Technical Specification -
Cell Selection Integrity Verification (CSIV)

Version 2.3

Michael "Mike" Curnow, defy.nt
RTP, North Carolina

September 7, 2025

Contents

Important Notice

Notice of Disclaimer & Limitation of Liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations. No recommendation as to products and services or vendors is made or should be implied. No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights. In no event shall authors of this document be held liable for loss of profits or any other incidental or consequential damages. Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and authors of this document shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

© 2022–2025 Michael (Mike) Curnow, C6. Personal use of this material is permitted, along with republishing, reprinting this material for advertising or promotional purposes, creating new collective works, provided that this document and its authors are cited in any redistributed pieces where this document or any portions of this document and any copyrighted and/or trademarked components are present.

1 Scope

This document specifies the Cell Selection Integrity Verification protocol (CSIV), which is directed at hardening the Cell Selection procedures defined in the Radio Resource Control (RRC)[1][3] protocol for 4G LTE and 5G NR.

The scope of this document also includes:

- The components of CSIV.
- The processes of each CSIV component.

2 References

[1] 3GPP TS 36.331 Release 15, version 15.2.4 - "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification."

[2] 3GPP TS 36.331 Release 17, version 17.1.0 - "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification."

[3] 3GPP TS 36.304 Release 16, version 16.1.0 - "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode."

[4] 3GPP TS 38.331 Release 17 - "5G; NR; Radio Resource Control (RRC); Protocol specification."

[5] 3GPP TS 38.304 Release 17, version 17.1.0 - "5G; NR; User Equipment (UE) procedures in idle mode and in RRC Inactive state."

3 Terminology

3.1 Cell Selection Concepts and Terminology

To understand the subject matter of the cell selection process, it is recommended that readers of this technical specification read at a minimum the following, as terminology and methodologies stated herein are drawn from these documents:

1. 3GPP TS 36.331 Release 15 - "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC); Protocol specification"
2. 3GPP TS 36.304 Release 16 - "LTE; Evolved Universal Terrestrial Radio Access (E-UTRA); User Equipment (UE) procedures in idle mode"
3. 3GPP TS 38.331 Release 17 - "5G; NR; Radio Resource Control (RRC); Protocol specification"
4. 3GPP TS 38.304 Release 17 - "5G; NR; User Equipment (UE) procedures in idle mode and in RRC Inactive state"

3.2 CSIV Terminology

This section includes terminology contextually relevant specifically to this document. CSIV terminology is as follows:

- **CSIV** - Cell Selection Integrity Verification
- **HAL** - Hardware Abstraction Layer
- **IE** - Information Element (for MIB SIBs)
- **LTE** - Long-Term Evolution, referring to the 4th generation of mobile network communications
- **MIB** - Master Information Block
- **NR** - New Radio, referring to the 5th generation of mobile network communications
- **OCS** - Onboard Cell-Selection Storage
- **PLMN** - Public Land Mobile Network
- **SIB** - System Information Block

- **UE** - User Equipment, equipment utilizing cellular network to communicate (i.e. cellular telephone, cellular modem, IoT devices, etc)
- **VA** - Verification Algorithm
- **VC** - Verification Condition
- **Suspicion Score** - Aggregated, weighted measure of a candidate cell's anomalous behavior that decays over time and influences state transitions.
- **Barred Cell** - A cell temporarily excluded from selection due to persistent or high-confidence evidence of misbehavior.
- **Probation** - The period after a barred cell's expiry during which it must demonstrate clean behavior to be reinstated.
- **RSRP** - Reference Signal Received Power. The average power (in dBm) measured by the UE of the cell's downlink reference signals over the full bandwidth.

4 Introduction

Modern cellular systems deliberately begin with a *trust-by-default* posture: a UE must first discover and evaluate cells using unauthenticated broadcast information before any mutual authentication or ciphering can occur. In this window, any transmitter that speaks plausible 3GPP MIB/SIB can attract a UE, influence reselection, or throttle access. This design choice is operationally convenient, but it also creates an attack surface exploited by rogue cells (a.k.a. cell-site simulators) to harvest identifiers, mislead mobility, or degrade and deny availability.

Cell Selection Integrity Verification (CSIV) is a UE-side verification and decision framework that raises the bar in this pre-authentication phase. CSIV consumes only information already available to the UE (e.g., MIB/SIB fields, measurements, and neighbor relations) and applies explicit verification conditions (VCs), scoring with decay, and well-defined state transitions (Clean/Suspect/Barred/Probation). The goal is conservative: reject or quickly escape from cells whose broadcast parameters are implausible, inconsistent, or adversarial, thereby reducing opportunities for surveillance and service disruption, while minimizing false positives for legitimate networks.

4.1 Current Cell Selection

In 3GPP idle/INACTIVE operation, selection and reselection follow a simple pipeline driven by broadcast information:

- **PLMN selection** — The UE filters detected cells by PLMN(s) allowed by its subscription and policy. (See 3GPP TS 36.304/38.304 §5.1.)
- **Cell selection** — Among allowable cells, the UE applies minimum signal/quality criteria (e.g., thresholds derived from SIB1 such as *q-RxLevMin*) and chooses a suitable serving cell. (See TS 36.304/38.304 §5.2.)
- **Cell reselection (idle mobility)** — As conditions change, the UE prefers cells/frequencies with higher configured priority or better measured suitability, guided by SIB parameters (e.g., reselection priorities, neighbor lists, timing).

Critically, all of the inputs above (MIB/SIB content, reselection priorities, neighbor lists, thresholds) are received *before* security is established and are therefore unauthenticated. A malicious transmitter can:

1. Advertise attractive thresholds or priorities to win selection,
2. Omit or falsify neighbor information to trap the UE,
3. Abuse standard reject/backoff behaviors to deny or delay service.

CSIV addresses this permissiveness by validating the plausibility and consistency of those very inputs prior to access attempts and by enforcing conservative escape logic when anomalies persist.

4.2 Rationale and Intent

CSIV was originally conceived as a baseband–integrated hardening profile that augments the UE's Cell Selection and Cell Reselection procedures defined in 3GPP idle/INACTIVE operation (e.g., LTE E-UTRA UE Procedures in Idle Mode [3] and 5G NR UE Procedures in Idle/INACTIVE [5]). That remains the *normative* target: performing verification before the UE transmits, so suspicious cells are never granted an opportunity to elicit sensitive identifiers, and either throttle or outright deny service.

In practice, however, the same verification conditions (VCs), scoring, and state machine can also be realized at other layers where implementations have control or visibility, including:

- **Baseband/stack integration** (ideal): VC evaluation occurs prior to RACH / NAS registration; failures map directly to cell barring and reselection per 3GPP behavior.
- **HAL / driver layer**: a platform-specific module that inspects broadcast system information (MIB/SIB) and measurement reports, applies CSIV logic, and steers the modem via supported controls (e.g., RAT/PLMN preference, ARFCN/PCI allow/deny, modem resets, quick reselection triggers).
- **Supervisory user-space agent** (routers or rooted devices): a policy daemon that ingests modem metrics, runs CSIV VCs, and enforces outcomes using available actuators (RAT locks, frequency/PCI locks where supported, temporary cell/PLMN barring, radio toggling, APN detach, SIM slot fail-over).

This layered view recognizes real-world deployment constraints while keeping the core objective unchanged: *pre-attach gating* of suspicious cells, conservative false-positive posture, and rapid escape from hostile coverage.

4.3 Document Positioning and Scope

CSIV is a **UE-side verification and decision framework**, not a new air-interface protocol. It defines local computations (VCs, scoring, decay, state transitions) using information already provided by 3GPP specifications (e.g., MIB/SIB, measurements). CSIV does not add new RRC/NAS messages or IEs, and it does not alter the semantics of existing procedures.

Table 1: What CSIV Is / Is Not

CSIV defines	VC formulas and thresholds; suspicion scoring and decay; immediate-bar combinations; barred/probation state machine; mappings to local enforcement (bar/avoid/reselect/lock).
CSIV does not define	Any new over-the-air messages, IEs, timers, or changes to RRC/NAS flows; network-side behavior; operator policies beyond local enforcement.
Implementation targets	Baseband (preferred), HAL/driver, or supervisory userspace-identical logic, different actuators.

Non-goals. CSIV SHALL NOT modify, replace, or extend 3GPP signaling on the air interface. CSIV decisions occur pre-attach where possible and map to standard-compliant local behaviors (e.g., cell bar/reselection).

4.4 Expected Use of CSIV

Implementers MAY adopt CSIV at one or more layers, with trade-offs:

- **Baseband profile (preferred):** Apply all VCs (sVer, tVer, dVer, lVer, pVer, nVer, spVer, qVer, rVer) before completing selection/registration. “Fatal” conditions and defined combination overrides result in immediate bar; “soft” deviations contribute ΔS per weights and decay.
- **HAL / driver profile:** Apply the same VC set using broadcast information and measurements exposed by the modem. Enforcement maps VC outcomes to supported controls (e.g., deny/avoid lists, reselection triggers, barring timers) with the goal of preventing RACH/NAS on suspect cells whenever possible.
- **User-space supervisory profile:** Where only limited controls exist, CSIV acts as an advisory and steering layer. Outcomes drive best-effort mitigations (e.g., RAT lock, PLMN preference, temporary radio off/on to force reselection, APN suspend) and produce audit logs for forensics. This profile does *not* modify 3GPP state machines but can materially reduce dwell time on rogue cells.

Across all profiles, implementers SHOULD:

1. Evaluate VCs using only pre-authentication information whenever available.
2. Treat defined *Immediate bar* combinations as non-negotiable denials.
3. Use weights and decay as specified to minimize false positives while remaining responsive to persistent anomalies.
4. Record decisions and inputs (audit trail) to support operator review and tuning.

This specification therefore describes CSIV as a *layer-agnostic* verification and decision framework with a preferred (baseband) realization and practical HAL/user-space realizations where firmware changes are not feasible.

4.5 Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119], which is defined below:

- **MUST** - This word, or the terms "REQUIRED" or "SHALL", mean that the definition is an absolute requirement of the specification.
- **MUST NOT** - This phrase, or the phrase "SHALL NOT", mean that the definition is an absolute prohibition of the specification.
- **SHOULD** - This word, or the adjective "RECOMMENDED", mean that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.
- **SHOULD NOT** - This phrase, or the phrase "NOT RECOMMENDED" mean that there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
- **MAY** - This word, or the adjective "OPTIONAL", mean that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option MUST be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality. In the same vein an implementation which does include a particular option MUST be prepared to interoperate with another implementation which does not include the option.

5 Components

Various components are required for this mechanism to operate. The overall components for CSIV include:

- **Onboard Cell-Selection Storage (OCS)** - An area of storage on the UE which stores data collected from a cell that has passed Public Land Mobile Network (PLMN) and Cell Selection Criteria. This storage includes data from the Synchronization Signal Block (SSB) and Cell information gathered from System Information Blocks (SIB) types 1-5 for 4G LTE and types 1-4 for 5G NR, and static data such as that enumerated in 3GPP standards (scheduling info, timing info, etc)

- **Verification Conditions (VC)** - Are computational checks which make use of information in OCS that serve to provide boolean value output indicating a Cell's ability to pass certain conditions pertaining to scheduling, location, identity, timing, neighbor, and signal power. These conditions include static checks to ensure parameter values adhere to 3GPP enumerated set of values, and dynamic checks to contrast a Cell's parameters to that of other suitable candidate cells (i.e. those that already passed previous selection criteria) to discern deviations indicative of malicious Cell configuration pathologies.

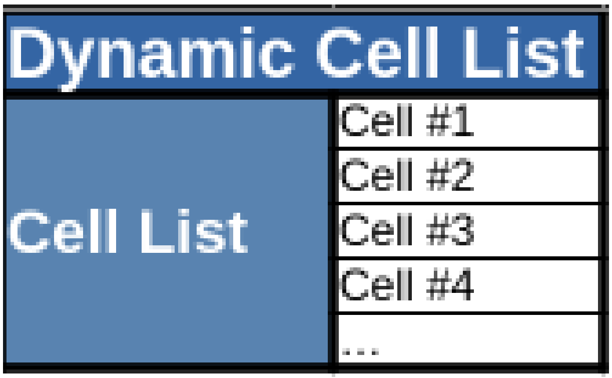
- **Verification Process** - The formula which determines whether a Cell should be added to the UE's barred cell list based on outcomes of various VC's. It performs computations of static and dynamic VC's to determine cell legitimacy after previous selection processes have succeeded (i.e. PLMN Selection, Cell Selection, Service Selection).

5.1 Onboard Cell-Selection Storage (OCS)

OCS storage is categorized into 3 types: Dynamic Cell List; Static Cell Lists; Dynamic Sets Lists. All of which are stored in UE's baseband processor for further computation.

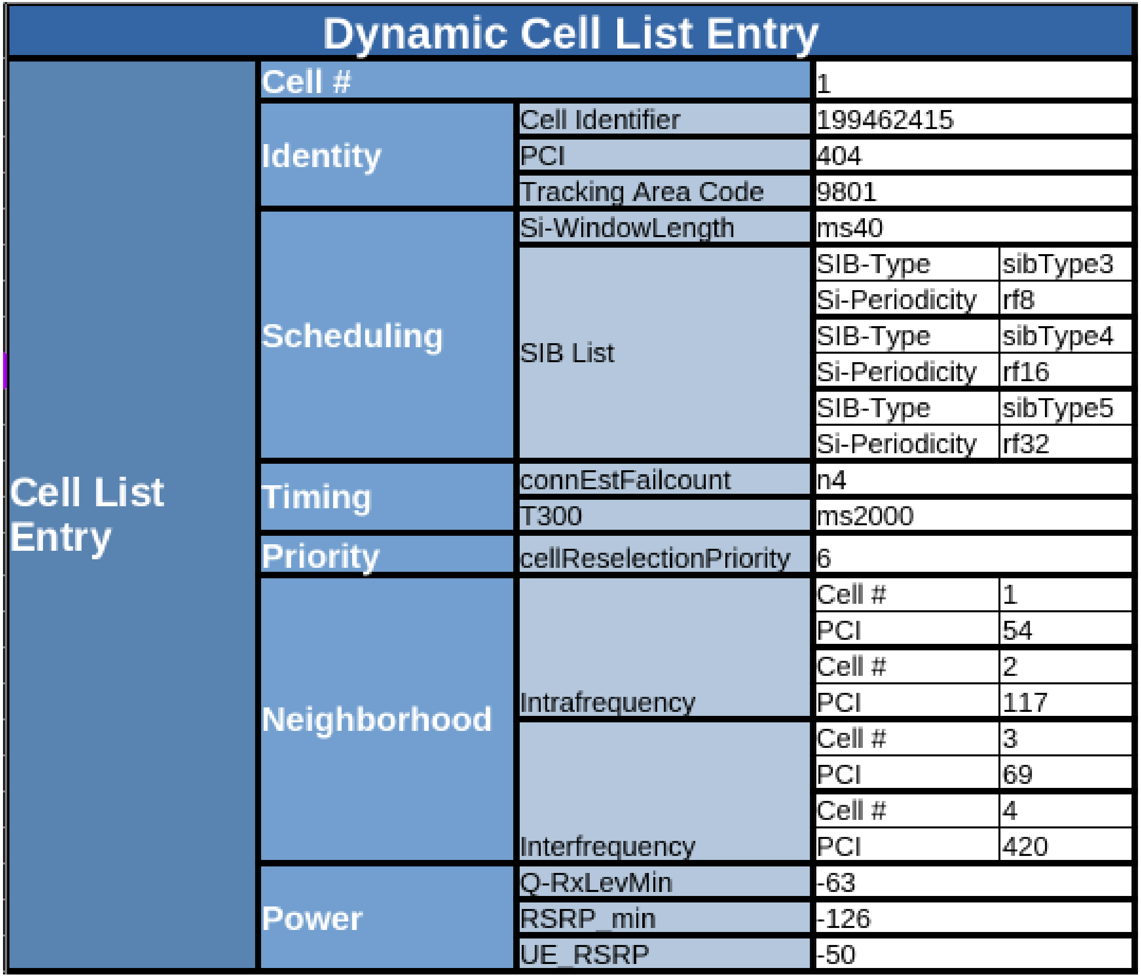
5.1.1 Dynamic Cell List

A list of cell entries, where each entry contains information elements of a particular cell, which is utilized for further computation.



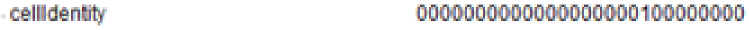
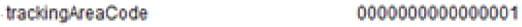
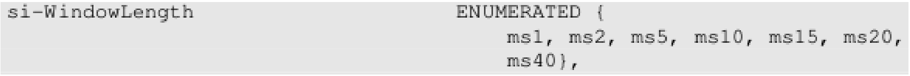
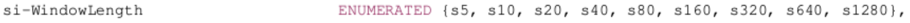
Dynamic Cell List	
Cell List	Cell #1
	Cell #2
	Cell #3
	Cell #4
	...

Figure 5.1.1-a. Dynamic Cell List

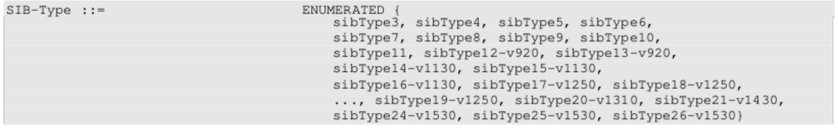


Dynamic Cell List Entry				
Cell List Entry	Cell #		1	
	Identity	Cell Identifier	199462415	
		PCI	404	
		Tracking Area Code	9801	
	Scheduling	Si-WindowLength	ms40	
		SIB List	SIB-Type	sibType3
			Si-Periodicity	rf8
			SIB-Type	sibType4
			Si-Periodicity	rf16
			SIB-Type	sibType5
			Si-Periodicity	rf32
	Timing	connEstFailcount	n4	
		T300	ms2000	
	Priority	cellReselectionPriority	6	
	Neighborhood	Intrafrequency	Cell #	1
			PCI	54
			Cell #	2
			PCI	117
		Interfrequency	Cell #	3
			PCI	69
			Cell #	4
			PCI	420
	Power	Q-RxLevMin	-63	
		RSRP_min	-126	
		UE_RSRP	-50	

Figure 5.1.1-b. Dynamic Cell List Entry

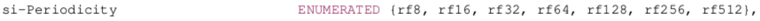
- **Cell Number/#** - Numeric identifier of a Dynamic Cell List Entry.
- **Identity** - Contains identity information for cell list entry.
  - Cell Identifier - Used to identify a cell uniquely within a Public Land Mobile Network (PLMN). In 4G LTE this is referred to as the E-UTRAN Cell Identifier (ECI), has a length of 28 bits and contains the eNodeB-Identifier (eNB-ID). In 5G NR this is referred to as the NR Cell Identity (NCI), has a length of 36 bits and contains the gNodeB-Identifier (gNB-ID).
    Example:
    
    ```
    cellIdentity                    0000000000000000000100000000
    ```
  - Physical Cell Identity (PCI) - The Primary Synchronization Signal (PSS) and Secondary Synchronization Signal (SSS) from the cell's Synchronization Signal Block (SSB), of which the UE is able to calculate the Physical Cell Identity (PCI). Formula: $PCI = 3(PSS) + SSS$
  - Tracking Area Code (TAC) - The BTS broadcasts its (TAC), which is a bit string (16 bits in 4G and 24 bits in 5G) used to indicate which Tracking Area the BTS belongs to, and the TAC is unique within a PLMN.
    Example:
    
    ```
    trackingAreaCode                0000000000000001
    ```
- **Scheduling** - Contains information relevant to scheduling of SIB messages from a cell to receiving UE within vicinity of cell's coverage.
  - si-WindowLength - The length of the SI scheduling window. In 4G this unit is in milliseconds, where ms1 denotes 1 millisecond, ms2 denotes 2 milliseconds and so on. In 5G this unit is measured in slots, where the value s5 corresponds to 5 slots, value s10 corresponds to 10 slots and so on. The network always configures si-WindowLength to be shorter than or equal to the si-Periodicity.
    Examples:
    4G LTE
    
    ```
    si-WindowLength                         ENUMERATED {
                                                ms1, ms2, ms5, ms10, ms15, ms20,
                                                ms40},
    ```
    5G NR
    
    ```
    si-WindowLength                 ENUMERATED {s5, s10, s20, s40, s80, s160, s320, s640, s1280},
    ```
  - SIB List - Contains the SIB-Type and si-Periodicity for each SIB the cell is broadcasting to UE's within vicinity of coverage.

    - SIB-Type - This value represents the mapping to a SIB type which information such as *si-periodicity* would apply to.
      Example:



```
SIB-Type ::=                        ENUMERATED {
                                        sibType3, sibType4, sibType5, sibType6,
                                        sibType7, sibType8, sibType9, sibType10,
                                        sibType11, sibType12-v920, sibType13-v920,
                                        sibType14-v1130, sibType15-v1130,
                                        sibType16-v1130, sibType17-v1250, sibType18-v1250,
                                        ..., sibType19-v1250, sibType20-v1310, sibType21-v1430,
                                        sibType24-v1530, sibType25-v1530, sibType26-v1530}
```

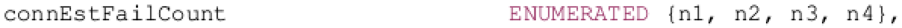
    - si-Periodicity - Periodicity of the System Information (SI) message in radio frames, such that rf8 denotes 8 radio frames, rf16 denotes 16 radio frames, and so on.
      Example:



```
si-Periodicity                      ENUMERATED {rf8, rf16, rf32, rf64, rf128, rf256, rf512},
```

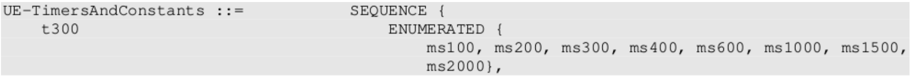
- **Timing** - Contains information relevant to timing operations of the cell and connection timeout configurations.

  - connEstFailCount - This is used to configure parameters for connection establishment failure control.
    Example:



```
connEstFailCount                    ENUMERATED {n1, n2, n3, n4},
```

  - T300 - A timing value from IE "UE-TimersAndConstants" used to establish limit for connection timeouts.
    Example:



```
UE-TimersAndConstants ::=           SEQUENCE {
    t300                                ENUMERATED {
                                            ms100, ms200, ms300, ms400, ms600, ms1000, ms1500,
                                            ms2000},
```

- **Power** - Broadcast minimum and measured received levels used by CSIV power-related checks.

  - **q-RxLevMin** (Q_{rxlevmin}) — Cell's advertised minimum received level (from SIB1), encoded in 2 dB steps. CSIV converts it to a dBm threshold as:

$$RSRP_{\text{min}}\,(\text{dBm}) = 2 \times Q_{\text{rxlevmin}}$$

    Example: $Q_{\text{rxlevmin}} = -58 \Rightarrow RSRP_{\text{min}} \approx -116\,\text{dBm}$.

  - $RSRP_{\text{min}}$ — Derived minimum RSRP threshold (dBm) computed from Q_{rxlevmin}; used by verification checks (e.g., *rVer*) when comparing measured level to the advertised minimum.

  - **UE measured RSRP** (UE_RSRP) — Latest filtered per-cell RSRP in dBm as measured by the UE and snapshotted here (full history/EWMA state may be kept in the *Dynamic Set Lists*).

- **Priority** - Denotes the degree of priority of other UE to camp on the cell.

  - connectionReselectionPriority - The absolute priority of the concerned carrier frequency, as used by the cell reselection procedure. Represented as integers between 0 (lowest) and 7 (highest). Example:

- **Neighborhood** - Contains information of a cell's intrafrequency and interfrequency cell neighbors.

  - Intrafrequency Neighbor PCI List - The intra-frequency neighbor list contains PCIs of all intra-frequency cells that are registered neighbors with the current cell whose integrity we're verifying.

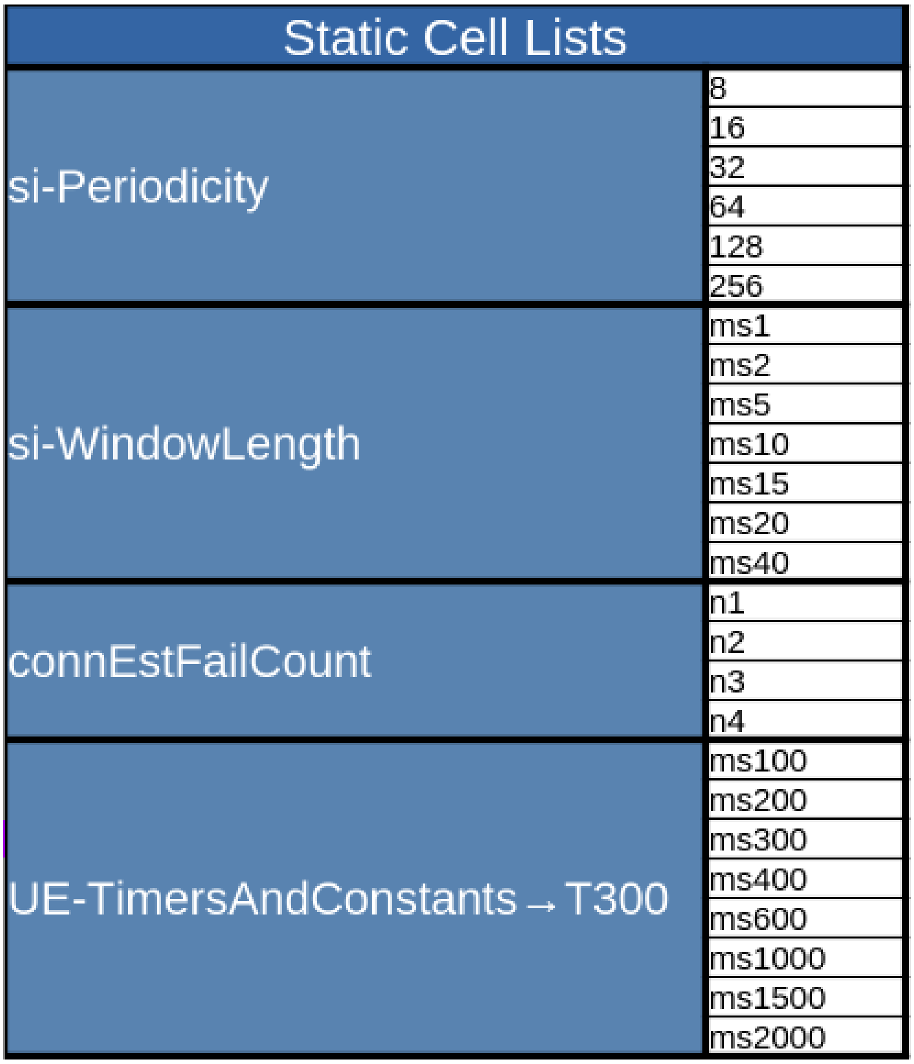
    * Cell Number/# - Numeric identifier of an Intrafrequency Neighbor Cell List Entry.
    * Physical Cell Identity (PCI) - The Primary Synchronization Signal (PSS) and Secondary Synchronization Signal (SSS) from the cell's Synchronization Signal Block (SSB), of which the UE is able to calculate the Physical Cell Identity (PCI). Formula: $PCI = 3(PSS) + SSS$

  - Interfrequency Neighbor PCI List - The inter-frequency neighbor list contains PCIs of all inter-frequency cells that are registered neighbors with the current cell whose integrity we're verifying.

    * Cell Number# - Numeric identifier of an Interfrequency Neighbor Cell List Entry.
    * Physical Cell Identity (PCI) - The Primary Synchronization Signal (PSS) and Secondary Synchronization Signal (SSS) from the cell's Synchronization Signal Block (SSB), of which the UE is able to calculate the Physical Cell Identity (PCI). Formula: $PCI = 3(PSS) + SSS$

5.1.2 Static Cell Lists

Multiple sets of static values corresponding to store enumerated values from the 3GPP standards.



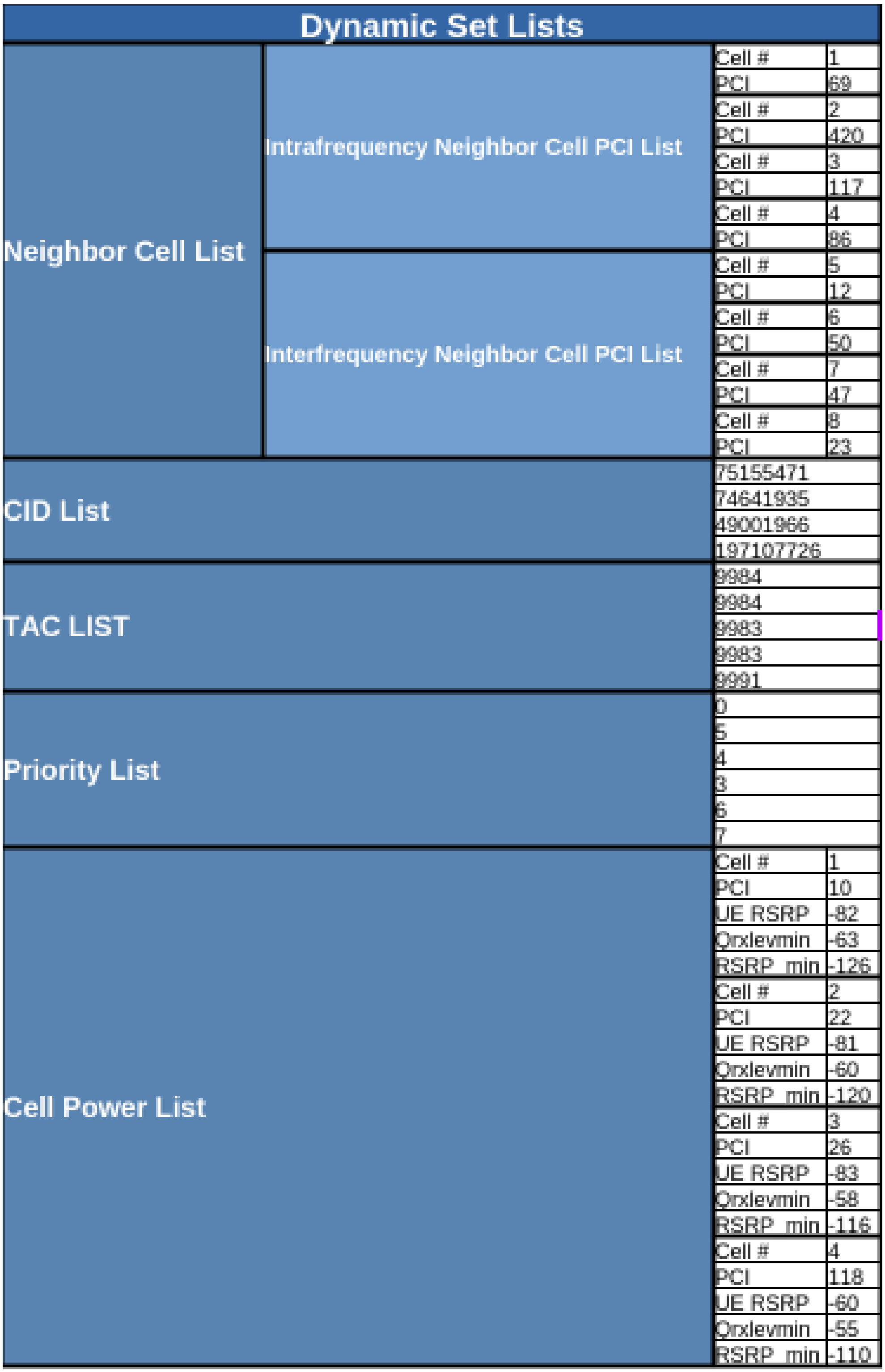
Static Cell Lists	
si-Periodicity	8
	16
	32
	64
	128
	256
si-WindowLength	ms1
	ms2
	ms5
	ms10
	ms15
	ms20
	ms40
connEstFailCount	n1
	n2
	n3
	n4
UE-TimersAndConstants → T300	ms100
	ms200
	ms300
	ms400
	ms600
	ms1000
	ms1500
	ms2000

Figure 5.1.2-a. Static Cell Lists

- **si-Periodicity List** - Valid enumerated values from the 3GPP 4G LTE & 5G NR Standards.
- **si-WindowLength List** - Valid enumerated values from the 3GPP 4G LTE & 5G NR Standards.
- **connEstFailCount List** - Valid enumerated values from the 3GPP 4G LTE & 5G NR Standards.
- **UE-TimersAndConstants/T300** - Valid enumerated values from the 3GPP 4G LTE & 5G NR Standards.

5.1.3 Dynamic Set Lists

Dynamic Set Lists reflect the current neighboring cells, priority parameters, and cell signal power for each cell that is currently a suitable candidate. Data within these lists are obtained through entries in the *Dynamic Cell List*, and will change according to the information and amount of Cells stored as entries. As suitable candidate cells are no longer feasible, the entry is removed from the *Dynamic Cell List Entries* and subsequently relevant information is removed from the varying Dynamic Set Lists correspondent to their respective PCIs, with the exception of the *Priority List*, as it's merely a collection of the various *cellReselectionPriority* values present throughout all *Dynamic Cell List* entries.



Dynamic Set Lists			
Neighbor Cell List	Intrafrequency Neighbor Cell PCI List	Cell #	1
		PCI	69
		Cell #	2
		PCI	420
		Cell #	3
		PCI	117
		Cell #	4
		PCI	86
	Interfrequency Neighbor Cell PCI List	Cell #	5
		PCI	12
		Cell #	6
		PCI	50
		Cell #	7
		PCI	47
		Cell #	8
		PCI	23
CID List		75155471	
		74641935	
		49001966	
		197107726	
TAC LIST		9984	
		9984	
		9983	
		9983	
		9991	
Priority List		0	
		5	
		4	
		3	
		6	
		7	
Cell Power List		Cell #	1
		PCI	10
		UE RSRP	-82
		Qrxlevmin	-63
		RSRP_min	-126
		Cell #	2
		PCI	22
		UE RSRP	-81
		Qrxlevmin	-60
		RSRP_min	-120
		Cell #	3
		PCI	26
		UE RSRP	-83
		Qrxlevmin	-58
		RSRP_min	-116
		Cell #	4
		PCI	118
		UE RSRP	-60
		Qrxlevmin	-55
		RSRP_min	-110

Figure 5.1.3-a. Dynamic Set Lists

- **Neighbor Cell List** – Contains the PCIs of current intra- and inter-frequency neighbor cells.
    - *Intrafrequency Neighbor Cell PCI List* – Intrafrequency neighbor PCIs.
        - **Cell Number/#** – Numeric identifier of list entry.
        - **Physical Cell Identity (PCI)** – PCI for the neighbor entry.
    - *Interfrequency Neighbor Cell PCI List* – Interfrequency neighbor PCIs.
        - **Cell Number/#** – Numeric identifier of list entry.
        - **Physical Cell Identity (PCI)** – PCI for the neighbor entry.
- **CID List** – Cell Identifiers (ECI/NCI) for all current suitable candidate cells.
- **TAC List** – Tracking Area Codes for all current suitable candidate cells.
- **Priority List** – *cellReselectionPriority* values drawn from current *Dynamic Cell List Entries*.
- **Cell Power List** – Per-cell snapshot used by power-related VCs (*spVer*, *rVer*).
    - **Cell Number/#** – Numeric identifier linking to the corresponding *Dynamic Cell List Entry*.
    - **Physical Cell Identity (PCI)** – PCI of the candidate cell.
    - **UE_RSRP (dBm)** – Latest filtered downlink reference-signal received power measured by the UE.
    - **q-RxLevMin** (Q_{rxlevmin}) – Advertised minimum received level from SIB1 (integer in 2 dB steps).
    - $RSRP_{\mathrm{min}}$ **(dBm)** – Derived minimum acceptable RSRP computed from Q_{rxlevmin}:
    $$RSRP_{\mathrm{min}} \;=\; 2 \times Q_{\mathrm{rxlevmin}}$$
    - **(Optional) Timestamp / EWMA state** – *last_update*, and if maintained for *spVer*, μ_t (mean) and v_t (variance).

5.2 Verification Conditions (VC)

5.2.1 Scheduling Verification

Ensuring that SIB scheduling IEs correspond with known enumerated values.

5.2.2 Timing Verification

Ensuring timing values correspond with known enumerated values.

5.2.3 Duplication Verification

Ensures a duplicate of cell doesn't exist.

5.2.4 Location Verification

Ensures the TAC of current cell matches already known TACs present in the *Dynamic Set Lists*' TAC List.

5.2.5 Priority Verification

Detects anomalously high `cellReselectionPriority` values relative to local peers. Such elevation is treated as suspicious and contributes to the aggregated suspicion score. Higher-than-median priorities produce proportional deviations, and strong elevation can participate in combinatorial escalation.

5.2.6 Neighborhood Verification

Ensuring the cell's neighboring cells correspond with neighbor cells from current suitable candidate cells.

5.2.7 Signal Power Verification

Ensure a cell's signal power isn't abnormally high when compared to those of other suitable candidate cells.

5.2.8 Minimum Required Signal Strength Verification

The lowest RSRP level (in dBm) that a cell advertises it will accept before a UE will camp on it. RSRP represents the mean power level of the specific reference signal resource elements, and is used by the UE for tasks such as cell selection, reselection, and handover decisions.

5.3 Verification Algorithm (VA)

This computation utilizes the nine *Verification Conditions* (VCs) to produce a continuously updated suspicion score for each candidate cell, apply combinatorial and immediate override logic, and manage the cell status state machine. The VCs are evaluated in parallel; their normalized deviations are weighted, accumulated with exponential decay, and used to drive escalation conservatively while still enabling adaptive sensitivity.

5.3.1 Policy Summary

The high-level design choices embodied in the VA are:

- **Conservative escalation:** Cells require persistent or multi-faceted evidence before being barred to keep false positives low.
- **Adaptive thresholds:** Statistical checks, especially signal-power deviations, use adaptive estimators so that noisy environments do not cause overreaction.
- **Automatic probation:** After the barred interval expires, cells enter probation for controlled reassessment.
- **Weighted and combinatorial severity:** Each VC contributes a normalized deviation; certain combinations of deviations amplify escalation or trigger immediate overrides.
- **Exponential decay:** Suspicion scores fade over time so that transient anomalies are forgotten unless they recur in the same geographic area.

5.3.2 Default Weighting Strategy

Each Verification Condition i produces a normalized deviation $d_i \in [0, 1]$. The system multiplies each by a weight w_i to reflect its baseline severity. A suggested initial hierarchy (tunable per deployment) is:

- **High weight:** Duplicate identity (`dVer`), neighbor inconsistencies (`nVer`) — strong indicators of rogue configuration.
- **Medium weight:** Priority anomalies (`pVer`), signal power deviations (`spVer`) — potentially legitimate under some conditions but dangerous in combination.
- **Lower weight:** Scheduling and timing anomalies (`sVer`, `tVer`) — useful early indicators but more likely to be transient or environment-driven.
- **Adjustments:** Combinatorial patterns (e.g., high priority + location mismatch) can apply multipliers or bonus increments above base weight accumulation.

Weights do not strictly have to sum to 1; implementers can normalize after tuning. Example starting values: $w_{\text{dVer}} = 1.5$, $w_{\text{nVer}} = 1.2$, $w_{\text{pVer}} = 1.0$, $w_{\text{spVer}} = 1.0$, $w_{\text{qVer}} = 1.0$, $w_{\text{rVer}} = 1.2$, $w_{\text{sVer}} = 0.7$, $w_{\text{tVer}} = 0.7$.

5.3.3 Detailed Mechanisms

Evaluation of Verification Conditions On each evaluation tick, all nine VCs are evaluated to obtain deviations:

$$d_{\text{sVer}}, d_{\text{tVer}}, d_{\text{dVer}}, d_{\text{lVer}}, d_{\text{pVer}}, d_{\text{nVer}}, d_{\text{spVer}}, d_{\text{qVer}}, d_{\text{rVer}}.$$

Each old suspicion decays by half every T_{half} seconds, then new deviations are added:

$$S \leftarrow S \cdot 2^{-\Delta t / T_{\text{half}}} + \sum_i w_i \, d_i$$

where T_{half} is the chosen half-life, Δt is the time since the last update, and S is the suspicion score.

Adaptive Threshold for Signal Power Signal-power deviations use the adaptive threshold described in the `spVer` VC definitions: exponential weighted moving averages track mean and variance, and the effective Z-threshold is scaled by the coefficient of variation to avoid overreacting in noisy environments.

Combination Overrides Certain combinations of VC outputs are treated as high-confidence rogue patterns and either amplify ΔS or bypass normal accumulation:

- **Immediate bar:**
  - *Duplicate identity* $\wedge$ *no neighbors advertised*: a cell impersonating another with no neighbor graph is almost certainly malicious.
  - *Out-of-range q-RxLevMin* $\wedge$ *duplicate identity*: a cell advertising an implausibly permissive threshold while spoofing identity.
  - *Rapid q-RxLevMin toggling* $\wedge$ *spVer failure*: unstable threshold announcements combined with abnormal signal deviation.
  - *TAC mismatch between MIB and SIB1* $\wedge$ *neighbor TACs differ*: inconsistent tracking area codes indicate mis-configuration or spoofing.
  - *Missing mandatory SIB* (e.g. no SIB2 when expected) $\wedge$ *timing anomaly*: omission of required information plus invalid timer values is highly suspicious.

- **Escalation bonuses:**
  - *High-priority flag* $\wedge$ *geographic mismatch*: elevated priority combined with cell location inconsistency.
  - *Extreme SIB periodicity outlier* $\wedge$ *low si-WindowLength*: abnormal broadcast schedule that deviates from standard timing.
  - *Duplicate PCI in neighbor list* $\wedge$ *abnormal neighbor–overlap*: PCI collisions without the expected shared neighbor graph.
  - *spVer failure* $\wedge$ *low T300*: signal power anomaly combined with too-short connection timeout—a signal/timing conflict.

State Transitions and Parameters Cells move through nominal states:

$$\textbf{Clean} \rightarrow \textbf{Suspect} \rightarrow \textbf{Barred} \rightarrow \textbf{Probation} \rightarrow \textbf{Clean}.$$

Suggested default parameters are summarized in Table 2.

Barred duration backoff formula:

$$T_{\text{barred}}(N) = \min\left(T_{\text{barred,base}} \cdot 2^{N-1},\ T_{\text{barred,max}}\right)$$

where N is the recent bar count for the cell.

Pseudocode

```
function evaluate_cell(candidate_cell, now):
    # 1. Decay existing suspicion (half-life decay)
    dt = now - candidate_cell.last_update
    candidate_cell.S *= 2 ** (-dt / T_half)
    candidate_cell.last_update = now

    # 2. Compute normalized deviations from each soft VC
    d_sVer  = compute_sVer_deviation(candidate_cell)
    d_tVer  = compute_tVer_deviation(candidate_cell)
    d_dVer  = compute_dVer_deviation(candidate_cell)
    d_lVer  = compute_lVer_deviation(candidate_cell)
    d_pVer  = compute_pVer_deviation(candidate_cell)
    d_nVer  = compute_nVer_deviation(candidate_cell)
    d_spVer = compute_spVer_deviation(candidate_cell)
    d_qVer  = compute_qVer_deviation(candidate_cell)
    d_rVer  = compute_rVer_deviation(candidate_cell)

    # 3. Aggregate with weights
    delta_S = (w_sVer * d_sVer +
               w_tVer * d_tVer +
               w_dVer * d_dVer +
               w_lVer * d_lVer +
               w_pVer * d_pVer +
               w_nVer * d_nVer +
               w_spVer * d_spVer +
               w_qVer * d_qVer +
               w_rVer * d_rVer)

    # 4. High-severity combinatorial adjustments (examples)
    if inconsistent_priority_and_location(candidate_cell):
        delta_S *= (1 + combo_priority_location_boost)
    if rapid_scheduling_oscillation_and_timing(candidate_cell):
        adjust_local_persistence_window(candidate_cell)
    if diverse_failure_cluster(candidate_cell):
        delta_S += cluster_bonus

    # 5. Immediate override checks
    if duplicate_identity(candidate_cell) and no_neighbors_advertised(candidate_
        escalate_to_barred(candidate_cell)
        return
    if d_qVer.range_check_failed and duplicate_identity(candidate_cell):
        escalate_to_barred(candidate_cell)
        return
    if sib_integrity_failure_with_missing_mandatory(candidate_cell):
        escalate_to_barred(candidate_cell)
        return
    # NEW: rVer hard fault — measured RSRP far below advertised min while admitt
    if rver_hard_violation(candidate_cell):  # implements R <= Q_rxlevmin - (M +
        escalate_to_barred(candidate_cell)
        return

    # 6. Update suspicion score
    candidate_cell.S += delta_S

    # 7. State transitions
    switch candidate_cell.state:
        case CLEAN:
            if candidate_cell.S >= theta_suspect:
                candidate_cell.state = SUSPECT
                start_suspect_timer(candidate_cell)
        case SUSPECT:
            if candidate_cell.S >= theta_barred for duration W2:
                candidate_cell.state = BARRED
                candidate_cell.recent_bar_count += 1
                candidate_cell.barred_expiry = now + compute_barred_duration(can
            else if candidate_cell.S < theta_clear:
                candidate_cell.state = CLEAN
        case BARRED:
            if now >= candidate_cell.barred_expiry:
                candidate_cell.state = PROBATION
                candidate_cell.probation_expiry = now + T_probation
                candidate_cell.clean_streak = 0
        case PROBATION:
            if passes_all_VCs(candidate_cell):
                candidate_cell.clean_streak += 1
                if candidate_cell.clean_streak >= m:
                    candidate_cell.state = CLEAN
                    candidate_cell.S = 0
            else:
                candidate_cell.state = BARRED
                candidate_cell.recent_bar_count += 1
                candidate_cell.barred_expiry = now + compute_barred_duration(can
```

5.3.4 Verification Algorithm Summary

This mechanism balances conservative escalation with adaptive sensitivity. Short-lived or low-confidence anomalies decay naturally. Persistent deviations, severe combinations, or repeated misbehavior lead to escalating consequences, with probation offering a controlled recovery path.

6 Processes

6.1 Onboard Cell-Selection Storage (OCS)

As a new cell undergoes selection procedures on the UE, data from the cell is stored in an ephemeral portion in the OCS block. If the cell passes cell selection muster, then this ephemeral data persists as an entry to the *Dynamic Cell List*. If the cell fails selection, then the temporary data is dropped. This process is repeated for each newly detected cell undergoing selection.

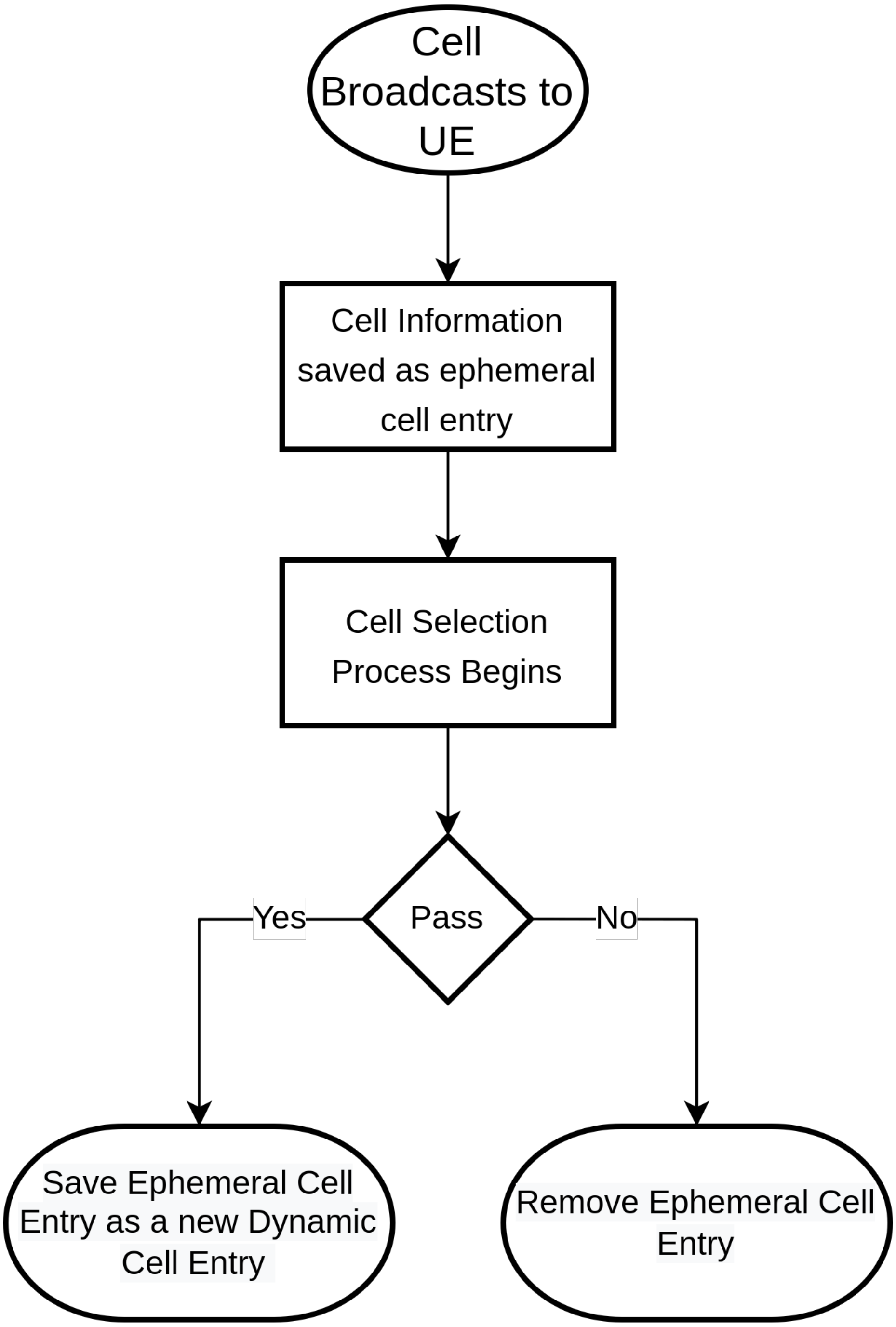

Cell Broadcasts to UE
Cell Information saved as ephemeral cell entry
Cell Selection Process Begins
Pass
Yes
No
Save Ephemeral Cell Entry as a new Dynamic Cell Entry
Remove Ephemeral Cell Entry

Figure 6.1-a. OCS Flow

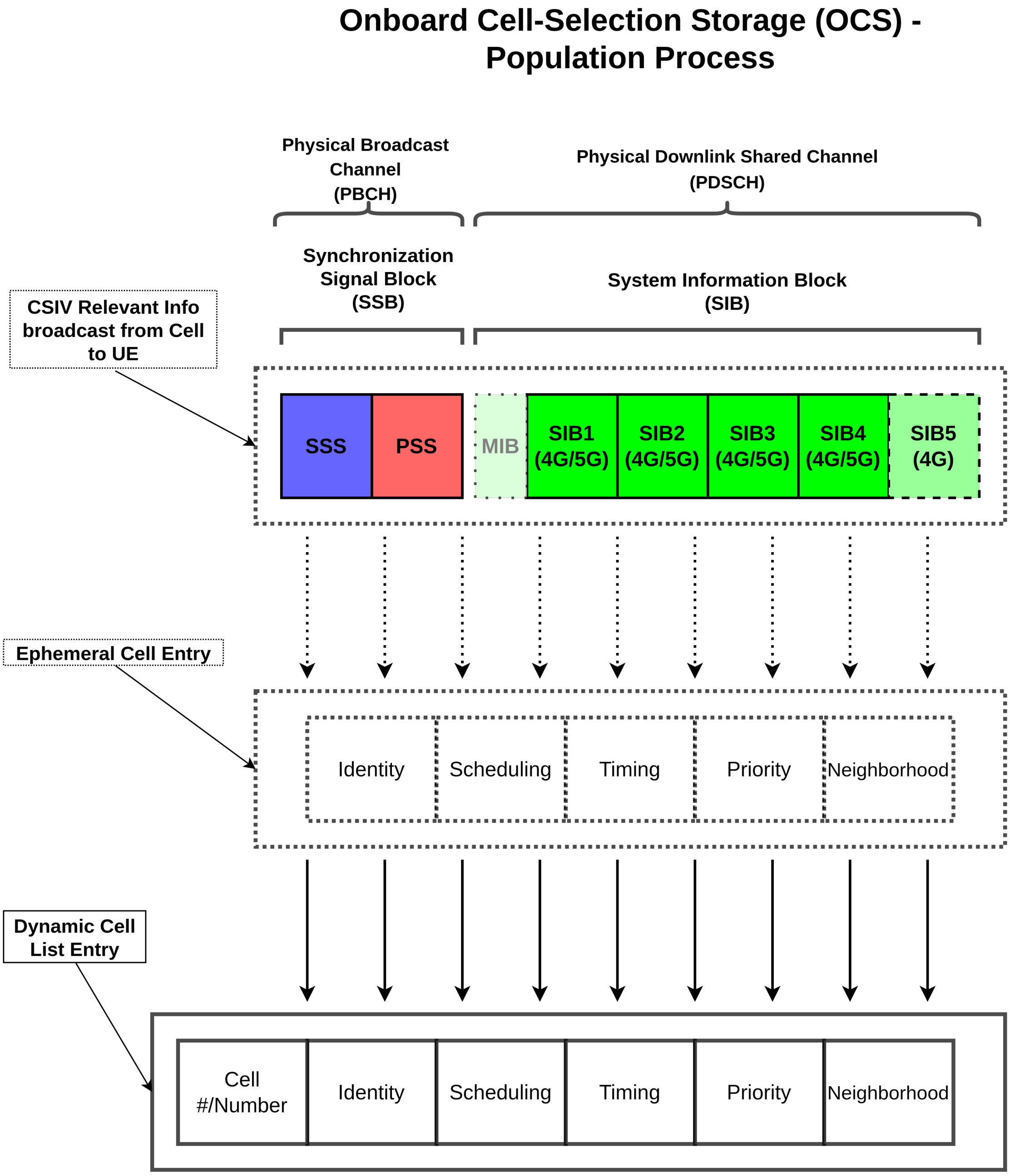


Figure 6.1-b. OCS Process (Cell Selection Succeeds)

6.2 Verification Conditions

6.2.1 Scheduling Verification (sVer)

Ensuring that SIB scheduling IEs correspond with known enumerated values.

Variables

Variable	Description	OBS Mapping
sip {...}	List of current *si-Periodicity* values in the Dynamic Cell List Entry.	*Dynamic Cell List Entry->Scheduling->SIB List->si-Periodicity*
siw	The *si-WindowLength* of the cell's scheduling information.	*Dynamic Cell List Entry->Scheduling->si-WindowLength*
pe	Enumerated valid *si-Periodicity* values from the standards.	*Static Cell Lists->si-Periodicity List*
we	Enumerated valid *si-WindowLength* values from the standards.	*Static Cell Lists->si-WindowLength List*

Verification Cases

- **sVc1** - Every reported periodicity is shorter than the scheduling window.
  Operation:
  - If $\forall p \in$ sip : $p <$ siw: sVc1 = True.
  - Else: sVc1 = False.

  Formula:

  $$sVc1 = \begin{cases} 1 & \text{if } \forall p \in \text{sip} : \ p < \text{siw} \\ 0 & \text{otherwise} \end{cases}$$

- **sVc2** - All periodicities are valid enumerated values.
  Operation:
  - If sip $\subseteq pe$: sVc2 = True.
  - Else: sVc2 = False.

Formula:

$$sVc2 = \begin{cases} 1 & \text{if sip} \subseteq pe \\ 0 & \text{otherwise} \end{cases}$$

- **sVc3** - The scheduling window length is a valid enumerated value.
  Operation:
  - If siw $\in$ *we*: sVc3 = True.
  - Else: sVc3 = False.

  Formula:

$$sVc3 = \begin{cases} 1 & \text{if siw} \in we \\ 0 & \text{otherwise} \end{cases}$$

sVer Formula:

$$sVer = \begin{cases} 1 & \text{if } sVc1 = sVc2 = sVc3 = 1 \\ 0 & \text{otherwise} \end{cases}$$

Composite deviation:

$$d_s = 1 - sVer$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\text{sVer}} = w_{\text{sVer}} \cdot d_s$$

6.2.2 Timing Verification (tVer)

Ensuring timing values correspond with known enumerated values.

Variables

Variable	**Description**	**OBS Mapping**
cfc	Value of *connEstFailCount*.	*Dynamic Cell List Entry->Timing->connEstFailCount*
t300	Value of *T300*.	*Dynamic Cell List Entry->Timing->UE-TimersAndConstants/T300*
ce	Enumerated valid *connEstFailCount* values.	*Static Cell Lists->connEstFailCount List*
te	Enumerated valid *T300* values.	*Static Cell Lists->UE-TimersAndConstants/T300*

Verification Cases

- **tVc1** - The reported *connEstFailCount* is a valid enumerated value.
  Operation:
  - If $cfc \in ce$: tVc1 = True.
  - Else: tVc1 = False.

  Formula:

$$tVc1 = \begin{cases} 1 & \text{if } cfc \in ce \\ 0 & \text{otherwise} \end{cases}$$

- **tVc2** - The reported *T300* is a valid enumerated value.
  Operation:

  - If $t300 \in te$: tVc2 = True.
  - Else: tVc2 = False.

Formula:

$$tVc2 = \begin{cases} 1 & \text{if } t300 \in te \\ 0 & \text{otherwise} \end{cases}$$

- **tVc3** - The *connEstFailCount* is not anomalously high among current candidates.
  Operation:
  - If $cfc \neq \max(ce)$: tVc3 = True.
  - Else: tVc3 = False.

  Formula:

$$tVc3 = \begin{cases} 1 & \text{if } cfc \neq \max(ce) \\ 0 & \text{otherwise} \end{cases}$$

- **tVc4** - The *T300* value is not anomalously high among current candidates.
  Operation:
  - If $t300 \neq \max(te)$: tVc4 = True.
  - Else: tVc4 = False.

  Formula:

$$tVc4 = \begin{cases} 1 & \text{if } t300 \neq \max(te) \\ 0 & \text{otherwise} \end{cases}$$

tVer Formula:

$$tVer = \begin{cases} 1 & \text{if } tVc1 = tVc2 = tVc3 = tVc4 = 1 \\ 0 & \text{otherwise} \end{cases}$$

Composite deviation:

$$d_t = 1 - tVer$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\text{tVer}} = w_{\text{tVer}} \cdot d_t$$

6.2.3 Duplication Verification (dVer)

Ensures a duplicate of cell doesn't exist.

Variables

Variable	**Description**	*OBS Mapping*
cid	The cell's Cell Identifier (ECI or NCI)	*Dynamic Cell List Entry->Identity->Cell Identifier*
pci	The cell's Physical Cell Identity	*Dynamic Cell List Entry->Identity->Physical Cell Identifier*
npl {...}	List of neighboring cells' PCIs.	*Dynamic Set Lists->Neighbor Cell List->Intrafrequency Neighbor Cell PCI List*; *Dynamic Set Lists->Neighbor Cell PCI List->Interfrequency Neighbor Cell PCI List*
cl {...}	List of neighboring cells' CIDs.	*Dynamic Set Lists->Neighbor Cell List->CID List*

Verification Cases

- **dVc1** - Ensuring the Cell Identifier isn't already present in current list of Cell Identifiers.
  Operation:
  - If cid is not in cl:

    - * dVc1 = True.
  - Else:
    - * dVc1 = False

  Formula:

$$dVc1 = \begin{cases} 1 & cid \notin cl \\ 0 & else \end{cases}$$

- **dVc2** - Ensuring the Physical Cell Identity isn't already present in current list of Physical Cell Identities.

  Operation:

  - If pci is not in npl:
    - * dVc2 = True.
  - Else:
    - * dVc2 = False

  Formula:

$$dVc2 = \begin{cases} 1 & pci \notin npl \\ 0 & else \end{cases}$$

dVer Formula:

$$dVer = \begin{cases} 1 & dVc1 = True \,\&\, dVc2 = True \\ 0 & else \end{cases}$$

Composite deviation:

$$d_d = 1 - dVer$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\text{dVer}} = w_{\text{dVer}} \cdot d_d$$

6.2.4 Location Verification (lVer)

Ensures the TAC of current cell matches known TACs in the *Dynamic Set Lists*' TAC List.

Variables

Variable	Description	OBS Mapping
tac	A cell's Tracking Area Code.	*Dynamic Cell List Entry->Identity->Tracking Area Code*
tl {...}	List of TACs for current suitable candidate cells.	*Dynamic Set Lists->TAC List*

Verification Case

- **lVc1** - Cell's TAC is present in the known TAC list.
  Operation:
  - If $tac \in tl$: lVc1 = True.
  - Else: lVc1 = False.

  Formula:

$$lVc1 = \begin{cases} 1 & \text{if } tac \in tl \\ 0 & \text{otherwise} \end{cases}$$

lVer Formula:

$$lVer = \begin{cases} 1 & \text{if } lVc1 = 1 \\ 0 & \text{otherwise} \end{cases}$$

Composite deviation:

$$d_l = 1 - lVer$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\mathrm{IVer}} = w_{\mathrm{IVer}} \cdot d_I$$

6.2.5 Priority Verification (pVer)

Detects anomalously high `cellReselectionPriority` values that deviate above the local norm, as such escalation is a potential indicator of malicious influence.

Variables

Variable	**Description**	**OBS Mapping**
crp	The candidate's *cellReselectionPriority*.	*Dynamic Cell List Entry->Priority->cellReselectionPriority*
neighbor_priorities	The set of *cellReselectionPriority* values for other current suitable candidate cells.	*Dynamic Set Lists->Priority List*
median_prio	The median of *neighbor_priorities*.	Derived
Δ_{priority}	Tuning threshold for flagging strong elevation.	Default: 1

Verification Cases

- **Priority deviation** - Elevated priority relative to peers.
  Operation:
  - Compute median_prio = median(neighbor_priorities).
  - If $crp >$ median_prio:

$$d_{\text{pVer}} = \frac{crp - \text{median\_prio}}{7 - \text{median\_prio}}$$

  - Else: $d_{\text{pVer}} = 0$.

- **High priority flag** - Strong elevation for combinatorial escalation.
  Operation:
  - If $crp - \text{median\_prio} \geq \Delta_{\text{priority}}$: high_priority_flag = true.
  - Else: high_priority_flag = false.

pVer Output:
The priority deviation $d_{\text{pVer}} \in [0, 1]$ is used as a weighted contribution to the suspicion score. The `high_priority_flag` can trigger combinatorial elevation when present alongside other significant anomalies.

Composite deviation:

$$d_p = d_{\text{pVer}}$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\text{pVer}} = w_{\text{pVer}} \cdot d_p$$

6.2.6 Neighborhood Verification (nVer)

Ensuring the cell’s neighboring information is consistent with current suitable candidate cells.

Variables

Variable	Description	OBS Mapping
cipl	Intrafrequency neighbor PCIs advertised by the candidate cell.	*Dynamic Cell List Entry->Neighborhood->Intrafrequency Neighbor PCI List*
copl	Interfrequency neighbor PCIs advertised by the candidate cell.	*Dynamic Cell List Entry->Neighborhood->Interfrequency Neighbor PCI List*
dipl	Trusted intrafrequency neighbor PCI list from current suitable candidate cells.	*Dynamic Set Lists->Neighbor Cell List->Intrafrequency Neighbor Cell PCI List*
dopl	Trusted interfrequency neighbor PCI list from current suitable candidate cells.	*Dynamic Set Lists->Neighbor Cell List->Interfrequency Neighbor Cell PCI List*
τ_1	Intersection threshold for intrafrequency lists.	Default: 2
τ_2	Intersection threshold for interfrequency lists.	Default: 2

Verification Cases

- **Intrafrequency consistency:** Candidate advertises intrafre-

quency neighbors and shares sufficient overlap with trusted intrafrequency neighbor list.
Condition: $cipl \neq \emptyset$ and $|cipl \cap dipl| \geq \tau_1$.

- **Interfrequency consistency:** Candidate advertises interfrequency neighbors and shares sufficient overlap with trusted interfrequency neighbor list.
Condition: $copl \neq \emptyset$ and $|copl \cap dopl| \geq \tau_2$.

nVer Formula:

$$nVer = \begin{cases} 1 & \text{if } (cipl \neq \emptyset \wedge |cipl \cap dipl| \geq \tau_1) \\ 1 & \text{if } (copl \neq \emptyset \wedge |copl \cap dopl| \geq \tau_2) \\ 0 & \text{otherwise} \end{cases}$$

Composite deviation:

$$d_n = 1 - nVer$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\text{nVer}} = w_{\text{nVer}} \cdot d_n$$

6.2.7 Signal Power Verification (spVer)

Ensure a cell's signal power deviation is statistically justified given the local variability; abnormal deviations increase suspicion.

Variables

Variable	**Description**	**OBS Mapping**
x_t	Current measured post-selection signal power (e.g., *RSRP*).	*Dynamic Set Lists->Cell Power List*
μ_{t-1}	Previous exponential weighted moving average of signal power mean.	Maintained state
v_{t-1}	Previous exponential weighted moving average of variance.	Maintained state
β	EWMA smoothing factor for mean/variance updates.	Default: 0.1
z_{base}	Base Z threshold.	Default: 2.0
α_{cv}	Coefficient of variation scaling factor.	Default: 0.5
ε	Small constant to avoid division by zero.	Default: 10^{-6}

Update Equations (maintained per cell):

$$\mu_t = (1-\beta)\mu_{t-1} + \beta x_t$$

$$v_t = (1-\beta)v_{t-1} + \beta(x_t - \mu_t)^2$$

$$\sigma = \sqrt{\max(v_t, \varepsilon)}$$

$$z = \frac{|x_t - \mu_t|}{\sigma}$$

$$\text{cv} = \frac{\sigma}{\max(\mu_t, \varepsilon)}$$

$$z_{\text{threshold}} = z_{\text{base}} \cdot (1 + \alpha_{\text{cv}} \cdot \text{cv})$$

Verification Case

- **spVc1** - Signal deviation is within adaptive tolerance.
  Operation:
  - If $z \leq z_{\text{threshold}}$: spVc1 = True.
  - Else: spVc1 = False.

  Formula:

$$spVc1 = \begin{cases} 1 & \text{if } z \leq z_{\text{threshold}} \\ 0 & \text{otherwise} \end{cases}$$

spVer Formula:

$$spVer = \begin{cases} 1 & \text{if } spVc1 = 1 \\ 0 & \text{otherwise} \end{cases}$$

Composite deviation:

$$d_{\text{sp}} = 1 - spVc1$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_{\text{sp}} = w_{\text{spVer}} \cdot d_{\text{sp}}$$

6.2.8 Minimum Required Signal Strength Verification (qVer)

Check the broadcast q-RxLevMin (from SIB1) for plausibility and local consistency to catch cells that abuse overly-permissive thresholds.

Variables

Variable	Description	Default
Q_{rxlevmin}	Advertised q-RxLevMin (in dBm).	—
$Q_{\text{min}}, Q_{\text{max}}$	Allowed q-RxLevMin range.	-130 dBm, -100 dBm
$\text{median}_{\text{nb}}$	Median q-RxLevMin of trusted neighbors.	Computed
Δ_{ref}	Reference deviation scale.	6dB
δ	Neighborhood tolerance (normalized).	1.0
w_q	Weight of q-RxLevMin deviation.	1.0

Checks:

- **qVc1: Reasonableness.**

$$qVc1 = \begin{cases} 1 & Q_{\text{min}} \leq Q_{\text{rxlevmin}} \leq Q_{\text{max}} \\ 0 & \text{otherwise} \end{cases}$$

- **qVc2: Neighborhood consistency.** Compute

$$d_{q2} = \min\left(1, \frac{\left|Q_{\text{rxlevmin}} - \text{median}_{\text{nb}}\right|}{\Delta_{\text{ref}}}\right).$$

Then

$$qVc2 = \begin{cases} 1 & d_{q2} \leq \delta \\ 0 & \text{otherwise} \end{cases}$$

Composite deviation:

$$d_q = \max\left(1 - qVc1,\ d_{q2}\right)$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_q = w_q \cdot d_q$$

Optionally, treat $\left(qVc1 = 0\right)$ paired with a high-confidence anomaly as an immediate override to **Barred**.

6.2.9 Received Level Consistency (rVer)

Check that the UE's measured received level is consistent with the cell's advertised minimum (q-RxLevMin) and not below a global plausibility floor.

Variables

Variable	Description	Default
R	Current measured RSRP (dBm).	—
Q_{rxlevmin}	Advertised minimum from SIB1, converted to dBm ($RSRP_{\text{min}} = 2 \cdot Q_{\text{rxlevmin}}$).	—
M	Tolerance margin for measurement/filters.	3 dB
Q_{floor}	Global plausibility floor for camping.	-130 dBm
Δ_{scale}	Normalization scale for deficits.	6 dB
w_{rVer}	Weight of rVer deviation.	1.0

Checks:

- **rVc1: Operational consistency.**
  Let $Q_{\text{min}}^{\text{eff}} = Q_{\text{rxlevmin}} - M$. Compute a normalized deficit:

$$d_{r1} = \min\left(1, \max\left(0, \frac{Q_{\text{min}}^{\text{eff}} - R}{\Delta_{\text{scale}}}\right)\right).$$

- **rVc2: Plausibility floor.**
  The measured RSRP should not be unrealistically low for a usable serving cell:

$$d_{r2} = \min\left(1, \max\left(0, \frac{Q_{\text{floor}} - R}{\Delta_{\text{scale}}}\right)\right).$$

Composite deviation:

$$d_r = \max\bigl(d_{r1}, d_{r2}\bigr)$$

Integration (S contribution): Add to the overall suspicion score:

$$\Delta S_r = w_{\mathrm{rVer}} \cdot d_r$$

Immediate override (optional): If $R \le Q_{\mathrm{rxlevmin}} - \bigl(M + 10\,\mathrm{dB}\bigr)$ while the cell still admits or holds the UE, treat as an integrity fault and transition to **Barred**.

6.3 Verification Algorithm

6.3.1 VA Process

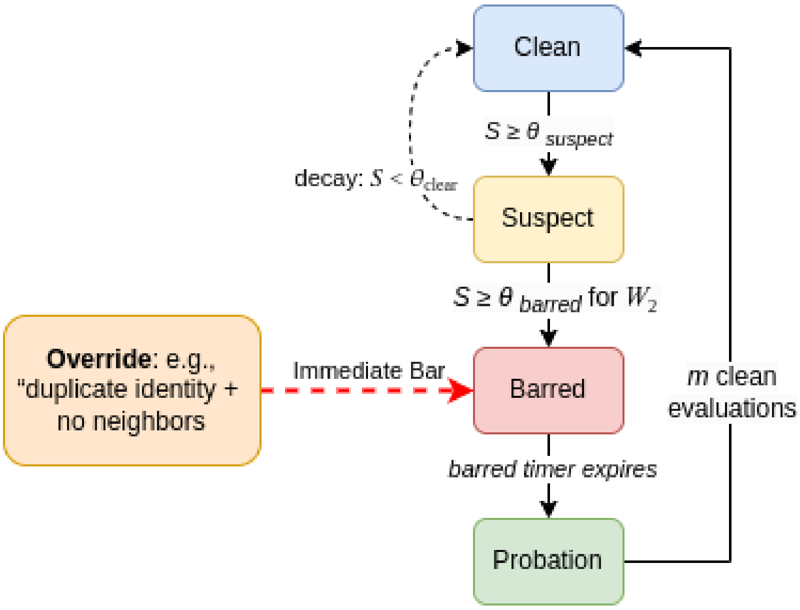

Clean
$S \geq \theta_{suspect}$
decay: $S < \theta_{\text{clear}}$
Suspect
$S \geq \theta_{barred}$ for W_2
Override: e.g., "duplicate identity + no neighbors
Immediate Bar
Barred
m clean evaluations
barred timer expires
Probation

Figure 6.3.1-a. CSIV Algorithm Operation Flow

When a new cell passes initial selection criteria, the Verification Algorithm evaluates all nine Verification Conditions in parallel, computes the normalized deviations, aggregates them into a suspicion score with exponential decay, applies combinatorial adjustments and immediate overrides, and updates the cell's state via the state machine described in Section 5.3.

Note: Figure 6.3.1-a illustrates the standard flow of the state machine. Also illustrated is how a detected override condition results in a cell being immediately barred.

Value Function The verification outcome is governed by the aggregated suspicion and state machine; a cell is considered legitimate when it resides in the **Clean** or **Probation** state with acceptable scores, and barred when elevated deviations persist or override conditions are met. Explicit gating of VCs in fixed order is replaced by the weighted accumulation framework to reduce brittleness and provide smoother degradation/recovery behavior.

6.3.2 Initial & Stored-Info Cell Selection

Initial Cell Selection is defined in 3GPP TS 36.304 & 3GPP TS 38.304; CSIV augments both initial and stored-info selection by using previously observed behavior combined with real-time deviations to inform cell legitimacy.

Table 2: Verification Algorithm Tunable Parameters

Parameter	Description	Default
θ_{suspect}	Threshold for entering **Suspect** state (moderate fraction of max aggregated deviation).	0.5 of maximum
θ_{barred}	Sustained elevated suspicion for **Barred**, held over persistence window W_2.	$> \theta_{\text{suspect}}$, W_2 = 30s
θ_{clear}	Suspicion level below which a **Suspect** cell returns to **Clean**.	0.25
T_{half}	Half-life for power-of-two decay of the suspicion score.	60s
$T_{\text{barred,base}}$	Base duration of **Barred** before probation.	5min
$T_{\text{barred,max}}$	Maximum barred duration after backoff.	30min
$T_{\text{probation}}$	Duration of **Probation** after barred expiry.	2min
m	Consecutive clean evaluations during **Probation** to return to **Clean**.	3
w_{sVer}	Weight for Scheduling VC deviation	0.7
w_{tVer}	Weight for Timing VC deviation	0.7
w_{dVer}	Weight for Duplication VC deviation	1.5
w_{lVer}	Weight for Location VC deviation	1.0
w_{pVer}	Weight for Priority VC deviation	1.0
w_{nVer}	Weight for Neighborhood VC deviation	1.2
w_{spVer}	Weight for Signal Power VC deviation	1.0
w_{qVer}	Weight for Min-Signal-Threshold (q-RxLevMin) VC deviation	1.0
w_{rVer}	Weight for Received Level Consistency VC deviation	1.2

7 Appendices

7.1 Appendix A - Tables

7.2 Appendix B - Changelog

Date	Change/Comments	New Version
6/7/2022	Version 0.1 drafted.	Version 0.1
6/17/2022	Correcting figure numbers for figures: • Figure 5.1.1-a. Dynamic Cell List • Figure 5.1.1-b. Dynamic Cell List Entry • Figure 5.1.2-a. Static Cell Lists • Figure 5.1.3-a. Dynamic Set Lists • Figure 6.1-a. OCS Flow • Figure 6.1-b. OCS Process (Cell Selection Succeeds)	Version 0.2

6/17/2022	Modified case 1 of piecewise notation case to include proper parentheses in Section 6.3.2.	Version 0.2
6/18/2022	Correct variable name "za" to "zs" in section 6.2.7.	Version 0.2
6/17/2022	Text edits to Sections: • 5.2.1 • 5.2.2 • 5.2.4 • 6.1	Version 0.2
6/18/2022	New figure images: • Figure 5.1.1-a. Dynamic Cell List • Figure 5.1.1-b. Dynamic Cell List Entry • Figure 5.1.2-a. Static Cell Lists • Figure 5.1.3-a. Dynamic Set Lists	Version 0.2

6/18/2022	Fixed Variables table in Section 6.2.5.	Version 0.2
6/18/2022	Modified width of Example images in Section 5.1.1.	Version 0.2
6/18/2022	Formatting modifications throughout document.	Version 0.2
6/21/2022	Publish Version 1.0.	Version 1.0
12/27/2023	Made following modifications: • Fix Figure 5.1.1-b - "Scheduling" listed twice. Replaced with proper text of "Timing" • Update scope with more accurate language.	Version 1.1
8/3/2025	Major VC & VA overhaul	Version 2.0
8/4/2025	Addition of qVer VC	Version 2.1

8/13/2025	Addition of rVer VC. Removal graphic in former Appendix A. Addition of a "Note" in section 6.3.1. Updated all graphics in section 5.1.	Version 2.2
8/18/2025	Modification of sections 4 & 4.1-4.3.	Version 2.2
9/7/2025	Editorial & grammatical fixes. Updated to latest C6 icon on cover page.	Version 2.3

8 Acknowledgements

This section is dedicated to expressing gratitude of the utmost sincerity to the following individuals for their generosity of time and expertise. Their general review, cellular domain review, sanity checks, and constructive feedback materially improved the clarity and accuracy of this specification:

- **Dr. Rodney LaLonde**
- **Anthony Candarini**
- **Mohamed Karoui**
- **Michael Nash**

Your thoughtful comments and willingness to challenge assumptions made this work stronger. Any remaining errors are mine alone.