SP 800-90B in Depth and Revision

Meltem Sönmez Turan, NIST

Random Bit Generation Workshop 2023

June 1, 2023

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

ITL INFORMATION TECHNOLOGY LABORATORY

Aim of the talk

- Provide a brief overview of SP 800-90B including general concepts, requirements, and entropy estimation process
- Explain the revision plan and next steps

NIST Special Publication 800-90B

Recommendation for the Entropy Sources Used for Random Bit Generation

Meltem Sönmez Turan
Elaine Barker
John Kelsey
Kerry A. McKay
Mary L. Baish
Mike Boyle

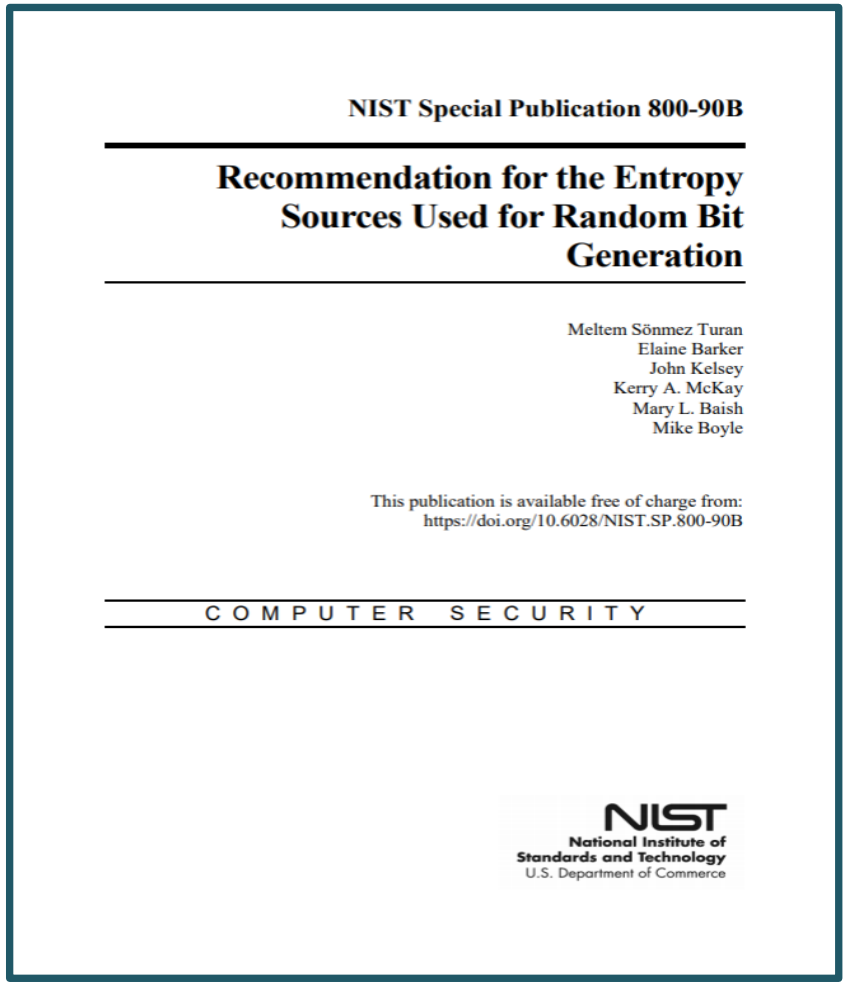
This publication is available free of charge from:
https://doi.org/10.6028/NIST.SP.800-90B

COMPUTER SECURITY

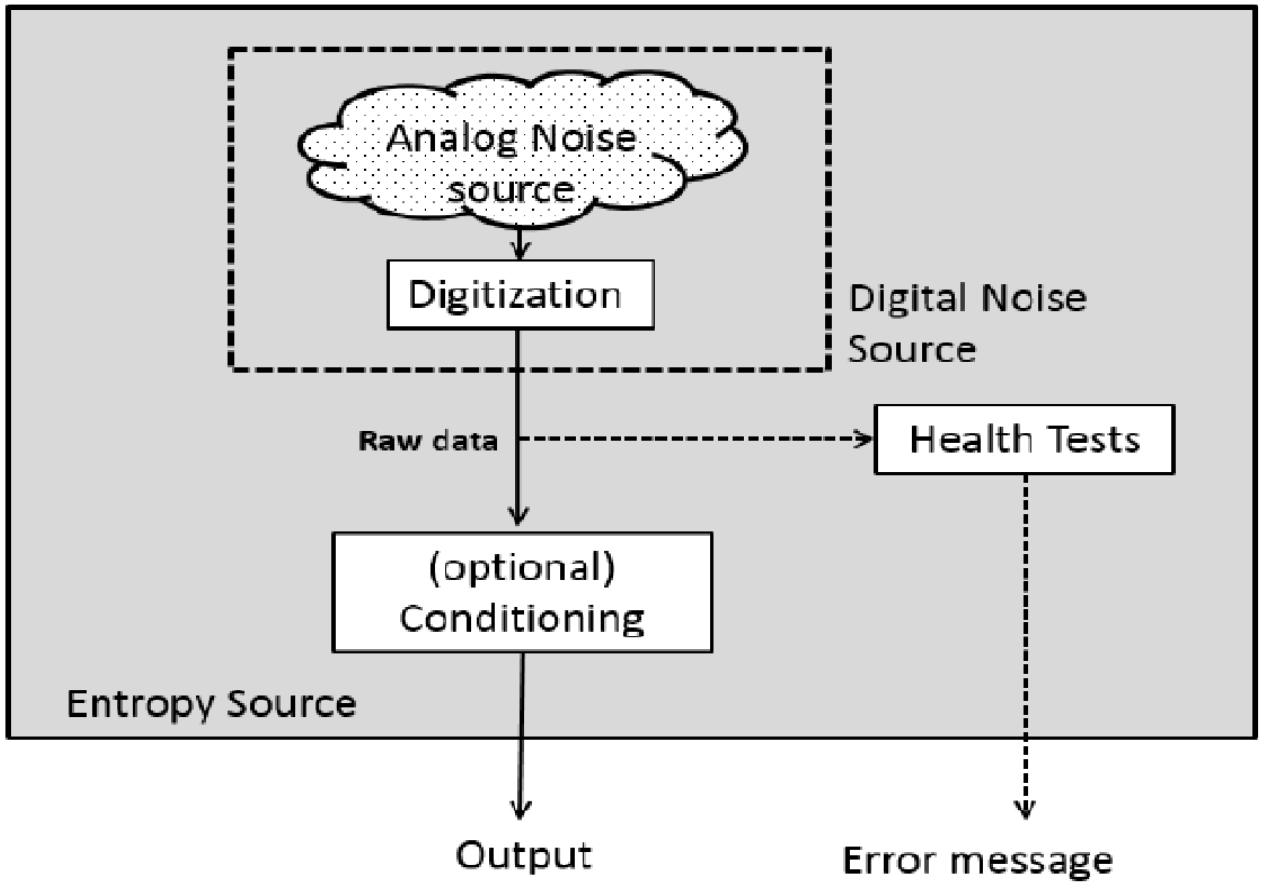
NIST
National Institute of Standards and Technology
U.S. Department of Commerce

SP 800-90B

- Defines the entropy source and provides a model.
- Specifies design principles for entropy source components.
- Lists requirements for the entropy source, including interactions between components.
- Specifies black-box entropy estimation techniques.
- Describes the validation process

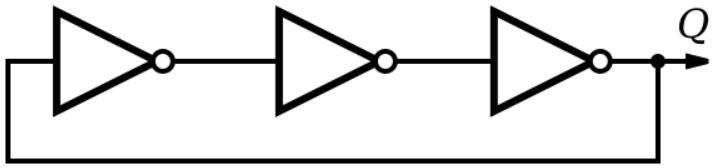


Entropy Source Model



Noise Source

A noise source provides bits or bitstrings of fixed length having approximately same amount of entropy per output.



Physical noise sources

- Dedicated hardware (e.g., ring oscillators)
- Behave similar for all copies of the RNG.
- Accurate modelling is suitable.

Non-Physical noise sources

- Use system data/human interaction (e.g., hard drive access times, mouse movements)
- Different behavior depending on the platform
- Do not allow accurate modelling.

Health Tests

Noise sources can be fragile and can be affected by the external conditions.

Health tests are required. They aim to detect deviations from intended behavior of the source during operation.

Types: Start-up, continuous and on-demand tests

Pre-defined health tests:

Adaptive proportion – detect when one value becomes much more common

Repetition count tests – detect when the source gets stuck on one output

User-defined tests tailored to the failure modes of the source

Conditioning Component

Noise source outputs may be statistically biased (e.g., biased coin), or correlated.

Conditioning component processes the noise source outputs to increase statistical quality of the outputs, or entropy rate.

90B specifies set of vetted conditioning components:

- Keyed: HMAC with any approved hash function, CMAC and CBC-MAC
- Unkeyed: Approved hash functions, hash_df (specified in 90A), Block_cipher_df (specified in 90A)

Vendors can use other conditioning components.

Conditioning is optional.

Conceptual Interfaces

90B describes three conceptual interfaces that can be used to interact with entropy source:

- GetEntropy inputs requested amount of entropy, and outputs the string that provides the requested entropy.
- GetNoise inputs requested number of samples from the noise source and receives the noise source outputs.
- HealthTest inputs the type of the tests to be performed.

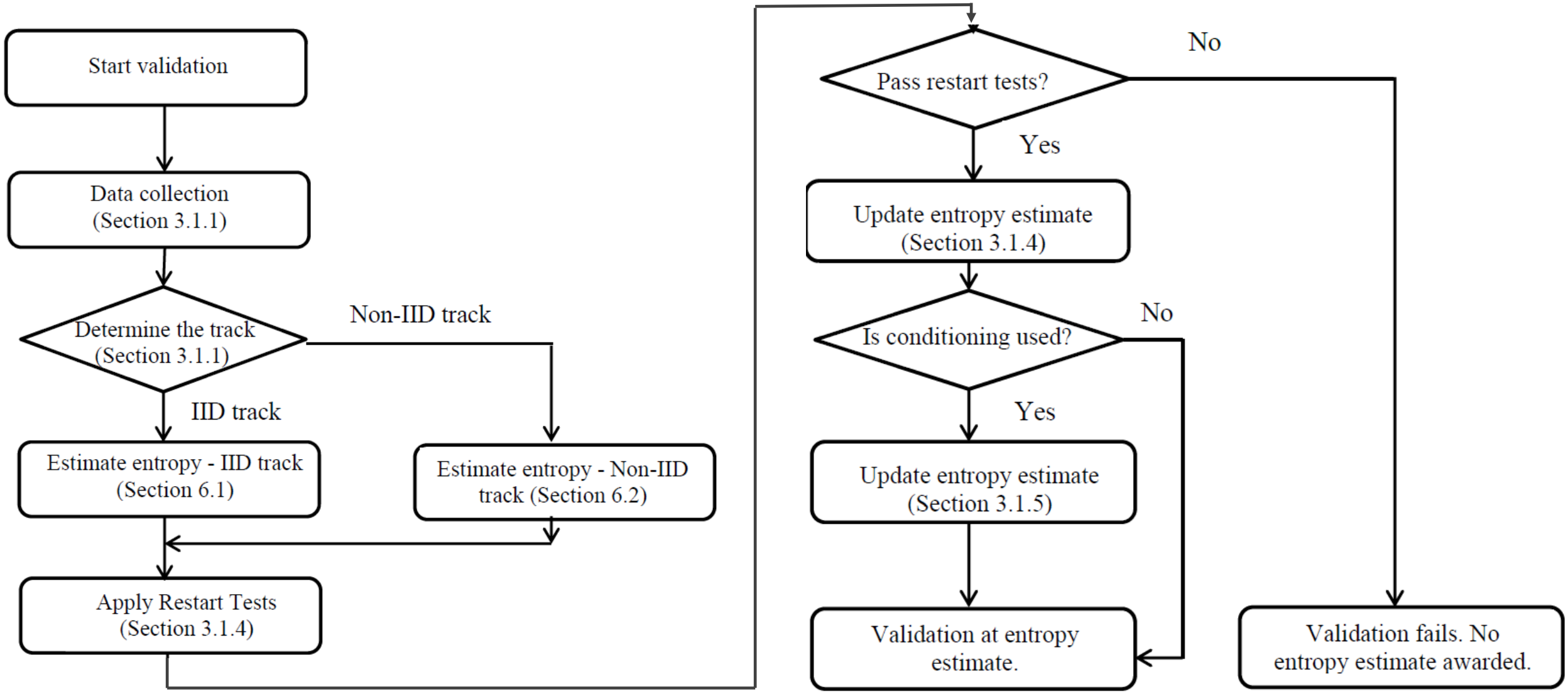
Basic requirements

- Documentation on the entire design of the entropy source including interactions between components, parameter selections
- Justification for why the noise source can be relied upon to produce bits with entropy
- Requirements on the noise source, conditioning component (correctness of the implementations etc.), health tests
- Requirements on data collection
- Range of operating conditions
- Entropy estimate from the submitter
- IID justification, if applicable
- etc.

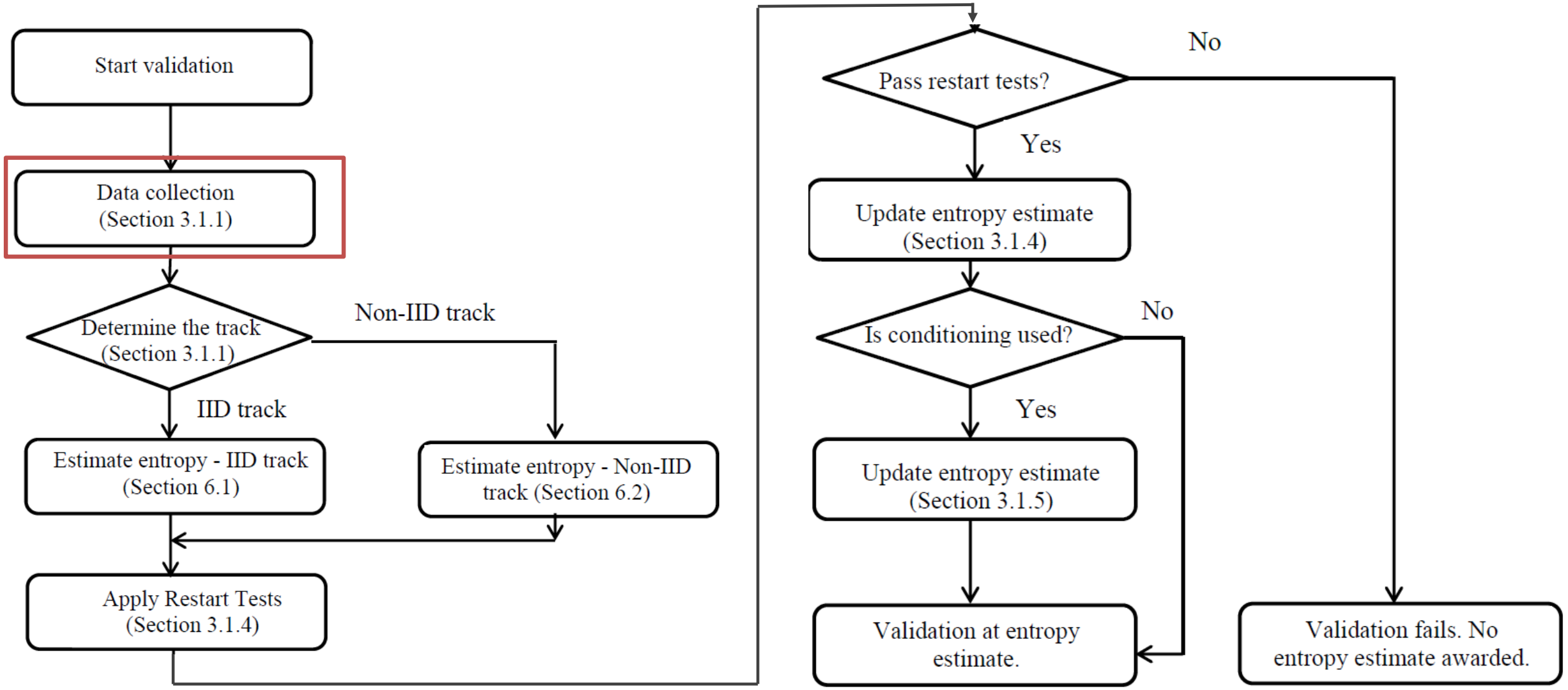
Validation Process

- Entropy source validation is necessary in order to obtain assurance that all relevant requirements are met.
- 90B provides requirements for validating an entropy source at a specified entropy rate.
- Validation consists of testing by an NVLAP-accredited laboratory against the requirements of 90B, followed by a review of the results by CAVP and CMVP.

Validating Entropy Sources



Validating Entropy Sources



Data Collection

Entropy estimation requires entropy source and noise source outputs.

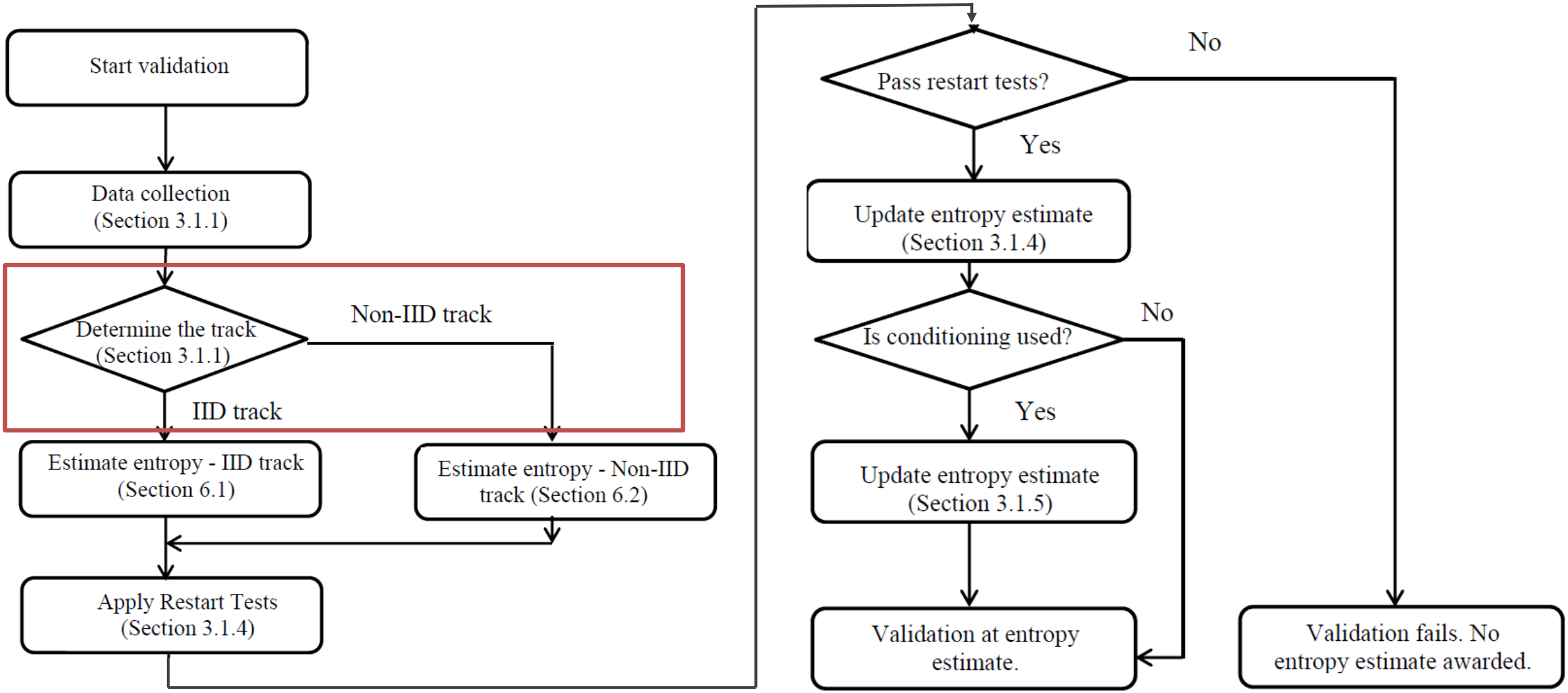
Two datasets from the noise source :

- Sequential dataset: 1,000,000 successive noise source outputs
- Restart dataset: 1,000 restarts, 1,000 samples from each restart (Restart = reboot, hard reset etc.)

One dataset for non-vetted conditioned outputs (if applicable):

- Conditioner dataset: 1,000,000 successive samples from conditioner

Validating Entropy Sources



Determining the Track

IID: independent and identically distributed

- each sample has the same probability distribution as every other sample, and
- all samples are mutually independent.

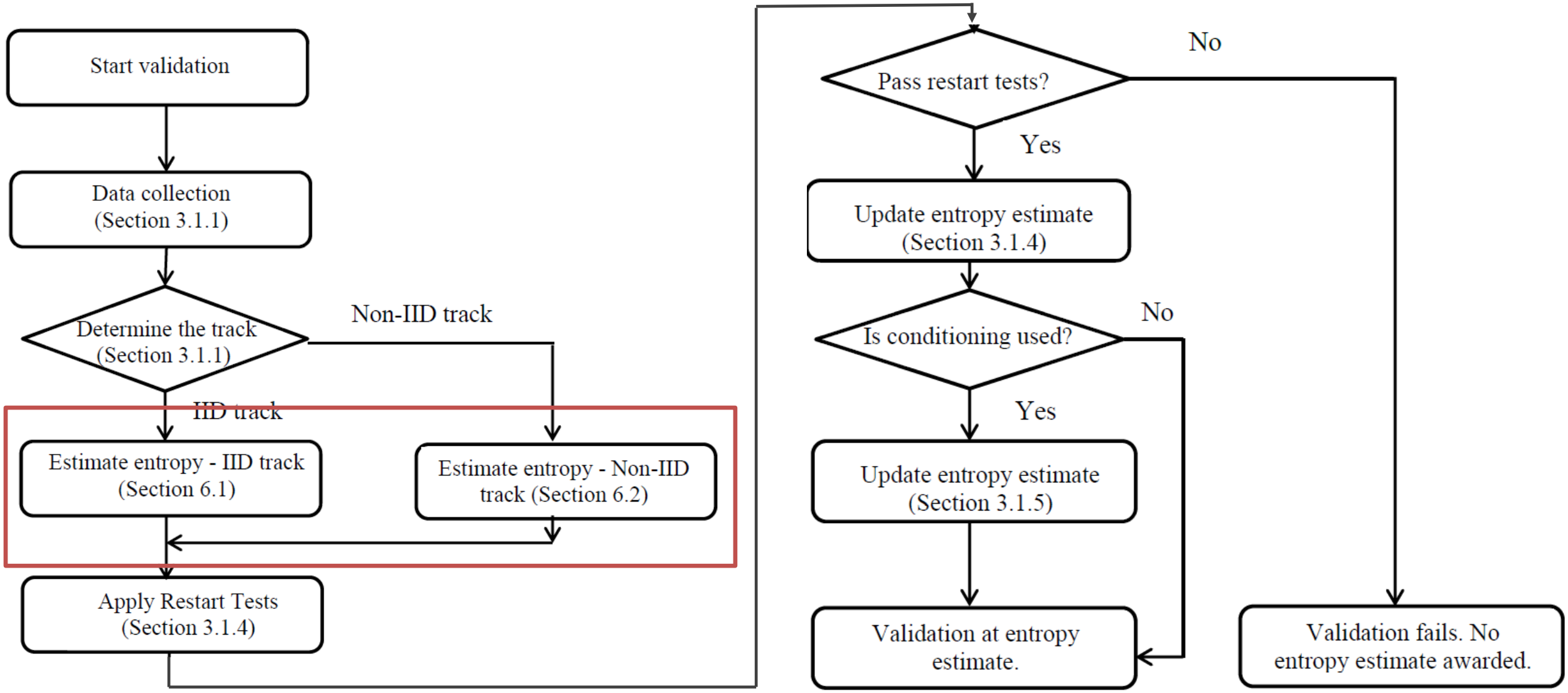
Two tracks: IID track and non-IID track

IID track is used only when both conditions are satisfied:

1. The submitter makes an IID claim on the noise source, based on the submitter's analysis of the design. The submitter provides rationale for the IID claim.

2. The input datasets (sequential, row, column and conditional) are tested using the statistical tests (permutation and chi-square) to verify the IID assumption, and the IID assumption is verified (i.e., there is no evidence that the data is not IID).

Validating Entropy Sources



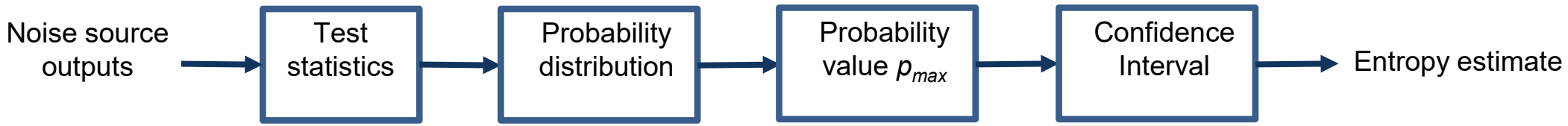
Estimating entropy

Main question: How unpredictable are the noise source outputs?

- Min-entropy is used to measure unpredictability.

Entropy estimation:

- Black-box approach, various estimation methods based on different statistical models (i.e., Markov models), and predictors



Two tracks:

- For IID track: based on the probability of the most common value.
- For non-IID track: multiple estimation methods are used.

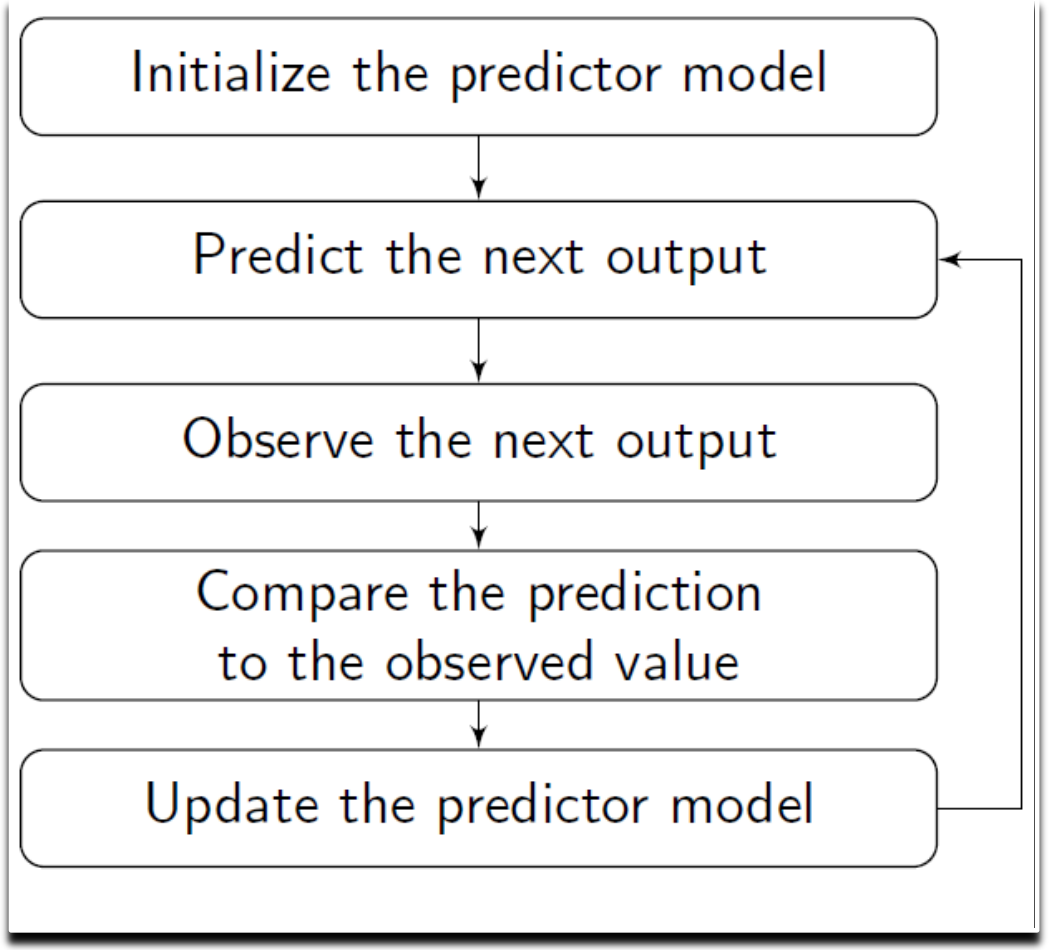
Estimating Entropy using Predictors

Predictors behave more like an attacker, who observes source outputs and makes guesses based on previous observations.

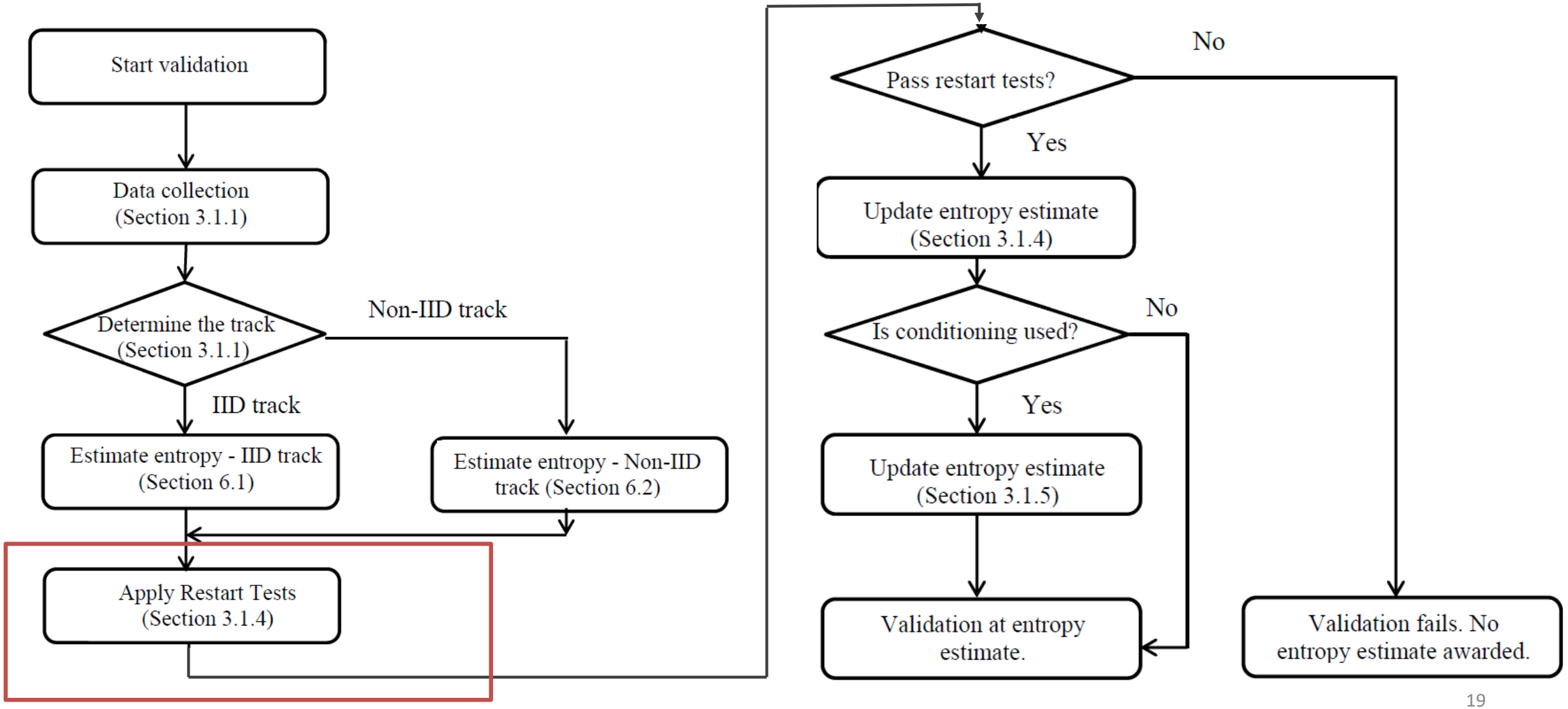
Generic predictors: Multi Most Common in Window Prediction, Lag Prediction, MultiMMC Prediction, LZ78Y Prediction

Global predictability: Number of correct predictions/Total number of predictions

Local predictability: Length of the longest run of correct predictions



Validating Entropy Sources



Restart Tests

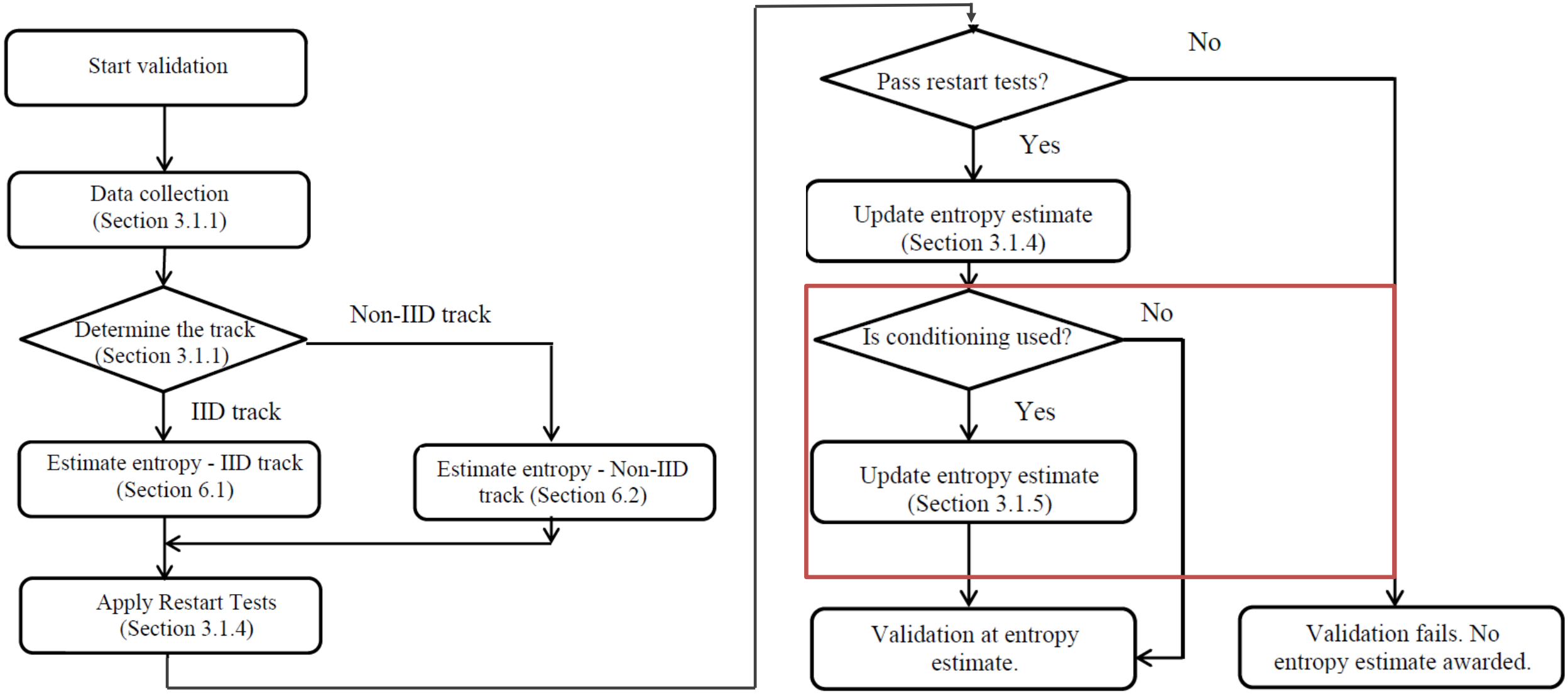
Noise sources may be predictable after restarts. (restart may be hard reset, reboot etc.)

Restart testing aims to detect predictability that only becomes apparent when examining many sequences generated by a source across restarts.

Restart dataset is a 1,000 x 1,000 matrix of samples: Row datasets and Column datasets

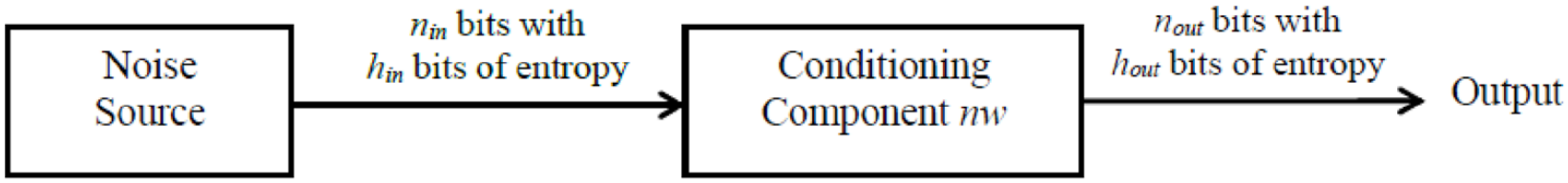
- Sanity test based on the frequency of the most common sample
- Entropy estimation using row and column datasets
- If the new estimate too low, validation fails.

Validating Entropy Sources



Conditioning

Entropy estimate is updated if a conditioning component is used.



Output_Entropy$(n_{in}, n_{out}, nw, h_{in})$

Let $P_{high} = 2^{-h_{in}}$ and $P_{low} = \frac{(1-P_{high})}{2^{n_{in}-1}}$.

$n = \min(n_{out}, nw)$.

$\psi = 2^{n_{in}-n} P_{low} + P_{high}$

$U = 2^{n_{in}-n} + \sqrt{2\, n(2^{n_{in}-n})\ln(2)}$

$\omega = U \times P_{low}$

Return $-\log_2(\max(\psi, \omega))$

For vetted conditioning:

$$h_{out} = \text{Output\_Entropy}(n_{in}, n_{out}, nw, h_{in})$$

For non-vetted conditioning:

$$\min(\text{Output\_Entropy}(n_{in}, n_{out}, nw, h_{in}), 0.999\text{n}_{out}, h' \times n_{out})$$

Caveats

- No set of general-purpose statistical tests can measure the entropy per sample in an arbitrary sequence of values.
- Better way is to understand the unpredictability of the noise source outputs, model it, and using the model to estimate the entropy. Run general purpose tests on outputs as a sanity check.
- NIST requires design documentation and an entropy estimate from the designer to support it, but there are limitations on available resources and required expertise from the labs for validation testing.

Revising NIST SP 800-90B

Revising NIST SP 800 90B

Since January 2018, received many comments and suggestions

- to improve technical quality of the standard
- to improve efficiency of the tests (e.g., permutation testing)
- to address issues with specific designs approaches
- editorial comments/typos/clarifications etc.

In August 2020, CMVP published the IG 7.19 *Interpretation of SP 800-90B Requirements*, based on the comments from CMUF Entropy WG.

Conditioning Components - Truncation

90B says: *Truncating vetted conditioning components is allowed, and entropy estimate is reduced to a proportion of the output*.

Ex: SHA-512 output with 128 bits of entropy truncated to 256 bits.

Consider as vetted: Output has 64-bits of entropy

Consider as non-vetted: Output has approx. 120 bits of entropy.

To avoid this inconsistency, truncated vetted conditioning components can be used as vetted components.

Bijective Conditioning Components

The output entropy formula models the conditioning as a random function, not a bijective function.

In random functions, due to internal collision, the output entropy decreases.

When a bijective function is used, and no need to apply the output entropy formula to reduce the entropy estimate.

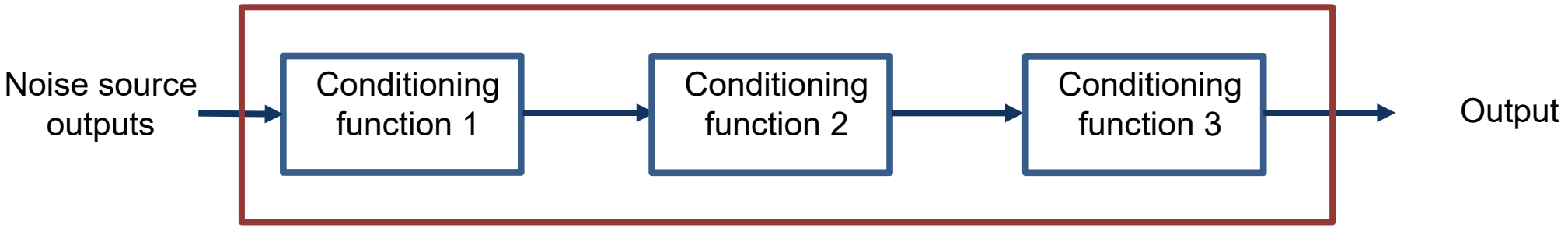
Input entropy can be assumed to be same as output entropy.

Output_Entropy($n_{in}, n_{out}, nw, h_{in}$)

Let $P_{high} = 2^{-h_{in}}$ and $P_{low} = \frac{(1-P_{high})}{2^{n_{in}-1}}$.
$n = \min(n_{out}, nw)$.
$\psi = 2^{n_{in}-n} P_{low} + P_{high}$
$U = 2^{n_{in}-n} + \sqrt{2\, n (2^{n_{in}-n}) \ln(2)}$
$\omega = U \times P_{low}$
Return $-\log_2(\max(\psi, \omega))$

Chaining Conditioning Components

When there are more than one conditioning components, 90B considers the function as a single non-vetted conditioning component as a combination of each. Not possible to achieve full entropy.



Now, chaining conditioning components, by calculating entropy estimate after each conditioning function is allowed. If the final conditioning function is vetted, it is possible to achieve full entropy.

Aligning NIST and BSI standards

BSI (Germany) also has similar standards:

- AIS 20: Functionality classes and evaluation methodology for deterministic random number generators
- AIS 31: Functionality classes and evaluation of physical random number generators

There are differences in the BSI's and NIST's validation process in terms of definitions, requirements, modeling and evaluation process.

Long term aim: Align NIST 90 Series and AIS 20, and AIS 31 by

- Understanding the source better using stochastic models (especially for physical sources).
- Differentiating between physical and non-physical sources
- Online health tests tailored to the characteristics of the design.

Thanks!

Contact NIST team

rbg_comments@nist.gov

Public forum

rbg-forum@list.nist.gov

Website

https://csrc.nist.gov/Projects/random-bit-generation