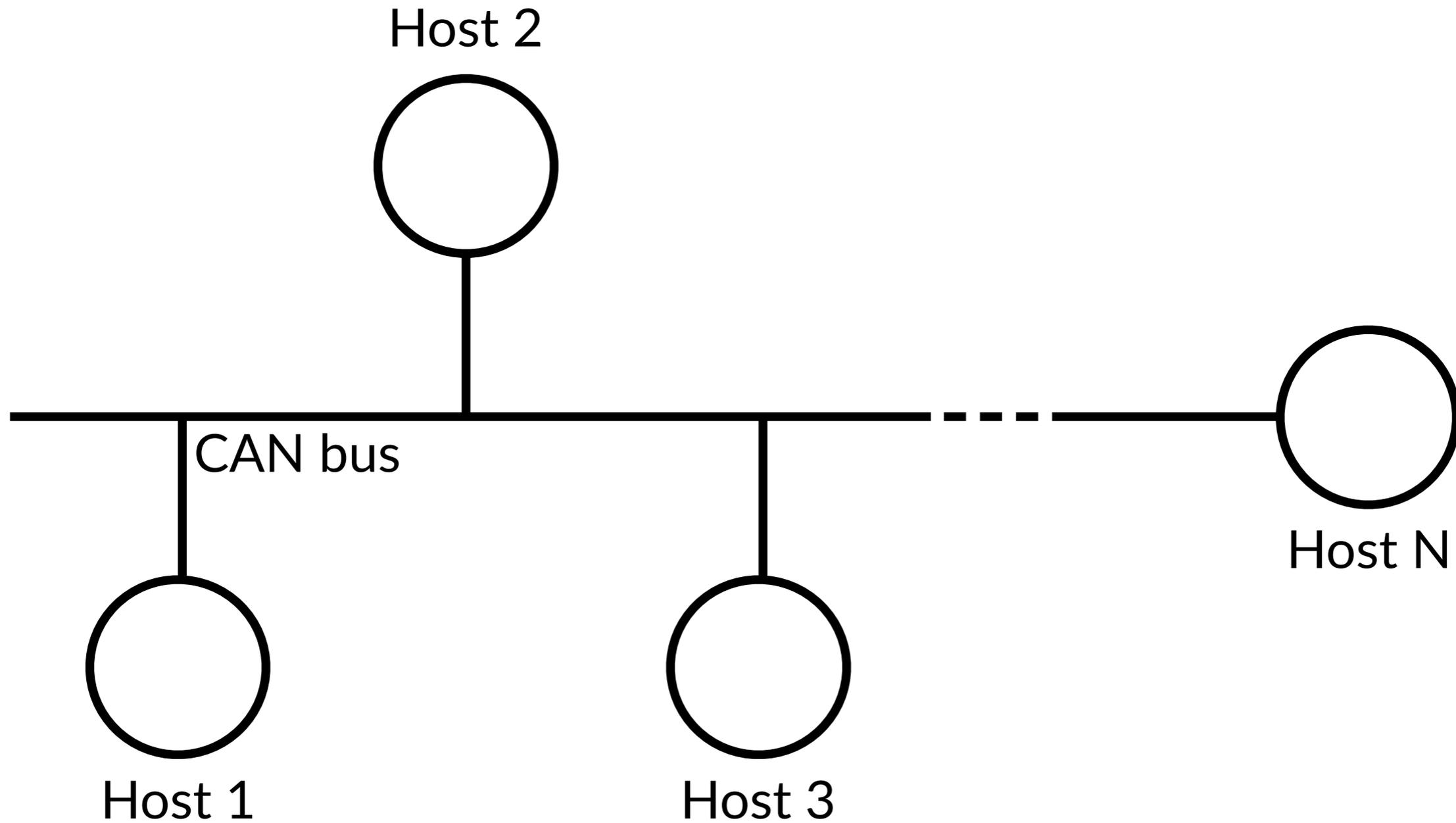
Crash failures
(due to fault-induced exceptions)



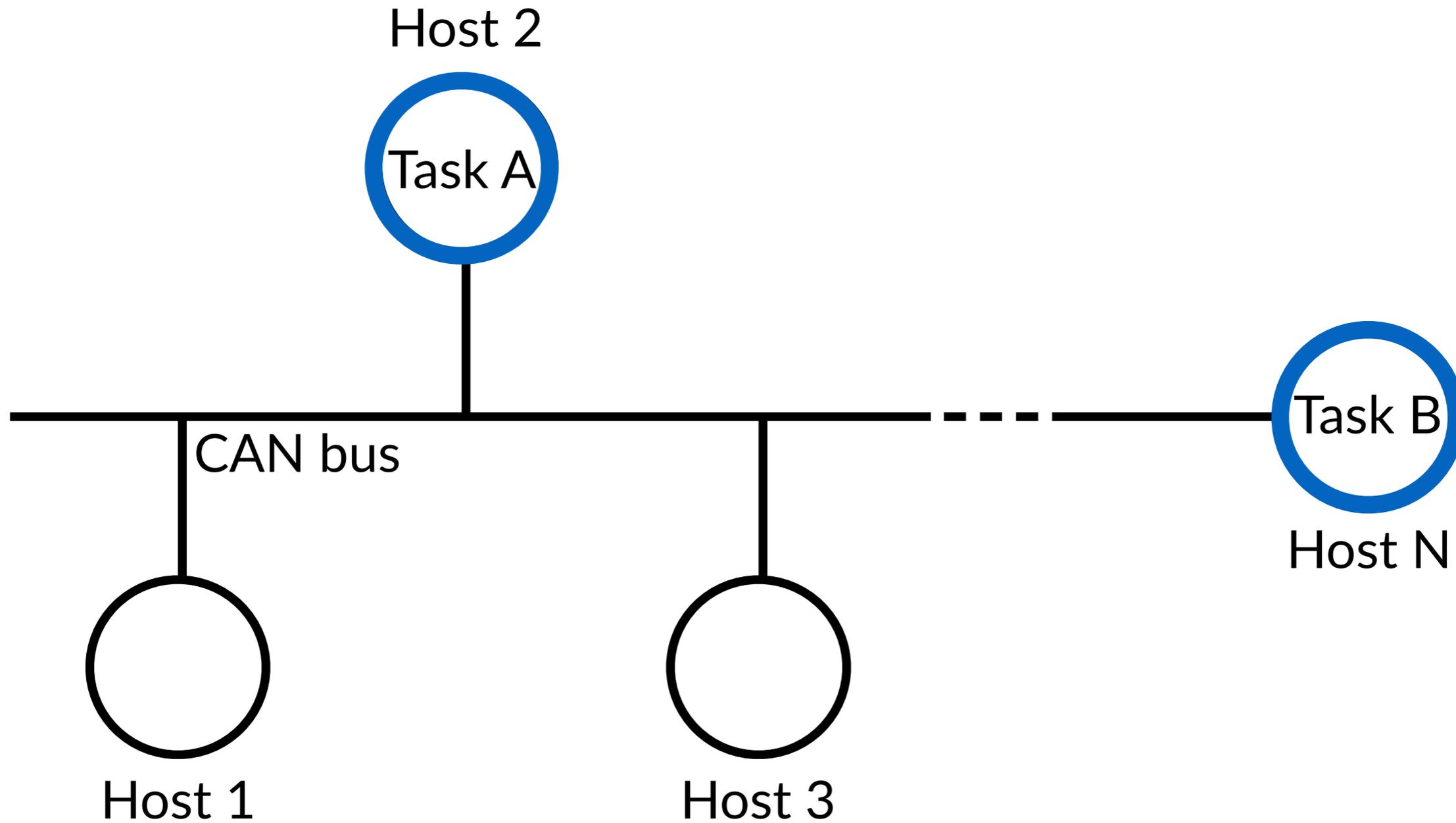
Probability that each message is omitted / corrupted / retransmitted

We do not consider **software defects...**

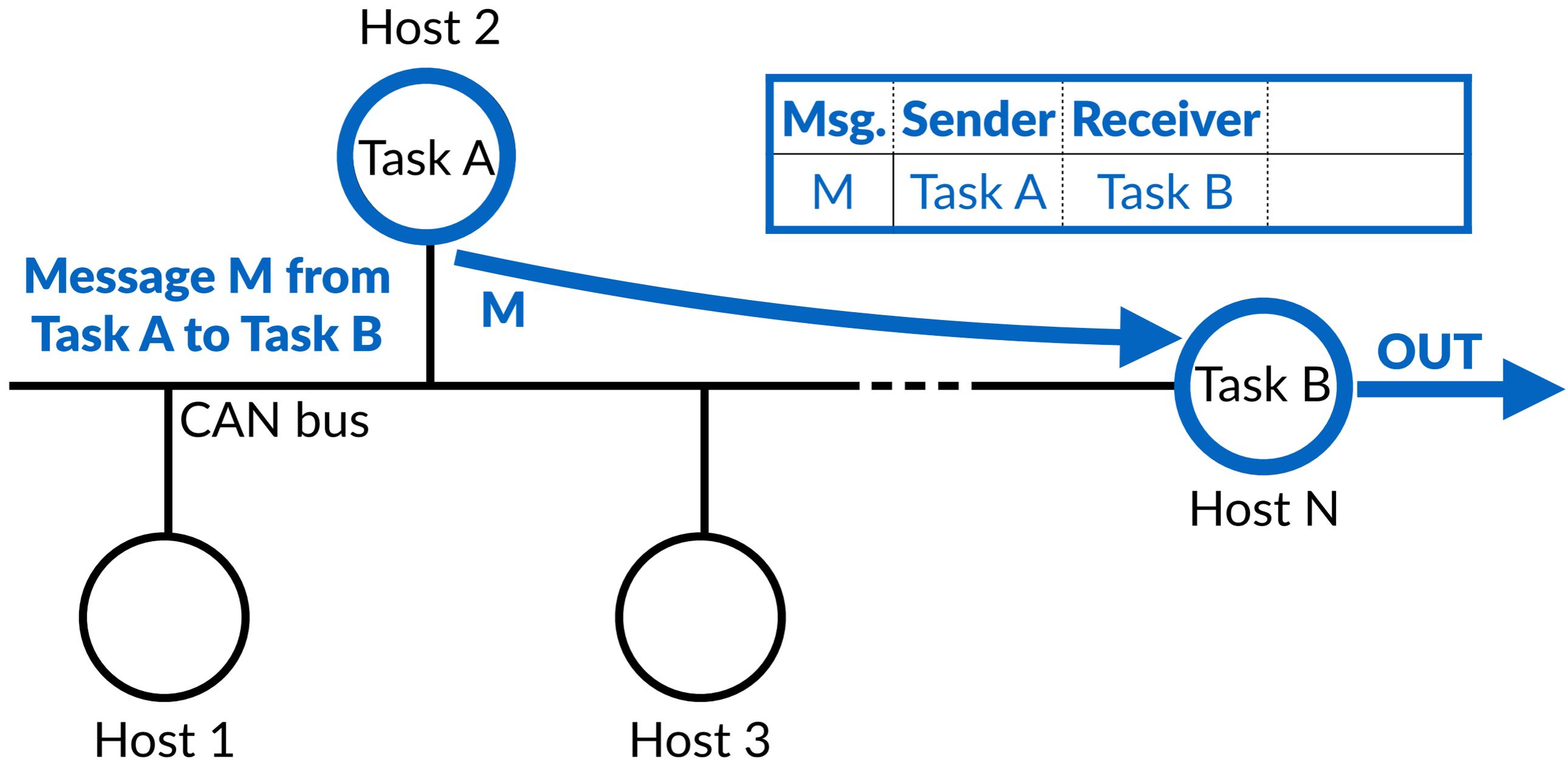
System Model



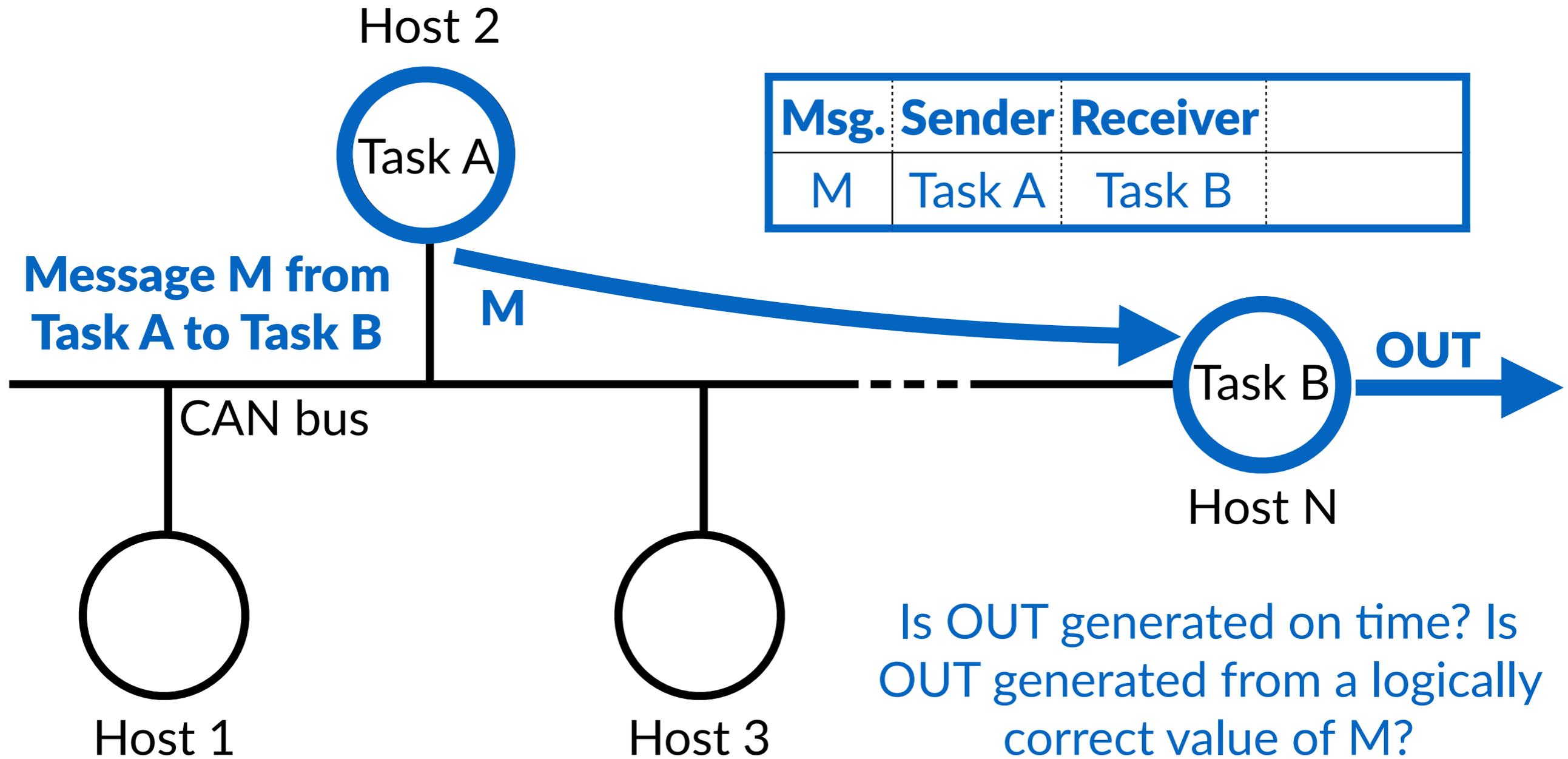
System Model



System Model

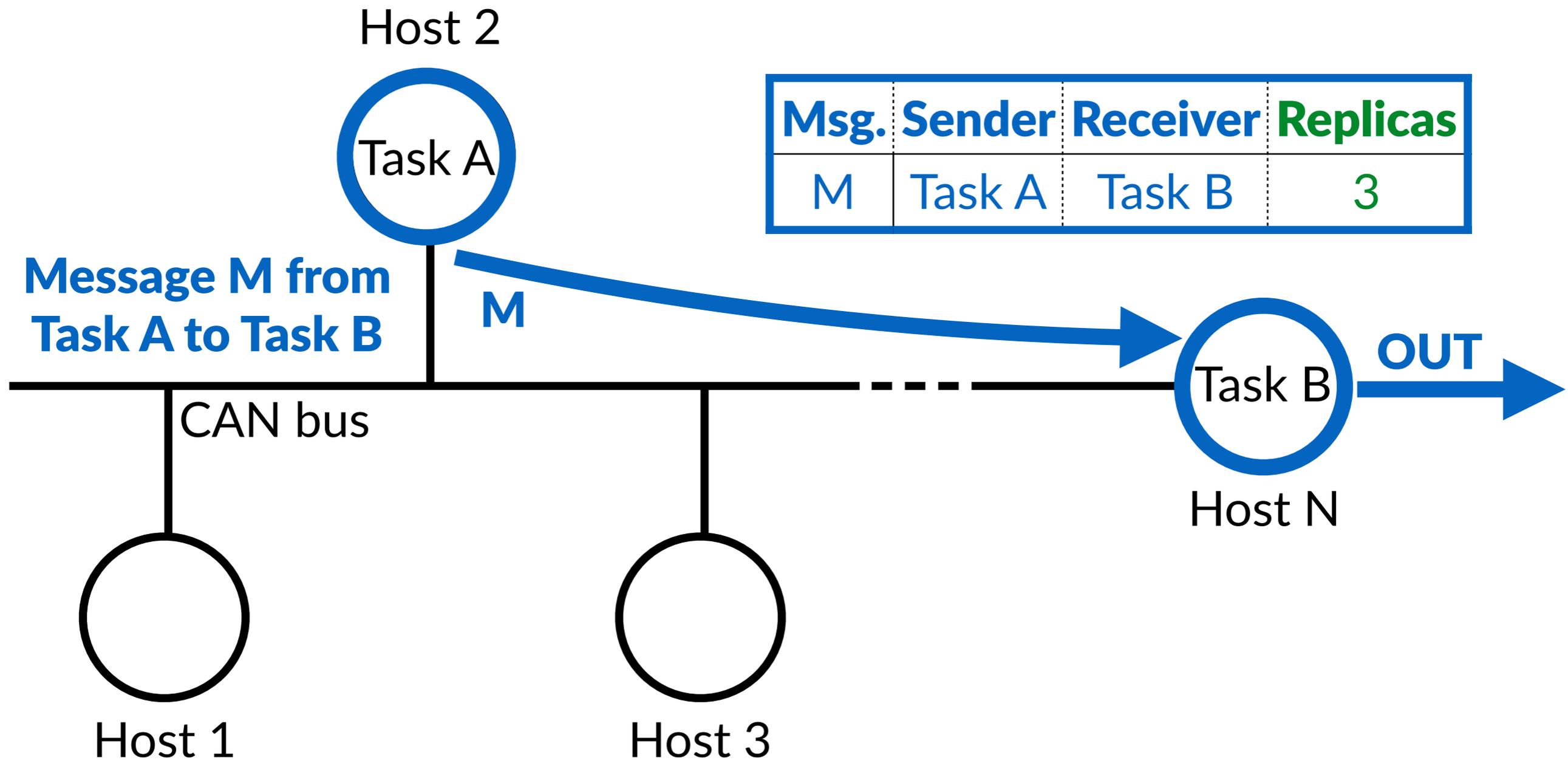


System Model



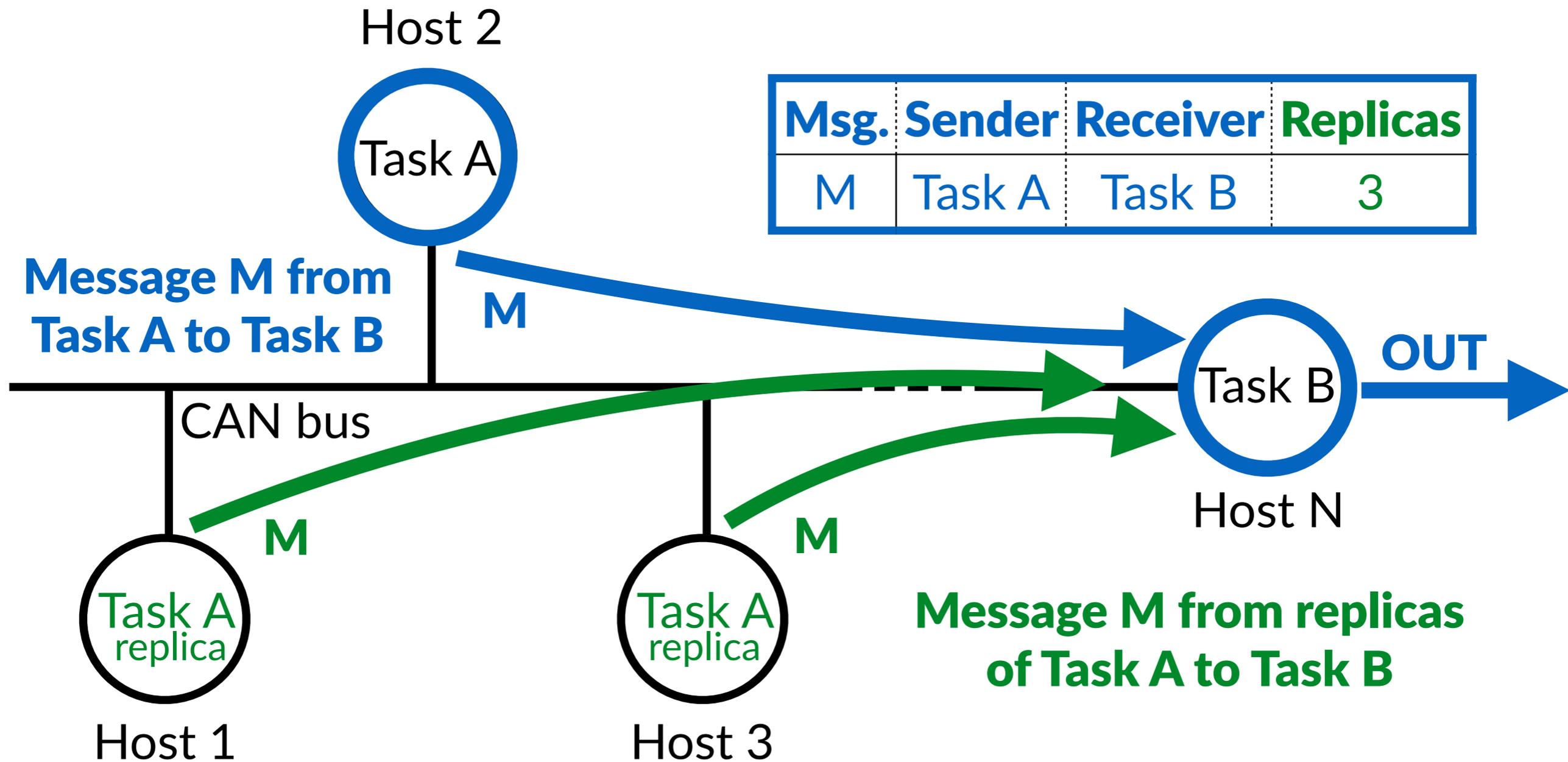
System Model

with Task Replication

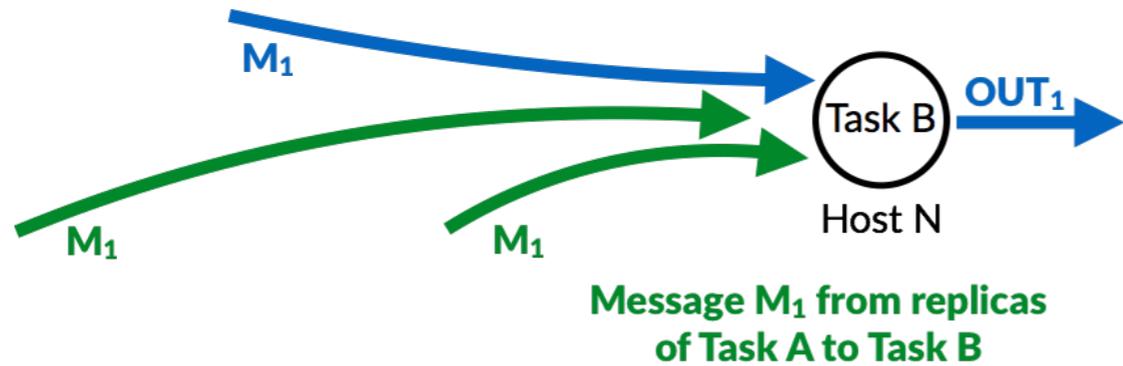


System Model

with Task Replication

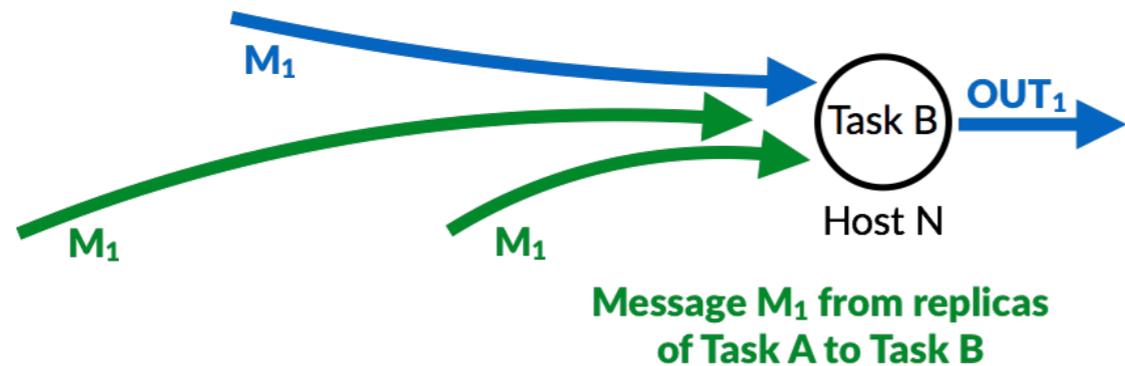


Aggregating the replicated messages



How & when to compute OUT
from multiple copies of M ?

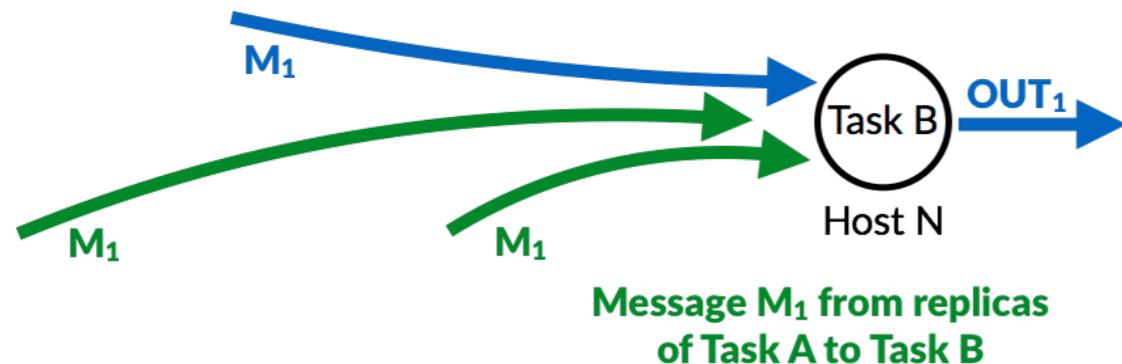
Aggregating the replicated messages



How & when to compute OUT
from multiple copies of M?

- Case 1: Synchronous Systems
 - **Common** global time base
 - e.g. majority value **at the absolute deadline**

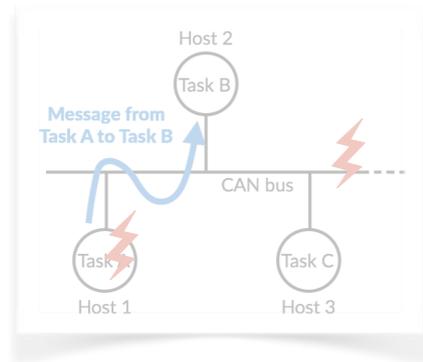
Aggregating the replicated messages



How & when to compute OUT
from multiple copies of M?

- Case 1: Synchronous Systems
 - **Common** global time base
 - e.g. majority value **at the absolute deadline**
- Case 2: Asynchronous Systems
 - **No** global time base
 - e.g. majority value **after “enough” copies have been received**

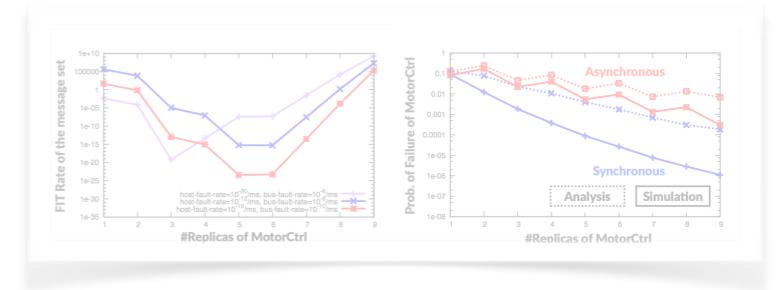
Overview



Model

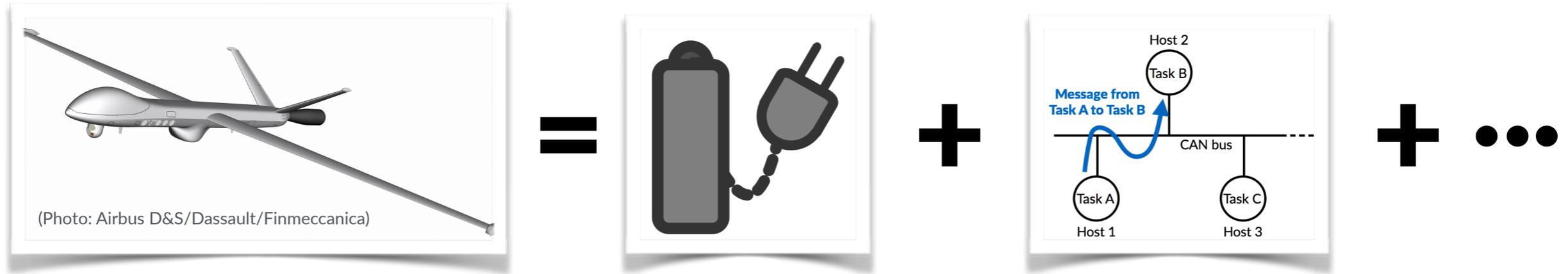
$$\sum_{\mathbb{H}' \subseteq \mathbb{H}} \Phi_{crash}^{\mathbb{H}'} \cdot \sum_{\mathbb{M}'_1 \subseteq \mathbb{M}_1} \left(\Phi_{timely}^{\mathbb{H}', \mathbb{M}'_1} \cdot \Phi_{correct}^{\mathbb{H}', \mathbb{M}'_1} \right)$$

Analysis

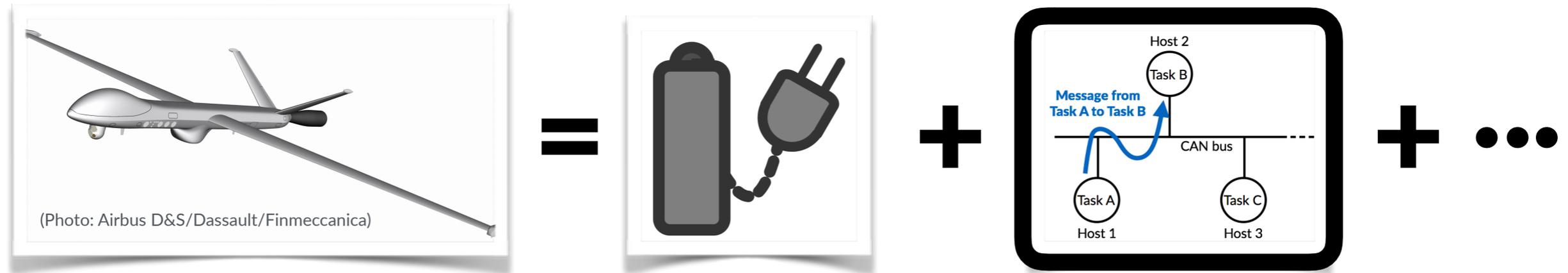


Evaluation

The Larger Picture...



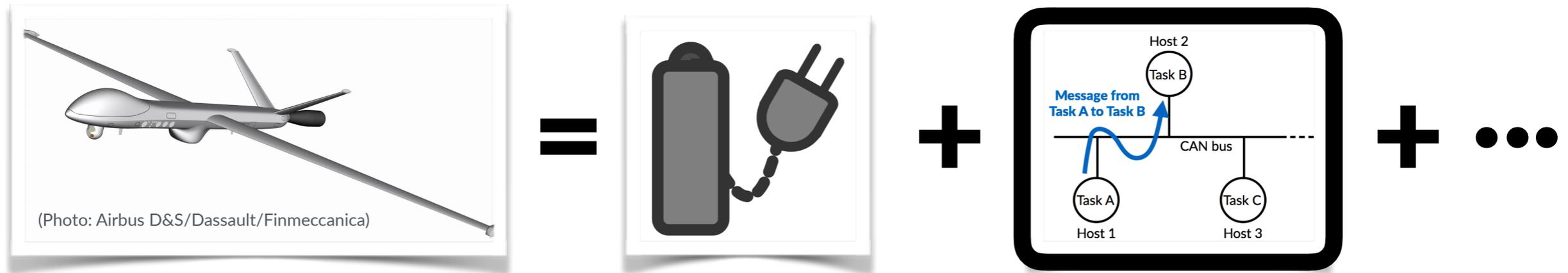
The Larger Picture...



Objectives:

- ➔ **A good replication strategy** for the CAN-based system
- ➔ **Compare** the reliability of the CAN-based system **with other components in the safety-critical system**

The Larger Picture...



Objectives:

- ➔ **A good replication strategy** for the CAN-based system
- ➔ **Compare** the reliability of the CAN-based system **with other components in the safety-critical system**

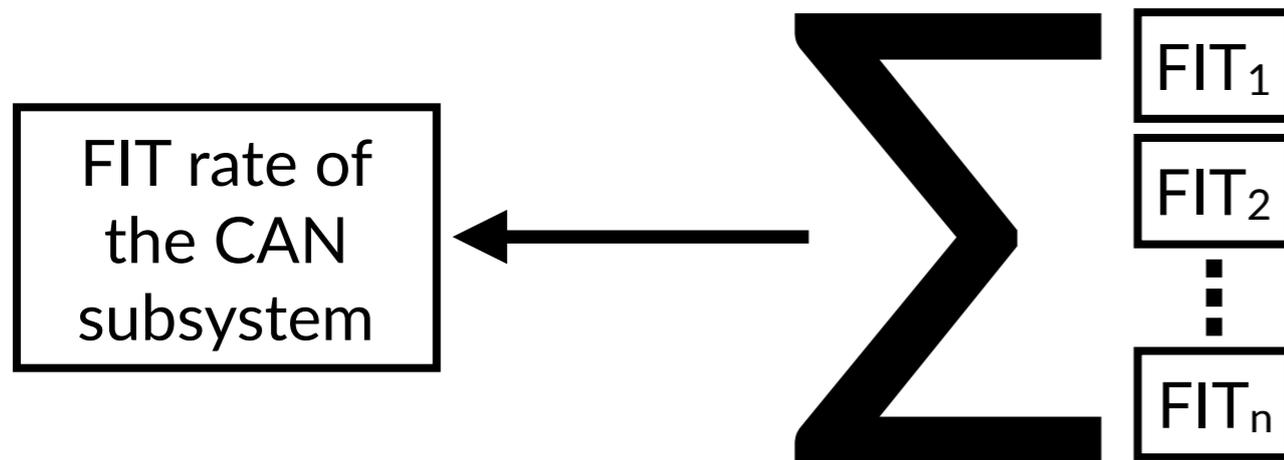
Solution: **FIT rate analysis**

- ➔ Using the **probabilistic analysis**

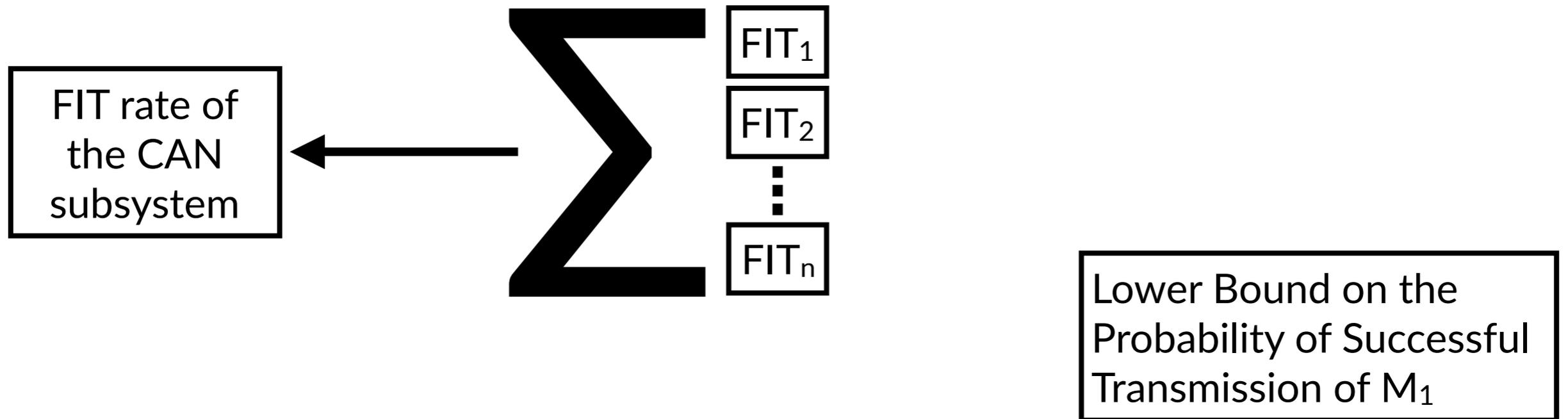
FIT Rate Analysis of the System

FIT rate of
the CAN
subsystem

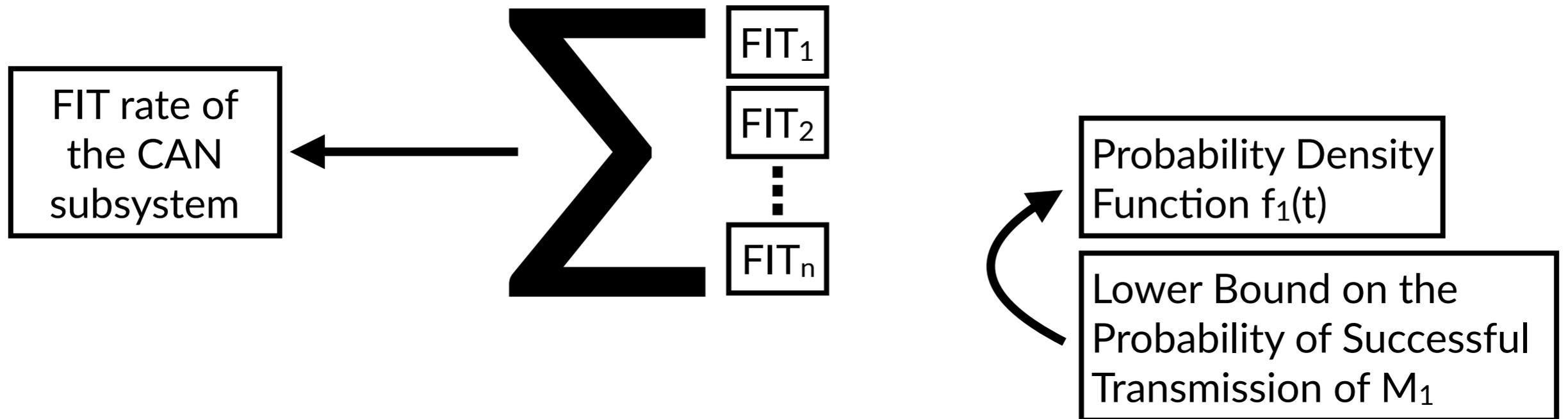
FIT Rate Analysis of the System



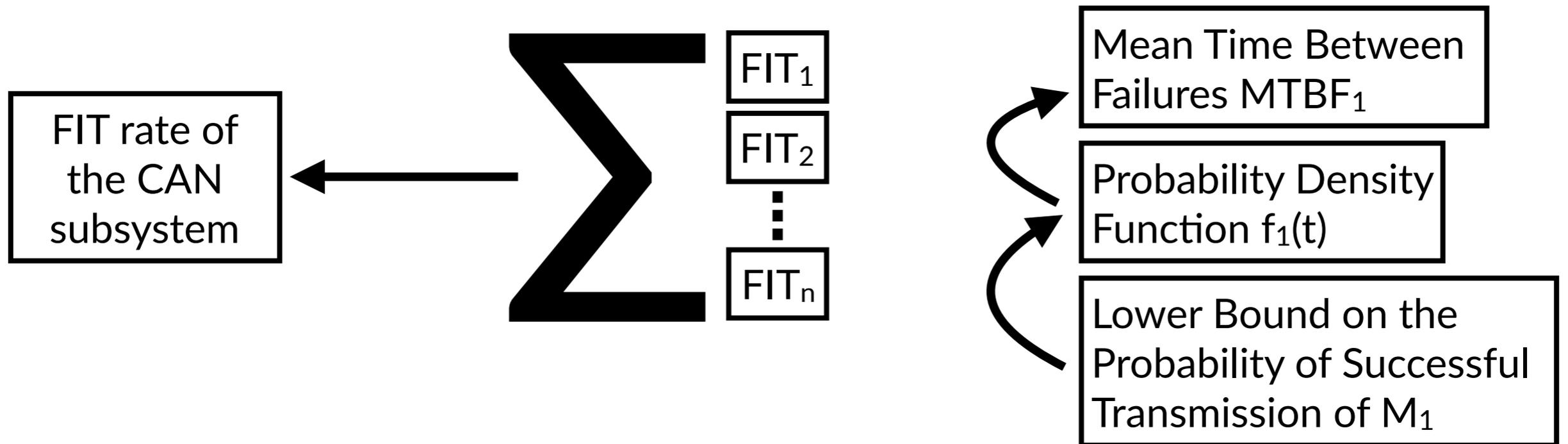
FIT Rate Analysis of the System



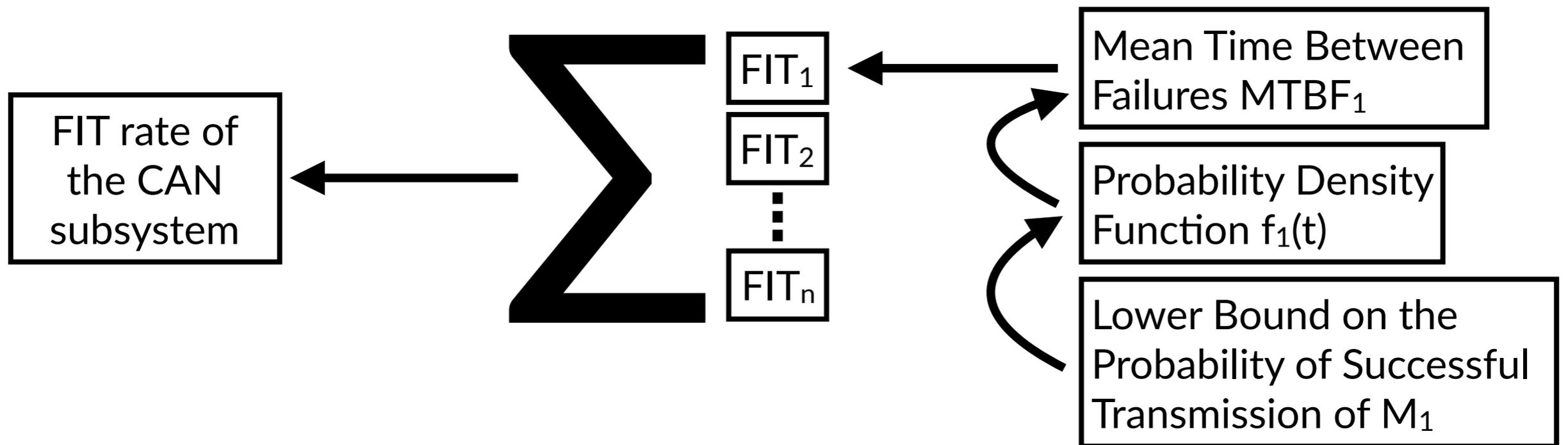
FIT Rate Analysis of the System



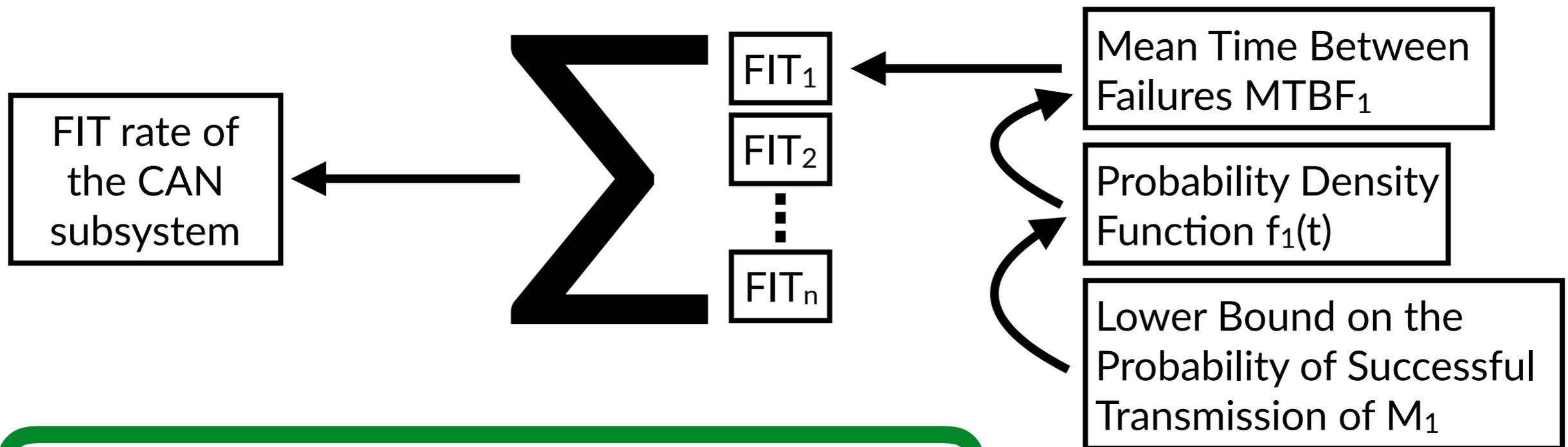
FIT Rate Analysis of the System



FIT Rate Analysis of the System

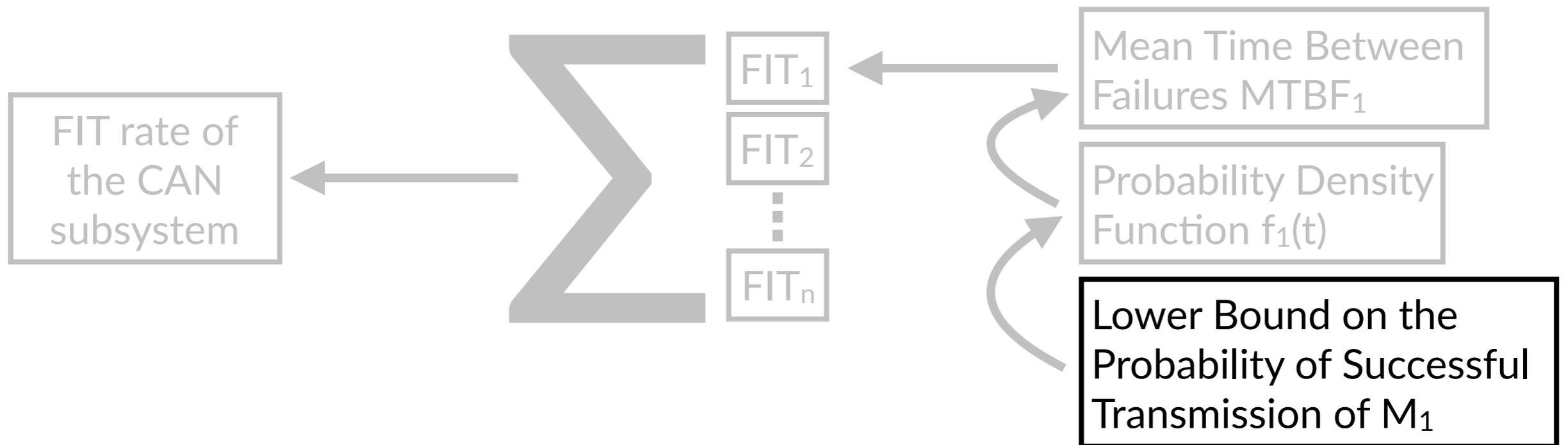


FIT Rate Analysis of the System

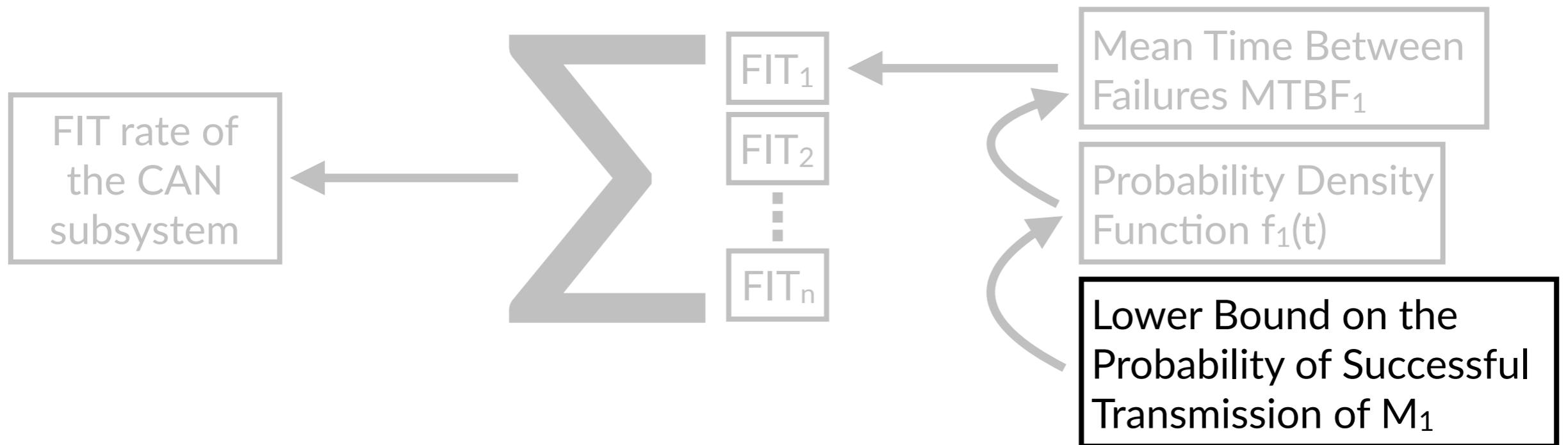


Standard procedure to compute FIT rates given the failure probabilities, but **tailored for real-time workloads**

FIT Rate Analysis of the System



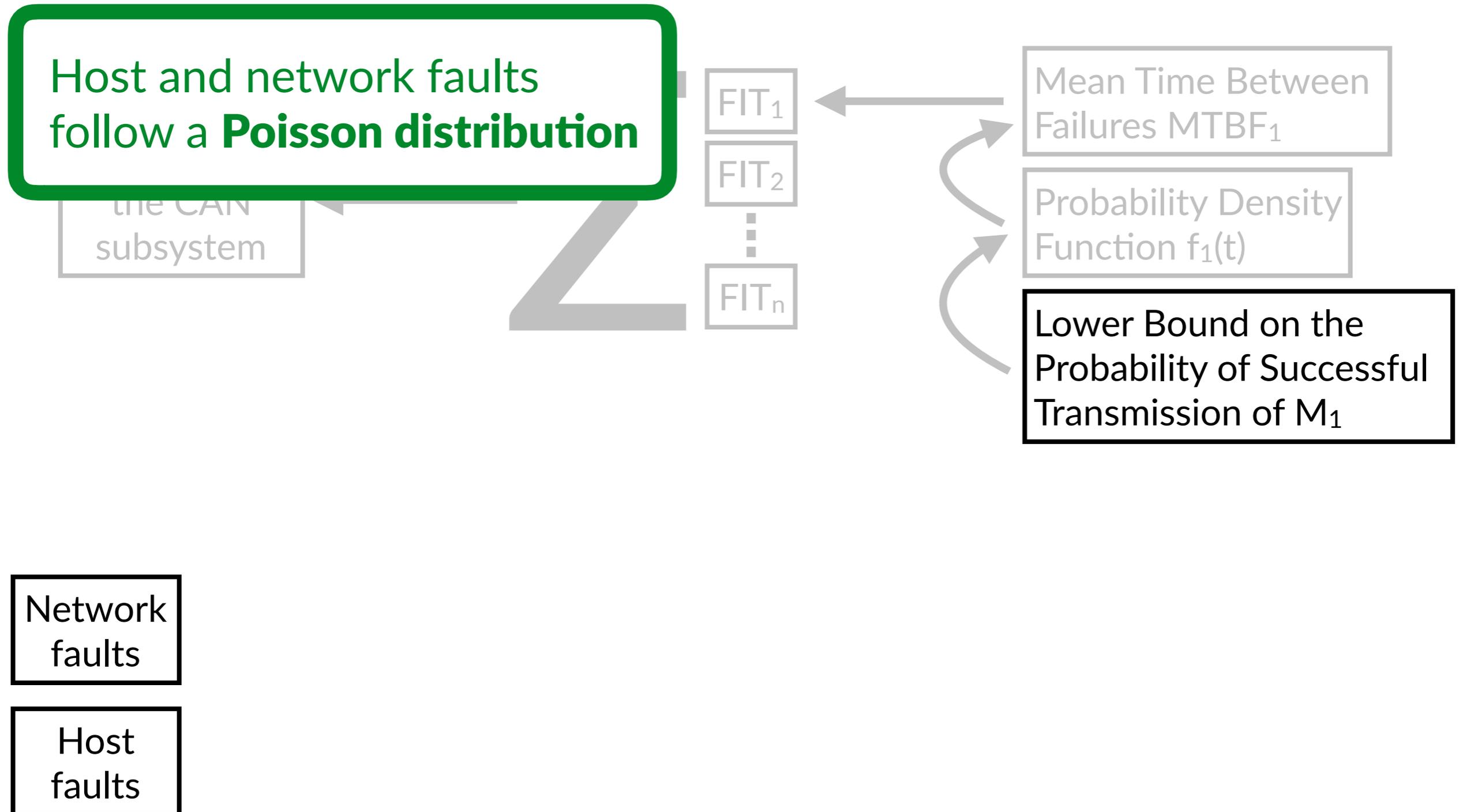
FIT Rate Analysis of the System



Network faults

Host faults

FIT Rate Analysis of the System



FIT Rate Analysis of the System

Host and network faults follow a **Poisson distribution**

FIT₁
FIT₂

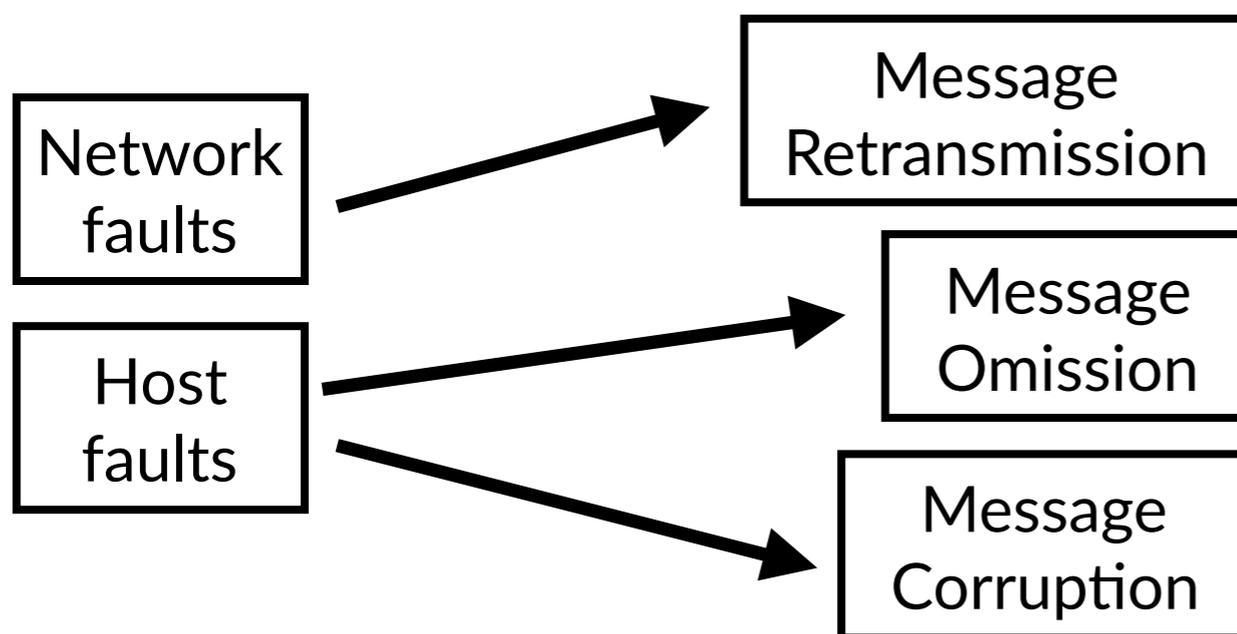
Probabilities that each message:

- retransmitted due to **transmission failures**
- omitted due to **crash failures**
- corrupted due to **commission failures**

Mean Time Between Failures MTBF₁

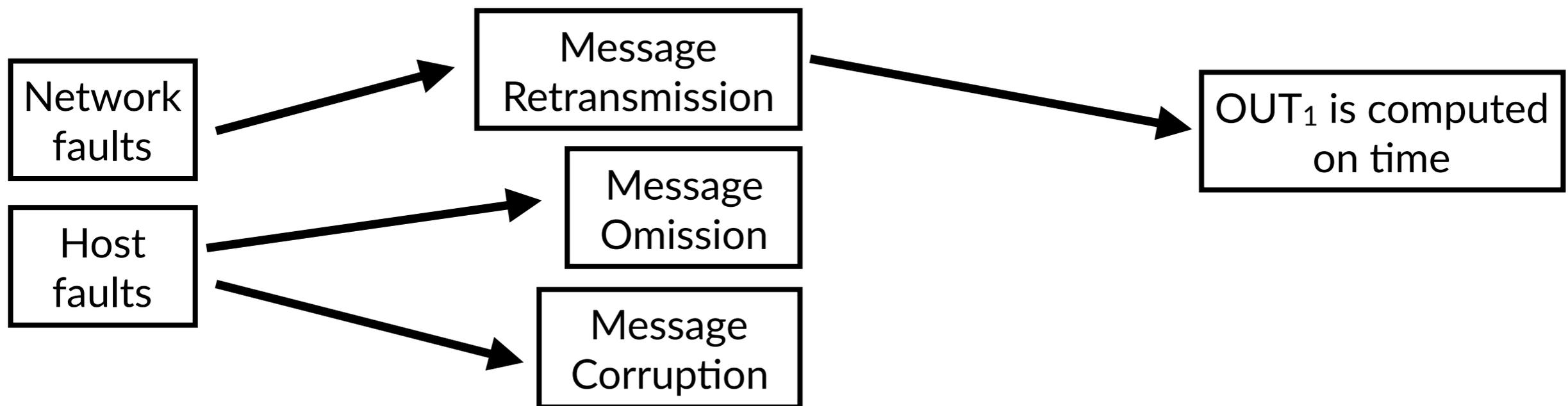
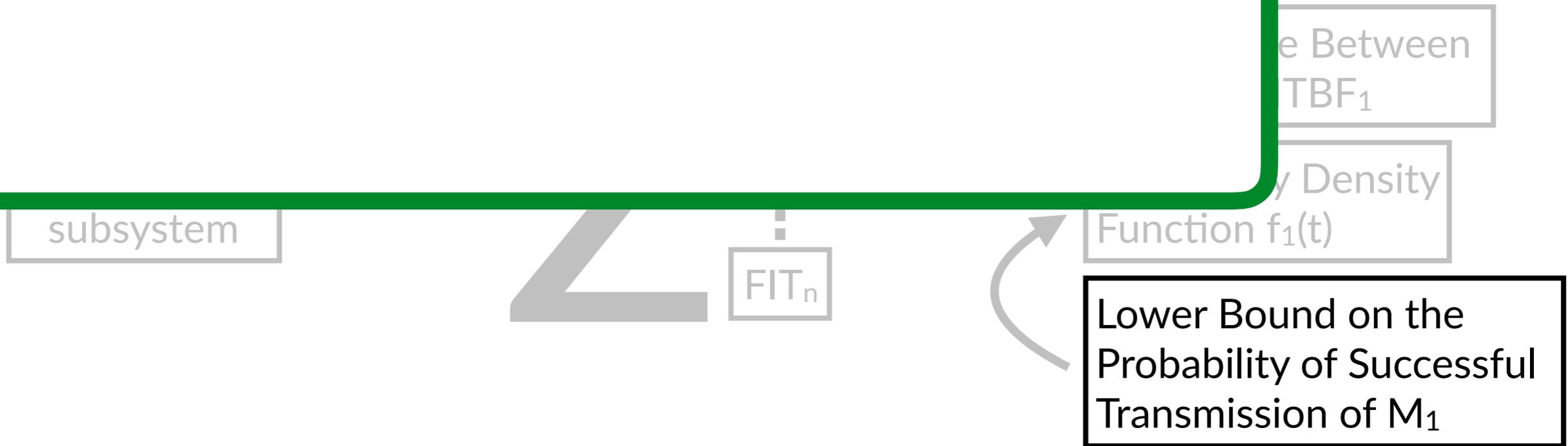
Probability Density Function $f_1(t)$

Lower Bound on the Probability of Successful Transmission of M_1



- Broster et al.'s **probabilistic response-time analysis***

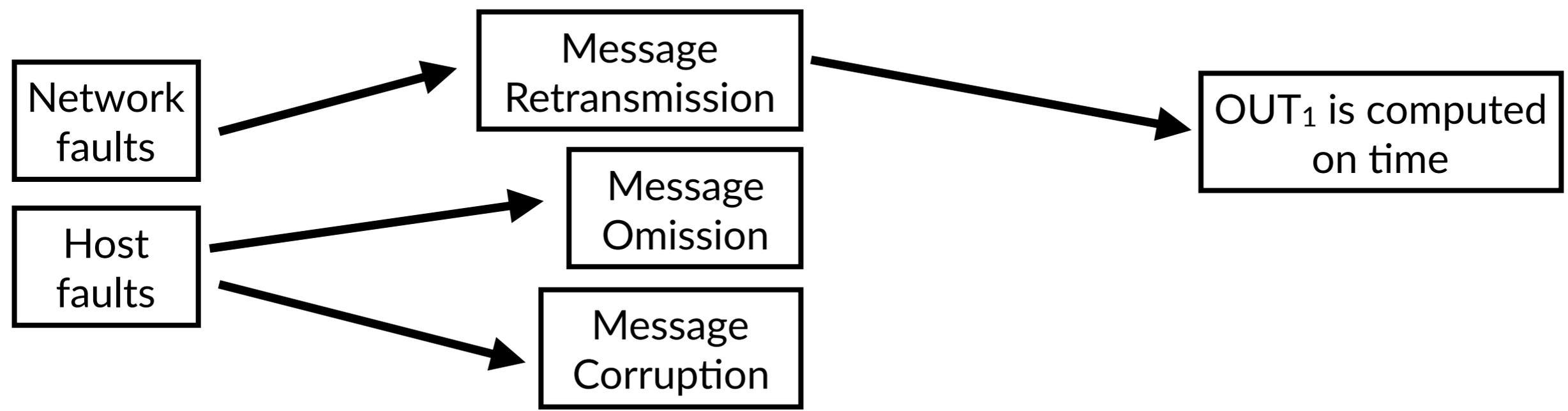
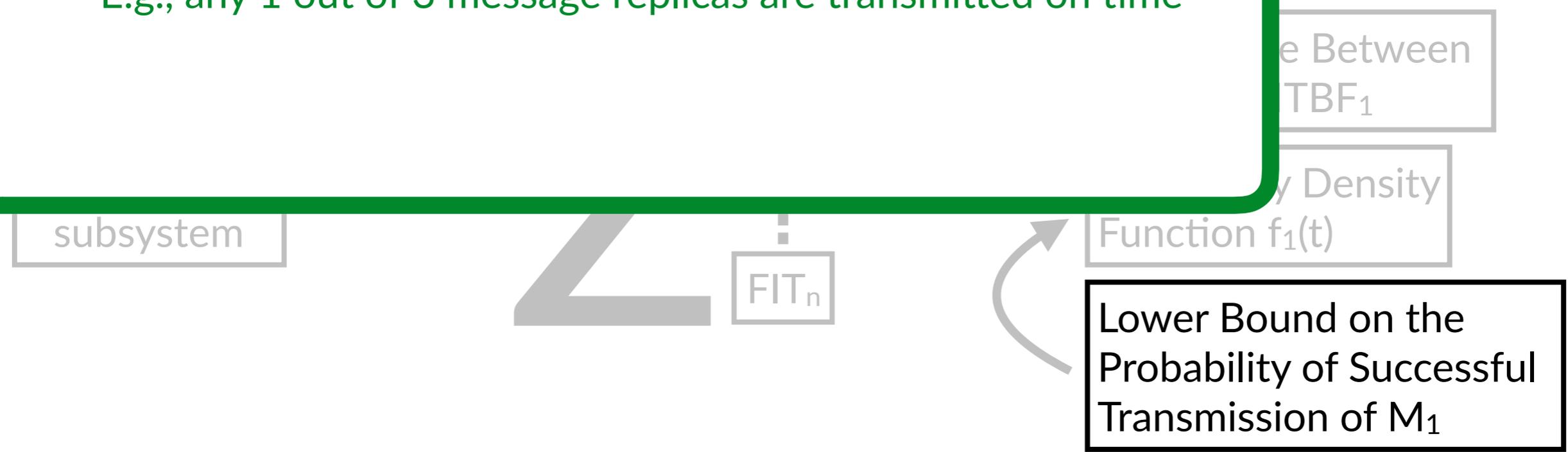
System



*Broster, Ian, Alan Burns, and Guillermo Rodriguez-Navas. "Timing analysis of real-time communication under electromagnetic interference." Real-Time Systems 30.1-2 (2005): 55-81.

System

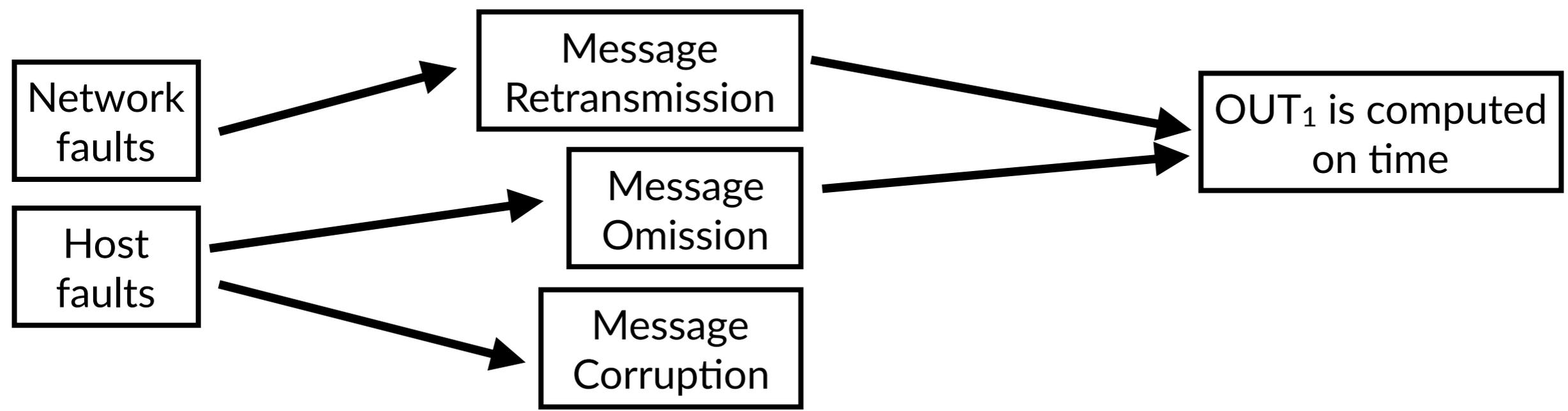
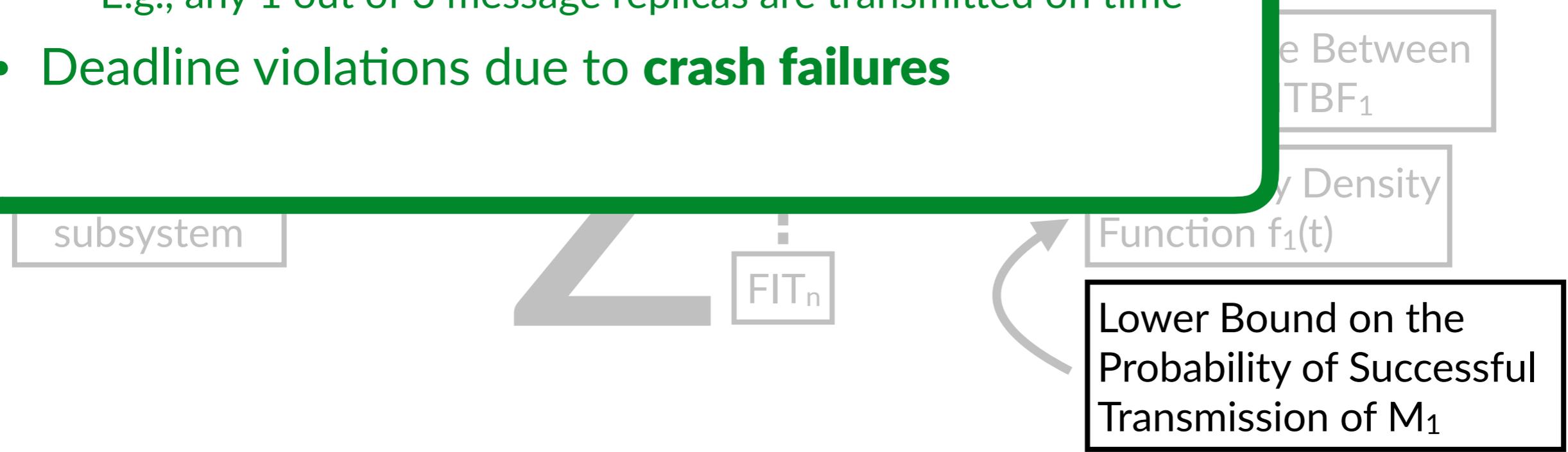
- Broster et al.'s **probabilistic response-time analysis***
- We extend the analysis for a **set of message replicas**
 - E.g., any 1 out of 3 message replicas are transmitted on time



*Broster, Ian, Alan Burns, and Guillermo Rodriguez-Navas. "Timing analysis of real-time communication under electromagnetic interference." Real-Time Systems 30.1-2 (2005): 55-81.

System

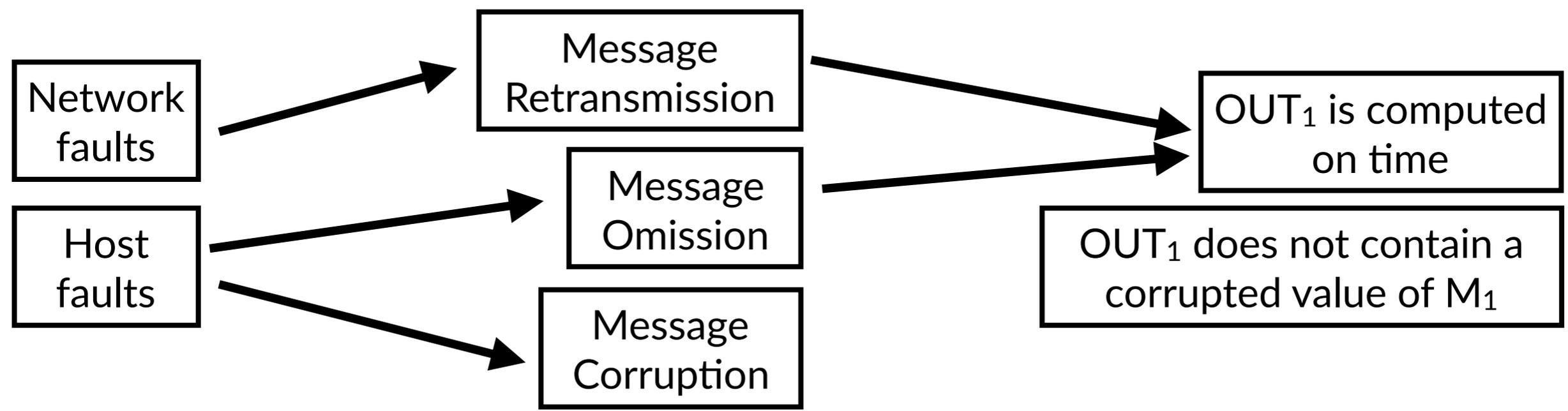
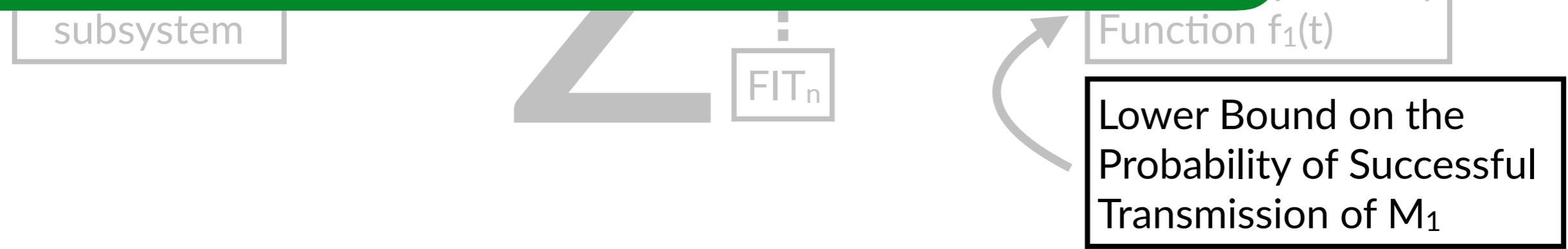
- Broster et al.'s **probabilistic response-time analysis***
- We extend the analysis for a **set of message replicas**
 - E.g., any 1 out of 3 message replicas are transmitted on time
- Deadline violations due to **crash failures**



*Broster, Ian, Alan Burns, and Guillermo Rodriguez-Navas. "Timing analysis of real-time communication under electromagnetic interference." Real-Time Systems 30.1-2 (2005): 55-81.

System

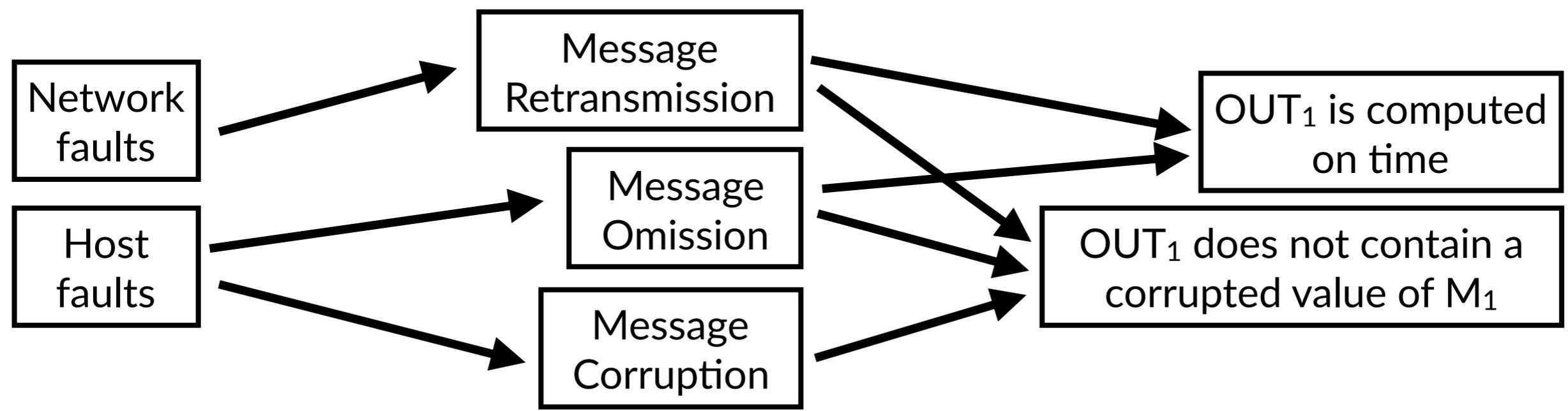
- Broster et al.'s **probabilistic response-time analysis***
- We extend the analysis for a **set of message replicas**
 - E.g., any 1 out of 3 message replicas are transmitted on time
- Deadline violations due to **crash failures**
- Incorrect output due to **commission failures**



*Broster, Ian, Alan Burns, and Guillermo Rodriguez-Navas. "Timing analysis of real-time communication under electromagnetic interference." Real-Time Systems 30.1-2 (2005): 55-81.

System

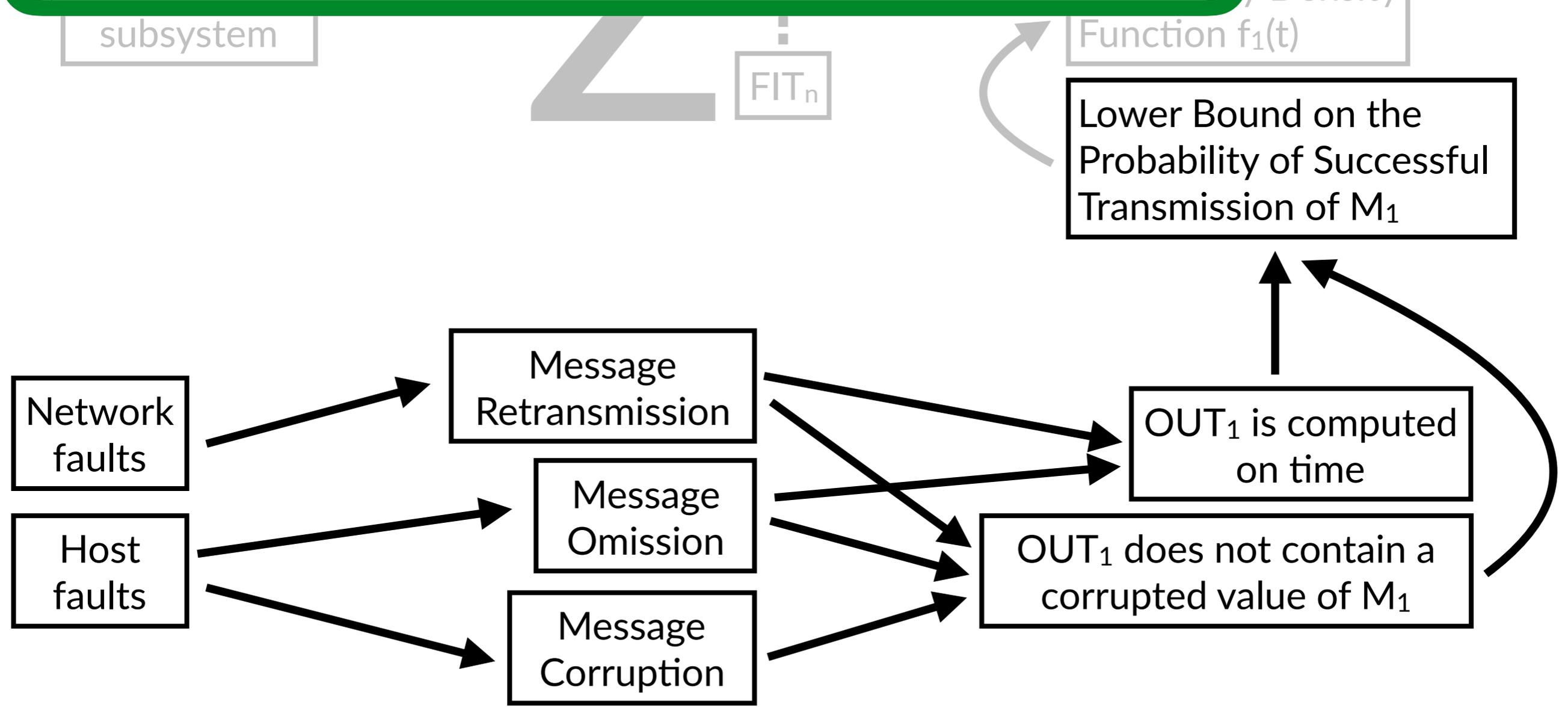
- Broster et al.'s **probabilistic response-time analysis***
- We extend the analysis for a **set of message replicas**
 - E.g., any 1 out of 3 message replicas are transmitted on time
- Deadline violations due to **crash failures**
- Incorrect output due to **commission failures**



*Broster, Ian, Alan Burns, and Guillermo Rodriguez-Navas. "Timing analysis of real-time communication under electromagnetic interference." Real-Time Systems 30.1-2 (2005): 55-81.

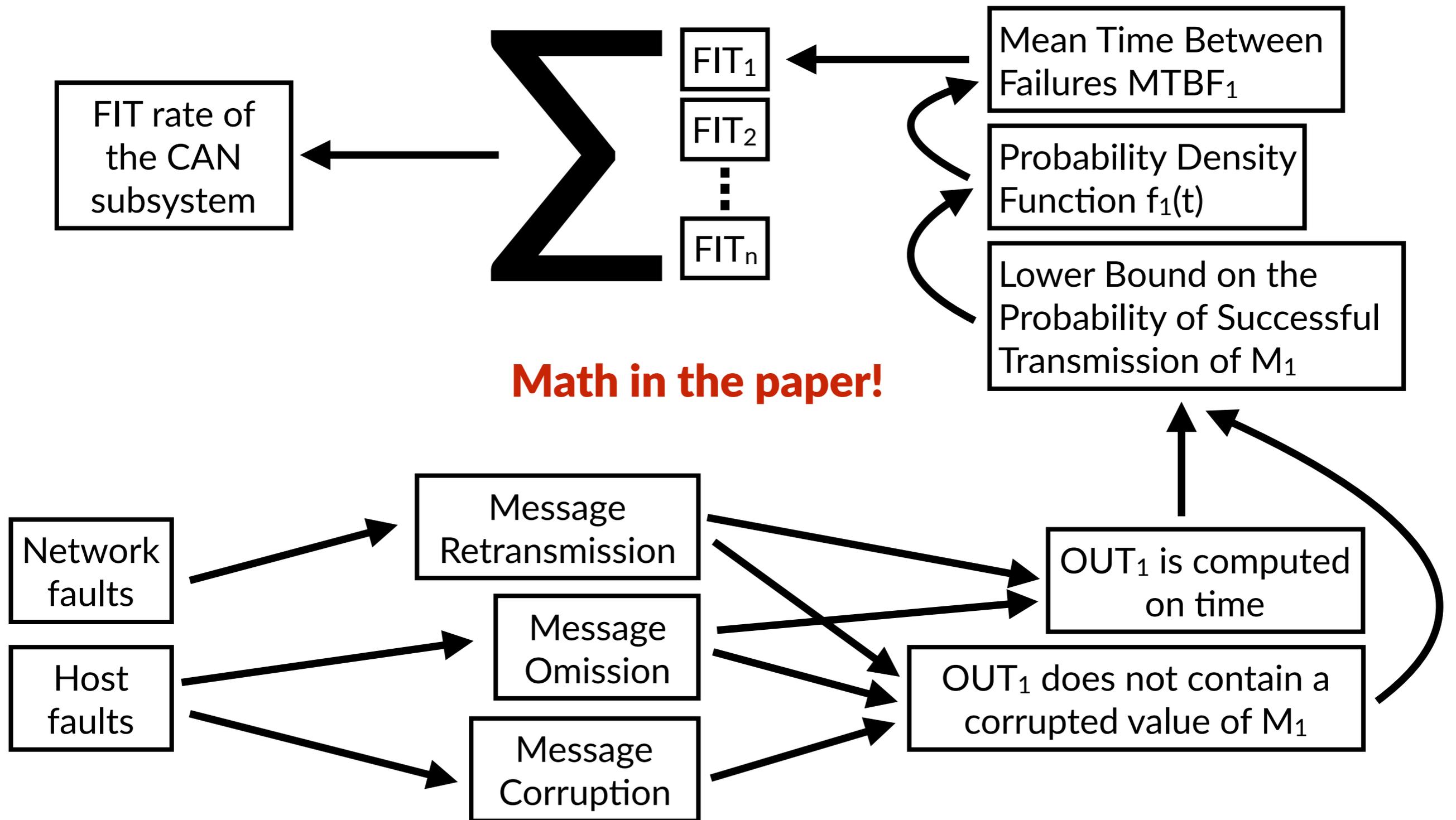
System

- Broster et al.'s **probabilistic response-time analysis***
- We extend the analysis for a **set of message replicas**
 - E.g., any 1 out of 3 message replicas are transmitted on time
- Deadline violations due to **crash failures**
- Incorrect output due to **commission failures**

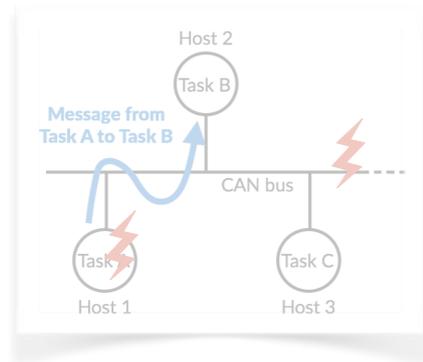


*Broster, Ian, Alan Burns, and Guillermo Rodriguez-Navas. "Timing analysis of real-time communication under electromagnetic interference." Real-Time Systems 30.1-2 (2005): 55-81.

FIT Rate Analysis of the System



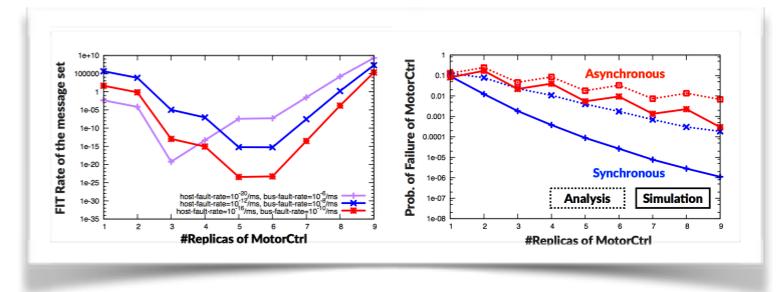
Overview



Model

$$\sum_{H' \subseteq H} \Phi_{crash}^{H'} \cdot \sum_{M'_1 \subseteq M_1} \left(\Phi_{timely}^{H', M'_1} \cdot \Phi_{correct}^{H', M'_1} \right)$$

Analysis



Evaluation

Mobile Robot Workload*

Task Name	Length (bytes)	Period (ms)	Deadline (ms)
MotorCtrl	2	2	2
Wheel1	3	4	4
Wheel2	3	4	4
RadiIn	8	8	8
Proximity	1	12	12
Logging	8	240	240

Mobile Robot Workload*

Only the **MotorCtrl** task is replicated
(#replicas vary from 1 to 9)

Task Name	Length (bytes)	Period (ms)	Deadline (ms)
MotorCtrl	2	2	2
Wheel1	3	4	4
Wheel2	3	4	4
RadiIn	8	8	8
Proximity	1	12	12
Logging	8	240	240

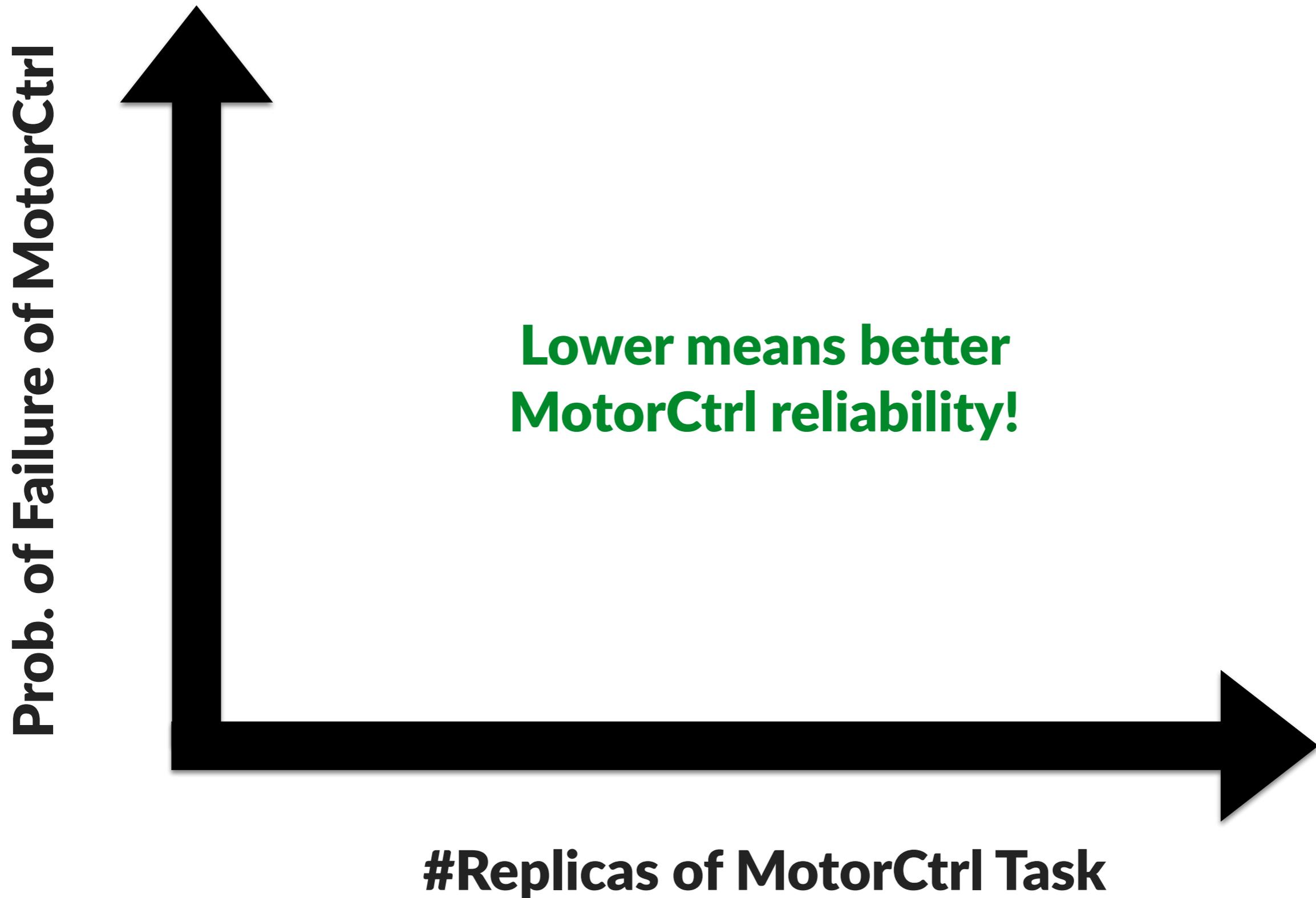
Evaluation

- Assess the proposed FIT rate derivation
 - ➔ Comparison with results from CAN bus simulation

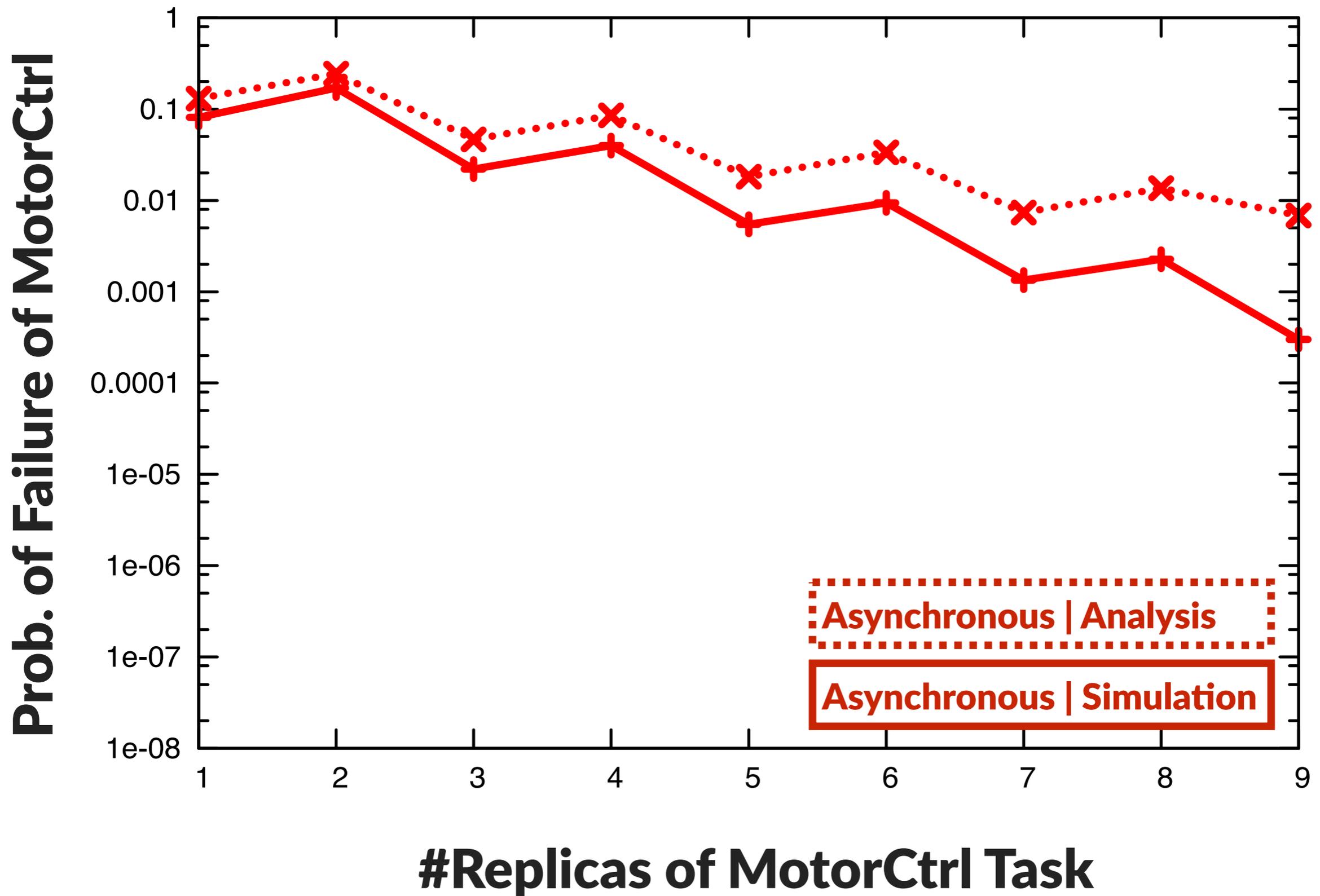
Evaluation

- Assess the proposed FIT rate derivation
 - ➔ Comparison with results from CAN bus simulation
- Is the FIT rate analysis too coarse-grained?
 - ➔ Analysis for various fault rates

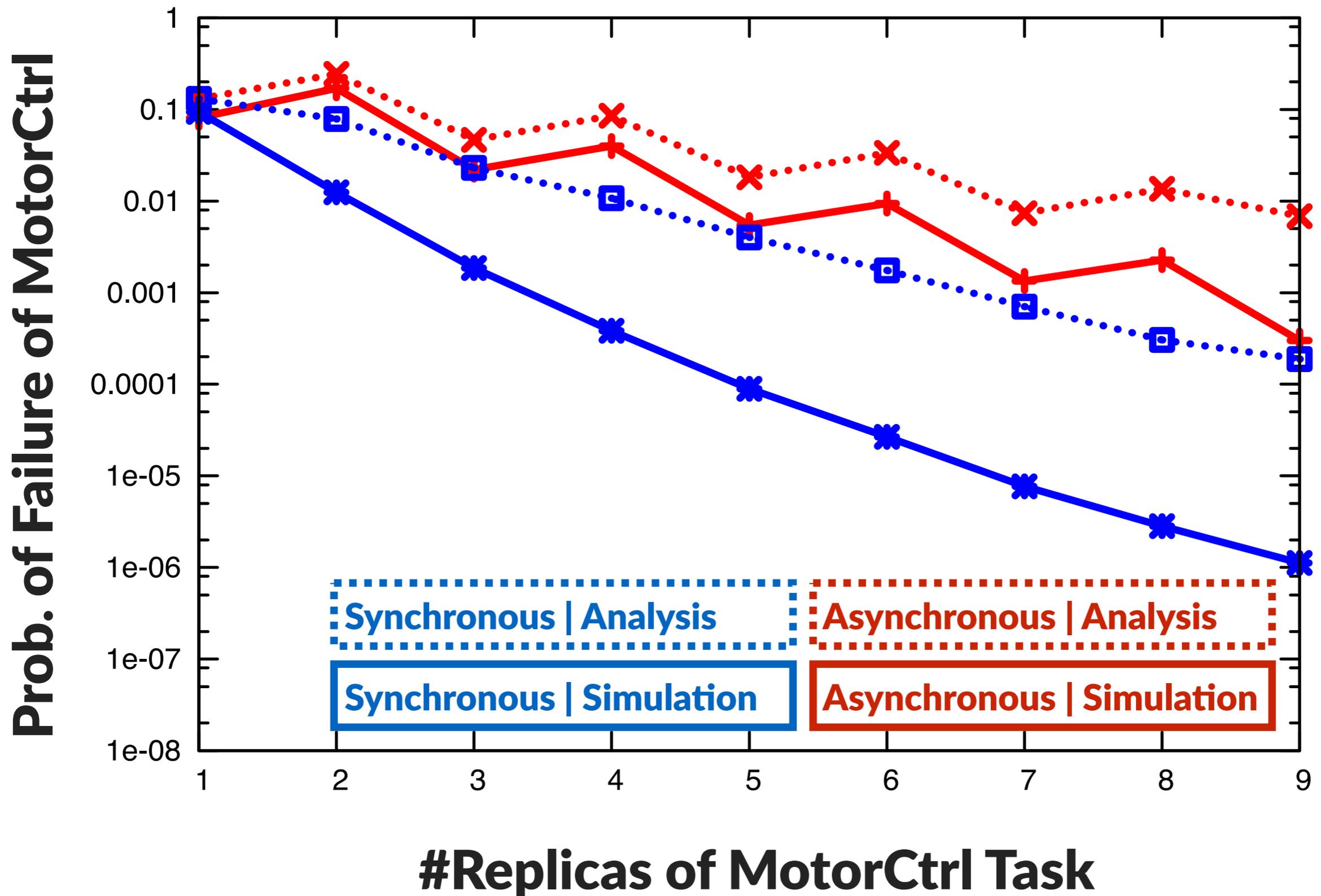
Analysis versus Simulation for MotorCtrl



Analysis versus Simulation for MotorCtrl



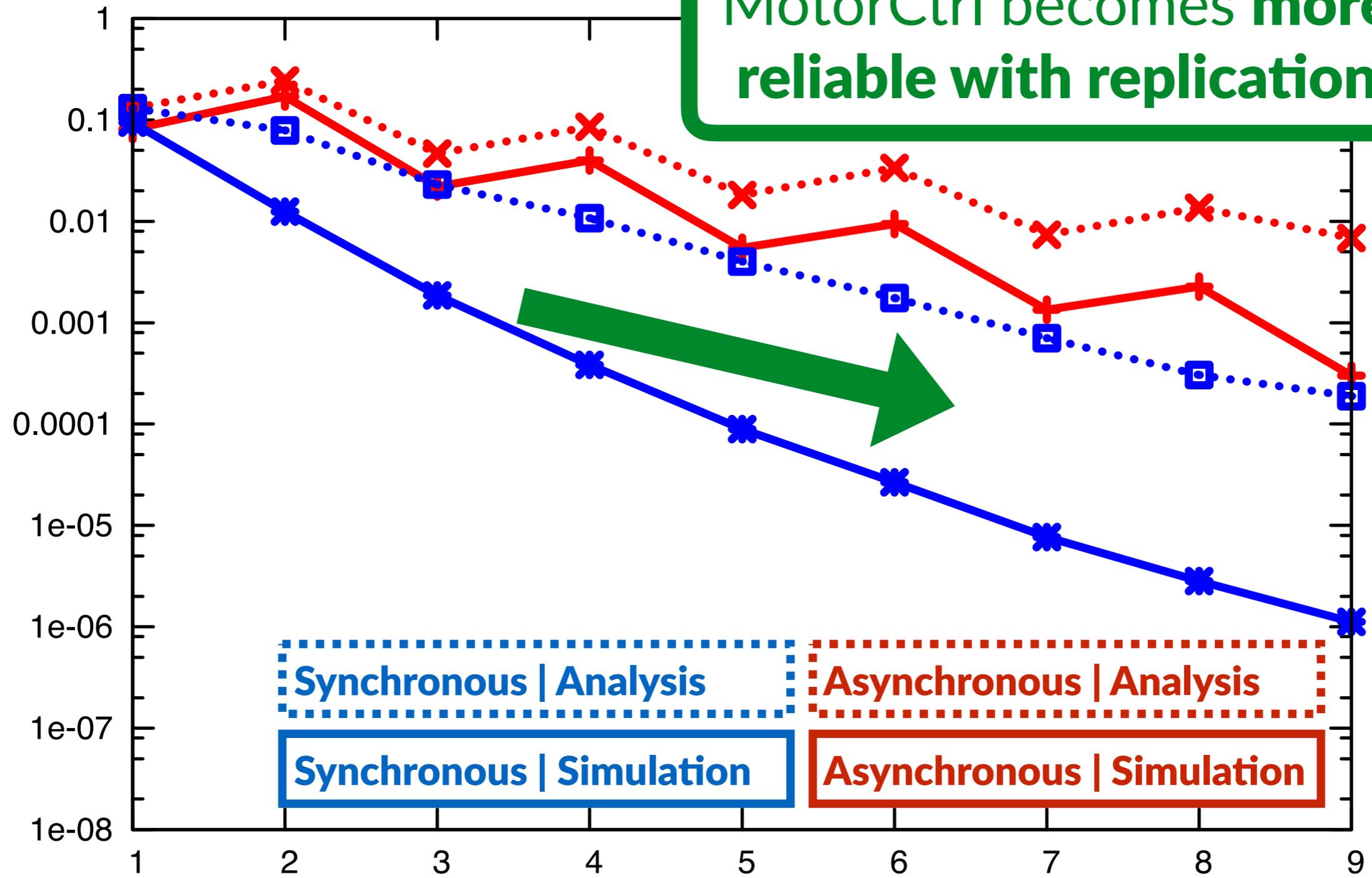
Analysis versus Simulation for MotorCtrl



Analysis versus Simulation for MotorCtrl

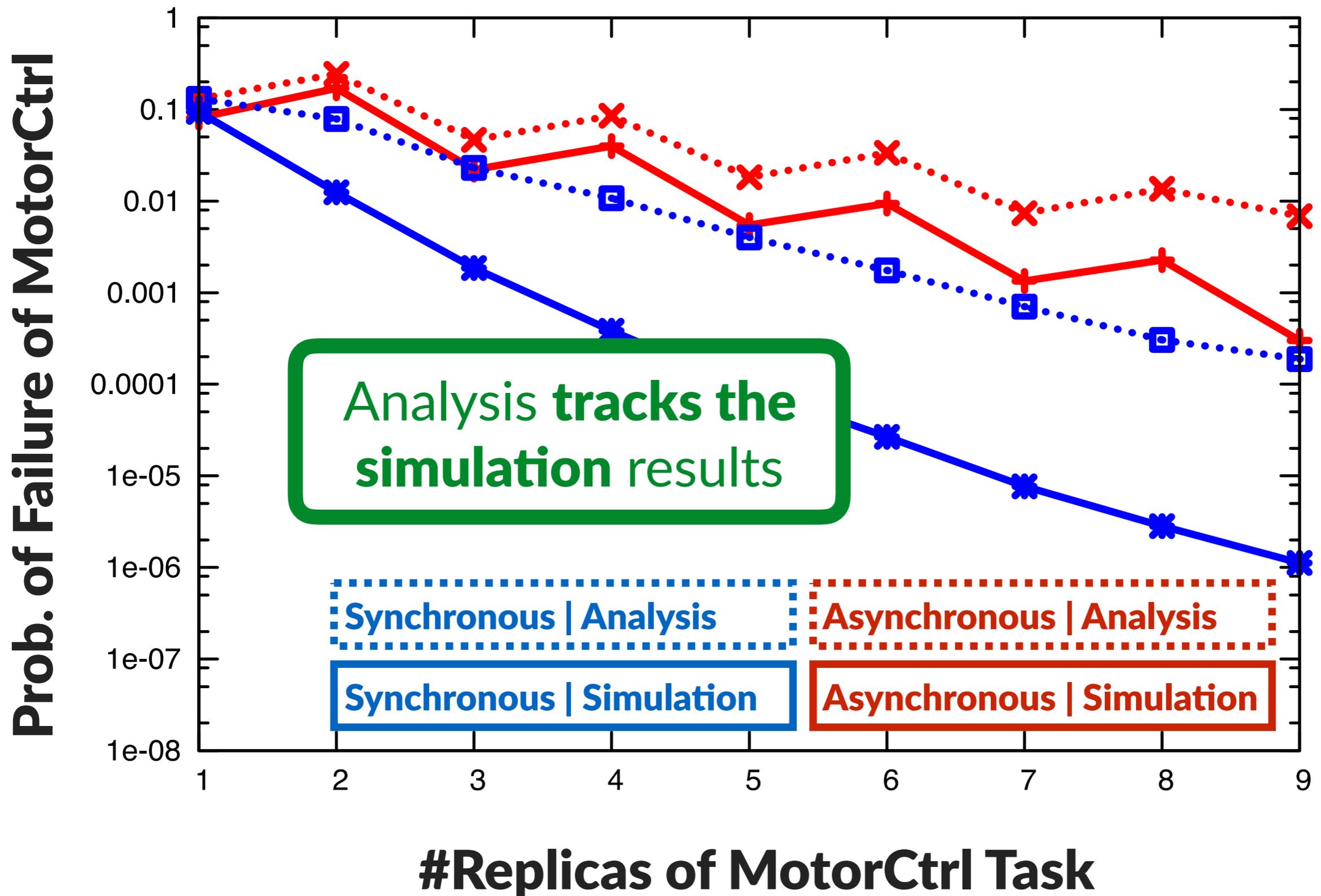
MotorCtrl becomes **more reliable** with replication

Prob. of Failure of MotorCtrl



#Replicas of MotorCtrl Task

Analysis versus Simulation for MotorCtrl

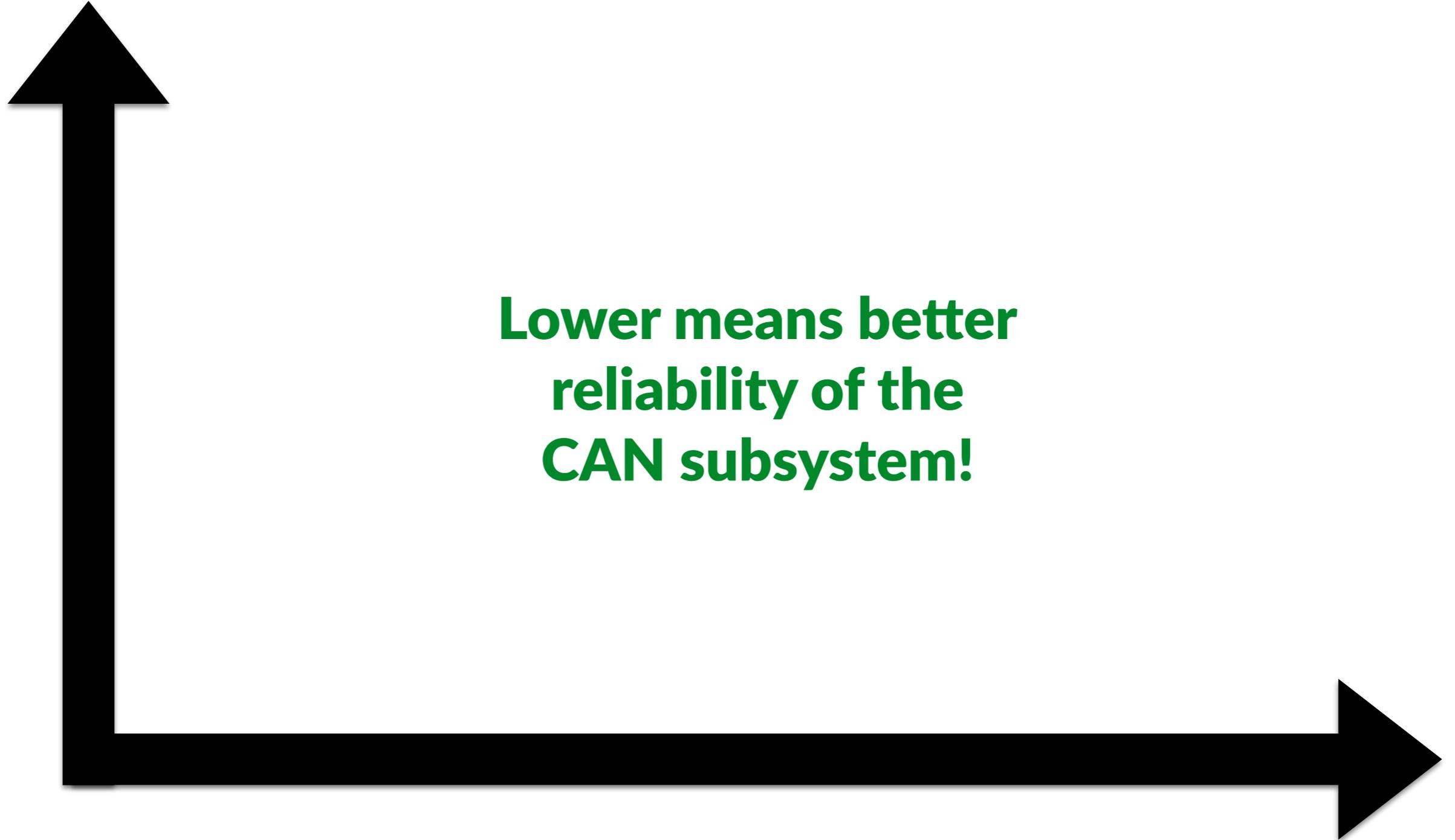


FIT Rate Analysis of the CAN Subsystem

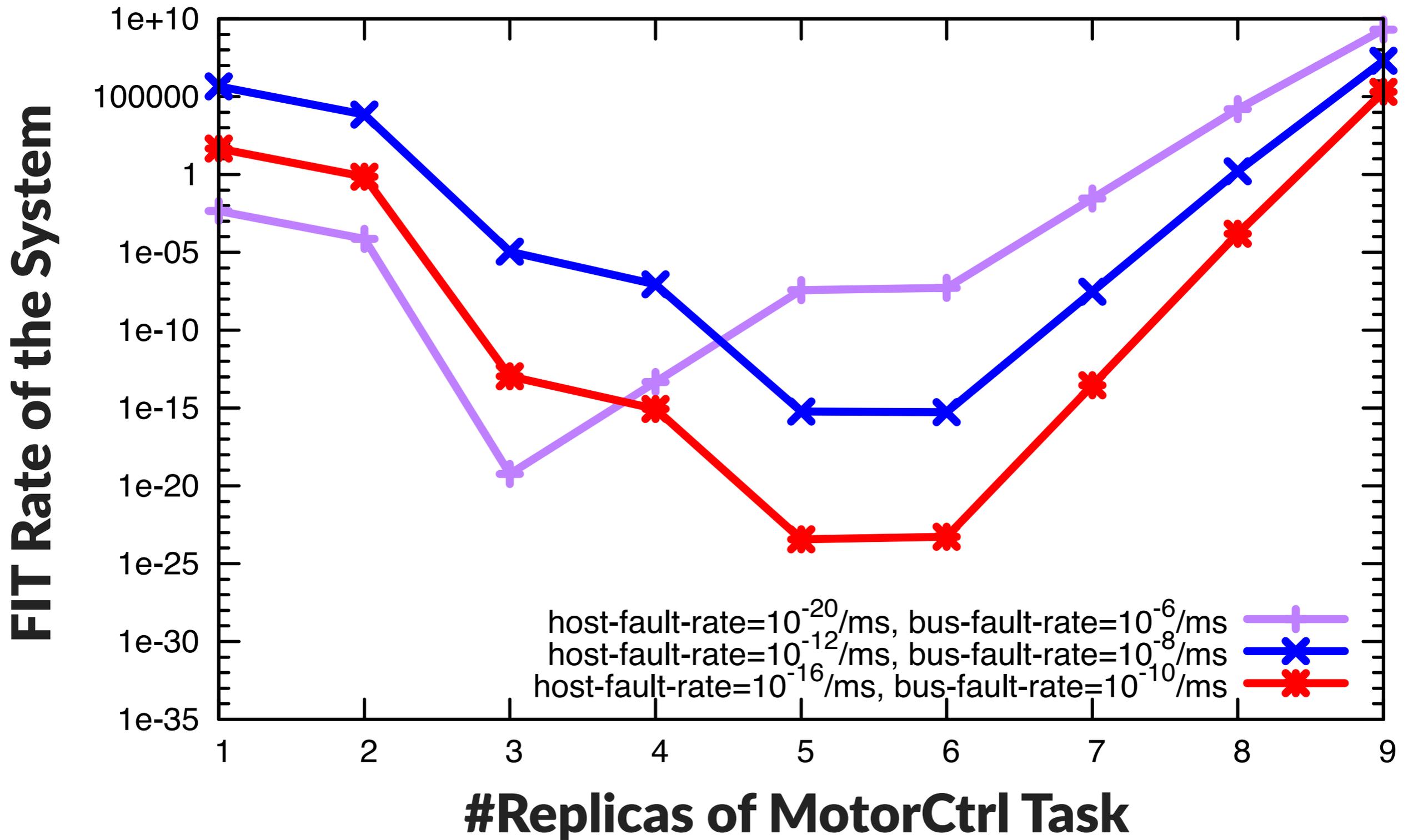
FIT Rate of the System

**Lower means better
reliability of the
CAN subsystem!**

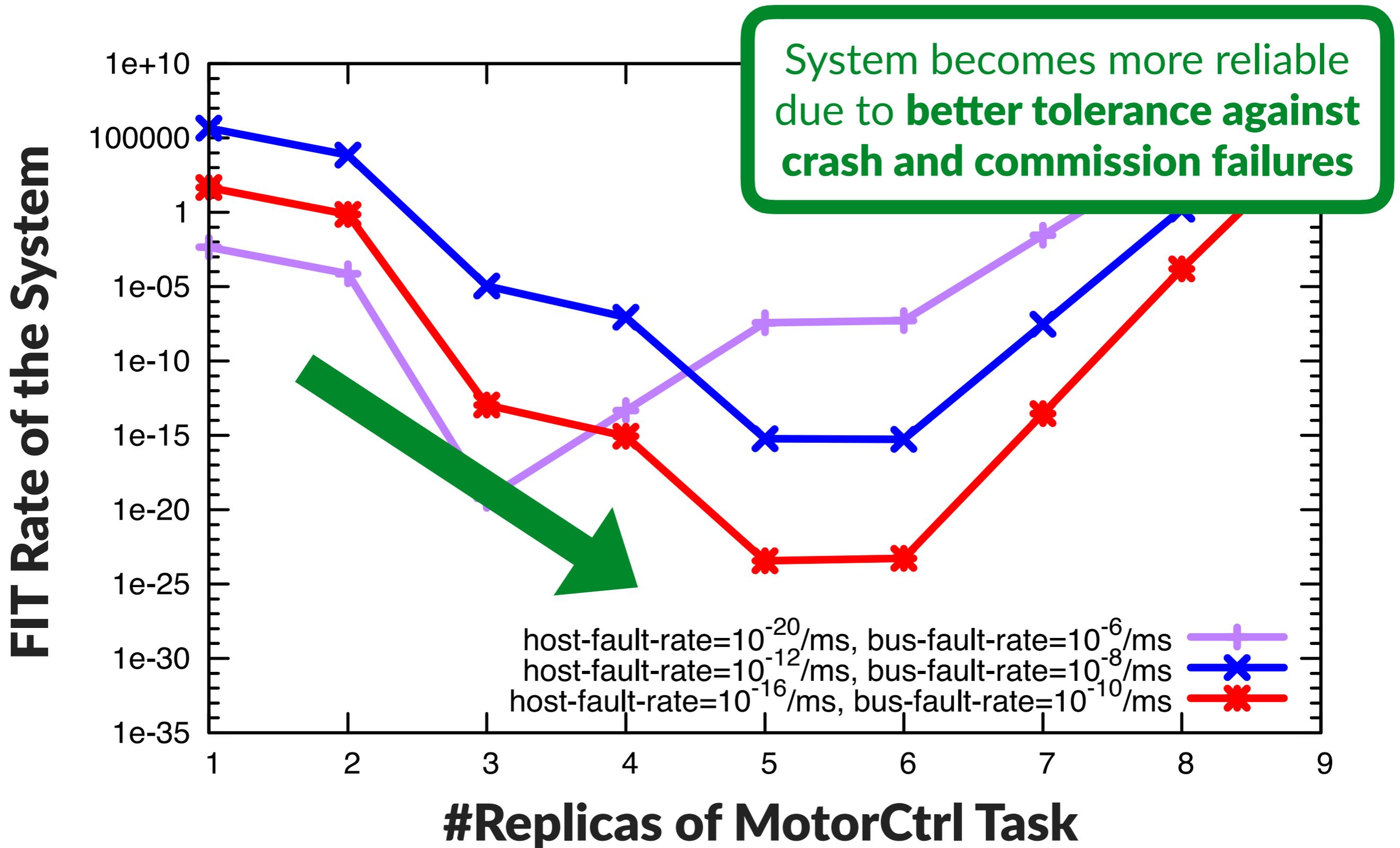
#Replicas of MotorCtrl Task



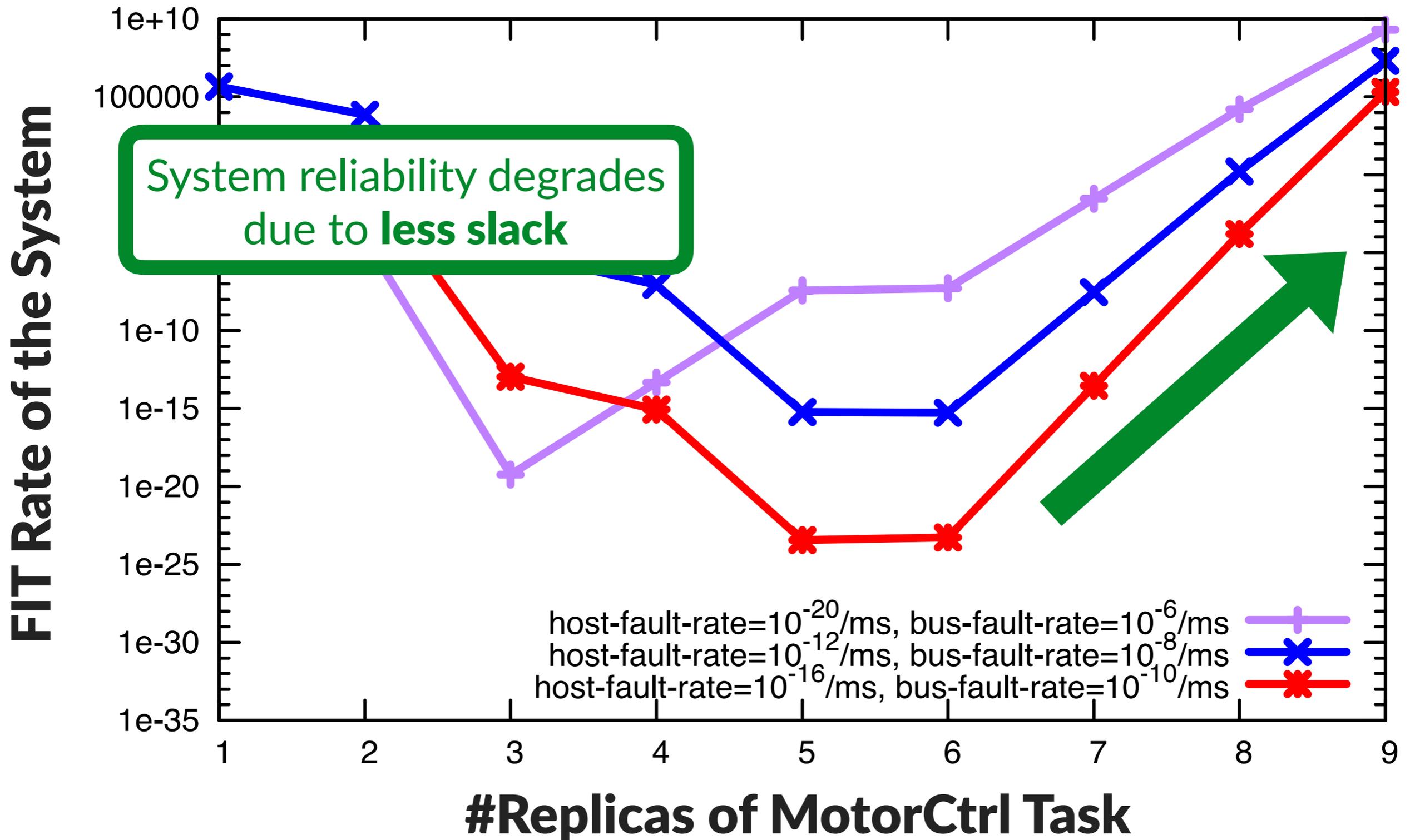
FIT Rate Analysis of the CAN Subsystem

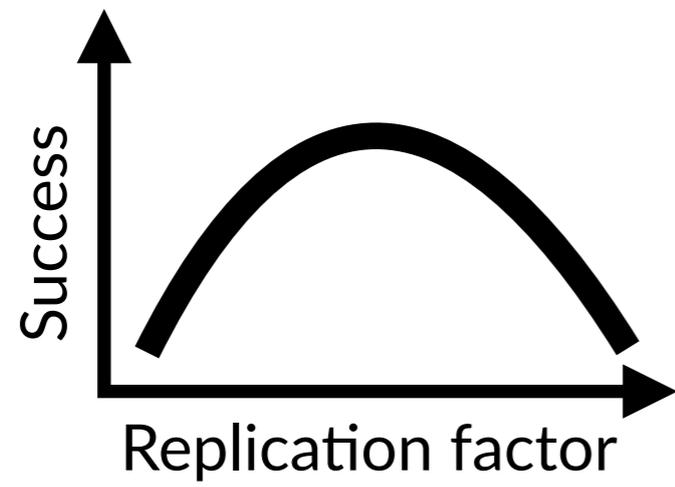


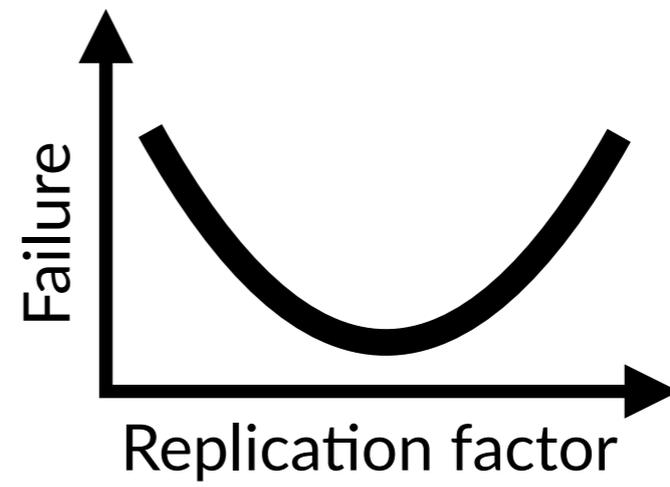
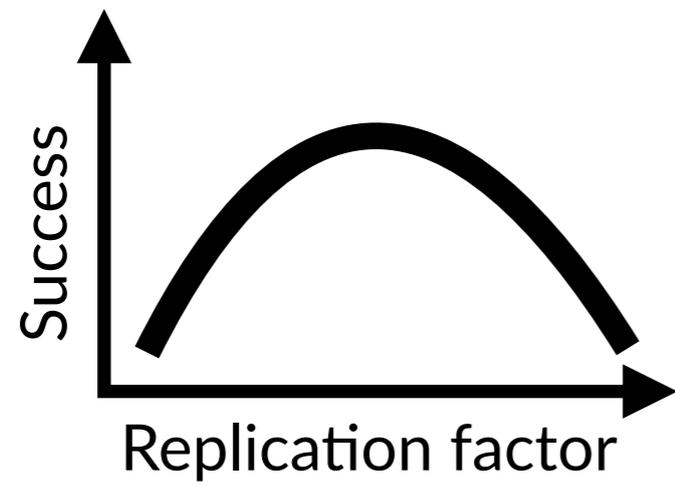
FIT Rate Analysis of the CAN Subsystem



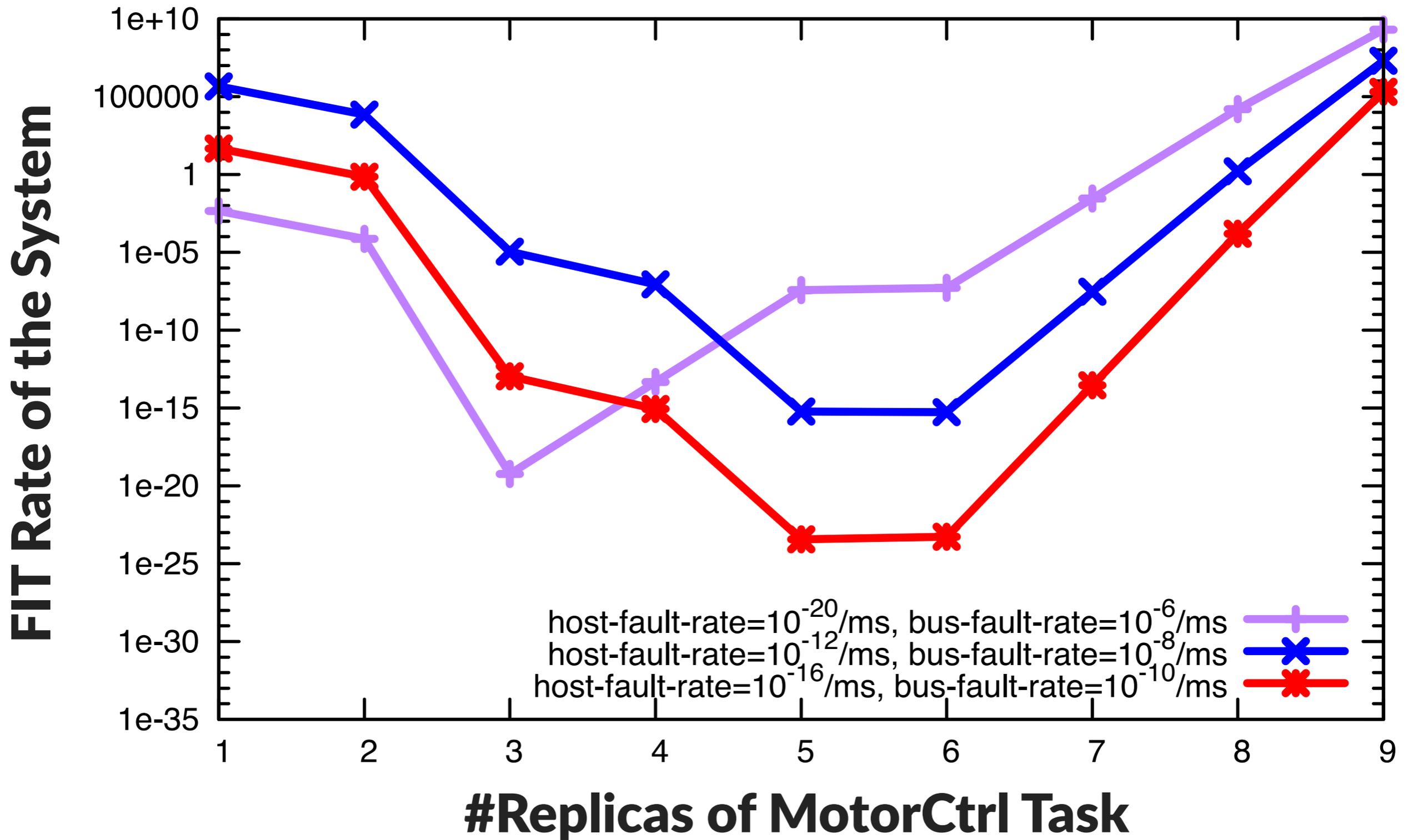
FIT Rate Analysis of the CAN Subsystem



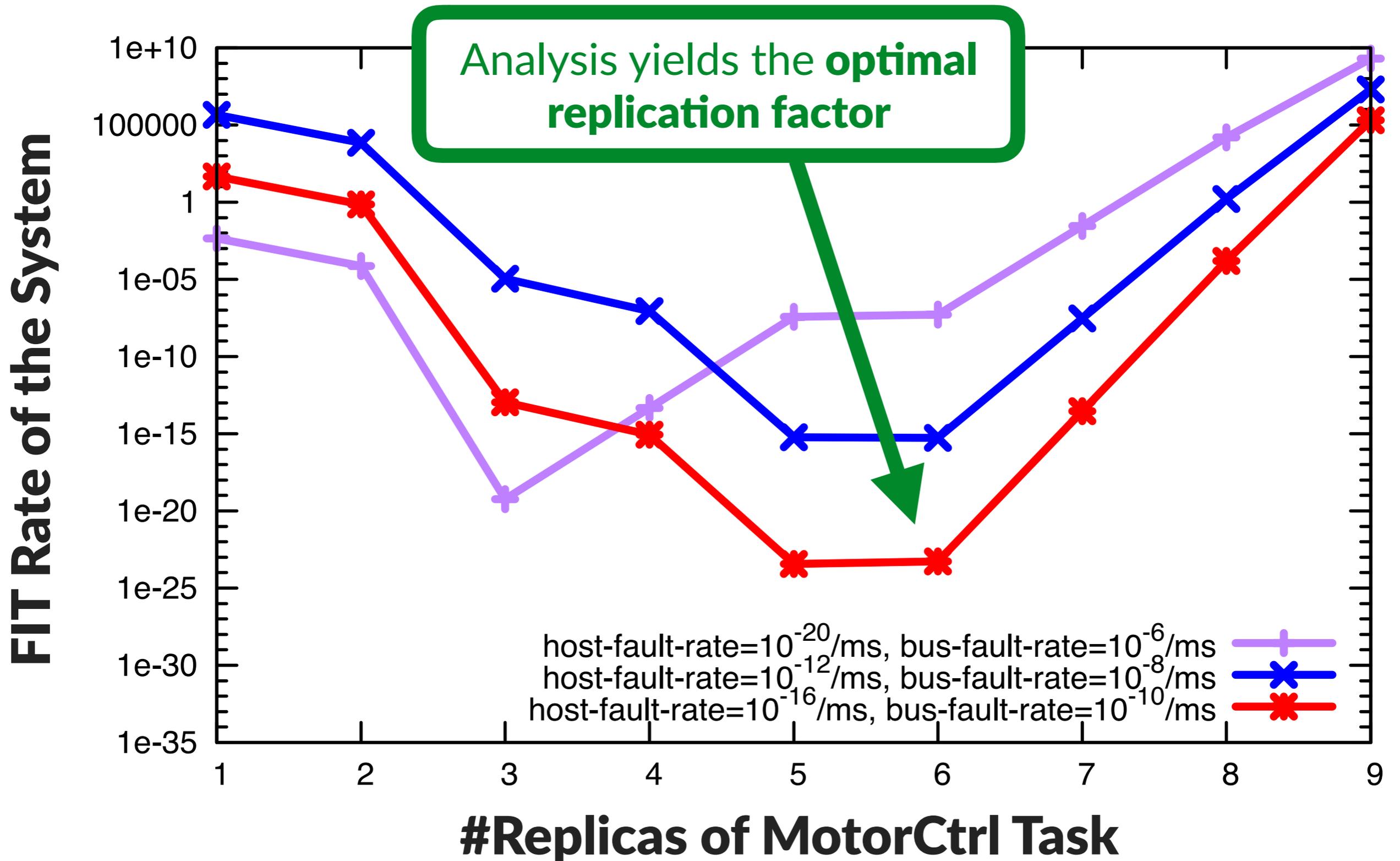




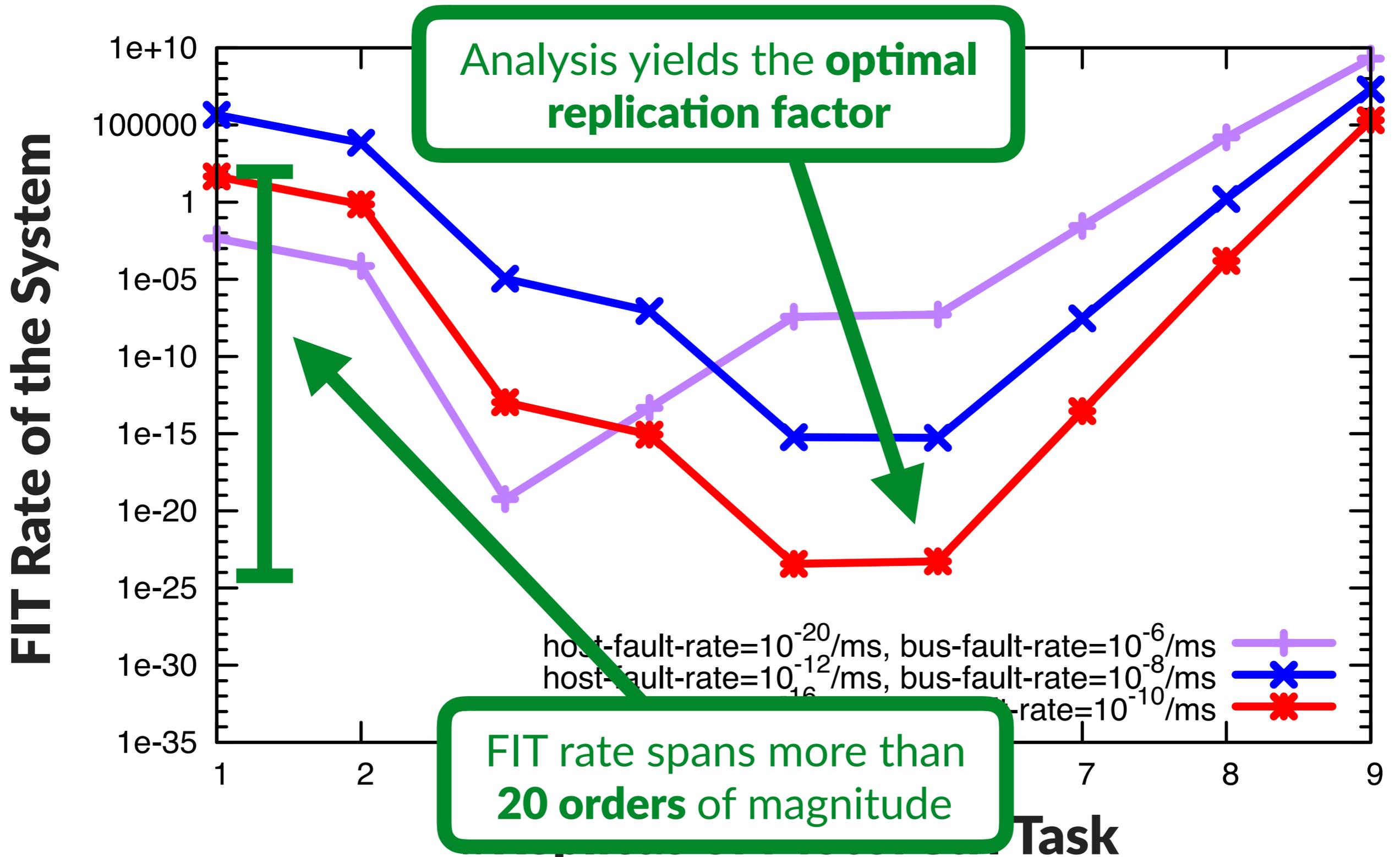
FIT Rate Analysis of the CAN Subsystem



FIT Rate Analysis of the CAN Subsystem



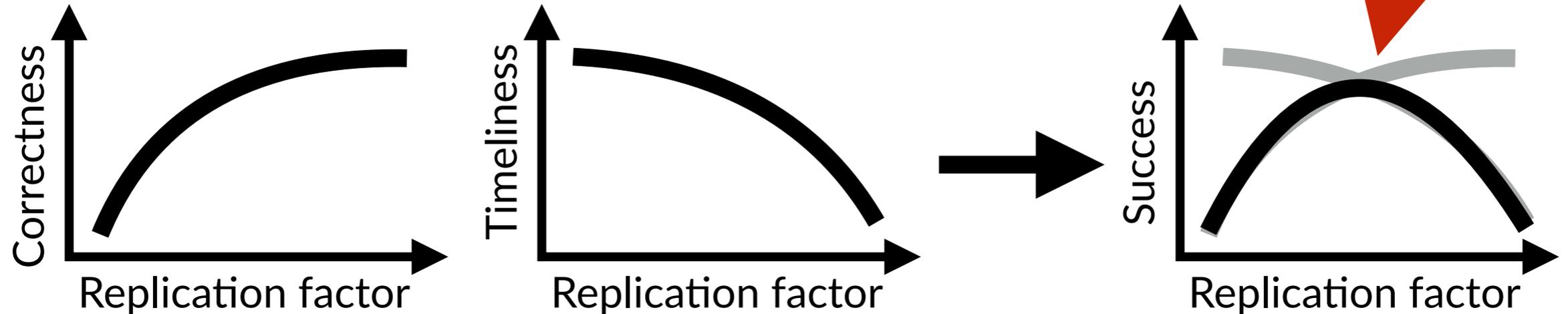
FIT Rate Analysis of the CAN Subsystem



Summary

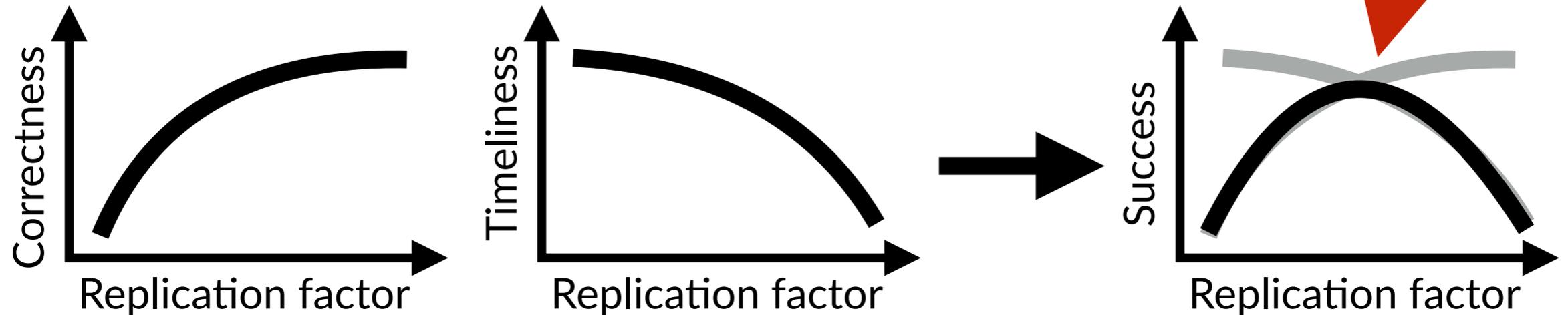
Summary

Find the best replication strategy for CAN-based systems

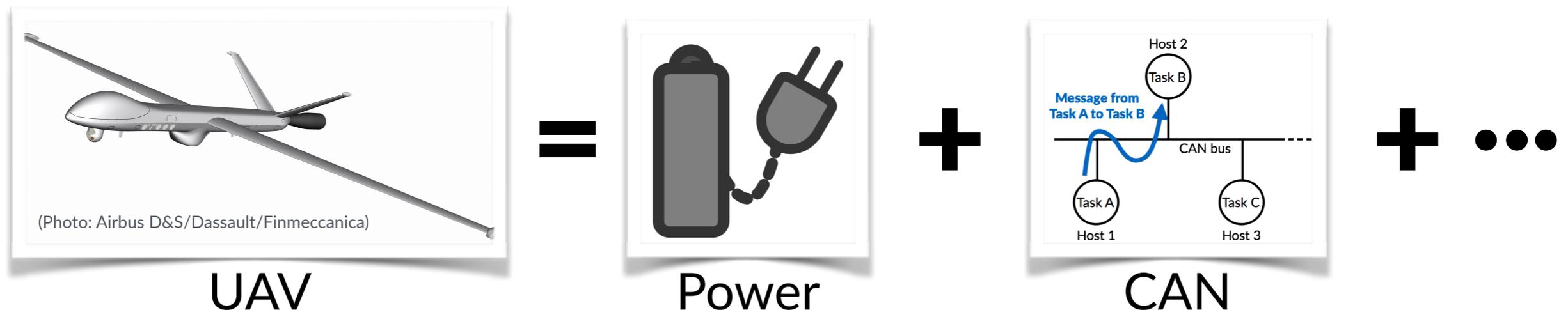


Summary

Find the best replication strategy for CAN-based systems



Compare reliability of the CAN-bus subsystem in the context of the larger safety-critical system



Future Work

- More complex system models
 - ➔ CAN-based systems bridged together
 - ➔ Sporadic DAG models

Future Work

- More complex system models
 - ➔ CAN-based systems bridged together
 - ➔ Sporadic DAG models
- Study of other technologies, e.g., Real Time Ethernet

Future Work

- More complex system models
 - ➔ CAN-based systems bridged together
 - ➔ Sporadic DAG models
- Study of other technologies, e.g., Real Time Ethernet
- <http://www.mpi-sws.org/~bbb/projects/schedcat>