POLITECNICO DI TORINO

Corso di Laurea Magistrale
in Ingegneria Matematica

Tesi di Laurea Magistrale

Generatori di numeri aleatori: relazioni tra test di casualità

Relatori
Prof. Danilo Bazzanella
Dott. Guglielmo Morgari (Telsy S.p.A.)
firma dei relatori

. . . . . .
. . . . . .

Candidato
Alessandro Giacchetto

firma del candidato

. . . . . .

Anno Accademico 2021-2022

Sommario

L'elevata richiesta di sequenze casuali in numerose applicazioni nel mondo reale motiva la creazione e l'utilizzo dei generatori di numeri casuali, ossia processi in grado di produrre sequenze che appaiano casuali. Per valutare la correttezza di un generatore vengono utilizzati i test di casualità; si tratta di test statistici che hanno il compito di verificare che determinate caratteristiche attese da una sequenza casuale siano effettivamente presenti nelle sequenze prodotte da un generatore. Al fine di valutare la qualità di un generatore è necessario applicare più test di casualità, in quanto le sequenze prodotte potrebbero deviare in più modi dal comportamento casuale atteso. A seconda del quantitativo di risorse computazionali a disposizione, è necessario individuare l'insieme di test che consenta di controllare quante più caratteristiche possibili. Questa esigenza implica la creazione di suite composte da test tra loro indipendenti.

Le ricerche condotte in questo documento sono volte a studiare le relazioni, in termini di dipendenza, presenti tra i test di casualità della Suite del NIST, la collezione di test maggiormente nota ed utilizzata a livello globale. In contrapposizione a quanto sostenuto dagli autori di questa batteria, sono state osservate diverse coppie di test tra loro dipendenti. In questo lavoro è riportata una trattazione completa sullo studio dell'indipendenza tra tutti i test di casualità appartenenti alla Suite del NIST e alcune analisi approfondite sulle molteplici versioni di determinati test. In aggiunta alle interazioni evidenziate da altri lavori svolti sullo stesso tema, sono state trovate dipendenze che ci risultano originali, le quali pongono le basi per possibili lavori futuri volti ad indagare analiticamente le cause delle relazioni osservate.

Ringraziamenti

Vorrei dedicare questo spazio a coloro che, con dedizione ed instancabile supporto, hanno contribuito alla realizzazione del presente elaborato.

In primo luogo vorrei ringraziare il Professor Danilo Bazzanella per avermi dato la possibilità di realizzare questa tesi ed aver suscitato in me l'interesse verso la crittografia. Vorrei inoltre ringraziare il Dott. Guglielmo Morgari, responsabile della ricerca della società Telsy S.p.A., per il tempo dedicato nei miei confronti. Le sue conoscenze e l'interesse verso questo tema sono state fondamentali per il compimento di questo studio.

Un ringraziamento va poi a tutte le persone che mi sono state vicine in questa fase importante della mia vita e che mi hanno sostenuto ed hanno sempre creduto in me.

Ringrazio i miei genitori e mia sorella Alice per essere sempre stati presenti ed avermi dato la forza per arrivare fino a questo punto. Grazie ai loro insegnamenti sono diventato la persona che sono oggi.

Ringrazio Claudia per essere sempre al mio fianco, pronta a consigliarmi e ad appoggiarmi in ogni mia scelta. È la persona che nei momenti più difficili mi ha dato il coraggio di andare avanti.

Ringrazio tutti i miei amici di infanzia e quelli che ho conosciuto durante questo percorso per i momenti di condivisione e spensieratezza vissuti insieme.

Infine, voglio dedicare questa tesi a nonna Agata, colei che più di tutti vorrei mi vedesse oggi.

Indice

Elenco delle tabelle 6

Elenco delle figure 8

1 Test per generatori di numeri pseudocasuali 13
- 1.1 Casualità . . . 13
- 1.2 Sequenze casuali . . . 14
- 1.3 Generatori di numeri casuali . . . 16
  - 1.3.1 Proprietà di un generatore di numeri pseudocasuali . . . 17
- 1.4 Test di casualità . . . 19
  - 1.4.1 Test di ipotesi . . . 20
  - 1.4.2 Interpretazione dei risultati . . . 21
  - 1.4.3 Batterie di test . . . 23

2 Studio delle relazioni tra test di casualità 27
- 2.1 Concetti di indipendenza . . . 28
- 2.2 Indipendenza tra test di casualità . . . 29
  - 2.2.1 Test indipendenti due a due . . . 30
  - 2.2.2 Test reciprocamente indipendenti . . . 34
  - 2.2.3 Indipendenza tra suite di test . . . 37
- 2.3 Test di casualità disgiunti . . . 39

3 Dipendenze tra test della Suite del NIST 43
- 3.1 Lavori precedenti . . . 44
- 3.2 Metodi di generazione delle sequenze . . . 47
  - 3.2.1 Generatori utilizzati . . . 48
  - 3.2.2 Validazione dei generatori . . . 51
- 3.3 Indipendenza a due a due tra i test della Suite del NIST . . . 54
  - 3.3.1 Suite NIST escludendo Non Overlapping Template Matching, Random Excursions e Random Excursions Variant Test . . . 55
  - 3.3.2 Non Overlapping Template Matching Test . . . 61
  - 3.3.3 Random Excursions Test . . . 63
  - 3.3.4 Random Excursions Variant Test . . . 65
  - 3.3.5 Suite NIST . . . 66

3.4 Indipendenza reciproca tra i test della Suite del NIST 68
3.5 Conclusioni 71

A Impatto del numero di cicli in una sequenza sui test della Suite del NIST 73
A.1 Numero di cicli in una sequenza casuale 73
A.2 Interazione tra il numero di cicli e il Frequency Monobit Test 77
A.3 Interazione tra il numero di cicli e il Cumulative Sums Test 79

B Distribuzione p-value 83
B.1 Correttezza del test sulla proporzione di successi 83

Elenco delle tabelle

1.1 Test di casualità presenti nella suite proposta dal National Institute of Standards and Technology (NIST). . . . 24
1.2 Lunghezza minima delle sequenze richiesta per ciascun test di casualità presente nella Suite del NIST. . . . 25
2.1 Tabella di contingenza tra due test di casualità. . . . 33
2.2 Frequenze teoriche in un test di indipendenza chi quadro, supponendo che i due test di casualità dividano lo spazio delle sequenze con lo stesso livello di significatività α. . . . 33
2.3 Frequenze teoriche in un test di indipendenza chi quadro, supponendo che i due test di casualità dividano lo spazio delle sequenze con due livelli di significatività differenti. . . . 34
2.4 Frequenze teoriche del test di indipendenza chi quadro, tra una suite composta da due test, entrambi con livello di significatività pari ad α_1, interpretata secondo la modalità (1) e un test di casualità con livello di significatività pari ad α_2. . . . 39
2.5 Frequenze teoriche del test di indipendenza chi quadro, tra una suite composta da due test, entrambi con livello di significatività pari ad α_1, interpretata secondo la modalità (2) e un test di casualità con livello di significatività pari ad α_2. . . . 39
2.6 Frequenze teoriche nel caso in cui due test di casualità siano disgiunti e siano condotti con livello di significatività differente. . . . 41
3.1 Coppie di test di casualità dipendenti trovate in [9]. . . . 45
3.2 Coppie di test di casualità dipendenti trovate in [21]. . . . 46
3.3 Coppie di test di casualità dipendenti trovate in [11]. . . . 47
3.4 Sottoinsieme di test di casualità della Suite del NIST, considerato nella Sezione 3.3.1, con i rispettivi parametri utilizzati. . . . 57
3.5 Statistiche relative agli esiti dei test di casualità riportati in Tabella 3.4 applicati ad un insieme di 100.000 sequenze generate tramite l'AES-DRBG. 58
3.6 Coppie composte da test di casualità tra loro dipendenti. . . . 61
3.7 Statistiche relative allo studio sull'indipendenza a due a due condotto tra i test effettuati con il Non Overlapping Template Matching Test implementato con parametro uguale a 9. . . . 62
3.8 Risultati dello studio sull'indipendenza a due a due condotto tra tutti i test appartenenti alla Suite del NIST. . . . 67

3.9 Esempi di suite di test di casualità, composte da test indipendenti a due a due e tali che ciascuno divide lo spazio delle sequenze testate in accordo con il livello di significatività α. 70
3.10 Risultati del test utilizzato per verificare che la coverage stimata dai dati sia statisticamente equivalente alla coverage attesa da una suite composta da test reciprocamente dipendenti. 70
B.1 Risultati del test sulla proporzione di successi applicato agli esiti del Discrete Fourier Transform Test, utilizzando diversi livelli di significatività. 85

Elenco delle figure

2.1 Disposizione delle regioni di accettazione e rifiuto di due test disgiunti (a) e di due test indipendenti (b), con entrambi livello di significatività pari a 1/2. 31
2.2 Rappresentazione della disposizione delle regioni di rifiuto di due test indipendenti nello spazio di tutte le sequenze testate. 32
2.3 Rappresentazione della disposizione delle aree di rifiuto di tre test tra loro indipendenti a due a due, ma non reciprocamente indipendenti. 36
2.4 Area di rifiuto di una suite composta da due test. 38
2.5 Rappresentazione della disposizione delle aree di rifiuto di tre test di casualità disgiunti. 40
3.1 Andamento del p-value relativo al test di indipendenza chi quadrato, effettuato tra il Frequency Test e il Longest Runs of Ones Test, al variare del numero di sequenze considerate. 53
3.2 Andamento del p-value relativo al test di indipendenza chi quadrato, effettuato tra il Frequency Test e il Longest Runs of Ones Test, al variare del numero di sequenze considerate. 54
3.3 Risultati del test di indipendenza chi quadro effettuato tra ogni coppia di test di casualità presenti in Tabella 3.4. 60
3.4 Risultati analisi di indipendenza a due a due tra i test relativi al Non Overlapping Template Matching Test condotto con parametro uguale a 9. . 63
3.5 Risultati dello studio sull'indipendenza a due a due svolta tra gli 8 test relativi al Random Excursions Test. 64
3.6 Risultati dello studio sull'indipendenza a due a due svolta tra i 18 test relativi al Random Excursions Variant Test. 66
A.1 Esempio di un cumulative sum random walk. 74
A.2 Distribuzione del numero di cicli in una sequenza al variare della sua lunghezza. 75
A.3 Distribuzione del numero di sequenze con un numero di cicli maggiore di 500 in un insieme composto da 100 sequenze, al variare della lunghezza delle sequenze. 76
A.4 Distribuzione della statistica S_n, calcolata all'interno del Frequency Test, rispetto al numero di cicli in una sequenza. 78
A.5 Distribuzione della statistica z, calcolata nel Cumulative Sums Test (modalità forward), rispetto al numero di cicli in una sequenza. 81

B.1 Distribuzione dei p-value ottenuti applicando il Discrete Fourier Transform Test a 100.000 sequenze casuali. 84

Introduzione

I test di casualità si collocano nel contesto della valutazione di un generatore di sequenze binarie casuali; un sistema in grado di produrre sequenze di bit che siano casuali. Il significato di casualità è un concetto ampiamente studiato e molto delicato. In letteratura esistono molteplici definizioni su cosa si intenda con il termine casualità ed in particolare con sequenze casuali. Idealmente, un generatore di numeri casuali dovrebbe essere in grado di produrre sequenze di bit, in cui ciascun elemento ha la stessa probabilità di essere 0 o 1. Gli ambiti in cui si ritrova un uso sempre maggiore di numeri casuali sono molteplici: dalla simulazione alla statistica, fino alla crittografia e al gioco. Dato il numero considerevole di richieste di sequenze casuali, sono stati introdotti generatori che si basano su processi deterministici e di conseguenza del tutto predicibili. La qualità di un buon generatore risiede nella capacità di produrre sequenze che appaiaono casuali, nonostante la presenza di componenti algoritmiche. Realizzare un generatore con prestazioni elevate e che sia in grado di produrre sequenze che risultino indistinguibili da sequenze prodotte tramite un generatore ideale è un compito molto difficile.

Al fine di verificare che le sequenze prodotte da un generatore appaiano casuali vengono utilizzati i test di casualità. Un test di casualità controlla che nelle sequenze prodotte da un generatore sia presente una determinata proprietà, caratteristica delle sequenze casuali. In letteratura esistono numerosi test di casualità; tuttavia, per motivi di risorse computazionali limitate, al fine di testare la casualità di un generatore, è necessario considerarne un numero ridotto. A tal proposito sono state realizzate delle collezioni di test, chiamate batterie di test di casualità, ciascuna con l'obiettivo di essere in grado di individuare quante più deviazioni possibili dalla casualità. Un requisito importante per una suite di test è il fatto di essere composta da test tra loro indipendenti.

La relazione tra i test di casualità sarà l'aspetto principale trattato nella presente tesi. In particolare, verrà affrontata questa tematica focalizzandosi sull'insieme di test realizzato dal National Institute of Standards and Technology (NIST); suite di test di casualità considerata lo standard a livello mondiale per la valutazione di generatori utilizzati in crittografia. L'indipendenza tra test di casualità può essere studiata utilizzando diversi approcci. Le strategie considerate in questo elaborato sono due e si basano entrambe sul concetto di indipendenza tra eventi di probabilità. È possbile definire la relazione tra un insieme di test di casualità tramite la nozione di indipendenza a due a due ed attraverso quella di indipendenza reciproca. Nel primo caso viene stabilita l'indipendenza tra ciascuna coppia di test presente nell'insieme di interesse, utilizzando un test di indipendenza chi quadrato. L'indipendenza reciproca è invece un concetto più forte rispetto al precedente:

non solo viene valutata l'indipendenza tra tutte le coppie di test ma sono richieste ulteriori condizioni su ciascun sottoinsieme possibile di test.

In questa tesi verrà condotta un'analisi completa sull'indipendenza a due a due tra tutti i test presenti nella Suite del NIST e saranno svolti alcuni studi concentrandosi su determinati sottoinsiemi di test costituti da diverse versioni dello stesso test di casualità. Durante le analisi effettuate sono originati diversi spunti di riflessione su alcuni fondamenti teorici e sono state individuate nuove interazioni tra i test della Suite del NIST meritevoli di ulteriori approfondimenti.

Questo documento è strutturato come segue. Nel Capitolo 1 sono presenti le nozioni indispensabili per introdurre il contesto in cui si sviluppano i test di casualità. In particolare, viene ricordato il concetto di casualità, sono descritti il funzionamento e le proprietà di un generatore di numeri casuali ed infine sono riportate le basi teoriche necessarie per definire i test di casualità. Nel Capitolo 2 viene introdotto e affrontato il problema dell'indipendenza tra test di casualità, trattando gli aspetti teorici dell'indipendenza a due a due e dell'indipendenza reciproca tra test. Nel Capitolo 3 sono presentati i risultati delle analisi di indipendenza svolte in questo lavoro. Le conclusioni ottenute sono state confrontate con quanto ricavato da altre ricerche condotte nello stesso ambito. Nelle Appendici A e B, sono infine riportati degli approfondimenti su alcune interazioni riscontrate, delle quali non sono state trovate evidenze note prima della stesura di questo documento.

Capitolo 1

Test per generatori di numeri pseudocasuali

In questo capitolo vengono introdotti alcuni concetti fondamenti per lo studio dell'indipendenza tra test di casualità. Oltre a definire i test di casualità, oggetti centrali in questa Tesi, è indispensabile motivarne la loro necessità. Dopo aver ricordato nelle Sezioni 1.1 e 1.2 le definizioni di casualità e sequenze casuali, nella Sezione 1.3 verranno descritti il funzionamento e le proprietà dei generatori di numeri casuali. Nella Sezione 1.4, infine, sono definiti i test di casualità e la metodologia utilizzata per valutare la correttezza di un generatore.

1.1 Casualità

Il concetto di casualità è comunemente noto, tuttavia spiegare formalmente che cosa si intende con casualità è tutt'altro che semplice. Solitamente, la parola 'casuale' viene associata a qualcosa di impredicibile, come l'estrazione del numero fortunato per la vincita di un montepremi oppure un'apparente mancanza di regolarità in un sistema.

Nell'antichità, la casualità era intrecciata al concetto di fato. Si credeva infatti che il verificarsi di un evento casuale, come il lancio di un dado, determinasse il destino di una persona. Con l'evolversi del tempo, il termine 'casuale' compare nel contesto del gioco, ambito in cui ancora oggi è utilizzato. Solo l'avvento della matematica del XVII secolo e l'introduzione di strumenti di calcolo avanzati hanno reso possibile lo studio approfondito del concetto di casualità, permettendo di formalizzare il significato di numero casuale associandolo alla nozione di variabile aleatoria. Negli anni recenti, la casualità è diventata un concetto fondamentale nella teoria dell'informazione, disciplina in cui le sequenze binarie casuali assumono un ruolo centrale. Si sono diffusi principalmente tre approcci per definire una sequenza di bit casuale. Il primo, introdotto da von Mises, si basa sul fatto che diversi elementi di una stringa casuale ed anche di sue specifiche sottostringhe debbano apparire con una *frequenza stabile*. L'approccio introdotto da Kolmogorov, nel 1963, lega il concetto di casualità con la teoria della *complessità*. Egli ha definito la casualità di una sequenza in termini della lunghezza della descrizione più breve possibile che permette di ricreare la

sequenza stessa. Infine, l'idea principale dell'approccio *quantitativo* diffuso da Martin-Löf nel 1966, è che una sequenza casuale dovrebbe avere un numero piccolo di regolarità e di conseguenza dovrebbe superare determinati test [20]. È importante sottolineare che questi approcci non sono indipendenti tra loro, ma profondamente collegati.

1.2 Sequenze casuali

L'oggetto principale di questa tesi è rappresentato dalle sequenze di bit casuali. Una sequenza di bit casuale può essere interpretata come il risultato del lancio di una moneta non truccata, in cui le facce sono etichettate con '0' e '1'. Siccome la moneta è non truccata, essa ha probabilità uguale, pari a 1/2, di produrre '0' o '1'. Il compito difficile è determinare se una sequenza di bit sia casuale o meno. Consideriamo ad esempio le seguenti stringhe di 23 bit:

00000000000000000000000
01101010000010011110011
11011110011101011111011

A prima vista, la seconda e la terza appaiono casuali, mentre la prima è considerata sospetta. Tuttavia, secondo la teoria delle probabilità, tutte e tre le stringhe hanno la stessa probabilità di realizzazione, che in questo caso è pari a $1/2^{23}$. Infatti, dato lo spazio di tutte le sequenze di 23 bit, ogni sequenza ha la stessa probabilità di realizzazione. La ragione per cui la prima sequenza non sembra casuale è legata alla percezione che si ha della casualità. Quando vediamo la prima stringa, il nostro cervello memorizza subito la struttura di questa sequenza. Essa è infatti molto semplice da ricordare in quanto consiste in tutti 0. Prendendo in considerazione la seconda stringa, questa sembra casuale dato che non mostra nessun pattern visibile, ed è quindi maggiormente difficile da memorizzare in modo compatto. Tuttavia, anche se sembra casuale, in realtà è l'espansione binaria di $\sqrt{2} - 1$, quindi è ottenuta tramite un processo deterministico, tutt'altro che casuale. Nonostante ciò, è rimarcabile la somiglianza con la terza stringa ottenuta invece attraverso una sequenza di lanci di una moneta. Questo esempio illustra due principali errori che le persone spesso commettono. Il primo è quello di pensare che un oggetto sia generato casualmente solo perché sembra casuale. Il secondo è pensare che pattern che accadono per puro caso in realtà sono dovuti ad altre ragioni.

Dal momento che l'utilizzo di sequenze di bit casuali è sempre più diffuso nelle applicazioni reali, si pone il problema di come generare sequenze che possano essere adatte per i diversi casi d'uso. Esistono diversi modi per generare sequenze di bit casuali e possono essere fondamentalmente raggruppati in due categorie: **Non-Deterministic Random Bit Generators** (NDRBGs) o equivalentemente **True Random Number Generators** (TRNGs) e **Deterministic Random Bit Generators** (DRBGs) o equivalentemente **Psuedo-Random Number Generators** (PRNGs). La distinzione tra queste due tipologie risiede nel processo di generazione. Nel primo caso le sequenze generate sono considerate 'veramente casuali', in quanto derivate a partire da processi impredicibili. Tuttavia generatori di questo tipo necessitano di una 'sorgente fisica' e proprio per questo

sono molto inefficienti dal punto di vista della scalabilità. Un esempio di generatore di questo tipo, potrebbe essere la generazione di una sequenza di bit, a partire dal lancio di una moneta. Nel caso di una moneta non truccata, indipendentemente dal numero di lanci effettuati, l'esito del prossimo lancio risulterà sempre impredicibile. I generatori appartenenti alla seconda tipologia producono sequenze attraverso un processo completamente deterministico e per questo motivo risultano molto veloci. Siccome le sequenze sono generate in maniera deterministica, queste vengono chiamate *pseudocasuali*. La caratteristica fondamentale di queste sequenze è che esse appaiono casuali anche se in realtà sono ricavate tramite un preciso algoritmo. Il punto chiave è che in linea di principio dovrebbe essere impossibile distinguere sequenze generate tramite un DRBG da sequenze veramente casuali. In questo contesto è utile introdurre la definizione di *indistinguibilità*. La nozione di indistinguibilità afferma che una stringa pseudocasuale non possa essere distinta da una stringa casuale, da nessun algoritmo polinomiale. Di conseguenza, le stringhe prodotte da un DRBG che possiede tale proprietà, possono essere utilizzate al posto di stinghe casuali, senza nessun effetto negativo. Di seguito è fornita una definizione formale di questo concetto. Interpretiamo un DRBG come una funzione $G : \mathbb{Z}_2^n \to \mathbb{Z}_2^l$, che mappa sequenze casuali di lunghezza ridotta, in stringhe pseudocasuali di lunghezza notevolmente maggiore ($l >> n$).

Definizione 1 *Si dice che il DRBG produce sequenze indistinguibili da sequenze casuali se un attaccante non supera il seguente test con probabilità maggiore di* $1/2$.

Verifier	*Attaccante*
Sceglie con probabilità uniforme $b \in \{0,1\}$	
Se $b = 0$ *, allora sceglie* $r \in \{0,1\}^l$*,*	
con probabilità uniforme	
Se $b = 1$ *, allora sceglie* $s \in \{0,1\}^n$*,*	
con probabilità uniforme e calcola $r = G(s)$	
	Riceve r *e invia al Verifier un bit* b'
Attaccante vince il gioco se $b' = b$	

Vedremo nei paragrafi successivi come la nozione di indistinguibilità sia un requisito fondamentale per un generatore di numeri casuali.

Prima di spiegare il funzionamento e i requisiti di un generatore di numeri casuali, motiviamone la necessità. I numeri casuali sono fondamentali in molti campi, in particolare gli usi più interessanti si trovano nella simulazione e nella crittografia. Nella simulazione i numeri casuali vengono utilizzati per generare campioni che rappresentano una popolazione o individuo di interesse. Ad esempio, per simulare il comportamento di un fenomeno fisico o di problemi decisionali di carattere finanziario. Nella crittografia la casualità è presente in quasi ogni protocollo: dalla generazione delle chiavi, agli schemi di cifratura, fino ai protocolli di firma digitale. Senza casualità la crittografia sarebbe impossibile perché tutte le operazioni diventerebbero predicibili e quindi insicure [2]. È bene precisare che in ambito crittografico, vengono utilizzati dei generatori appositi, con requisiti di sicurezza maggiori rispetto ai semplici PRNG. Si ha infatti una distinzione tra PRNG e PRNG crittograficamente sicuri (**Cryptographically Secure Pseudo-Random Number Generator**, CSPRNG), che verrà affrontata in dettaglio nella Sezione 1.3.1.

1.3 Generatori di numeri casuali

In questo paragrafo vengono descritti i generatori di bit casuali, introdotti nella sezione precedente. Un generatore di bit casuali, proprio come suggerisce il nome, è un sistema in grado di fornire bit casuali. Attualmente sono disponibili diversi generatori di numeri casuali, caratterizzati da prestazioni e proprietà differenti. Nel 2010, il National Institute of Standards and Technology (NIST) ha prodotto una serie di raccomandazioni su come si dovrebbe progettare un buon generatore di numeri casuali [10]. L'impatto di queste pubblicazioni è stato considerevole in quanto negli anni precedenti, non essendo presente nessuna guida per realizzare in modo corretto un generatore, erano diffusi generatori con un livello di sicurezza non adeguato all'uso che se ne faceva. Ci sono principalmente due modi per generare casualità. La prima strategia è produrre bit in modo non deterministico. In questo caso ogni bit è generato attraverso un processo fisico impredicibile. Questa classe di generatori è comunemente chiamata Non-Deterministic Random Bit Generators (NRBGs) o equivalentemente True Random Number Generators (TRNGs). La seconda modalità è generare i bit in maniera deterministica, usando un algoritmo. Questo tipo di generatori vengono chiamati Deterministic Random Bit Generators (DRBGs) o in modo equivalente Pseudo-Random Number Generators (PRNGs). Di seguito sono esposte le caratteristiche principali di queste due tipologie.

In crittografia, la casualità proviene solitamente da un TRNG, una componente di tipo software o hardware che manipola l'entropia del mondo analogico in modo da produrre bit impredicibili. Specificare una sorgente di entropia è un problema complicato. Questa difficoltà è dovuta in parte ad una confusione legata al concetto di entropia ed in parte al fatto che le sorgenti di entropia, rispetto alle altre componenti di un generatore che sono strettamente algoritmiche, dipendono solamente da processi fisici. L'entropia è definita relativamente alla conoscenza che si ha del risultato di un esperimento prima che questo venga osservato. In un certo senso riflette l'incertezza associata al predire il risultato dell'esperimento: più è alta l'entropia, maggiore è l'incertezza nel predire il valore di un'osservazione. Le sorgenti di rumore che un TRNG può sfruttare per generare bit casuali, possono essere diverse. Alcuni TRNG sfruttano hardware dedicati per generare casualità, quali ad esempio strumenti che misurano la temperatura, il rumore acustico o gli spostamenti d'aria. Altri TRNG, invece, estraggono entropia a partire da dati di sistema, come l'output di funzioni applicative, gli accessi alla memoria RAM o l'attività su disco, oppure basandosi su comportamenti umani (come ad esempio i tasti premuti sulla tastiera o i movimenti del mouse). Questi sistemi sono buone sorgenti di entropia, tuttavia se progettati male presentano delle fragilità e possono essere manipolati da un attaccante. Inoltre, queste sorgenti di entropia potrebbero non essere sempre disponibili, rendendo lento il processo di generazione dei bit casuali. Si pensi ad esempio ad un TRNG che trasforma i movimenti del mouse in bit casuali. Esso smetterà di funzionare ogni qualvolta il mouse sarà fermo. Per queste motivazioni i TRNG vengono combinati insieme ai PRNG.

I PRNG si basano su un meccanismo di generazione di bit pseudocasuali e includono una sorgente di casualità, che spesso è rappresentata da un TRNG. I PRNG sono utilizzati per trasformare pochi bit casuali altamente affidabili, in una lunga sequenza di bit pseudo-casuali, che possono essere utilizzati nelle applicazioni. I PRNG producono velocemente

bit che 'appaiono' casuali a partire da una sorgente digitale, in maniera deterministica e con la massima entropia. Il meccanismo di generazione di bit pseudocasuali comprende un algoritmo (*algoritmo DRBG*) che produce sequenze di bit da un valore iniziale determinato a partire da un seed, prodotto utilizzando l'output di una sorgente di casualità. Una volta che viene fornito il seed, è possibile determinare il valore iniziale e istanziare il PRNG. A questo punto il PRNG è pronto e può iniziare a produrre sequenze di bit pseudocasuali. È molto importante che il seed utilizzato per istanziare il PRNG contenga un'entropia sufficiente da assicurare la randomicità delle sequenze generate. Siccome l'algoritmo DRBG è completamente deterministico, ogni volta che viene fornito lo stesso input la sequenza prodotta sarà uguale. Per generare sequenze uniche è quindi necessario fornire ogni volta un input diverso. Il meccanismo DRBG è costituito da tre processi principali: il processo di *istanziazione*, il processo di *generazione* e il processo di *reseed*. La funzione di istanziazione acquisice entropia dalla sorgente di casualità e crea il seed da cui viene derivato lo stato interno iniziale del generatore. Lo stato interno è la memoria del generatore e comprende tutti i parametri e le variabili che il meccanismo DRBG usa o modifica. La funzione di generazione ha il compito di generare bit casuali quando richiesto. Partendo dallo stato interno corrente e possibilmente qualche dato aggiuntivo vengono prodotti bit casuali. Al termine del processo, lo stato interno viene aggiornato. L'uso di dati aggiuntivi nel processo di generazione è utile per fornire entropia aggiuntiva allo stato interno del DRBG, in modo che si abbia maggior sicurezza sul raggiungimento del livello minimo di entropia richiesto. La funzione reseed è utilizzata per acquisire nuova entropia che viene combinata con lo stato interno corrente in modo da produrre un nuovo stato interno. Aggiornare periodicamente lo stato interno del DRBG è fondamentale per garantire determinate proprietà di sicurezza del generatore.

1.3.1 Proprietà di un generatore di numeri pseudocasuali

La proprietà fondamentale per un PRNG è che esso sia in grado di produrre sequenze che appaiano casuali. In altre parole, bisogna assicurare che le sequenze generate soddisfino le propietà che una sequenza casuale dovrebbe possedere. Una dimostrazione teorica della randomicità di un generatore è quasi impossibile da dare. Per cui, al fine di verificare l'uniformità di un generatore, vengono svolte analisi statistiche sulle sequenze da esso prodotte. A tale scopo esistono diversi test statistici che consentono di verificare la casualità delle sequenze. Ciascun test verifica una caratteristica precisa della sequenza, valutando se questa sia in accordo o meno con ciò che ci si aspetta ipotizzando l'uniformità della stringa. Ad esempio, se si considera una sequenza di n bit, con n sufficientemente grande, ci si aspetta che il numero di '0' (e di conseguenza '1') sia circa $n/2$. Nel caso in cui ci sia un'evidenza statistica tale da affermare il contrario, la sequenza non passerà il test. Notare che, tuttavia, questa proprietà non può essere valutata su una singola sequenza. Un test di casualità, per essere attendibile, deve essere condotto utilizzando un campione di più sequenze e a seconda della proporzione di sequenze che superano il test, il generatore viene considerato buono oppure no. Infatti, è possibile che un buon generatore, ogni tanto produca alcune sequenze che non rispettano la proporzione attesa di '0' e '1'. Per valutare correttamente la bontà di un generatore è quindi necessario applicare il test ad un numero sufficiente di sequenze da esso prodotte. Maggiori dettagli sui test di casualità

sono riportati nel Paragrafo 1.4. Oltre a questo requisito fondamentale, esistono altre caratteristiche che determinano la qualità di un generatore. Tra i parametri principali vi sono:

- **Periodo**: poiché lo spazio degli stati interni che un DRBG può assumere a partire da un seme è finito, prima o poi lo stato del generatore tornerà allo stato iniziale. Dal momento che il processo di trasformazione dello stato interno nel bit generato è deterministico, tutti gli output generati si ripeteranno. Il valore più piccolo per cui si realizza un ritorno allo stato iniziale è chiamato periodo del PRNG. Ovviamente più il periodo è lungo, più il generatore è di buona qualità. L'ideale sarebbe avere periodo prossimo alla cardinalità dello spazio degli stati che il generatore può assumere;

- **Efficienza**: l'efficienza di un PRNG è misurata in termini di memoria allocata e velocità di calcolo. Un buon generatore dovrebbe utilizzare un quantitativo ridotto di risorse computazionali;

- **Ripetibilità**: partendo dallo stesso stato iniziale i PRNG devono essere in grado di riprodurre la stessa sequenza di bit casuali;

- **Portabilità**: è importante che un PRNG non sia progettato per funzionare solamente su determinati hardware o software, ma che sia possibile implementarlo ed utilizzarlo in diversi contesti.

Oltre alle caratteristiche appena descritte, vi sono ulteriori proprietà che un generatore utilizzato in applicazioni crittografiche deve soddisfare. Un CSPRNG deve resistere ad attacchi di crittoanalisi anche quando parte del suo stato iniziale o corrente sia disponibile ad un attaccante. Le proprietà richieste sono le seguenti:

- **Prediction Resistance**: la garanzia che una compromissione dello stato interno del DRBG non abbia nessun effetto sulla sicurezza degli output futuri del DRBG. Notare che se un avversario viene a conoscenza dello stato interno all'iterazione x e conosce il meccanismo usato dal DRBG, esso ha abbastanza informazioni per calcolare tutta la sequenza degli stati a partire da $x + 1$, e di conseguenza tutti gli output futuri. L'unico modo per garantire questa proprietà è quello di aggiornare il seed tra ogni richesta di generazione, aggiungendo nuova entropia tra un output e l'altro. La quantità di entropià aggiunta dovrebbe essere almeno pari al livello di sicurezza richiesto. Nel caso in cui l'entropia nuova sia poca, nonostante l'attaccante non possa direttamente calcolare gli stati successivi conoscendo quello attuale, ha comunque solo un numero limitato di possibilità. Infatti, a seguito della compromissione dello stato, il nuovo livello di sicurezza dipende dalla quantità di entropia nuova introdotta. Per soddisfare questa proprietà un DRBG dovrebbe avere a disposizione una sorgente di entropia e, ogni qualvolta gli è possibile, fornirne nuova al processo di generazione;

- **Backtracking Resistance**: la garanzia che le sequenze in output di un generatore rimangano indistinguibili da sequenze casuali ideali, anche nel caso in cui il DRBG venga compromesso in futuro. In pratica la backtracking resistance assicura che un attaccante che conosce lo stato x, non è in grado di distinguere tutti gli output

generati dagli stati $1, 2, \ldots, x-1$, da output casuali ed inoltre non è in grado di ricostruire gli stati precedenti. Questa proprietà può essere garantita progettando l'algoritmo di generazione in modo che sia una funzione one-way. Una funzione one-way (o funzione unidirezionale) è una funzione facile da calcolare ma difficile da invertire. Ciò significa che a partire dallo stato interno x è molto facile ottenere lo stato successivo $x+1$. Tuttavia, conoscendo lo stato $x+1$ non esiste nessun algoritmo polinomiale in grado di ricavare lo stato precedente x.

Un CSPRNG è quindi progettato per resistere alla crittoanalisi. È importante sottolineare che un CSPRNG è anche un PRNG, ma il viceversa non vale. Esistono PRNG che soddisfano tutti i test di casualità ma che sono vulnerabili ad attacchi di ingegneria inversa. Si faccia riferimento, ad esempio, al Linear Congruential Generator (LCG); un generatore molto noto e comunemente utilizzato per le simulazioni, che tuttavia è predicibile e quindi totalmente insicuro per scopi crittografici.

1.4 Test di casualità

Dato un generatore, si vuole verificare che esso sia indistinguibile da un generatore ideale. Un requisito essenziale per un PRNG è infatti che esso sia in grado di produrre delle sequenze che siano indistinguibili da sequenze casuali. Per valutare che il generatore utilizzato soddisfi questa proprietà, vengono utilizzati dei test statistici, chiamati **test di casualità**, che consentono di trovare certi tipi di debolezze che un generatore potrebbe avere. Idealmente per testare la distribuzione delle sequenze generate potremmo utilizzare un singolo test che verifica se ogni sequenza prodotta dal generatore sia equiprobabile. Poiché il numero di possibili sequenze è pari a 2^n, dove n è la lunghezza in bit di una sequenza, all'aumentare di n questo test diventa insostenibile. Per accertare la randomicità di un PRNG viene quindi svolta un'analisi di casualità sugli output prodotti dal generatore, la quale consiste nell'applicare diversi test di casualità sulle sequenze prodotte. I test hanno il compito di valutare se ciascuna sequenza possiede o meno un certo attributo che una sequenza veramente casuale dovrebbe avere. Ogni test cerca la presenza o meno di 'pattern' che, se trovati, indicano che la sequenza non è casuale. In letteratura, esistono centinaia di test statistici, già formalizzati e implementati, che possono essere utilizzati per testare la randomicità di un generatore. L'uso di un singolo test non è abbastanza per verificare la randomicità di una sequenza, perché la sequenza potrebbe possedere diversi tipi di non randomicità. Se un generatore passa un numero elevato di test diversi invece, la confidenza nella sua casualità aumenta. Proprio per questo motivo si sono diffuse in letteratura collezioni che comprendono più test, generalmente chiamate *suites* o *batterie* di test di casualità. Ogni test all'interno della suite ha il compito di controllare una caratteristica di casualità diversa. Se tutti o una determinata proporzione dei test nella batteria concludono che il PRNG produce numeri casuali, allora il PRNG viene considerato di buona qualità. Il compito difficile è determinare quali e quanti test sia meglio applicare e come stabilire la proporzione minima, di test superati rispetto al totale dei test applicati, affinchè il generatore sia considerato di buona qualità. Partendo dal presupposto che potenzialmente esistano infiniti test e che siano tutti ugualmente validi, nessun insieme finito di test può essere considerato 'completo'. Ogni insieme di test finito, in qualche

modo non controlla alcuni difetti che un generatore potrebbe avere. Da quanto appena osservato si deduce che anche applicando tutti i test diffusi in letteratura non si avrebbe la certezza assoluta che il generatore sia effettivamente casuale. Nella pratica, tuttavia si ha un quantitativo di risorse limitato, per cui è necessario selezionare un numero finito di test ragionevole. Solitamente la scelta è fatta in accordo con i requisiti richiesti dall'applicazione per la quale il generatore è utilizzato. L'altra domanda a cui è difficile dare una risposta è quali test inserire all'interno di una suite. Questa domanda presuppone l'esistenza di test migliori rispetto ad altri. Tuttavia, per quanto detto finora, ogni test dovrebbe essere ugualmente valido. Ciascun test infatti, permette di stabilire se un generatore sia casuale o meno e quindi, indipendentemente dalla caratteristica analizzata dal test, un generatore che produce numeri veramente casuali dovrebbe soddisfarla.

1.4.1 Test di ipotesi

Un test di ipotesi è una procedura che si basa su un campione di dati che viene utilizzato per supportare il processo di decisione sulla distribuzione teorica di una popolazione. In ogni test d'ipotesi ci sono due ipotesi contradditorie prese in considerazione. L'ipotesi nulla, denotata con H_0 è l'affermazione inizialmente considerata vera. L'ipotesi alternativa, denotata con H_a, è l'asserzione che è contradditoria a H_0. I possibili risultati di un test d'ipotesi sono: accettare H_0 o accettare H_a. L'ipotesi nulla verrà rifiutata in favore dell'ipotesi alternativa solo se vi è un'evidenza campionaria che suggerisce che H_0 sia falsa. Per stabilire se H_0 sia falsa viene utilizzata una test statistic: una funzione dei dati campionati. La statistica test è utilizzata per calcolare un p-value che riassume la forza dell'evidenza contro l'ipotesi nulla. Il p-value è definito come la probabilità, calcolata assumendo che l'ipotesi nulla sia vera, di ottenere un valore della test statistic ugualmente o meno probabile di quello osservato dai dati. Una conclusione è raggiunta selezionando un numero α, chiamato livello di significatività del test, che è ragionabilmente vicino a 0. L'ipotesi nulla sarà rifiutata se il p-value è minore o uguale ad α, mentre non sarà rifiutata se p-value è maggiore di α. Più piccolo è il p-value, maggiore è l'evidenza contro l'ipotesi nulla. Ci sono 2 tipi di errori che possono essere fatti durante un'analisi di questo tipo:

- Errore di tipo I, consiste nel rifiutare l'ipotesi nulla quando in realtà è vera;
- Errore di tipo II, consiste nel non rifiutare l'ipotesi nulla quando in realtà è falsa.

È possibile dimostrare che il livello di significatività α corrisponde alla probabilità di commettere errori di tipo I: $\mathbf{P}(rifiutare\ H_0|H_0 \text{ è } vera) = \alpha$. La probabilità di un errore di tipo II è invece denotata con $\beta = \mathbf{P}(accettare\ H_0|H_0 \text{ è } falsa)$. Solitamente la probabilità di errori di tipo I è fissata. Valori comuni per α in crittografia appartengono all'intervallo $[0{,}001; 0{,}01]$. Diversamente da α, β non è un valore fisso ma può assumere molti valori differenti, in quanto l'ipotesi H_0 può essere falsa per infinite ragioni.

Quando si vuole valutare la randomicità di un generatore vengono utilizzati i test di casualità, che sono a tutti gli effetti dei test d'ipotesi. Nei test di casualità, l'ipotesi nulla è che la sequenza sotto esame sia casuale, mentre quella alternativa è che la sequenza non lo sia. L'ipotesi nulla verrà rifiutata in favore dell'ipotesi alternativa qualora ci sia evidenza sperimentale che H_0 sia falsa. Valutare la randomicità di un PRNG applicando un test

statistico significa applicare il test ad un certo numero di sequenze da esso generate e studiare la proporzione di sequenze che superano o meno il test. Una sequenza supera il test di casualità, nel caso in cui il p-value corrispondente sia maggiore o uguale al livello di significatività scelto. Ipotizzando che generatore produca bit casuali, ci si aspetta che circa una proporzione di $1-\alpha$ delle sequenze testate passeranno il test, mentre la restante α no. Proprio per definizione di test statistico, quindi, alcune sequenze testate non supereranno il test anche se il generatore è corretto. La quantità di sequenze rifiutate è pari ad α, in quanto α è la probabilità di commettere errori di tipo I.

1.4.2 Interpretazione dei risultati

Applicando un test di casualità ad un insieme di n sequenze di lunghezza m fissata, otteniamo un insieme di n p-value. Fissato un livello di significatività, α, per il test di casualità, è possibile stabilire se ciascuna sequenza ha passato o meno il test. Nel caso in cui $p-value \geq \alpha$, la sequenza passa il test, altrimenti no. È possibile, quindi, costruire un vettore $S = (s_1, \ldots, s_n)$, di lunghezza n, in cui l'elemento s_i assume valore $True$ o $False$ a seconda che la sequenza i-esima abbia superato o meno il test di casualità ($False$ indica che la sequenza non ha superato il test). Nella documentazione del NIST sono proposti due approcci per interpretare e valutare i risultati ottenuti:

- Esaminare la proporzione di sequenze che passano un test statistico;
- Controllare la distribuzione dei p-value.

Per concludere che il generatore ha superato il test statistico sottoposto è necessario che le due verifiche sopracitate siano in accordo con i risultati teorici attesi. Di seguito, esaminiamo in dettaglio la motivazione e la formulazione di questi due controlli necessari.

Proporzione di sequenze che passano un test

Come riportato nella Sezione 1.4.1, se un generatore produce sequenze realmente casuali ci si aspetta che la proporzione di sequenze che passano il test sia pari ad α. Utilizziamo un test statistico per valutare se la proporzione delle sequenze che passano un certo test di casualità, calcolata empiricamente, $\hat{p}$, è statisticamente uguale alla proporzione attesa, $1-\alpha$. Consideriamo la variabile aleatoria X, che rappresenta il numero di sequenze che superano il test di casualità. Assumendo che l'ipotesi nulla sia vera, il numero atteso di sequenze che superano il test dovrebbe essere pari a $n(1-\alpha)$, dove n è il numero delle sequenze testate. Infatti, $X = \sum_{i=1}^{n} s_i$, dove, sotto l'ipotesi nulla H_0, s_i assume valore 1 con probabilità $1-\alpha$. Se n è piccolo, la distribuzione della variabile aleatoria X, è una Binomiale $X \sim Bin(n,1-\alpha)$, in quanto somma di Bernoulli con probabilità di successo pari a $1-\alpha$. Tuttavia, se il numero di sequenze considerato è sufficientemente grande allora la variabile casuale X può essere approssimata con una distribuzione Normale di media $n(1-\alpha)$ e varianza $n(1-\alpha)\alpha$, per cui $X \sim N(n(1-\alpha), n(1-\alpha)\alpha)$. Questa approssimazione è valida quando valgono le seguenti condizioni: $n(1-\alpha) \geq 10$ e $n\alpha \geq 10$. Considerando che in contesto crittografico α è scelto pari a $\alpha = 0{,}01$, per sfruttare questa approssimazione è necessario utilizzare almeno 1000 sequenze. Lo stimatore utilizzato per

condurre il test sulla proporzione di successi è $\hat{p} = X/n$, ovvero la proporzione di successi sul totale delle seqeunze testate. Supponendo che $X \sim N(n(1-\alpha), n(1-\alpha)\alpha)$ allora $\hat{p} \sim N((1-\alpha), (1-\alpha)\alpha/n)$. Di conseguenza si ottiene che la variabile

$$Z = \frac{\hat{p} - (1-\alpha)}{\sqrt{(1-\alpha)\alpha/n}},$$

segue una distribuzione normale standard: $Z \sim N(0,1)$. A questo punto, dopo aver calcolato empiricamente la proporzione di successi, $\hat{p}$, è possibile calcolare il p-value, che nel caso di una normale standard si calcola utilizzando la formula,

$$p - value = 2P(Z \geq z) = 2[1 - \phi(z)],$$

dove $z = \frac{\hat{p}-(1-\alpha)}{\sqrt{(1-\alpha)\alpha/n}}$ è la statistica del test e $\phi(x)$ è la funzione di ripartizione della normale standard. Fissato un livello di significatività, α_1, relativo al test sulla proporzione, è possibile stabilire, sulla base del p-value ottenuto se la proporzione empirica si discosta in modo eccessivo da quella attesa. L'intervallo di fiducia attorno alla proporzione di successi $\hat{p}$:

$$(1-\alpha) - z_{\alpha_1/2}\sqrt{\frac{(1-\alpha)\alpha}{n}} \leq \hat{p} \leq (1-\alpha) + z_{\alpha_1/2}\sqrt{\frac{(1-\alpha)\alpha}{n}}.$$

Valori di $\hat{p}$ che cadono all'esterno di questo intervallo causano il rifiuto dell'ipotesi nulla e il conseguente fallimento del test. Nelle raccomandazioni del NIST il quantile $z_{\alpha_1/2}$ è preso pari a 3, che corrisponde ad un livello di significatività $\alpha_1 \approx 0{,}26\%$.

Distribuzione uniforme dei p-value

La seconda verifica proposta dal NIST, per confermare il superamento di un test di casualità da parte di un PRNG, è il controllo dell'uniformità dei p-value. È noto dalla teoria della statistica che in un test d'ipotesi, se l'ipotesi nulla è vera e la distribuzione cumulativa della statistica del test è invertibile, allora i p-value sono distribuiti in maniera uniforme nell'intervallo $[0, 1]$. In un test di casualità, quindi, se le sequenze sono veramente random e la statistica del test è distribuita in modo continuo, allora i p-value calcolati dovrebbero avere distribuzione uniforme tra $[0, 1]$.

Il fatto che la statistica di un test di casualità segua una distribuzione continua non è in generale vero. Durante le analisi svolte in questo lavoro è stato osservato che esistono test di casualità che generano test statistic e di conseguenza p-value che non sono caratterizzati da una distribuzione continua. In questi casi, l'uniformità dei p-value non è assicurata da nessun risultato teorico. Nell'Appendice B è riportato uno studio approfondito sull'uniformità dei p-value di alcuni test di casualità riportati nella Suite del NIST.

Supponendo che siano soddisfatte le ipotesi che garantiscono l'uniformità dei p-value, un metodo proposto per verificare che la distribuzione di questi ultimi sia uniforme nell'intervallo $[0, 1]$, è quello di dividere l'intervallo tra 0 e 1 in 10 sottointervalli e classificare i p-value in base all'intervallo in cui cadono. Successivamente, si applica un test chi quadro per valutare se la quantità di p-value in ogni categoria sia approssimativamente la stessa.

Il test chi quadro "χ^2" è ampiamente utilizzato in statistica per verificare che le frequenze di alcuni valori osservati si adattino alle frequenze teoriche di una distribuzione di probabilità prefissata. Nel nostro contesto, siccome la distribuzione di probabilità dei p-value dovrebbe essere uniforme, la frequenza teorica in ciascuno dei 10 sottointervalli dovrebbe essere pari a $n/10$, dove n è il numero di p-value disponibili. Consideriamo il vettore F, dove $F_i, \quad i = 1 \dots 10$ corrisponde alla frequenza empirica nel $i-$esimo sottointervallo e definiamo la statistica chi quadro come:

$$\chi^2 = \sum_{i=1}^{10} \frac{(F_i - n/10)^2}{n/10}.$$

La distribuzione di questa statistica è una chi quadro con $\nu = 9$ gradi di libertà. È possibile quindi calcolare il p-value relativo alla statistica test utilizzando la funzione gamma incompleta,

$$pValue = igamc\left(\frac{9}{2}, \frac{\chi^2}{2}\right) = \frac{1}{\Gamma(9/2)} \int_{\chi^2/2}^{\infty} e^{-t}\ t^{9/2-1} dt,$$

dove $\Gamma(x) = \int_0^\infty t^{x-1}\ e^{-t} dt$. Il livello di significatività da utilizzare per confrontare il p-value, suggerito dal NIST, è pari a $\alpha_2 = 0{,}1\%$. È possibile quindi trovare il valore massimo che la statistica χ^2 può assumere in modo che l'ipotesi nulla sia accettata: $\chi^2_{max} = igamc^{-1}(\frac{9}{2}, \alpha_2) \approx 33{,}72$. Infine, è importante sottolineare il fatto che per garantire la validità del test chi quadro, la frequenza attesa di ogni categoria dovrebbe essere almeno pari a 10 [8].

1.4.3 Batterie di test

La qualità di un generatore non è valutata sulla base di un singolo test ma su un insieme di test. Solitamente, è diffusa l'idea che più test vengono superati da un generatore, più la confidenza che si ha nella sua randomicità aumenta. In realtà questo pensiero non è però del tutto corretto. Infatti, basandosi sulle raccomandazioni del NIST, dato un insieme di test di casualità, un generatore potrebbe non passarne alcuni anche se in realtà produce sequenze casuali. Il test sulla propozione di successi, che viene sottoposto ai risultati di ciascun test di casualità, è un test statistico, con un suo livello di significatività, α_1. All'aumentare del numero di test di casualità condotti, dovrebbe esserci una percentuale pari a α_1, di test che non superano il controllo sulla proporzione attesa di successi. Dato un insieme di test di casualità, il generatore verrà considerato valido, nel caso in cui vengano superati in media $(1 - \alpha_1)$ % test.

In letteratura sono diffuse diverse batterie di test di casualità. Una delle prime batterie, è stata presentata da Donal Knuth, nel 1969 [13]. Questa batteria è stata utilizzata per molto tempo ed era considerata come una batteria di test di base. Attualmente è ormai datata e poco utilizzata in quanto è stato dimostrato che generatori considerati scarsi passano i test presenti. Questa batteria è stata adattata ed estesa da Marsaglia nel 1995, il quale ha introdotto test maggiormente stringenti, nel senso che sono più difficili da passare. La suite di Marsaglia è chiamata DIEHARD [16]. Queste due suite sono state sviluppate

per testare generatori utilizzati in simulazione. Nel 2001, viene rilasciata una suite da parte del NIST che comprende 15 test sviluppati per testare la randomicità di sequenze binarie arbitrariamente lunghe, prodotte da generatori progettati per applicazioni crittografiche [1]. La suite comprende sia test già esistenti, sia nuovi test appositamente sviluppati. Nel 2007, Pierre L'Ecuyer e Richard Simard introducono una libreria implemenata in C, chiamata TestU01 [15]. Questa libreria include molti test di casualità disponibili in letteratura e consente di applicarli sperimentando diverse combinazioni. Essa comprende al suo interno sei batterie predefinite, in grado di testare sequenze di numeri o di bit casuali. TestU01, può essere intesa come una suite di batterie di test. La libreria include anche dei test proposti dal NIST ma non tutti. Oltre a queste batterie di test ne esistono altre, che però hanno avuto un impatto pressochè limitato. Attualmente la suite del NIST è lo standard a livello mondiale utilizzato per testare generatori di numeri casuali. Per questo motivo nello svolgimento di questa tesi verrà considerata proprio questa suite. I test che fanno parte di questa suite sono riportati in Tabella 1.1. Per ciascun test è stata

Test	Caratteristica analizzata
Frequency (Monobit)	numero di 0 e 1 nella sequenza
Frequency within a Block	numero di 0 e 1 nella sequenza
Runs	Oscillazioni di 0 e 1 troppo veloci o troppo lente
Longest Run of Ones in a Block	Oscillazioni di 0 e 1 troppo veloci o troppo lente
Binary Matrix Rank	Deviazioni dalla distribuzione attesa del rango
Discrete Fourier Transform (Spectral)	Pattern ripetitivi
Non-overlapping Template Matching	Occorrenze irregolari di uno specifico template
Overlapping Template Matching	Occorrenze irregolari di uno specifico template
Mauer's Universal Statistical	Incompressibilità della sequenza
Linear Complexity	Linear feedback shift register (LFSR) troppo corti
Serial	Uniformità della distribuzione di pattern di lunghezza data
Approximate Entropy	Uniformità della distribuzione di pattern di lunghezza data
Cumulative Sums (Cusums)	Troppi 0 o 1 all'inizio o alla fine delle sequenze
Random Excursions	Deviazione dalla distribuzione del numero di visite di un random walk in un certo stato
Random Excursions Variant	Deviazione dalla distribuzione del numero di visite di più random walk in un certo stato

Tabella 1.1. Test di casualità presenti nella suite proposta dal National Institute of Standards and Technology (NIST). Per ciascun test è riportata la caratteristica analizzata.

riportata in breve la caratteristica che viene controllata. Precisiamo fin da subito che alcuni test presenti nella suite calcolano più di un p-value. In particolare, questi test sono il Serial e il Cumulative sums test, che calcolano entrambi due p-value e il Random

Excursions e Random Excursions Variant Test, che generano rispettivamente 8 e 18 p-value. Nella documentazione del NIST non è precisato se questi p-value vadano considerati individualmente o pesati in qualche maniera. Nella Sezione 3.3, si è deciso di trattare ogni p-value come il risultato di un test, a pari livello degli altri. Ogni test presente nella suite del NIST richiede in input una sequenza di bit di una lunghezza minima in modo che la distribuzione di riferimento utilizzata nel test sia valida. In Tabella 1.2 è riportata la lunghezza minima per ciascun test.

Test	Lunghezza minima della sequenza (bit)
Frequency (Monobit)	100
Frequency within a Block	100
Runs	100
Longest Run of Ones in a Block	128
Binary Matrix Rank	38.912
Discrete Fourier Transform (Spectral)	1000
Non-overlapping Template Matching	1.000.000
Overlapping Template Matching	1.000.000
Mauer's Universal Statistical	387.840
Linear Complexity	1.000.000
Serial	128
Approximate Entropy	128
Cumulative Sums (Cusums)	100
Random Excursions	1.000.000
Random Excursions Variant	1.000.000

Tabella 1.2. Lunghezza minima delle sequenze richiesta per ciascun test di casualità presente nella Suite del NIST.

Capitolo 2

Studio delle relazioni tra test di casualità

Per valutare la randomicità di un generatore è necessario applicare diversi test di casualità. A tal proposito, come spiegato nel paragrafo 1.4.3, esistono alcune collezioni di test. Quando si utilizzano più test statisitici bisogna prestare attenzione a due problemi: il *multiple testing problem*, chiamato anche *multiplicity problem* e alla *dipendenza tra test*. Il primo problema, affrontato da Demirhan e Bitirim, in [7], mostra che l'applicazione di un numero maggiore di test di casualità può incidere negativamente sulla decisione riguardo la randomicità di un PRNG. Questo problema in realtà dipende fortemente dal criterio con cui viene stabilita l'approvazione o meno del generatore. Un'assunzione diffusa in letteratura, è quella di rifiutare un generatore nel caso in cui non superi anche solo un test ed è proprio in questo contesto che si orgina il multiplicity problem. Per comprendere questo problema consideriamo k test statistici in una suite e supponiamo che ciascuno sia condotto, indipendentemente dagli altri ad un livello di significatività α. Dalla teoria della probabilità si ha il seguente risultato:

$$P(\{fallire\ almeno\ un\ test\}) = 1 - P(\{non\ fallire\ nessun\ test\}) = 1 - (1-\alpha)^k.$$

La probabilità di fallire almeno un test viene calcolata modellizzando la suite tramite una variabile Binomiale. Infatti, dal momento che ciascun test può essere descritto da una variabile aleatoria Bernoulliana con probabilità di successo $1-\alpha$ la suite che supponiamo sia composta da k test statistici indipendenti, segue una distibuzione Binomiale di parametri k e $1-\alpha$. Se ad esempio, $k = 10$ e il livello di significatività $\alpha = 0{,}01$, si ha circa una probabilità del 10% di concludere che la sequenza generata dal PRNG non è casuale in almeno un test, anche se tutti i test indicano che le sequenze generate sono casuali. Se il numero di test considerati è $k = 16$, e il livello di significatività sempre pari a $\alpha = 0{,}01$, questa probabilità sale al 15%. Si noti che all'aumentare del numero di test di casualità che vengono utilizzati, la probabilità di concludere che il generatore non sia casuale cresce, anche se esso in realtà genera sequenze casuali. Un modo per ridurre questa probabilità è quello di diminuire il livello di significatività α; tuttavia in questo caso diminuisce anche la regione di rifiuto di ciascun test e diventa sempre più difficile rifiutare l'ipotesi nulla. Per

evitare questo problema, viene consigliato di utilizzare batterie composte da un numero ridotto di test [15]. L'uso di batterie piccole non solo consente di diminuire la probabilità di stabilire erroneamente la non casualità di un generatore, ma comporta anche rilevanti vantaggi dal punto di vista computazionale. Notare tuttavia, che questo problema è dovuto all'assunzione secondo la quale un generatore supera la suite, solamente nel caso in cui superi tutti i test in essa presenti. Se infatti si considerasse superata una suite nel caso in cui venga superata una determinata proporzione di test, questo problema si risolve. Questa seconda strategia di valutazione, che è la stessa che viene proposta dal NIST, risulta più adeguata. Dal momento che per stabilire se un generatore supera o meno un test statistico, viene condotto un test sulla proporzione di successi (che è anch'esso un test statistico con un suo livello di significatività) potrebbe accadere che, applicando diversi test di casualità, si abbia un certo numero di test che presenta una proporzione di successi statisticamente diversa da quella attesa, nonostante l'ipotesi nulla sia corretta (l'ipotesi nulla del test sulla proporzione di successi è che la proporzione di sequenze che superano un test sia pari al livello di significatività con cui viene condotto il test di casualità). La porzione di falsi negativi nel caso in cui l'ipotesi nulla sia corretta è pari al livello di significatività con cui viene condotto il test sulla proporzione. Se ad esempio si considerasse una suite composta da 100 test di casualità e il livello di significatività del test sulla proporzione di successi di ciascun test fosse pari a 0,01, allora dovrebbe capitare che in media un test di casualità non rispetti la proporzione di successi, per affermare che il generatore produca sequenze che appaiono casuali. Utilizzando questo criterio di superamento di una suite, la probabilità che il generatore non superi la suite non è pari alla probabilità che anche solo un test non venga superato, ma si tiene in considerazione la possibillità che un certo numero di test non vengano superati dal generatore. Di conseguenza, la probabilità di rifiutare il generatore anche se esso produce sequenze che appaiono casuali scende notevolmente.

Il secondo problema riguarda la dipendenza tra i test presenti all'interno di una suite. Dal momento che il numero di test inseriti in una batteria è limitato, è importante avere test di natura diversa, in modo che l'insieme scelto sia in grado di identificare quante più possibili deviazioni dall'uniformità. Questo è il motivo per cui i test all'interno di una suite dovrebbero essere indipendenti. Esistono diverse strategie per valutare l'indipendenza tra i test di casualità in una suite e le idee riportate nel presente capitolo rientrano tra queste. Le altre strategie diffuse in letteratura sono invece discusse nel Paragrafo 3.1.

2.1 Concetti di indipendenza

Il concetto di indipendenza si ritrova nella teoria della probabilità. Prima di arrivare ad una definizione di indipendenza tra due test statistici introduciamo alcune definizioni di base.

Definizione 2 (Eventi indipendenti) *Dato uno spazio di probabilità* (Ω, F, P)*, due eventi qualsiasi* E *e* F*, si dicono indipendenti se e solo se* $P(E \cap F) = P(E)P(F)$

Nel caso in cui i due eventi non siano indipendenti, allora si dice che essi sono *dipendenti*. È importante notare fin da subito che la definizione di indipendenza non implica che i due

insiemi siano *incompatibili* (o *disgiunti*). A tal proposito, introduciamo la definizione di eventi incompatibili.

Definizione 3 (Eventi incompatibili) *Due eventi qualsiasi E e F sono detti incompatibili se non possono verificarsi contemporaneamente, ovvero se $E \cap F = \emptyset$.*

Poiché la probabilità dell'evento impossibile è nulla, se due eventi oltre ad essere incompatibili sono anche indipendenti allora si ha

$$0 = P(\emptyset) = P(E \cap F) = P(E)P(F) = 0$$

Da questa osservazione segue che se E e F sono eventi con probabilità diverse da zero, allora incompatibilità e indipendenza non possono verificarsi allo stesso tempo. Generalizziamo ora la nozione di eventi indipendenti, nel caso in cui siano considerati tre o più eventi. In questo caso esistono due definizioni di indipendenza: una più debole che prende il nome di *indipendenza a due a due* e una più forte, detta *indipendenza reciproca* (o *mutua indipendenza*).

Definizione 4 (Eventi indipendenti a due a due) *Gli eventi $E_1, E_2, \ldots, E_n$ si dicono indipendenti a due a due se, comunque se ne scelgono due, essi sono eventi indipendenti. In formule:*

$$E_1, E_2, \ldots, E_n \text{ indipendenti a due a due } \iff P(E_i \cap E_j) = P(E_i)P(E_j)\ \forall i, j$$

Definizione 5 (Eventi reciprocamente indipenendenti) *Gli eventi $E_1, E_2, \ldots, E_n$ si dicono reciprocamente indipendenti se e solo se*

$$P\left(\bigcap_{i \in I} E_i\right) = \prod_{i \in I} P(E_i),\ \forall \text{ sottoinsieme } I \subseteq \{1, 2, \ldots, n\}$$

Notare che per verificare che $n \geq 3$ eventi siano reciprocamente indipendenti bisogna verificare che la probabilità della loro intersezione sia uguale al prodotto delle loro probabilità e che la probabilità dell'intersezione di tutte le possibili famiglie di eventi sia uguale al prodotto delle singole probabilità. Da queste definizioni si deduce che se n eventi sono reciprocamente indipendenti allora sono indipendenti a due a due, ma in generale il viceversa non vale.

2.2 Indipendenza tra test di casualità

Nel paragrafo precedente sono state enunciate le definizioni principali di indipendenza, tuttavia queste nozioni sono relative ad eventi di probabilità. In questo paragrafo tradurremo queste definizioni in modo da definire e studiare l'indipendenza tra test di casualità. Nel contesto in esame si hanno n sequenze prodotte da un generatore di numeri casuali che vengono sottoposte a k test di casualità. È possibile costruire per ogni test un vettore di lunghezza n, in cui ogni cella assume valore 0 o 1 a seconda che la sequenza non abbia superato il test o abbia superato il test, rispettivamente. Interpretando i risultati dei test

di casualità in questo modo è possibile studiare la relazione, in termini di dipendenza, che esiste tra i diversi test. Come introdotto nel paragrafo precedente esistono diverse nozioni di indipendenza. Nel nostro caso potremmo decidere di studiare l'indipendenza tra tutte le possibili coppie di test. In questo caso parleremo di indipendenza a due a due. Un altro approccio possibile potrebbe essere quello di considerare più di due test e studiare l'indipendenza reciproca tra essi, arrivando al concetto di *coverage*. Infine è di particolare interesse osservare la dipendenza tra suite di test. Nei paragrafi seguenti verranno trattate separatemente queste tre possibilità.

2.2.1 Test indipendenti due a due

Il primo modo in cui è possibile verificare l'indipendenza di un certo numero di test appartenenti ad una suite è quello di controllare l'indipendenza tra tutte le possibili coppie di test. Nel paragrafo precedente è stata enunciata la definizione di eventi indipendenti a due a due (Definizione 4), tuttavia è importante precisare che cosa si intende per test di casualità indipendenti a due a due. L'idea è sostanzialmente quella di interpretare un test statistico come una variabile aleatoria che può avere due possibili outcome: successo o insuccesso. In particolare abbiamo già visto che un test statistico può essere rappresentato tramite una variabile di Bernoulli, T, che nel caso in cui l'ipotesi nulla del test statistico sia corretta, ovvero il generatore produce numeri casuali, ha probabilità di successo $1-\alpha$; $T \sim B(1-\alpha)$.

Per comprendere meglio il concetto di indipedenza tra test di casualità è utile introdurre due insiemi. Supponiamo di applicare il test T_i ad un insieme di n sequenze $S = \{s_1, \ldots, s_n\}$. Il test può essere interpretato come un algoritmo che divide lo spazio delle sequenze testate nei seguenti insiemi:

$$A_i = \{s_j \in S : T_i(s_j) = accetto\}; \tag{2.1}$$

$$R_i = \{s_j \in S : T_i(s_j) = rifiuto\}, \tag{2.2}$$

dove A_i e R_i sono la regione di accettazione e di rifiuto del test i, rispettivamente. Siccome vogliamo studiare l'indipendenza a due a due, possiamo limitarci a considerare solamente due test. Come si dispongono le due aree di rifiuto nel caso in cui i test siano indipendenti? Una risposta errata è dire che le due aree di rifiuto devono essere totalmente disgiunte, perché, se così fosse, conoscendo il risultanto di un test avremmo informazioni maggiori sull'esito dell'altro. Per comprendere meglio questo discorso supponiamo che l'area della zona di rifiuto per entrambi i test sia $1/2$ dell'area totale. Ossevare che l'ampiezza della regione di rifiuto è α, quindi in questo caso stiamo ipotizzando $\alpha = 1/2$. Avere le due regioni di rifiuto disgiunte significherebbe ricadere nella situazione (a) in Figura 2.1. In questo caso, conoscendo l'esito di un test, sappiamo con certezza stabilire l'esito dell'altro test e quindi i due test sono tutt'altro che indipendenti. Per dire che due test sono indipendenti, non dovrebbe essere possibile, conoscendo l'esito di un test, predire l'esito del secondo test con una probabilità maggiore rispetto a quella che avremmo senza conoscere l'esito del primo test. Nel caso in cui l'area della zona di rifiuto sia $1/2$ dell'area totale, per dire che i due test sono indipendenti dovremmo essere nel caso (b) della Figura 2.1. Infatti,

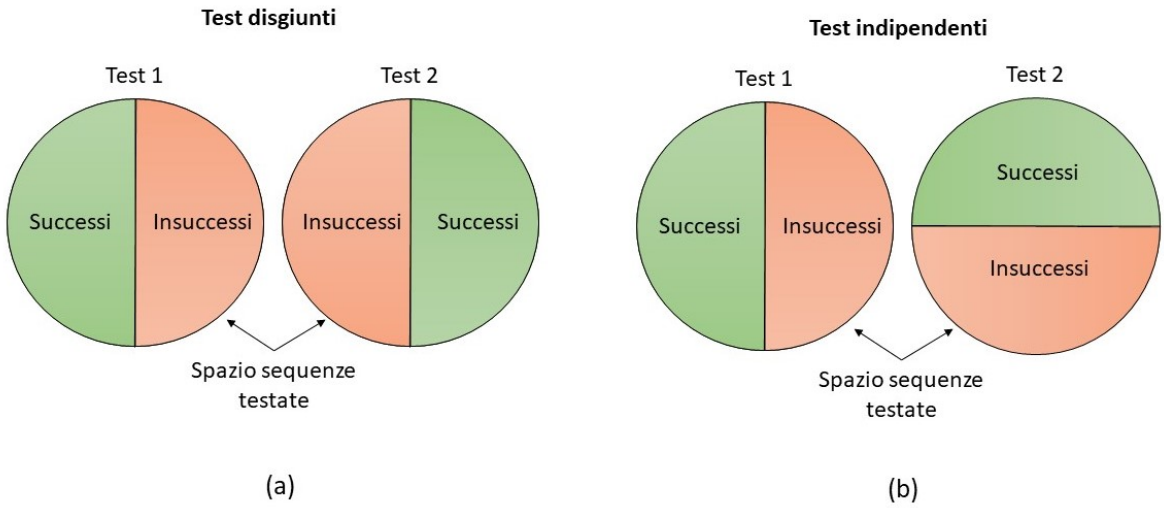

Figura 2.1. Disposizione delle regioni di accettazione e rifiuto di due test disgiunti (a) e di due test indipendenti (b), con entrambi livello di significatività pari a 1/2.

in questo caso, conoscendo l'esito del test 1, la probabilità che il test 2 sia superato o meno è sempre pari a 1/2. Notare che l'ampiezza della regione in cui entrambi i test non vengono superati è pari a 1/4 dell'area totale, che è equivalente al prodotto tra la probabilità di rifiutare il test 1 (uguale a 1/2) e la probabilità di rifiutare il test 2 (anch'essa pari a 1/2). Questa osservazione vale anche per tutte le altre regioni individuate dall'interazione tra i due test.

Generalizzando il discorso con un α generico, ci aspettiamo che l'ampiezza dell'intersezione delle due regioni di rifiuto sia pari a $n\alpha^2$, l'ampiezza dell'intersezione tra le due regioni di accettazione pari a $n(1-\alpha)^2$ e l'ampiezza dell'intersezione tra una regione di rifuto e una di accettazione pari a $n\alpha(1-\alpha)$. Le regioni di accettazione/rifiuto di due test indipendenti si dispongono, quindi, come in Figura 2.2. Notare che chiedendo solamente che l'ampiezza dell'intersezione tra le due regioni di rifiuto sia pari a $n\alpha^2$, automaticamente vengono soddisfatte le condizioni di tutte le altre regioni di interazione, in quanto le regioni di rifuto e accettazione di un singolo test sono una la complementare dell'altra. È possibile quindi limitare il discorso alle sole regioni di rifiuto. Possiamo quindi dare una definizione di test a due a due indipendenti nel seguente modo. Siano R_1 e R_2 le regioni di rifiuto relative a due test di casualità, diciamo che i due test sono indipendenti due a due se

$$P(R_1 \cap R_2) = P(R_1)P(R_2). \tag{2.3}$$

Operativamente questa condizione è equivalente a verificare che l'area dell'intersezione delle regioni di rifiuto sia pari a $n\alpha^2$. La definizione precedente è analoga alla Definizione 4, in cui vengono considerati come eventi: "rifiuto test 1" e "rifiuto test 2".

Dal punto di vista matematico, per dimostrare l'indipendenza tra due test statistici,

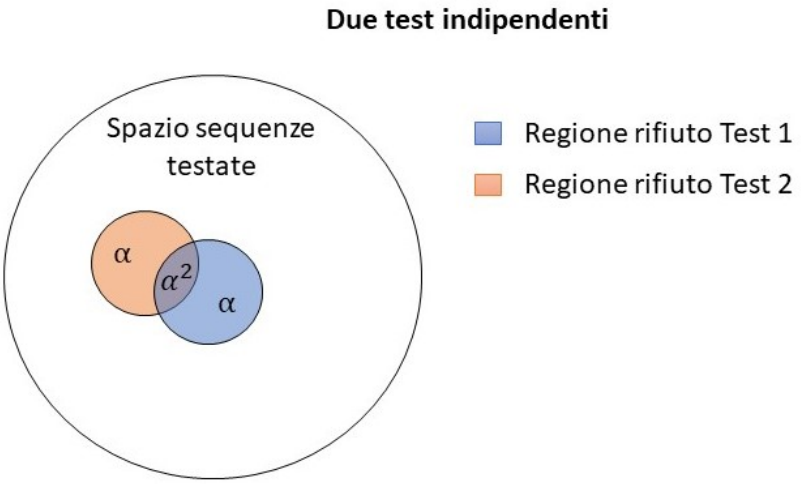

Figura 2.2. Rappresentazione della disposizione delle regioni di rifiuto di due test indipendenti nello spazio di tutte le sequenze testate.

utilizziamo il test di indipendenza chi quadro. Esso ci consente di verificare che l'intersezione tra le aree di accettazione e rifiuto relative a due test siano pari a quelle attese. Il test chi quadro è infatti utile quando si hanno a disposizione i conteggi di variabili categoriche e si vuole verificare la correlazione tra le due variabili. Consideriamo due test di casualità e supponiamo di essere in grado di ricavare per ogni test il numero di successi e insuccessi e la quantità di sequenze che superano entrambi i testi, solamente uno dei due o non li superano entrambi. A tal proposito introduciamo le seguenti variabili:

- $n_{1.}$ il numero di successi del test di casualità 1
- $n_{0.}$ il numero di insuccessi del test di casualità 1
- $n_{.1}$ il numero di successi del test di casualità 2
- $n_{.0}$ il numero di insuccessi del test di casualità 2
- n_{11}, il numero di sequenze che superano entrambi i test;
- n_{10}, il numero di sequenze che superano il test 1 ma non il test 2;
- n_{01}, il numero di sequenze che superano il test 2 ma non il test 1;
- n_{00}, il numero di sequenze che non superano nè il test 1 nè il test 2.

Conoscendo tutti questi dati è possibile costruire una tabella di contingenza come quella riportata in Tabella 2.1. A questo punto è possibile applicare un test chi quadro per valutare l'indipendenza tra i due test statistici. Il test chi quadro è esattamente come quello utilizzato per verificare che la distribuzione dei p-value sia uniforme, descritto

	Successi Test 2	Insuccessi Test 2	Tot
Successi Test 1	n_{11}	n_{10}	$n_{1.}$
Insuccessi Test 1	n_{01}	n_{00}	$n_{0.}$
Tot	$n_{.1}$	$n_{.0}$	n

Tabella 2.1. Tabella di contingenza tra due test di casualità.

nel paragrafo 1.4.2. Tuttavia, in questo caso le frequenze osservate empiricamente sono $n_{11}, n_{10}, n_{01}, n_{00}$ e le frequenze teoriche attese, $\bar{n}_{11}, \bar{n}_{10}, \bar{n}_{01}, \bar{n}_{00}$, sono calcolate supponendo che i due test statistici siano indipendenti. La statistica del test si ottiene tramite la formula:

$$X^2 = \frac{(n_{11} - \bar{n}_{11})^2}{\bar{n}_{11}} + \frac{(n_{10} - \bar{n}_{10})^2}{\bar{n}_{10}} + \frac{(n_{01} - \bar{n}_{01})^2}{\bar{n}_{01}} + \frac{(n_{00} - \bar{n}_{00})^2}{\bar{n}_{00}}. \tag{2.4}$$

La distribuzione di questa statistica è una chi quadro con $\nu = 1$ gradi di libertà, in quanto i gradi di libertà in un test di indipendenza si ricavano tramite la formula:

$$\textit{gradi di libertà} = (\textit{numero di colonne} - 1)(\textit{numero di righe} - 1).$$

Nel caso di test di indipendenza chi quadro tra due varibili dicotomiche, le frequenze attese sono calcolabili in maniera piuttosto semplice, conoscendo la probabilità di successo o insuccesso di ciascun test. L'aver supposto che il generatore produca numeri casuali è fondamentale, in quanto ci consente di conoscere la probabilità di successo (e di conseguenza di insuccesso) di un test di casualità. Facendo riferimento alla teoria dei test statistici abbiamo che la probabilità di successo di ciascun test di casualità è pari a $1-\alpha$, mentre la probabilità di insuccesso è pari a α. Supponendo che i due test di casualità siano indipendenti e che abbiano lo stesso livello di significatività, α, si trova che il numero atteso di sequenze che passano entrambi i test è pari a $\bar{n}_{11} = n(1-\alpha)^2$, il numero atteso di sequenze che superano un test mentre l'altro no è pari a $\bar{n}_{10} = \bar{n}_{01} = n\alpha(1-\alpha)$ e il numero di sequenze che non passano nessuno dei due test è uguale a $\bar{n}_{00} = n\alpha^2$. Le frequenze attese sono riassunte nella Tabella 2.2. Conoscendo le frequenze attese è possibile calcolare la statistica chi quadro 2.4, e il p-value ad esso associata. Più il p-value

	Successi Test 2	Insuccessi Test 2	
Successi Test 1	$(1-\alpha)^2$	$(1-\alpha)\alpha$	$1-\alpha$
Insuccessi Test 1	$\alpha(1-\alpha)$	α^2	α
	$1-\alpha$	α	1

Tabella 2.2. Frequenze teoriche in un test di indipendenza chi quadro, supponendo che i due test di casualità dividano lo spazio delle sequenze con lo stesso livello di significatività α.

calcolato sarà piccolo, maggiore sarà l'evidenza della dipedenza tra i due test di casualità.

Le frequenze teoriche riportate in Tabella 2.2 sono valide solamente nel caso in cui i due test di casualità abbiano lo stesso livello di significatività. Nel caso più generale, in cui il test 1 ha livello di significatività α_1 e il test 2 ha livello di significatività α_2, seguendo un ragionamento analogo al precedente, le frequenze teoriche attese che si trovano sono pari a quelle riportate in Tabella 2.3.

	Successi Test 2	Insuccessi Test 2	
Successi Test 1	$(1-\alpha_1)(1-\alpha_2)$	$(1-\alpha_1)\alpha_2$	$1-\alpha_1$
Insuccessi Test 1	$\alpha_1(1-\alpha_2)$	$\alpha_1\alpha_2$	α_1
	$1-\alpha_2$	α_2	1

Tabella 2.3. Frequenze teoriche in un test di indipendenza chi quadro, supponendo che i due test di casualità dividano lo spazio delle sequenze con due livelli di significatività differenti.

È importante precisare che il test di indipendenza chi quadro non fornisce nessuna informazione sull'intensità della dipendenza trovata. Un valore elevato della statistica X^2 indica che vi è una forte evidenza nei confronti della dipendenza trovata ma non implica che questa associazione sia forte, dal momento che per un dato grado di associazione il valore X^2 aumenta all'aumentare della numerosità campionaria. Per questo motivo potrebbe essere utile, per indagare la forza dell'associazione tra due test, introdurre una misura relativa che assume valore nell'intervallo $[-1, 1]$: l'indice ϕ, conosciuto come media quadratica delle contingenze. Nel caso di un test chi quadro 2x2 l'indice ϕ è definito tramite:

$$\phi = \frac{n_{11}n_{00} - n_{10}n_{01}}{\sqrt{n_{1.}n_{0.}n_{.1}n_{.0}}}.$$

In questo caso è possibile dimostrare che l'indice ϕ è equivalente al coefficiente di correlazione di Pearson tra le variabili che sono considerate nel test chi-quadro. Questo significa che è possibile, nel nostro caso, ricavare il coefficiente di correlazione tra due test di casualità a partire dal test chi-quadro.

2.2.2 Test reciprocamente indipendenti

Con l'approccio presentato nel paragrafo precedente, studiamo l'indipendenza tra i test all'interno di una suite, controllando la dipendenza tra tutte le coppie di test possibili. Questa strategia deriva dalla definizione di eventi indipendenti a due a due. Tuttavia abbiamo visto che esiste anche un concetto più forte, ossia quello di eventi reciprocamente indipendenti. In questo paragrafo cercheremo di tradurre questa definizione nel caso di un insieme di test di casualità. Come prima, immaginiamo un test di casualità come un processo decisionale che divide lo spazio di tutte le sequenze testate in due regioni: una di accettazione e una di rifiuto. Nel caso in cui l'ipotesi nulla sia soddisfatta, siamo in grado di quantificare la dimensione di entrambe le regioni. Ipotizzando un livello di significatività del test, α, la regione di rifiuto avrà dimensione $n\alpha$, mentre quella di accettazione $n(1-\alpha)$, dove n è il numero di sequenze a cui viene sottoposto il test. Ragionando sulle aree di rifiuto

e di conseguenza sulle aree di accettazione, abbiamo visto che due test sono indipendenti due a due quando l'area dell'intersezione delle regioni di rifiuto è pari a $n\alpha^2$, e l'intersezione delle aree di accettazione è $n(1-\alpha)^2$. Nel caso in cui la suite sia costituita da solamente due test, l'indipendenza a due a due equivale all'indipendenza reciproca. Se supponiamo di avere $m \geq 3$ test, l'indipendenza a due a due tra tutte le coppie di test, non è più sufficiente per affermare che i test siano anche reciprocamente indipendenti.

Definiamo il concetto di mutua indipendenza tra m test di casualità nel seguente modo. Siano $R_1, \ldots, R_m$, le regioni di rifiuto relative a ciascun test, diciamo che i test sono reciprocamente indipendenti se

$$P\left(\bigcap_{i\in I} R_i\right) = \prod_{i\in I} P(R_i),\ \forall\ \textit{sottoinsieme}\ I \subseteq \{1,2,\ldots,m\}.$$

Ciò equivale a verificare che l'area dell'intersezione di ogni sottoinsieme $R \subseteq \{R_1, R_2, \ldots, R_m\}$, sia pari a $\alpha^{|R|}n$, dove $|R|$ indica la cardinalità dell'insieme R. Notare che, per motivazioni analoghe a quelle esposte nel paragrafo 2.2.1, è sufficiente concentrarsi sulle regioni di rifiuto. Per comprendere meglio il concetto di indipendenza reciproca e la differenza con l'indipendenza a due a due, consideriamo per semplicità il caso $m = 3$. Per poter dire che i tre test siano reciprocamente indipendenti è necessario che essi siano indipendenti a due a due e che l'intersezione di tutte e tre le aree di rifiuto sia pari a $\alpha^3 n$ (ragionando sulle regioni di accettazione, si avrebbe che l'intersezione delle tre aree di accettazione deve essere pari a $(1-\alpha)^3 n$). Il concetto di indipendenza reciproca è più forte rispetto a quello di indipendenza a due a due. Si ha infatti che l'indipendenza reciproca implica l'indipendenza a due a due, ma il viceversa non vale. È possibile costruire esempi di test indipendenti a due a due ma non reciprocamente indipendenti. Si consideri ad esempio il caso in cui le regioni di rifiuto di tre test di casualità si dispongono come in Figura 2.3

In questo contesto è utile introdurre la definizione di coverage di una suite. La coverage di una suite è definita come il rapporto tra il numero di tutte le sequenze che falliscono almeno un test presente nella suite sul numero totale di sequenze testate. La coverage, in termini assoluti, può essere pensata come l'ampiezza dell'unione di tutte le regioni di rifiuto di ogni test presente nella suite. Nel caso in cui i test all'interno della suite siano reciprocamente indipendenti è possibile ricavare la coverage attesa della suite, grazie al seguente teorema.

Teorema 1 *Sia α il livello di significatività di m test di casualità reciprocamente indipendenti. Allora la coverage attesa di questa collezione di test è:*

$$\sum_{i=1}^{m} (-1)^{i+1} \binom{m}{i} \alpha^i = 1 - (1-\alpha)^m. \tag{2.5}$$

Osservare che, in generale, il viceversa non vale; infatti è possibile riportare esempi di suite che hanno coverage pari a 2.5, ma i test da cui sono composte non sono tra loro reciprocamente indipendenti. L'unica affermazione valida è che se, attraverso i dati ricavati da una suite, si trova una coverage diversa da quella attesa, allora sicuramente i test non saranno reciprocamente indipendenti. Di seguito, viene data un giustificazione informale a questo

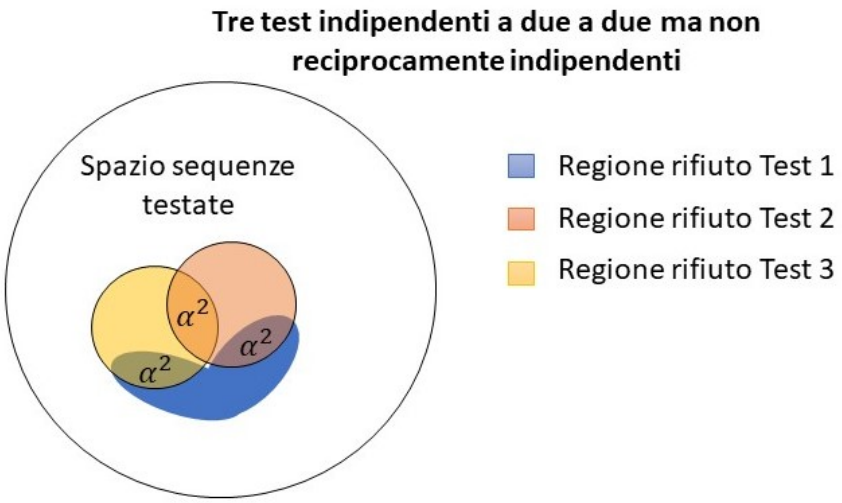

Figura 2.3. Rappresentazione della disposizione delle aree di rifiuto di tre test tra loro indipendenti a due a due, ma non reciprocamente indipendenti.

teorema. Date $R_1, \ldots, R_m$, le regioni di rifiuto di m test, interpretiamo la coverage come l'ampiezza dell'unione di queste aree:

$$\left| \bigcup_{i=1}^{m} R_i \right|, \tag{2.6}$$

dove con la notazione $|S|$ indichiamo la cardinalità dell'insieme S. Per il *principio di inclusione-esclusione* [17], l'unione di un numero finito di insiemi può essere riscritta come,

$$\left| \bigcup_{i=1}^{m} R_i \right| = \sum_{i=1}^{m} |R_i| - \sum_{1 \le i < j \le m} |R_i \cap R_j| + \sum_{1 \le i < j < k \le m} |R_i \cap R_j \cap R_k| - \ldots + (-1)^{m+1} |R_1 \cap \ldots \cap R_m|, \tag{2.7}$$

che in forma compatta, diventa,

$$\left| \bigcup_{i=1}^{m} R_i \right| = \sum_{\emptyset \neq J \subseteq \{1,\ldots,m\}} (-1)^{|J|+1} \left| \bigcap_{j \in J} R_j \right|. \tag{2.8}$$

Un caso speciale di questa formula è quando la dimensione dell'intersezione delle regioni di rifiuto dipende solamente dal numero di regioni considerate. In altre parole è possibile ricavare una forma semplificata, quando $R_J := \bigcap_{j \in J} R_j$ ha la stessa cardinalità ($a_k = |R_J|$), per ogni sottoinsieme $J \subseteq \{1, \ldots, m\}$, di grandezza k. In questo caso, l'equazione 2.8, diventa:

$$\left| \bigcup_{i=1}^{m} R_i \right| = \sum_{k=1}^{m} (-1)^{k+1} \binom{m}{k} a_k. \tag{2.9}$$

Finora abbiamo considerato le regioni di rifiuto come dei veri e propri insiemi. Tuttavia nel nostro caso sarebbe corretto interpretarle come eventi di probabilità. Esiste una formulazione probabilistica analoga del principio di inclusione-esclusione, valida nel caso in cui la probabilità dell'intersezione di un qualsiasi sottoinsieme di k eventi dipenda solamente dal numero di eventi considerati. In questo caso si considera la probabilità dell'intersezione delle regioni di rifiuto e si dimostra che essa è pari a:

$$\mathbf{P}\left(\bigcup_{i=1}^{m} R_i\right) = \sum_{k=1}^{m} (-1)^{k+1} \binom{m}{k} r_k, \tag{2.10}$$

dove $r_k = \mathbf{P}(R_J)\ \forall J \subset \{1, \ldots, m\}$ con $|J| = k$. Osservare che, nelle ipotesi del teorema, questa riscrittura è possbile. Infatti siccome i test sono reciprocamente indipendenti e hanno lo stesso livello di significatività α, ogni insieme R_J di cardinalità k, ha probabilità α^k, per definizione di eventi reciprocamente indipendenti. Sostituendo $r_k = \alpha^k$, nell'equazione 2.10, si ricava proprio la formula della coverage attesa.

2.2.3 Indipendenza tra suite di test

In quest'ultimo paragrafo sull'indipendenza tra test di casualità si vuole estendere il discorso al concetto di indipendenza tra suite di test. Il caso più semplice in questo contesto potrebbe essere quello di determinare l'indipendenza tra una suite di test ed un test di casualità. Si pensi ad esempio di possedere una suite composta da m test, tra loro reciprocamente indipendenti e di volerne aggiungere uno purché esso sia indipendente dall'insieme di test già presenti. Utilizzando le nozioni introdotte nei paragrafi precedenti, un modo per verificare l'indipendenza di questo test aggiuntivo rispetto alla suite è quello di verficare che esso sia indipendente a due a due con ciascun test all'interno della suite o che il nuovo insieme di test, comprendente quello aggiuntivo, sia ancora composto da test reciprocamente indipendenti. Tuttavia, questo metodo richiede molto sforzo computazionale, tanto più il numero di test presenti all'interno della suite è grande. Inoltre, questa strategia diventa più complicata quando si vuole determinare l'indipendenza tra due suite di test, o più in generale l'indipendenza di un insieme di suite.

In questo contesto è utile considerare ogni suite come se fosse un singolo test; in questo modo, per verificare l'indipendenza tra più suite di test si ricade nel casi descritti nei paragrafi precedenti. Precisiamo fin da subito che le suite tra cui vogliamo valutare la dipendenza devono essere composte da test tra loro reciprocamente indipendenti. Questa assunzione si rivelerà utile in seguito. Inoltre, è necessario specificare come rappresentare la suite tramite un singolo test. In particolare, bisogna stabilire un criterio per determinare quando un generatore supera o no una suite di test. Le possibilità sono fondamentalmente tre:

1) Il generatore supera la suite nel caso in cui ogni test di casualità presente all'interno della suite venga superato;

2) Il generatore supera la suite se almeno un test di casualità presente all'interno della suite viene superato;

3) Il generatore supera la suite se una certa porzione di test di casualità presenti all'interno della suite vengono superati.

La terza opzione è una via di mezzo tra le prime due, più il numero di successi richiesti si avvicina al totale più ci si avvicina all'opzione 1. Viceversa, se la percentuale di test superati richiesta diminuisce, ci si avvicina all'opzione 2. Osservare inoltre che nel primo caso, la regione di rifiuto della suite è l'unione delle regioni di rifiuto di ciascun test, mentre, nel secondo caso, la regione di rifiuto della suite è l'intersezione delle regioni di rifiuto di ciascun test. Nella Figura 2.4 sono riportate le regioni di accettazione e rifiuto di una suite,

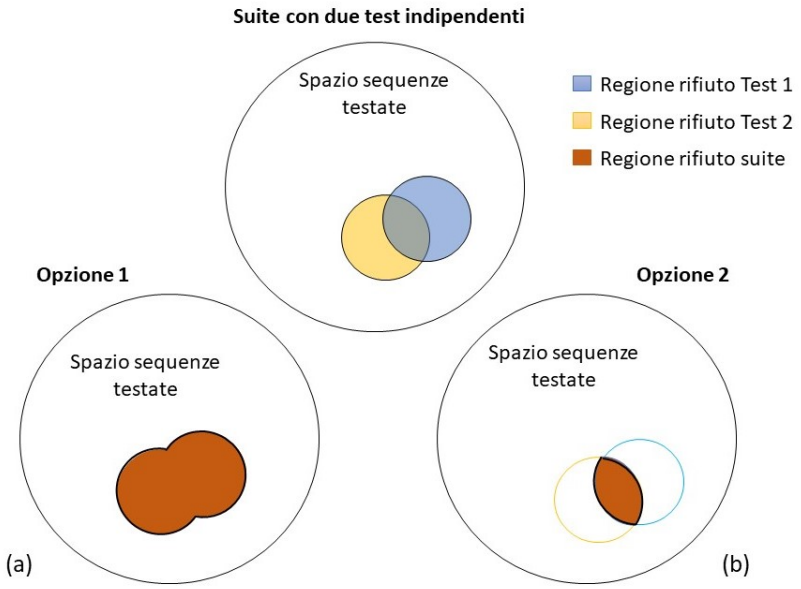

Figura 2.4. Area di rifiuto di una suite composta da due test. Il caso (a) corrisponde alla scelta di considerare superata una suite solamente se entrambi i test sono superati. Il caso (b) corrisponde alla scelta di considerare superata una suite se almeno uno dei due test è superato.

composta da due test, nel caso in cui venga interpretata come un test seguendo l'opzione 1 (grafico a) o l'opzione 2 (grafico b). Una volta scelta la modalità con cui tradurre la suite in un singolo test, siccome abbiamo supposto che i test di casualità all'interno della suite sono indipendenti, possiamo determinare l'ampiezza attesa dell'area di accettazione, e di conseguenza anche quella dell'area di rifiuto. A questo punto, conoscendo la proporzione di successo/insuccesso del test che rappresenta la suite, possiamo applicare le definizioni di indipendenza tra test di casualità riportate nelle sezioni precedenti.

Per comprendere meglio l'idea di questa strategia facciamo un esempio. Supponiamo di voler determinare la dipendenza tra una suite, composta da due test reciprocamente indipendenti, ed un singolo test di casualità che ha livello di significatività pari a α_2. La suite è composta da due test di casualità, entrambi con livello di significatività pari ad α_1. Per determinare l'ampiezza della regione di accettazione e rifiuto della suite è

necessario scegliere la modalità con cui stabilire se una sequenza passa o meno la suite. Supponendo di considerare la prima opzione, l'ampiezza della regione di accettazione sarà pari a $(1-\alpha_1)^2$ mentre quella della regione di rifiuto uguale a $1-(1-\alpha_1)^2 = 2\alpha_1 - \alpha_1^2$. Per determinare la dipendenza tra la suite e il test singolo, applichiamo un test di indipendenza chi quadro. Le frequenze teoriche nel caso in cui la suite e il test siano indipendenti sono riportate in Tabella 2.4. Considerando invece la seconda opzione l'area di accettazione è

	Successi Test	Insuccessi Test	
Successi Suite	$(1-\alpha_1)^2(1-\alpha_2)$	$(1-\alpha_1)^2\alpha_2$	$(1-\alpha_1)^2$
Insuccessi Suite	$(2\alpha_1-\alpha_1^2)(1-\alpha_2)$	$(2\alpha_1-\alpha_1^2)\alpha_2$	$(2\alpha_1-\alpha_1^2)$
	$1-\alpha_2$	α_2	1

Tabella 2.4. Frequenze teoriche del test di indipendenza chi quadro, tra una suite composta da due test, entrambi con livello di significatività pari ad α_1, interpretata secondo la modalità (1) e un test di casualità con livello di significatività pari ad α_2.

pari a $1-\alpha_1^2$ e quella di rifuto è α_1^2. La tabella delle frequenze teoriche in questo caso è la Tabella 2.5.

	Successi Test	Insuccessi Test	
Successi Suite	$(1-\alpha_1^2)(1-\alpha_2)$	$(1-\alpha_1^2)\alpha_2$	$1-\alpha_1^2$
Insuccessi Suite	$\alpha_1^2(1-\alpha_2)$	$\alpha_1^2\alpha_2$	α_1^2
	$1-\alpha_2$	α_2	1

Tabella 2.5. Frequenze teoriche del test di indipendenza chi quadro, tra una suite composta da due test, entrambi con livello di significatività pari ad α_1, interpretata secondo la modalità (2) e un test di casualità con livello di significatività pari ad α_2.

È importante osservare che l'indipendenza tra una suite e l'altra, trovata seguendo la strategia proposta in questo paragrafo, non implica in alcun modo che l'insieme composto dai test di casualità di entrambe le suite sia formato da test reciprocamente indipendenti.

2.3 Test di casualità disgiunti

Una domanda a cui non è facile dare una risposta è la seguente: una suite composta da test di casualità indipendenti tra loro è da preferirsi ad una suite composta da test tra loro disgiunti, o vale il viceversa? Nel paragrafo 2.2 abbiamo compreso cosa significa avere un insieme di test tra loro indipendenti; avendo capito i ragionamenti fatti, si dovrebbe essere in grado di intuire anche il significato di test disgiunti, tuttavia prima di provare a rispondere alla domanda iniziale, definiamo in maniera precisa questo concetto.

Diciamo che un insieme è composto da test disgiunti, nel caso in cui ogni sequenza non supera al più un test. In altre parole, utilizzando l'interpretazione di test di casualità

come un processo decisionale in grado di suddividere lo spazio delle sequenze in due regioni, una di accettazione e l'altra di rifiuto, possiamo affermare che un insieme è composto da test disgiunti nel caso in cui l'intersezione delle aree di rifiuto sia nulla. In Figura 2.5 è riportato un esempio di test disgiunti. Osserviamo che l'intersezione delle aree di rifiuto è nulla, mentre le regioni di accettazione sono inevitabilmente sovrapposte. Ciò dipende

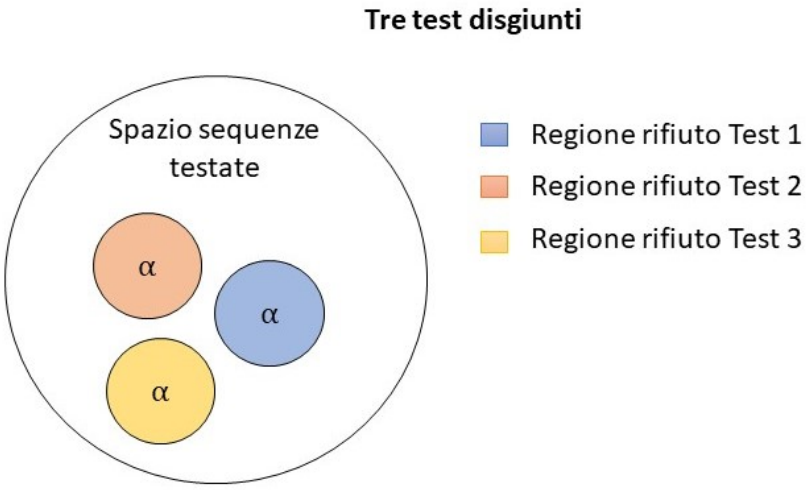

Figura 2.5. Rappresentazione della disposizione delle aree di rifiuto di tre test di casualità disgiunti.

dall'ampiezza delle regioni di rifiuto e quindi dal livello di significatività α. Notare inoltre che se α fosse maggiore di $1/m$, dove m è il numero di test considerati, allora l'insieme di m test non potrà mai essere formato da test digiunti. Tuttavia, siccome useremo solitamente valori di $\alpha \in [0{,}001; 0{,}01]$, suite con un numero di test m minore di 100 (o addirittura 1000 nel caso di $\alpha = 0{,}001$), potrebbero essere composte da test disgiunti.

Per verificare che una suite sia composta da test disgiunti, è necessario controllare che la regione di rifiuto di ogni test non sia sovrapposta alla regione di rifiuto di un altro. In altre parole possiamo limitarci a controllare che per ogni possibile coppia di test, l'intersezione delle due aree di rifiuto sia nulla. Per valutare se due test sono disgiunti possiamo utilizzare un test chi quadro in cui le frequenze teoriche attese sono pari a quelle riportate in Tabella 2.6.

Ora che abbiamo precisato il significato di test di casualità disgiunti torniamo alla domanda che ci siamo posti all'inizio di questo paragrafo. È preferibile una suite composta da test indipendenti tra loro o una suite composta da test tra loro disgiunti? In realtà non c'è una risposta precisa. L'idea maggiormente diffusa in letteratura è che una suite di test di casualità debba essere composta da test tra loro indipendenti, tuttavia avere una suite composta da test tra loro disgiunti ci permette di osservare la casualità del generatore da maggiori punti di vista. Il grande problema di avere test disgiunti, tuttavia, è che l'esito di un test fornisce informazioni sull'esito di altri test. Inoltre, siccome per valori di α piccoli, la sovrapposizione delle regioni di rifiuto di test tra loro indipendenti è particolarmente

	Successi Test	Insuccessi Test	
Successi Suite	$1 - \alpha_1 - \alpha_2$	α_2	$1 - \alpha_1$
Insuccessi Suite	α_1	0	α_1
	$1 - \alpha_2$	α_2	

Tabella 2.6. Frequenze teoriche nel caso in cui due test di casualità siano disgiunti e siano condotti con livello di significatività differente.

ridotta, l'area di copertura individuata da una suite composta da test tra loro disgiunti non è molto più grande di quella individuata da una suite con lo stesso numero di test tra loro indipendenti.

Capitolo 3

Dipendenze tra test della Suite del NIST

L'obiettivo di questo documento è quello di ottenere uno studio completo delle dipendenze tra i test di casualità presenti all'interno della Suite del NIST [1]. Gli approcci con cui sono state cercate le dipendenze tra i test di casualità sono quelli descritti nel Capitolo 2. In questa trattazione è stata condotta un'analisi completa sull'indipendenza a due a due tra tutte le coppie di test e un'analisi parziale sull'indipendenza reciproca. A differenza degli studi simili diffusi in letteratura, la suite del NIST è stata considerata nella sua interezza; infatti, sebbene i test proposti dal NIST siano 15, alcuni di essi hanno la possibilità di essere condotti utilizzando parametri diversi, dando origine a molteplici versioni dello stesso test di casualità. In questo studio è stato deciso di considerare le differenti versioni di un test di casualità come dei test distinti. Prima di includere nello studio tutti i test di casualità della Suite del NIST sono state svolte delle analisi approfondite su alcuni sottoinsiemi di test, composti da tutte le possibili versioni di un test di casualità. Tra questi, vi sono i test derivati dal Non Overlapping Template Matching Test, dal Random Excursions Test e dal Random Excursions Variant Test. Il campione di sequenze utilizzato per condurre lo studio sulle dipendenze tra test è composto da 100.000 sequenze di lunghezza pari ad un milione di bit.

Il presente capitolo è strutturato come segue. La Sezione 3.1 include alcuni studi presenti in letteratura, riguardanti la dipendenza tra i test di casualità della Suite del NIST. Nel Paragrafo 3.2 è riportato il metodo con il quale sono state prodotte le sequenze utilizzate durante lo svolgimento delle analisi di indipendenza. Nei paragrafi successivi sono esposti gli studi effettuati per valutare la dipendenza tra i test di casualità presenti nella Suite del NIST. Nella Sezione 3.3 la dipendenza tra test è studiata utilizzando il concetto di indipendenza a due a due, mentre nella Sezione 3.4 è descritto un'approccio indiretto per valutare l'indipendenza reciproca tra test. Non si tratta di una vera e propria analisi completa dell'indipendenza reciproca tra test, in quanto la procedura descritta non sempre è in grado di stabilire se l'insieme considerato sia composto o meno da test reciprocamente indipendenti; tuttavia, si ritiene che questa strategia possa essere molto utile come punto di partenza per verifica dell'indipendenza reciproca tra test. Infine, nel

Paragrafo 3.5 sono riassunti i risultati ottenuti dagli studi effettuati.

3.1 Lavori precedenti

Il problema dell'indipendenza tra i test inclusi nella suite del NIST è stato affrontato da diversi matematici nel corso degli ultimi anni. Nelle stesse pubblicazioni del NIST è introdotta la problematica relativa alla dipendenza tra i test proposti e viene condotto un breve studio che afferma l'indipendenza dei test all'interno della suite [1].

Per comprendere le dipendenze tra i test di casualità, il NIST effettua un'analisi ai fattori sui p-value risultanti dall'applicazione dei test di casualità. Questa analisi, conosciuta anche come Principal Component Analysis (PCA), è una tecnica che consente di mappare un insieme di dati appartenenti ad uno spazio di dimensione maggiore, in un nuovo spazio la cui dimensione è molto minore. Questo processo è strettamente connesso con il concetto della compressione, nella teoria dell'informazione. Quando viene effettuata una PCA ad un insieme di dati, solitamente si ottengono poche componenti principali, in grado di esprimere una grande proporzione della variabilità dei dati. Nel contesto dei test di casualità, il numero di componenti principali sufficienti ad esprimere una grande proporzione di variabilità può essere utilizzato per quantificare il numero di dimensioni di non casualità analizzate dalla suite. Svolgendo questo tipo di analisi sull'insieme composto da tutti i test nella Suite, il NIST ha concluso che non vi è un grande ridondanza nei test da esso proposti. Tuttavia, questa affermazione non è del tutto corretta, dal momento che studi successivi, basati su approcci differenti, hanno dimostrato l'esistenza di alcune dipendenze tra i test di casualità della Suite del NIST. Inoltre, in aggiunta alle relazioni tra test già osservate da alcuni lavori, durante le analisi svolte in questa tesi, sono state riscontrate ulteriori coppie di test di casualità tra loro dipendenti. I contributi aggiuntivi di questo lavoro sono riportati nelle sezioni successive. Di seguito, vengono invece riportati i risultati maggiormente rilevanti effettuati sullo studio delle dipendenze tra i test di casualità del NIST prima della stesura di questa tesi.

Nel 2008, Turan, Doganaksoy e Boztas [23] pubblicano un primo studio approfondito, che tratta la dipendenza tra test di casualità. Essi definiscono due test indipendenti, nel caso in cui le loro regioni di rifiuto siano indipendenti per qualsiasi livello di significatività α. In particolare, per esprimere la dipendenza del test i sul test j, viene calcolato il rapporto tra il numero di sequenze che falliscono entrambi i test e il numero di sequenze che falliscono il test i. In formule la dipendenza tra i due test è espressa mediante $\frac{|R_i \cup R_j|}{|R_i|}$, dove $|R_i|$ è la cardinalità della regione di rifiuto del test $i - esimo$. In questo studio viene svolta un'analisi di dipendenza tra alcuni test appartenenti alla Suite del NIST, utilizzando un campione di sequenze di lunghezza pari a 30 bit. Vista la dimensione ridotta delle sequenze, gran parte dei test del NIST risultano imprecisi, in quanto la lunghezza minima richiesta per poter considerare attendibili i test è molto più grande (vedi Tabella 1.2). Le dipendenze trovate, pertanto, risultano irrilevanti. Tuttavia, è importante sottolineare che le idee introdotte in questo articolo hanno posto le basi per lo studio della dipendenza tra i test di casualità.

Successivamente, Sulak et al. [22] pubblicano uno studio maggiormente dettagliato che si basa sullo stesso concetto di indipendenza introdotto nel lavoro precedente ([23]).

Il numero di sequenze considerate in questo caso è pari a 100.000 e ogni sequenza ha lunghezza pari a 38.912 bit, in modo da poter applicare correttamente nove test della suite del NIST (test che in Tabella 1.2 hanno un requisito di lunghezza minima inferiore o uguale a 38.912 bit). Oltre a valutare la dipendenza tra tutte le coppie di test, gli autori confrontano la coverage di una suite con la coverage attesa, introducendo un nuovo concetto: la *coverage efficiency* di una suite di test. Essi definiscono la coverage efficiency come il rapporto tra la coverage della suite calcolata empiricamente e la coverage teorica, ottenuta tramite la formula: $1 - (1 - \alpha)^k$, dove k è il numero di test nella suite. Questa misura, tuttavia, favorisce suite composte da test disgiunti. Nel caso in cui vi siano test disgiunti, infatti, la coverage calcolata empiricamente sarà maggiore di quella teorica, determinando un rapporto maggiore di 1. Inoltre, è possibile avere un rapporto maggiore di 1 anche nel caso in cui la proporzione di sequenze che hanno p-value minore di α è maggiore di α, per qualche test.

Nel 2015, Doganaksoy et al. [9] propongono un metodo alternativo per valutare la dipendenza tra test di casualità, basato sull'utilizzo del coefficiente di correlazione di Pearson. Dato un insieme di k test di casualità ed un insieme di n sequenze casuali, è possibile ottenere un vettore di p-value di lunghezza n per ciascun test di casualità, applicando i test alle sequenze a disposizione. Una volta collezionati gli n p-value relativi ad ogni test, nei vettori $P_1, \ldots, P_k$, si definisce la dipendenza tra due test sulla base della correlazione presente tra i vettori di p-value P_i e P_j, relativi ai due test. Più il coefficiente di correlazione sarà lontano da 0, maggiore sarà la dipendenza tra i due test. In questo documento vengono distinti due gruppi di test. Un primo composto da test che possono essere applicati anche a sequenze di lunghezza ridotta e un secondo insieme composto da test che invece richiedono un lunghezza maggiore delle sequenze. Sono stati condotti due esperimenti: nel primo caso sono state considerate 200.000 sequenze, lunghe 2^{10} bit, in cui vengono analizzate le dipendenze tra i test che possono essere applicati a sequenze ridotte, mentre nel secondo caso 200 sequenze di lunghezza 2^{20}, in modo da poter valutare la dipendenza tra tutti i 17 test principali presenti nella suite del NIST. Oltre ai 15 test del

Coppie di test di casualità dipendenti
Frequency − Cumulative Sums (forward)
Frequency − Cumulative Sums (backward)
Cumulative Sums (forward) − Cumulative Sums (backward)
Serial 1 − Serial 2
Approximate Entropy − Serial 1
Approximate Entropy − Serial 2
Random Excursions (x = 1) − Random Excursions Variant (x = 1)

Tabella 3.1. Coppie di test di casualità dipendenti trovate in [9].

NIST riportati in Tabella 1.1, sono stati considerati come test distinti le due versioni del Serial Test e il Cumulative Sums Test è stato applicato in entrambe le modalità (forward e backward). In entrambi i casi le sequenze testate sono state generate tramite AES

(Advanced Encryption Standard). Tramite gli esperimenti condotti sono state trovate delle dipendenze, le quali sono riportate in Tabella 3.1

Nel 2019, Emil Simion e Paul Burciu [21] si pongono come obiettivo quello di migliorare i risultati trovati in [9]. Il metodo proposto per valutare l'indipendenza tra due test i e j è il seguente. Per entrambi i test definiamo le variabili aleatorie T_i e T_j, che seguono una distribuzione di Bernoulli, e assumono valore 1 se il test è superato, altrimenti valore 0. Successivamente viene stimato il valore di: $P(T_i, T_j) - P(T_i)P(T_j)$. Se i due test sono indipendenti allora questa quantità dovrebbe essere vicina a zero. Le dipendenze tra i

Coppie di test di casualità dipendenti
Frequency − Cumulative Sums (forward)
Frequency − Cumulative Sums (backward)
Cumulative Sums (forward) − Cumulative Sums (backward)
Serial 1 − Serial 2
Random Excursions (x = 1) − Random Excursions Variant (x = 1)

Tabella 3.2. Coppie di test di casualità dipendenti trovate in [21].

test del NIST ottenute, sono riportate in Tabella 3.2 e si basano su un campione di 100 sequenze. In particolare sono state valutate le dipendenze tra i test variando la lunghezza delle sequenze presenti nel campione, considerando stringhe lunghe 1, 2 e 5 milioni bit. In tutti i casi le dipendenze trovate sono le stesse.

I recenti lavori sullo studio dell'indipendenza dei test di casualità all'interno della Suite del NIST si basano su approcci differenti. Un esempio è l'articolo scritto da Karell-Albo et al. nel 2020 [11]. Essi propongo di stabilire la dipendenza tra i test utilizzando la *mutua informazione*. Il principale vantaggio è che questa misura è in grado di catturare anche le dipendenze di tipo non lineare, rispetto al coefficiente di correlazione di Pearson usato nei lavori precedenti. Il metodo proposto consiste nel calcolare la mutua informazione tra le sequenze di p-value (o delle test statistic) relative a diversi test di casualità. Per stimare la mutua informazione tra tutte le coppie (P_i, P_j) di sequenze di p-value, si usa l'espressione della mutua informazione basata sull'entropia. La mutua informazione tra due variabili X e Y, è definita come

$$I(X,Y) = \int_Y \int_X p(x,y) log\left(\frac{p(x,y)}{p(x)p(y)}\right)\ dx\ dy, \tag{3.1}$$

dove $p(x,y)$ è la funzione di probabilità congiunta di X e Y, e $p(x)$ e $p(y)$ sono le distribuzioni di probabilità marginali di X e Y, rispettivamente. La mutua informazione tra due variabili X e Y, può anche essere espressa in termini di entropia, attraverso,

$$I(X,Y) = H(X) + H(Y) - H(X,Y), \tag{3.2}$$

dove $H(X)$ e $H(Y)$ sono le entropie marginali delle variabili X e Y, rispettivamente e $H(X,Y)$ è l'entropia congiunta di entrambe le variabili. Per stabilire l'indipendenza tra i

Coppie di test di casualità dipendenti
Frequency − Cumulative Sums (forward)
Frequency − Cumulative Sums (backward)
Frequency − Random Excursions (x = -4)
Frequency − Random Excursions Variant (x = -9)
Frequency − Runs
Cumulative Sums (forward) − Cumulative Sums (backward)
Cumulative Sums (forward) − Random Excursions (x = -4)
Cumulative Sums (forward) − Random Excursions Variants (x = -9)
Cumulative Sums (forward) − Runs
Cumulative Sums (backward) − Non Overlapping Template
Cumulative Sums (backward) − Random Excursions (x = -4)
Cumulative Sums (backward) − Random Excursions Variants (x = -9)
Cumulative Sums (backward) − Runs
Serial 1 − Serial 2
Random Excursions (x = -4) − Random Excursions Variant (x = -9)
Longest Run of Ones − OverlappingTemplate

Tabella 3.3. Coppie di test di casualità dipendenti trovate in [11].

due test viene formulato il seguente test statistico,

$$H_0 : I(P_i, P_j) = 0$$
$$H_a : I(P_i, P_j) > 0$$

dove l'ipotesi nulla H_0 indica l'indipendenza tra i test. Le dipendenze trovate all'interno della Suite del NIST, seguendo l'approccio descritto, sono riportate in Tabella 3.3. I risultati sono ottenuti utilizzando un campione di 10.000 sequenze di lunghezza pari a un milione di bit, generate tramite diversi generatori, quali SHA-1, Linear Congruential, Micali-Schnorr e Blum-Blum-Shub. È possibile notare come in questo caso le dipendenze trovate siano maggiori rispetto a quelle individuate tramite gli approcci precedenti.

3.2 Metodi di generazione delle sequenze

Per poter valutare correttamente la dipendenza tra i test di casualità della Suite del NIST, utilizzando gli approcci descritti nel Capitolo 2, è necessario avere a disposizione delle sequenze binarie casuali. Il numero di sequenze considerate dovrebbe essere sufficiente per ottenere dei risultati attendibili durante le analisi svolte. In particolare, per valutare correttamente l'indipendenza a due a due è necessario avere almeno 100.000 sequenze casuali. Infatti, dal momento che per osservare la dipendenza tra due test viene applicato un test chi quadro, bisogna accertarsi che le frequenze teoriche attese siano almeno 10 [8]. Il test di indipendenza chi quadro è condotto tra i risultati di successo/insuccesso di due

test di casualità eseguiti con un livello di significatività $\alpha = 0{,}01$. Facendo riferimento alla tabella delle frequenze teoriche attese 2.2, riportata nel Capitolo 2, si nota che il numero minimo di sequenze per avere una frequenza attesa di 10 sequenze che non superano entrambi i test è pari a 100.000. Inoltre, la lunghezza delle sequenze considerate deve essere pari ad almeno un milione di bit, dal momento che alcuni test di casualità presenti nella Suite del NIST sono formalizzati in modo che le distribuzioni di riferimento siano valide soltanto con sequenze di tale lunghezza minima (vedi Tabella 1.2). Riassumendo, per le analisi successive sono necessarie almeno 100.000 sequenze di lunghezza minima pari ad un milione di bit.

Per disporre di questo quantitativo di sequenze è necessario utilizzare un generatore di numeri pseudocasuali. La scelta di tale generatore deve essere svolta con particolare attenzione, in quanto le analisi sulla dipendenza tra test si basano sul fatto che le sequenze utilizzate siano casuali. È indispensabile, quindi, avere un DRBG che sia indistinguibile da un generatore ideale. Dalla teoria, siamo a conoscenza che l'AES-DRBG è un generatore considerato indistinguibile da uno ideale. Supponendo che esso si comporti come un generatore ideale, è possibile utilizzare le sequenze da esso prodotte come se fossero delle sequenze casuali. Tuttavia, non vi è nessuna certezza assoluta sulla veridicità di queste supposizioni e soprattutto sul fatto che l'implementazione del generatore utilizzata consenta di generare sequenze indistinguibili da sequenze ideali. Un modo per avere una verifica indiretta sulla randomicità posseduta dalle sequenze prodotte dall'AES-DRBG è quello di prendere un altro generatore, basato su una logica completamente differente, anch'esso considerato indistinguibile da un generatore ideale e verificare che essi si comportino nella stessa maniera. In altre parole, se i risultati complessivi sulle statistiche di superamento dei test di casualità o sulle statistiche relative alla dipendenza tra test, ottenute utilizzando due generatori considerati indistinguibili da uno ideale sono le stesse, allora è possibile concludere con molta confidenza che entrambi i generatori siano effettivamente indistinguibili da uno ideale. Notare che comunque non si ha ancora una certezza assoluta, in quanto i due generatori potrebbero deviare dal comportamento ideale nello stesso modo. Tuttavia, questa ipotesi è molto meno probabile, in quanto esistono infinite deviazioni dal comportamento ideale. Una verifica formale di questo tipo, anche se non fornisce una certezza assoluta, ci consente di ottenere una maggiore confidenza riguardo l'affidabilità delle sequenze utilizzate. Questo studio è stato svolto utilizzando come secondo generatore lo SHA-DRBG, anch'esso considerato indistinguibile da un generatore ideale.

Nella Sezione 3.2.1 vengono descritti i meccanismi e le caratteristiche dei due generatori utilizzati: l'AES-DRBG e lo SHA-DRBG. Successivamente, nella Sezione 3.2.2 vedremo come i risultati ottenuti utilizzando sequenze prodotte dai due generatori tendano allo stesso valore all'aumentare del numero di sequenze considerate.

3.2.1 Generatori utilizzati

Le sequenze utilizzate per ricavare i risultati riportati nelle sezioni seguenti sono state generate utilizzando due PRNG di natura diversa:

- CTR-AES-128 DRBG

- HMAC-SHA512 DRBG

Il primo PRNG appartiene alla famiglia di generatori, il cui meccanismo interno è basato sui cifrari a blocchi. In questo caso, l'algoritmo di cifratura utilizzato è l'Advanced Encryption Standard (AES) con chiavi di 128 bit. Il secondo generatore invece è un PRNG basato su una funzione hash, in questo caso SHA-512. Di seguito viene descritto, per entrambi i PRNG, il processo di generazione delle sequenze. Ciascun insieme di sequenze considerato negli esperimenti successivi è stato generato utilizzando la seguente procedura:

1) Generazione di una stringa casuale di 256 bit tramite il generatore di numeri casuali implementato nel modulo OS in Python 3.11.0 [18]. La stringa è utilizzata come fonte di entropia per istanziare in un caso l'AES-DRBG e nell'altro lo SHA-DRBG e produrre una sequenza di lunghezza $m = numero\_sequenze \cdot entropy\_len$, dove $numero\_sequenze$ è il numero di sequenze che si vogliono generare e $entropy\_len$ è il numero di bit richiesti per l'entropia fornita in input al processo di istanziazione del generatore. La sequenza di lunghezza m è suddivisa, rispetto al numero di sequenze, in modo da ottenere 256 bit di entropia per la generazione di ciascuna sequenza casuale.

2) Generazione delle sequenze casuali in modo indipendente, utilizzando come entropia in input al processo di istanziazione un segmento ricavato della sequenza generata al punto (1). Il generatore viene, quindi, istanziato nuovamente prima della generazione di ogni sequenza.

Di seguito viene descritto il funzionamento dei due generatori considerati.

AES CTR DRBG

Il generatore appartiene alla classe dei DRBG basati su cifrari a blocchi. La sicurezza di questi generatori, oltre a dipendere dall'implementazione, si basa fortemente sulla sicurezza del cifrario a blocchi utilizzato. In questo caso, la primitiva crittografica utilizzata è l'AES con chiavi di 128 bit. La progettazione di questo tipo di generatori è descritta nella documentazione del NIST [10]. L'implementazione utilizzata di questo generatore è scritta in Python [6] e segue con precisione le raccomandazioni del NIST. Come descritto nella Sezione 1.3, ogni PRNG possiede 3 funzioni principali: funzione di istanziazione, di reseed e di generazione. Per gli scopi di questo documento, la funzione di reseed non è stata utilizzata, in quanto viene effettuata una nuova istanziazione ogni volta che è necessario generare una sequenza. La funzione di istanziazione richiede in input una quantità di entropia pari a 256 bit e opzionalmente una "personalization string": una stringa scelta dall'utente di lunghezza pari a 256 bit. La personalization string utilizzata negli esperimenti è quella di default: $personalization\_string = 0^{256bit}$. La funzione di istanziazione è riassunta nello pseudocodice 1. Una volta istanziato il DRBG è possibile generare le sequenze chiamando ripetutamente la funzione di generazione. La funzione di generazione richiede in input solamente la lunghezza in bit della sequenza casuale. Il funzionamento del processo di generazione è descritto nello pseudocodice 2.

Algorithm 1 AES_DRBG Instantiate function

Require: $Entropy\_input$, $personalization\_string$
Ensure: Key, V

```
seed ← entropy_input ⊕ personalization_string
Key ← 0^keylen                                  ▷ keylen = 128 bit
V ← 0^keylen                                    ▷ keylen = 128 bit
temp ← b' '
while len(temp) ≤ seedlen do                    ▷ seedlen = 256 bit
    V ← (V + 1) mod 2^128
    output_block ← AES.Encrypt(Key, V)
    temp ← temp + output_block
end while
temp ← temp ⊕ seed
Key ← leftmost(temp, 128)
V ← rightmost(temp, 128)
```

Algorithm 2 AES_DRBG Generate function

Require: $Byte\_required > 0$, V, Key
Ensure: $Random\_sequence$

```
temp ← b' '
while len(temp) ≤ Byte_required do
    V ← (V + 1) mod 2^128
    output_block ← AES.Encrypt(Key, V)
    temp ← temp + output_block
end while
Random_sequence ← temp
```

HMAC SHA DRBG

Questo secondo generatore fa parte di una classe di generatori basati su una keyed hash function. Una keyed hash function, conosciuta anche come HMAC (Hash-based Message Authentication Code), è un algoritmo che utilizza una chiave e una funzione hash per produrre un codice che garantisce l'integrità e l'autenticità di un messaggio. Per il funzionamento del generatore è necessario, quindi, utilizzare una funzione di hash. Nel generatore di riferimento scelto, la funzione hash considerata è SHA-512. Siccome la sicurezza di questo generatore è riposta fondamentalmente nella funzione hash utilizzata, questa dovrebbe permettere di raggiungere o superare il livello di sicurezza richiesto dal caso d'uso. Come con il generatore precedente è stata utilizzata un'implementazione in Python, in linea con le raccomandazioni del NIST [14]. Anche in questo caso per la generazione di ciascuna sequenza si è preferito istanziare il generatore ogni volta. La funzione di istanziazione e di generazione sono differenti rispetto al caso precedente. Uno pseudocodice che descrive il meccanismo di funzionamento della funzione di istanziazione è riportato nello schema 3, mentre per quanto riguarda la funzione di generazione il processo è descritto nello schema 4.

Algorithm 3 HMAC_DRBG Instantiate function

Require: $Seed$
Ensure: Key, V
$Key \leftarrow b'\backslash x00' * 64$ ▷ $outlen = 512$ bit
$V \leftarrow b'\backslash x01' * 64$ ▷ $outlen = 512$ bit
$Key \leftarrow HMAC(Key, V \oplus b'\backslash x00' \oplus seed)$
$V \leftarrow HMAC(Key, V)$

Algorithm 4 HMAC_DRBG Generate function

Require: $Byte\_required > 0$, V, Key
Ensure: $Random\_sequence$
$temp \leftarrow b''$
while $len(temp) \leq Byte\_required$ **do**
$V \leftarrow HMAC(Key, V)$
$temp \leftarrow temp + V$
end while
$Random\_sequence \leftarrow temp$

3.2.2 Validazione dei generatori

L'obiettivo di questo paragrafo è quello di verificare che il generatore AES-DRBG si comporti come un generatore ideale. La strategia utilizzata per avere maggiore confidenza sulla casualità delle sequenze utilizzate si basa su una verifica indiretta, in cui si accerta che il comportamento del generatore AES-DRBG sia uguale a quello di un altro generatore considerato ideale: lo SHA-DRBG. Partendo dal presupposto che entrambi i generatori siano

indistinguibili da un generatore ideale, dovrebbe essere impossibile trovare una differenza tra di essi. Nel nostro caso, valuteremo il comportamento dei due generatori basandoci sul risultato del test di indipendenza a due a due tra i test di casualità. Qualora venga osservato un comportamento differente utilizzando le sequenze prodotte da uno dei due generatori, sarebbe possibile stabilire con una probabilità maggiore di 1/2 se una sequenza sia stata prodotta da un generatore piuttosto che un altro. In questo caso cadrebbe un assunto forte della crittografia, in quanto i due generatori sono considerati indistinguibili da quello ideale e di conseguenza indistinguibili tra loro. Ipotizzando che questo ragionevolmente non dovrebbe succedere, data la quantità di studi fatti in letteratura su questi due generatori, le ulteriori possibili spiegazioni di un comportamento differente potrebbero essere dovute ad un'implementazione scorretta di uno o entrambi i generatori oppure al fatto che il numero di sequenze utilizzate, per calcolare le diverse statistiche, non è sufficiente per poter osservare un comportamento uguale tra i due generatori. Per ovviare a questo possibile problema è stata condotta un'analisi considerando un numero di sequenze progressivamente maggiore. Gli studi effettuati sono stati condotti, per motivi di risorse computazionali limitate, soltanto su una coppia di test di casualità. In particolare, è stato verificato che il risultato ottenuto dal test di indipendenza svolto considerando la coppia formata dal Frequency Monobit Test e il Longest Run of Ones Test, utilizzando un insieme di sequenze prodotto tramite AES-DRBG ed uno formato da sequenze generate tramite lo SHA-DRBG, converga all'aumentare del numero di sequenze considerate.

In Figura 3.1 è riportato l'andamento del p-value relativo al test di indipendenza chi quadro effettuato tra i due test di casualità, all'aumentare del numero si sequenze considerate. La curva rossa indica l'andamento del p-value nel caso in cui le sequenze utilizzate per condurre il test di indipendenza siano prodotte tramite l'AES-DRBG. La curva blu invece è relativa ai risultati ottenuti utilizzando sequenze generate attraverso lo SHA-DRBG. È possibile osservare che all'aumentare del numero di sequenze considerate le due curve convergano. I risultati riportati in Figura 3.1 sono ottenuti conducendo il test di indipendenza chi quadro con frequenze teoriche attese calcolate secondo la Tabella 2.2, utilizzando $\alpha = 0{,}01$. Infatti, in linea teorica quando viene applicato un test di casualità ad un insieme di sequenze casuali, lo spazio delle sequenze testate viene diviso in due insiemi di proporzione $1 - \alpha$ e α, dove α è il livello di significatività del test di casualità. Prima di condurre il test di indipendenza chi quadro bisognerebbe controllare che questa proporzione venga rispettata; verifica fatta tramite il test sulla proporzione di successi descritto nel paragrafo 1.4.2. Nel caso in cui il test sulla proporzione di successi non venga superato, per poter ugualmente condurre il test di indipendenza chi quadro in modo corretto, è necessario utilizzare la proporzione esatta di insuccessi per calcolare le frequenze teoriche attese. In questo modo, il test di indipendenza è condotto correttamente, correggendo eventuali errori nella definizione delle frequenze teoriche attese dovuti all'utilizzo di una proporzione scorretta di insuccessi per qualche test di casualità. Nella Sezione 3.3.1 sono trattate con maggior riguardo le possibili cause di questo problema. Dal momento che la proporzione esatta di successi per il Frequency Test e il Longest Run of Ones Test ricavata mediante l'utilizzo di sequenze prodotte dai due generatori non è esattamente pari ad $1 - \alpha$, conviene condurre il test di indipendenza chi quadro utilizzando le proporzioni di successo corrette, calcolate sperimentalmente.

In Figura 3.2 è riportato il p-value relativo al test di indipendenza chi quadro svolto

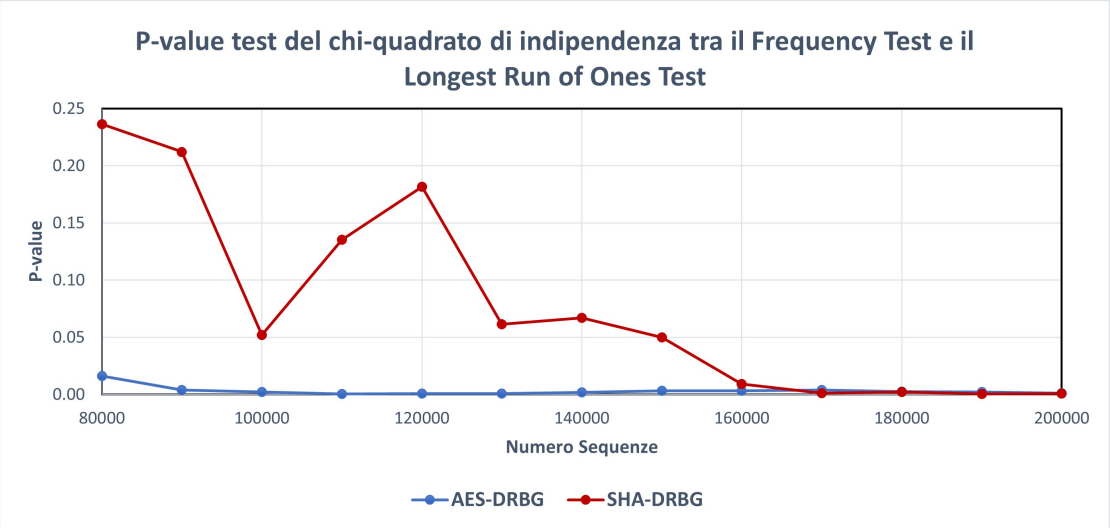

Figura 3.1. Andamento del p-value relativo al test di indipendenza chi quadrato, effettuato tra il Frequency Test e il Longest Runs of Ones Test, al variare del numero di sequenze considerate. La linea rossa corrisponde al caso in cui il test di indipendenza è effettuato utilizzando sequenze prodotte tramite l'AES-DRBG, mentre la linea blu corrisponde al test di indipendenza effettuato utilizzando sequenze generate tramite lo SHA-DRBG. Il test di indipendenza chi quadrato è effettuato utilizzando $\alpha = 0{,}01$ per ottenere le frequenze teoriche attese nel caso in cui i due test siano indipendenti, come specificato nella Tabella 2.2.

tra il Frequency Test e il Longest Run of Ones Test, condotto utilizzando le proporzioni di successo/insuccesso esatte per calcolare le frequenze teoriche attese. Come è possibile osservare dalla figura, all'aumentare del numero di sequenze considerate, il valore del p-value ottenuto mediante l'utilizzo di sequenze prodotte tramite AES-DRBG e quello ottenuto considerando le sequenze ottenute mediante lo SHA-DRBG, tendendono a convergere. Da questi risultati, possiamo concludere che all'aumentare del numero di sequenze considerate i due generatori si comportano allo stesso modo, per quanto riguarda il test di indipendenza tra la coppia considerata, come ci si aspetta da due generatori indistinguibili da generatori ideali. Nonostante il comportamento uguale per i due generatori sia stato confermato soltanto per una coppia di test, il risultato trovato garantisce una particolare confidenza nella casualità delle sequenze generate tramite l'AES-DRBG. Infatti, è bene tenere presente che anche confrondando i due generatori con tutti i test diffusi in letteratura non si avrebbe comunque una certezza assoluta sul fatto che essi siano indistinguibili. D'altra parte, quindi, anche una sola coppia di test è una validazione significativa.

A seguito di questa evidenza sperimentale indiretta, possiamo assumere con buona confidenza che il generatore AES-DRBG ed in particolare l'implementazione utilizzata, sia in grado di produrre sequenze indistinguibili da sequenze casuali. Notare che per le ragioni esposte in precedenza lo stesso discorso è valido per quanto riguarda lo SHA-DRBG. Tuttavia, per ottenere i risultati riportati nelle sezioni successive si è deciso di

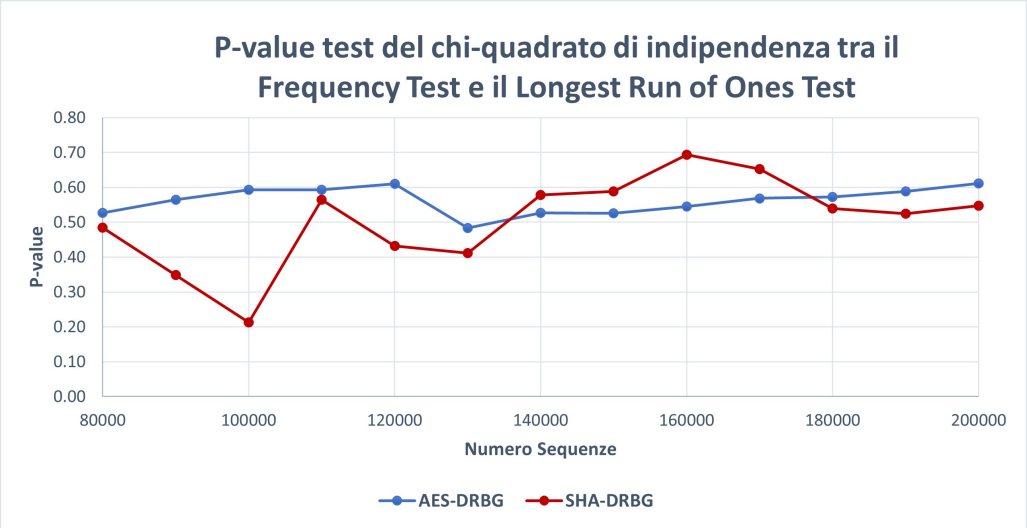

Figura 3.2. Andamento del p-value relativo al test di indipendenza chi quadrato, effettuato tra il Frequency Test e il Longest Runs of Ones Test, al variare del numero di sequenze considerate. La linea rossa corrisponde al caso in cui il test di indipendenza è effettuato utilizzando sequenze prodotte tramite l'AES-DRBG, mentre la linea blu corrisponde al test di indipendenza effettuato utilizzando sequenze generate tramite lo SHA-DRBG. Il test di indipendenza chi quadrato è effettuato utilizzando le proporzioni di successo calcolate sperimentalmente, per ottenere le frequenze teoriche attese nel caso in cui i due test siano indipendenti.

utilizzare per praticità un solo generatore ed è stato scelto l'AES-DRBG.

3.3 Indipendenza a due a due tra i test della Suite del NIST

In questo paragrafo sono riportati i risultati dello studio condotto per valutare l'indipendenza a due a due tra i test di casualità presenti nella Suite del NIST. Le sequenze utilizzate per ottenere tutti i risultati esposti in questa sezione sono state generate mediante l'AES-DRBG. Il campione di sequenze generato ha numerosità pari a 100.000 e ciascuna sequenza ha una lunghezza di un milione di bit.

Lo studio dell'indipendenza a due a due tra i test presenti all'interno della Suite del NIST è stato condotto su diversi sottoinsiemi di test:

1) Insieme composto da 14 test. Comprende Frequency (Monobit), Frequency within a Block, Runs, Longest Run of Ones in a Block, Binary Matrix Rank, Discrete Fourier Transform (Spectral), Overlapping Template Matching, Mauer's Universal Statistical, Linear Complexity, Serial Version1 e Serial Version2, Approximate Entropy, Cumulative Sums in forward e in backward mode.

2) Insieme composto da 148 test: Non Overlapping Template Test condotto con tutti i possibili template non periodici di lunghezza 9 bit.

3) Insieme composto da 8 test relativi al Random Excursions Test. Ogni test corrisponde al controllo del numero di visite ad un determinato stato.

4) Insieme composto da 18 test relativi al Random Excursions Variant Test. Ogni test corrisponde al controllo del numero di visite ad un determinato stato.

5) Insieme composto da 188 test. Questo insieme comprende tutti i test presenti nei precedenti insiemi: i 14 test presenti nell'insieme (1), i 148 test della famiglia del Non Overlapping Template Matching Test, gli 8 test della famiglia del Random Excursions Test e i 18 test della famiglia del Random Excursions Variant Test.

Nel primo insieme sono stati esclusi il Non Overlapping Template Matching Test, il Random Excursions Test e il Random Excursions Variant Test. Il motivo dell'esclusione del Non Overlapping Template Matching Test è dovuto al fatto che esso comprende più test di casualità. Risulta conveniente, quindi, analizzare a parte questa famiglia di test, prima di considerarli insieme ai restanti test della suite. Il secondo insieme, infatti, è formato dal Non Overlapping Template Matching test utilizzando diversi template. Lo scopo è quello di osservare se vi sono dipendenze tra questo test condotto ricercando un template piuttosto che un altro. La causa dell'esclusione del Random Excursions e Random Excursions Variant è invece dovuta ad un vincolo sulle sequenze richiesto da questi due test. Il Random Excursions e Random Excursions Variant Test sono applicabili solamente a sequenze che presentano un numero di cicli inferiore a 500. Il numero di cicli presenti in una sequenza è pari al numero di visite allo stato 0, interpretando la sequenza come un cumulative sum random walk. Una definizione formale del numero di cicli in una sequenza è fornita nella sezione A.1. Gli insiemi (3) e (4) sono costituiti dai test appartenenti, rispettivamente, alla famiglia del Random Excursions e del Random Excursions Variant e sono utilizzati per studiare la relazione presente tra le diverse versioni di questi test. In particolare, si vuole osservare se ci sia qualche connessione tra il numero di visite ai diversi stati del random walk associato alla sequenza. Per poter valutare la dipendenza tra il Random Excursions e il Random Excursions Variant e i restanti test della suite è necessario che tutte le sequenze possiedano il numero di cicli minimo richiesto. L'insieme (5), che comprende tutti i test della suite, è analizzato considerando un campione composto da sequenze con un numero di cicli maggiore di 500. Vedremo che considerare solamente sequenze che soddisfano questa proprietà incide sulle performance di alcuni test.

3.3.1 Suite NIST escludendo Non Overlapping Template Matching, Random Excursions e Random Excursions Variant Test

L'obiettivo finale di questa tesi è quello di avere una caratterizzazione completa delle dipendenze presenti all'interno della Suite del NIST. All'interno di questa batteria, però, esistono alcuni test che necessitano di una trattazione approfondita prima di poter essere

considerati insieme a tutti gli altri. Tra questi, vi è il Non Overlapping Template Matching Test che può essere condotto in numerose versioni, ciascuna focalizzata sulla ricerca di un template differente. Data la lunghezza del template, il Non Overlapping Template Matching Test viene applicato più volte utilizzando tutti i possibili pattern aperiodici della lunghezza specificata. In realtà, quindi, il Non Overlapping Template Matching test comprende più test. Scegliendo, ad esempio, la lunghezza dei template pari a 9 (lunghezza di default) si hanno 148 versioni del Non Overlapping Template Matching Test. Ovviamente, maggiore sarà la lunghezza del template specificata, maggiore sarà il numero possibile di template aperiodici e di conseguenza crescerà il numero di test effettuati. Le lunghezze dei template consigliate dal NIST sono 9 o 10 bit. Utilizzando una di queste due lunghezze, l'insieme di tutte le versioni del Non Overlapping Tempate Matching Test diventa considerevole. Motivo per il quale, nei lavori precedenti, riportati nel Paragrafo 3.1, solitamente è utilizzata una sola versione di questo test, condotto utilizzando il template $'000000001'$ di lunghezza 9 bit. In linea di principio, però, fissata la lunghezza del pattern ogni possibile template dovrebbe essere ugualmente valido. Sarebbe corretto, quindi, utilizzare tutti i template possibili di una determinata lunghezza. Per il momento questo test sarà escluso e prima di considerarlo insieme ai restanti test della suite verrà condotta un'analisi dettagliata sulla famiglia dei Non Overlapping Template Matching test con template di lunghezza pari a 9, nel Paragrafo 3.3.2.

Gli altri test esclusi in questa analisi sono il Random Excursions e il Random Excursions Variant Test. Il motivo di questa esclusione è legato al fatto che questi due test, a differenza di tutti gli altri presenti nella Suite del NIST, vengono applicati ad una sequenza soltanto se il numero di cicli che possiede è maggiore di 500. L'esistenza di questo vincolo è dovuta all'esigenza di avere una frequenza attesa almeno pari a 5, in un test chi quadro presente all'interno del Random Excursions Test. Inoltre, per essere sicuri che le approssimazioni utilizzate per caratterizzare le statistiche presenti nel Random Excursions e Random Excursions Variant Test siano valide è necessario che il numero di cicli in una sequenza sia maggiore di: $\max\{0{,}005\sqrt{n}, 500\}$, dove n è la lunghezza in bit della sequenza considerata. Osservare che considerando sequenze di lunghezza pari a un milione, il vincolo sul numero di cicli minimo in una sequenza è pari a 500. Per poter confrontare questi test con i restanti, presenti nella Suite del NIST, è indispensabile utilizzare solamente sequenze con un numero di cicli maggiore di 500. Tuttavia, il fatto di considerare sequenze di questo tipo incide con i risultati di alcuni test, come vedremo nel Paragrafo 3.3.5. In particolare, il Frequency Monobit Test e il Cumulative Sums Test se applicati a sequenze con un numero di cicli maggiore di 500 tendono ad essere superati in proporzione maggiore rispetto a quanto atteso. Nell'Appendice A è riportata una trattazione approfondita riguardo questa interazione osservata.

Avendo motivato la momentanea esclusione del Non Overlapping Template Matching, del Random Excursions e del Random Excursions Variant Test, possiamo studiare la dipendenza tra i test rimanenti. I test considerati in questo studio, con i relativi parametri, sono riportati in Tabella 3.4. Le dipendenze tra questi test di casualità sono state valutate utilizzando un insieme composto da 100.000 sequenze prodotte tramite l'AES-DRBG. Dato un insieme di sequenze casuali, per valutare l'indipendenza a due a due di una batteria di test di casualità, si procede con i seguenti passi:

1) Si applicano i test di casualità, presenti nell'insieme di interesse, alle sequenze prodotte;

2) Si raccolgono i risultati di successo/insuccesso di ogni sequenza, per ciascun test di casualità;

3) Si verifica che, per ciascun test, la proporzione di successi sia corretta; ovvero sia statisticamente uguale alla propozione attesa $\alpha = 0{,}01$ (tutti i test di casualità vengono applicati utilizzando un livello di confidenza pari a 0,01);

4) Si applica il test di indipendenza chi quadro, descritto nella Sezione 2.2.1, a ciascuna coppia di test, utilizzando come frequenze attese quelle presenti in Tabella 2.2, specificando $\alpha = 0{,}01$. Il test di indipendenza produrrà un p-value, che consentirà di stabilire se i due test sotto esame siano tra loro dipendenti o meno.

Test	Parametri
Frequency (Monobit)	block length(M) = 128 bit
Frequency within a Block	
Runs	
Longest Run of Ones in a Block	
Binary Matrix Rank	
Discrete Fourier Transform (Spectral)	
Overlapping Template Matching	template = $'111111111'$
Mauer's Universal Statistical	
Linear Complexity	block length (M) = 500
Serial 1	block length (m) = 16
Serial 2	block length (m) = 16
Approximate Entropy	block length (m) = 10
Cumulative Sums	mode forward
Cumulative Sums	mode backward

Tabella 3.4. Sottoinsieme di test di casualità della Suite del NIST, considerato nella Sezione 3.3.1, con i rispettivi parametri utilizzati.

È importante sottolineare che nel caso in cui la verifica presente nel terzo step non venga soddisfatta per alcuni test, è comunque possibile proseguire con lo studio sulla dipendenza utilizzando però degli accorgimenti. Infatti, potrebbe capitare che alcuni test di casualità non superino il test sulla proporzione di successi. Nella Tabella 3.5 sono riportate le statistiche del test sulla proporzione di successi condotto per ciascuno dei 14 test considerati, utilizzando 100.000 sequenze generate tramite AES-DRBG. Si può notare come alcuni di questi test di casualità non superino il test sulla proporzione di sucessi attesa, tra cui il Discrete Fourier Transform e il Mauer's Universal Statistical Test. Condurre un test di indipendenza chi quadro come descritto nel quarto step, considerando un test di casualità che non rispetta la proporzione di successo potrebbe portare a conclusioni

scorrette, in quanto le frequenze teoriche attese riportate in Tabella 2.2 risulterebbero errate. Le frequenze in Tabella 2.2 sono infatti calcolate assumendo che entrambi i test sotto esame dividano l'insieme di tutte le sequenze in due insiemi di proporzione $1 - \alpha$ e α; cosa che non accade nel caso in cui un test non supera il controllo sulla proporzione di successi. Al fine di risolvere questo problema, per i test di casualità che non hanno una percentuale di insuccessi statisticamente uguale ad 0,01 viene utilizzata la proporzione di insuccessi calcolata sperimentalmente per ricavare le frequenze teoriche attese del test di indipendenza chi quadro. In questi casi si utilizza la Tabella 2.3, che consente di ottenere le frequenze attese nel caso in cui i due test dividano lo spazio delle sequenze con una proporzione diversa. In questo modo è possibile valutare correttamente le dipendenze tra test di casualità che dividono effettivamente lo spazio delle sequenze in accordo con il livello di significatività e test di casualità che deviano da questa ipotesi.

Test di casualità	Numero successi	Proporzione insuccessi	P-value test proporzione successi
Frequency (Monobit)	99.056	0,00944	0,0751
Frequency within a Block	99.014	0,00986	0,6563
Runs	99.052	0,00948	0,0984
Longest Run of Ones in a Block	98.913	0,01087	0,0057
Binary Matrix Rank	98.964	0,01036	0,2526
Discrete Fourier Transform [1]	98.818	0,01182	$7{,}2798 \cdot 10^{-09}$
Overlapping Template Matching	98.906	0,01094	0,0028
Mauer's Universal Statistical [1]	98.800	0,01200	$2{,}0651 \cdot 10^{-10}$
Linear Complexity	98.967	0,01033	0,2943
Serial 1	98.970	0,01030	0,3404
Serial 2	99.044	0,00956	0,1620
Approximate Entropy	98.987	0,01013	0,6795
Cumulative Sums (mode forward)	99.035	0,00965	0,2660
Cumulative Sums (mode backward)	99.026	0,00974	0,4086

Tabella 3.5. Statistiche relative agli esiti dei test di casualità riportati in Tabella 3.4 applicati ad un insieme di 100.000 sequenze generate tramite l'AES-DRBG.

Il problema appena affrontato è lo stesso evidenziato nel paragrafo 3.2.2. Di seguito, cercheremo di spiegarne le possibili cause. Facendo riferimento alle riflessioni svolte nel Capitolo 2, potrebbe capitare che, anche nel caso in cui le sequenze utilizzate siano casuali, il test sulla proporzione di successi non venga superato, in quanto, essendo un test statistico, ci potrebbero essere dei falsi negativi. Tuttavia, se in un insieme di test di casualità ridotto, come quello considerato in questa sezione, il test sulla proporzione di successi non

[1] Test di casualità che non superano il controllo sulla proporzione di successi. Per comprendere questi test nell'analisi di indipendenza viene utilizzata la proporzione di successi esatta (ricavata sperimentalmente) per calcolare le frequenze teoriche attese del test chi quadro.

viene superato da 2 test, si ha il sospetto che le sequenze considerate non siano casuali. Il test sulla proporzione di successi è condotto utilizzando un livello di significiatività pari a 0,0026 (come descritto nel Paragrafo 1.4.2); per cui, il numero di falsi negativi, nel caso in cui l'ipotesi nulla sia corretta, dovrebbe essere mediamente 2 o 3 su 1000 test considerati. Osservando i risultati del test sulla proporzione di successi applicato ai 14 test di casualità considerati in questa sezione, potremmo essere condotti a credere che il generatore utilizzato non produca sequenze che appaiono casuali. Tuttavia, è importante tenere presente che questa non è l'unica spiegazione possibile. Infatti, oltre ad esserci un problema nelle sequenze generate e quindi nei generatori o nelle loro implementazioni, la responsabilità di questa deviazione potrebbe essere dovuta alla formulazione dei test di casualità o alla loro implementazione. Per escludere il caso in cui il problema sia dovuto alle sequenze, è stata valutata la proporzione di successi per ciascun test, utilizzando sequenze prodotte tramite un altro generatore: lo SHA-DRBG. I risultati ottenuti tramite questo generatore sono sovrapponibili a quanto trovato utilizzando l'AES-DRBG. Questa verifica contribuisce ad aumentare la fiducia sul fatto che l'AES-DRBG e di conseguenza anche lo SHA-DRBG siano indistinguibili da un generatore ideale; assunzione verificata indirettamente già nella sezione 3.2.2. È possibile quindi supporre che il problema dell'insuccesso di diversi test sulla proporzione di successi non sia dovuto alle sequenze ma piuttosto ai test di casualità. Nella Sezione 1.4.2 e successivamente nell'Appendice B è stato osservato che la distribuzione dei p-value di alcuni test di casualità non è uniforme e i possibili valori assunti dai p-value sono ridotti rispetto al numero di campioni considerati. Proprio quest'ultimo problema potrebbe influenzare la corretta esecuzione di un test di casualità. Infatti, nel caso in cui un test di casualità venga condotto con un livello di significatività α che non è presente tra i valori possibili che il p-value relativo alla statistica del test può assumere, la proporzione di successi/insuccessi, anche nel caso in cui l'ipotesi nulla sia corretta, potrebbe non essere pari a quella attesa. Come osservato nell'Appendice B, questo fenomeno si verifica per i test di casualità Discrete Fourier Transform e Mauer's Universal Statistical.

Per poter proseguire nell'analisi sull'indipendenza a due a due tra i test di casualità, per i test che non soddisfano la proporzione di successi o comunque la soddisfano ma con un p-value vicino al limite della regione di rifiuto del test sulla proporzione di successi, utilizzeremo la proporzione di successi calcolata sperimentalmente per ricavare le frequenze teoriche attese del test di indipendenza chi quadro.

Di seguito sono riportati i risultati ottenuti. In Figura 3.3 è riportata una tabella che riassume gli esiti del test di indipendenza chi quadrato per ciascuna coppia possibile di test di casualità presenti nell'insieme considerato. Per ciascuna coppia di test è riportato il p-value relativo al test di indipendenza chi quadrato. Valori inferiori a 0,01 indicano l'evidenza di una dipendenza tra i test considerati. In Tabella 3.6 sono riassunte le dipendenze trovate con il relativo grado di evidenza espresso dal p-value. Valori del p-value molto piccoli indicano un'alta confidenza nella dipendenza trovata. Nella Tabella 3.6 è inoltre riportato per ciascuna coppia di test dipendenti, il valore della variabile ϕ, descritta nella Sezione 2.2.1. La variabile ϕ esprime l'intensità della dipendenza trovata; valori di ϕ prossimi a 1 indicano una forte intensità dell'associazione tra i due test di casualità. È interessante osservare che tra le diverse coppie di test dipendenti, l'intensità della correlazione non è la stessa: alcune coppie presentano una ϕ vicina a 0,7, mentre altre una ϕ

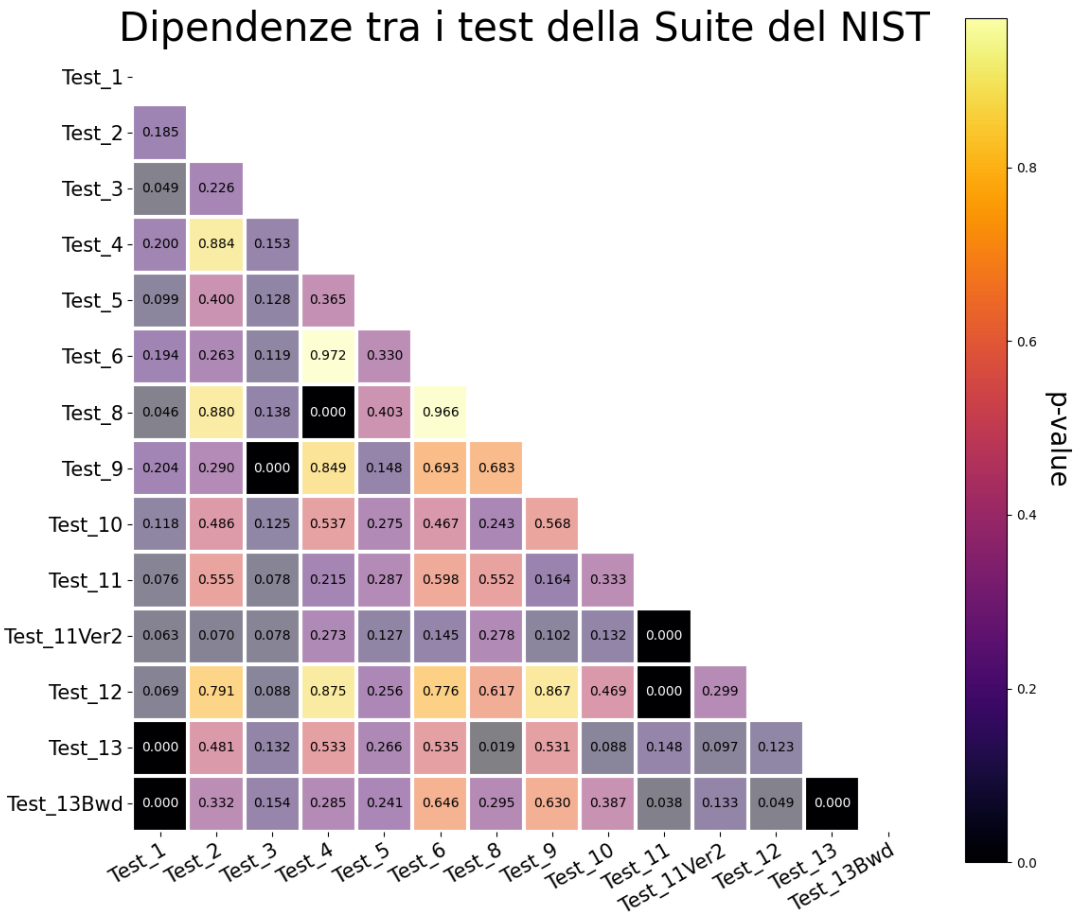

Figura 3.3. Risultati del test di indipendenza chi quadro effettuato tra ogni coppia di test di casualità presenti in Tabella 3.4. Per ciascuna coppia di test è riportato il valore del p-value relativo al test di indipendenza chi quadro. Una dipendenza tra due test di casualità, viene evidenziata nel caso in cui il p-value sia minore di 0,01. Valori bassi di p-value indicano una forte evidenza della dipendenza tra i due test di casualità considerati.

che supera di poco 0,01. Avere una ϕ prossima a 0,01, significa che la dipendenza trovata è di piccola intensità. Considerando, ad esempio, la coppia composta dai test di casualità Runs e Mauer's Universal Statistical, la correlazione tra essi è pari a 0,0126, valore che non dista molto da $\phi = 0$ che indicherebbe correlazione nulla tra i test. Tuttavia, seppure con una piccola intensità i due test sono correlati; infatti, osservando il numero di sequenze che non superano entrambi i test si trova che esso è pari a 29. Questo dato evidenzia, con un certo grado di confidenza, che i due test non siano indipendenti, in quanto il numero atteso di sequenze che non passano entrambi i test, nel caso in cui essi siano indipendenti, dovrebbe essere pari a 10 (considerando un campione di 100.000 sequenze).

Infine, è utile capire come le dipendenze tra i test di casualità trovate tramite questa

Coppie di test di casualità dipendenti	P-value test indipendenza	ϕ
Frequency − Cumulative Sums (forward)	0	0,7282
Frequency − Cumulative Sums (backward)	0	0,7160
Runs − Mauer's Universal Statistical	$9{,}3968 \cdot 10^{-5}$	0,0126
Longest Runs of Ones − Overlapping Template Matching	$3{,}4425 \cdot 10^{-6}$	0,0159
Serial 1 − Serial 2	0	0,2537
Serial 1 − Approximate Entropy	$8{,}4329 \cdot 10^{-8}$	0,0177
Cumulative Sums (forward) − Cumulative Sums (backward)	0	0,6228

Tabella 3.6. Coppie composte da test di casualità tra loro dipendenti. Per ciascuna coppia è riportato il p-value del test di indipendenza chi quadrato e il valore della media quadratica delle contingenze, ϕ (in questo caso l'indice ϕ è equivalente alla correlazione tra i due test di casualità).

analisi si relazionino con le dipendenze ottenute dai lavori precedenti, riportati nel Paragrafo 3.1. I risultati ottenuti sono in accordo con le dipendenze registrate attraverso i lavori precedentemente svolti. In particolare, tutte le dipendenze trovate dai lavori che utilizzano una strategia simile o comunque legata al test di indipendenza chi quadrato sono state ritrovate anche in questa analisi. A queste coppie di test dipendenti si aggiungono quella composta dal Longest Run of Ones e Overlapping Template Test, evidenziata solamente dal lavoro [11], che utilizza una strategia differente rispetto agli altri lavori e la coppia formata dal Runs Test e il Mauer's Universal Statistical Test, che invece non è riportata in nessuno dei testi analizzati.

3.3.2 Non Overlapping Template Matching Test

In questo paragrafo studiamo in maniera approfondita la dipendenza tra le diverse versioni del Non Overlapping Template Matching Test. Il Non Overlapping Template Matching Test ha come obiettivo quello di controllare che le occorrenze di uno specifico template non siano distanti rispetto al numero che ci si aspetta da una sequenza puramente casuale. In questo test, i template vanno scelti in modo che siano aperiodici, ovvero in modo che non sia possibile ricreare il template ripetendo più volte un template di lunghezza minore. Il Non Overlapping Template Matching test implementato nella Suite del NIST consente di specificare solamente la lunghezza dei template che si vogliono testare. Fissata la lunghezza del template vengono quindi applicati tanti test quanti sono i pattern aperiodici di tale lunghezza. Ad esempio, se la lunghezza del template fosse scelta pari a 4, verranno effettuati 6 test, in quanto 6 è il numero di pattern aperiodici di lunghezza 4: $0001, 0011, 0111, 1000, 1100, 1110$. Nei risultati riportati in questo paragrafo la lunghezza dei template è stata scelta pari a 9, lunghezza consigliata dal NIST. I template analizzati e di conseguenza i test effettuati sono quindi pari a 148. Per verificare l'indipendenza tra questo insieme di test procediamo in maniera analoga a come fatto nel paragrafo precedente. Applichiamo il Non Overlapping Template Matching Test 148 volte, utilizzando

ogni volta un template diverso, alle 100.000 sequenze generate tramite l'AES-DRBG. In questo modo otteniamo per ciascun test un vettore di lunghezza pari a 100.000, in cui l'elemento i-esimo è pari a 0 nel caso in cui la sequenza i-esima non abbia superato il test mentre è pari a 1 nel caso in cui essa abbia superato il test. Utilizzando questi risultati è possibile verificare in primo luogo, che la proporzione di successi/insuccessi per ciascun test sia rispettata e successivamente studiare l'indipendenza a due a due come spiegato nella Sezione 2.2.1.

Dall'analisi degli esiti dei 148 Non Overlapping Template Matching Test si osserva che due test presentano una proporzione di successi significativamente differente da $1 - \alpha$, dove $\alpha = 0{,}01$ è il livello di significatività con cui vengono condotti i Non Overlapping Template Matching Test. Per le motivazioni esposte nella Sezione 3.3.1 questo comportamento non mette in dubbio la casualità delle sequenze utilizzate. Infatti, è del tutto normale che all'aumentare di test di casualità effettuati alcuni non superino il controllo sulla proporzione di successi. In questo caso, avendo applicato 148 test di casualità, e utilizzando un livello di significatività sul test relativo alla proporzione di successi, pari a 0,01, ci si aspetta che in media uno o due test non superino il controllo sui successi. Per quanto detto nel paragrafo precedente, quando la proporzione di successi relativa ad un test di casualità è statisticamente differente dal livello di significatività atteso, è consigliato calcolare questa proporzione empiricamente e utilizzare quella quando si studiano dipendenze che includono quel test di casualità. Risultati grafici come quelli riportati nel paragrafo precedente in questo caso risultano complicati da visualizzare, in quanto il numero di possibili coppie tra cui viene valutata l'indipendenza è pari a $\frac{148 \cdot 147}{2} = 10.878$. In Tabella 3.7 sono riportati alcuni dati sui risultati ottenuti dal test di indipendenza tra

	Numero di coppie
Test di casualità indipendenti	10.501
Test di casualità dipendenti	377

Tabella 3.7. Statistiche relative allo studio sull'indipendenza a due a due condotto tra i test effettuati con il Non Overlapping Template Matching Test implementato con parametro uguale a 9.

tutte le possibili coppie di test. Un risultato che si nota chiaramente è l'elevato numero di dipendenze trovate. Infatti, essendo il test di indipendenza chi quadro un test statistico, nel caso in cui l'ipotesi nulla sia vera (cioè la coppia sia composta da test indipendenti), dovremmo avere circa $10.878 \cdot \alpha_\chi$ falsi negativi, dove α_χ è il livello di significatività del test chi quadro. Negli esperimenti α_χ è scelto pari a 0,01 e quindi il numero di errori di tipo I dovrebbe essere circa 100, nel caso in cui l'ipotesi nulla sia corretta. Tuttavia, il numero di dipendenze trovate è molto maggiore. Questo risultato consente di concludere che le diverse versioni del Non Overlapping Template Matching test sono tra loro dipendenti. In Figura 3.4 è riportato per ciascuna versione del test il numero di test con i quali presenta una dipendenza. È possibile notare come ci siano due test che presentano un gran numero di dipendenze. I due test sono relativi ai template $'000001111'$ e $'001101101'$.

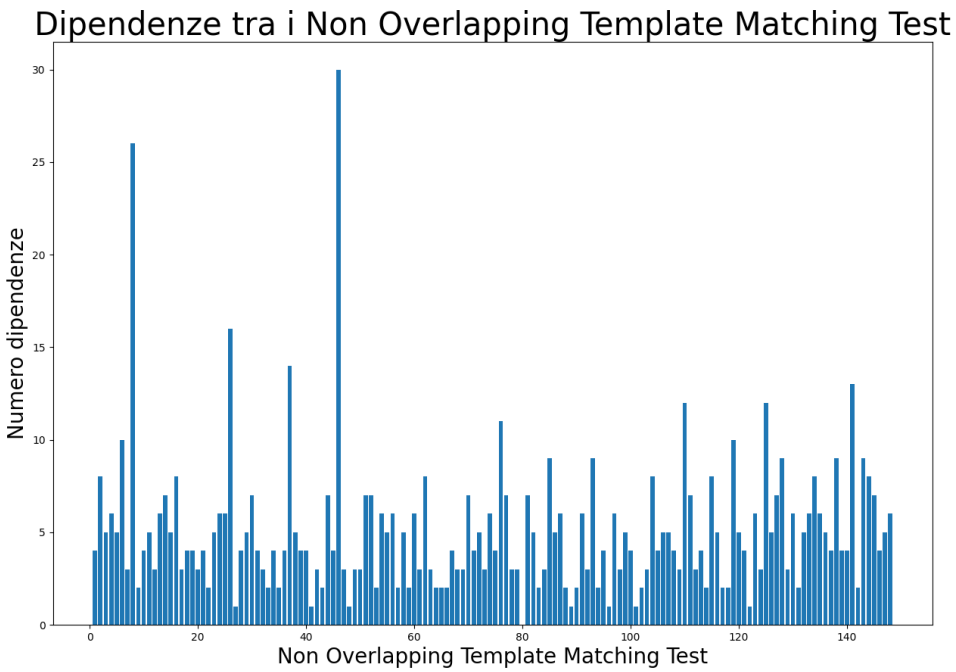

Figura 3.4. Risultati analisi di indipendenza a due a due tra i test relativi al Non Overlapping Template Matching Test condotto con parametro uguale a 9. Per ciascun test è riportato il numero di test con il quale esso risulta dipendente.

3.3.3 Random Excursions Test

Il test su cui è focalizzata l'attenzione in questo paragrafo è il Random Excursions test. Il test, per come è strutturato e implementato dal NIST, può essere visto come l'insieme di 8 test distinti. Il test si focalizza sul numero di cicli che hanno esattamente una certa quantità di visite nel cumulative sum random walk associato ad una sequenza. Il cumulative sum random walk di una sequenza è costruito a partire dalle somme parziali, dopo che la sequenza di $(0, 1)$ è stata traformata in una sequenza di $(-1, +1)$. In altre parole, il cumulative sum random walk è individuato dagli elementi della successione $\{S\}_k = X_1 + X_2 + \ldots, +X_k$, dove X_i è pari a 1 se l'i-esimo bit della sequenza è '1', mentre è pari a -1 nel caso in cui l'i-esimo bit sia '0'. Lo scopo di questo test è quello di verificare se il numero di visite ad uno stato, all'interno di un ciclo, sia statisticamente vicino al numero atteso di visite che ci si aspetterebbero in una sequenza casuale (un ciclo in una sequenza è individuato da un elemento nullo della successione $\{S\}_k$). Per riuscire nell'intento, viene fatto un test chi quadro in cui le frequenze osservate sono relative al numero di cicli in cui un determinato stato occorre esattamente k volte, dove $k = 0, 1, \ldots, 5$. Gli stati che vengono considerati in questo test sono 8: $-4, -3, -2, -1$ e $+1, +2, +3, +4$. In totale, quindi, si avranno 8 p-value e di conseguenza 8 test, ciascuno relativo ad uno

stato diverso. Vista la similitudine tra questi test è interessante chiedersi se il numero di visite ad uno stato, in qualche modo, sia connesso con il numero di visite ad un altro stato. In altre parole si tratta di studiare la dipendenza tra i diversi Random Excursions Test. A differenza dei casi precedenti, applichiamo gli 8 Random Excursions test, non all'insieme di tutte le sequenze prodotte, ma alle sole sequenze che possiedono un numero di cicli maggiore di 500, per i motivi esposti nel Paragrafo 3.3.1. Estraendo dall'insieme di 100.000 sequenze generate le sole sequenze che possiedono un numero di cicli maggiore di 500, queste diventano circa 60.000 (il numero esatto di sequenze con un numero di cicli maggiore di 500 nell'insieme delle sequenze generate è pari a 61.539). I risultati riportati in questo paragrafo e nel prossimo sono quindi basati su un insieme di sequenze ridotto rispetto all'insieme originale composto da 100.000 sequenze. In Figura 3.5 sono riportati i

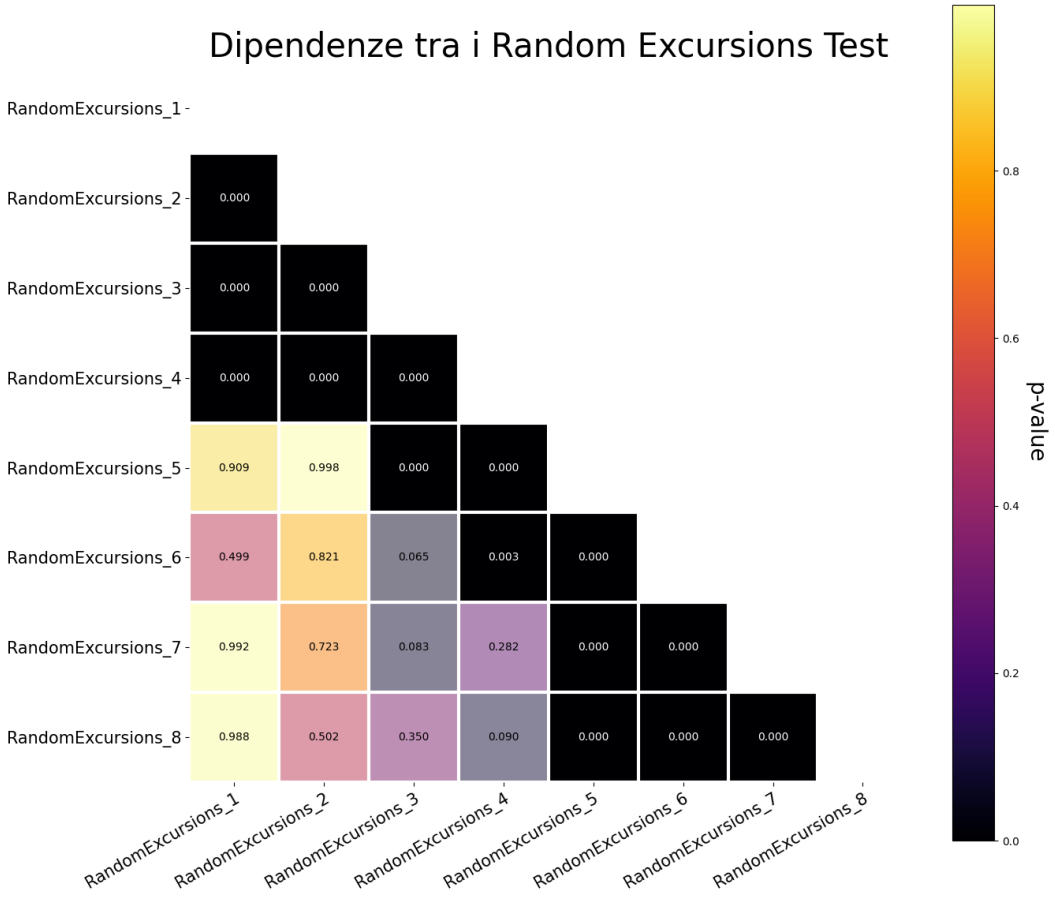

Figura 3.5. Risultati dello studio sull'indipendenza a due a due svolta tra gli 8 test relativi al Random Excursions Test. Per ciascuna coppia di test è riportato il p-value corrispondente al test di indipendenza chi quadrato. Valori del p-value inferiori a 0,01 evidenziano una dipendenza tra la coppia considerata.

risultati del test di indipendenza chi quadrato effettuato tra ciascuna coppia dei Random Excursions Test. Quello che si nota da questi risultati è la presenza di due gruppi composti da test fortemente correlati: i Random Excursions 1, 2, 3, 4 e i Random Excursions 5, 6, 7, 8. È bene precisare che i due insiemi individuati corrispondono rispettivamente ai test applicati considerando gli stati $-4, -3, -2, -1$ e ai test in cui si valutano le occorrenze sugli stati $+1, +2, +3, +4$. Si deduce, quindi, che vi è una dipendenza tra le visite agli stati 'negativi' e una dipendenza tra le visite agli stati 'positivi'. Oltre a forti dipendenze trovate all'interno di questi due insiemi di test, si riscontra una forte relazione tra il Random Excursions Test relativo allo stato -1 e quello relativo allo stato 1. Il motivo di questa dipendenza probabilmente è dovuto al fatto che gli stati -1 e 1, anche se uno negativo e l'altro positivo, sono tra loro vicini. Le cause delle dipendenze osservate in questo paragrafo sono meritevoli di ulteriori approfondimenti, che vengono delegati a studi successivi.

3.3.4 Random Excursions Variant Test

Il Random Excursions Variant Test è simile al Random Excursions Test, tuttavia, in questo caso, il numero di visite a ciascuno stato è calcolato considerando l'intero cumulative sum random walk, senza suddividere il conteggio all'interno di ogni ciclo. L'obiettivo è quello di individuare eventuali deviazioni dal numero atteso di visite a diversi stati nel caso di una sequenza casuale. A partire dalla distribuzione del numero di visite ad un particolare stato, all'interno di un ciclo in un cumulative sum random walk, ricavata in [3], è possibile calcolare il valore atteso delle occorrenze per i diversi stati e condurre un test statistico in cui si valuta quanto il valore ottenuto empiricamente dista dal valore atteso. Gli stati considerati in questo test sono 18: $-9, -8, \ldots, -1$ e $+1, +2, \ldots, +9$. Per cui, il numero di test di casualità appartenenti alla famiglia del Random Excursions Variant Test è pari a 18. Per le ragioni elencate nei paragrafi precedenti essi dovrebbero essere considerati come test di casualità distinti, ognuno ugualmente valido. Come nel caso dei Random Excursions Test, anche i Random Excursions Variant Test per poter essere applicati necessitano di sequenze che possiedono un numero di cicli maggiore di 500.

In Figura 3.6 sono riportati i risultati del test di indipendenza chi quadrato effettuato tra ciascuna coppia dei Random Excursions Variant Test. Il comportamento osservato per i Random Excursions Test si manifesta anche in questo caso. Dall'analisi delle dipendenze risulta che esistono due sottoinsiemi di test, tra tutti i Random Excursions Variant Test, composti da test dipendenti a due a due: l'insieme formato dai Random Excursions Variant $1, \ldots, 9$ e l'insieme formato dai Random Excursions Variant $10, \ldots, 18$. I due gruppi di test si riferiscono, rispettivamente, ai Random Excursions Variant Test focalizzati sul numero di occorrenze agli stati $-9, \ldots, -1$ e $+1, \ldots, +9$. Si riscontra, quindi, una dipendenza tra le visite agli stati 'negativi' e tra le visite agli stati 'positivi'. Inoltre, come nel caso precedente, sono presenti dipendenze anche tra i test che valutano il numero di occorrenze degli stati vicini allo stato 0 (Random Excursions Variant Test 8, 9, 10, 11).

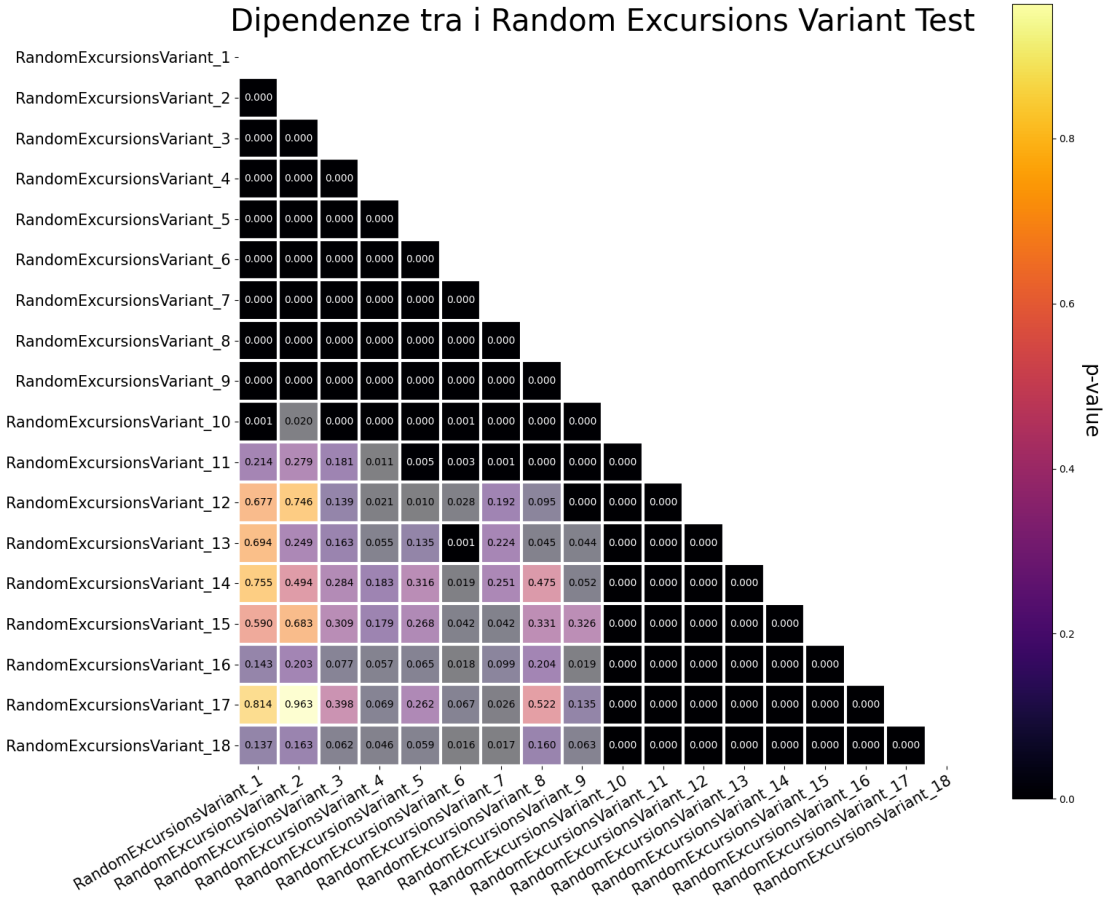

Figura 3.6. Risultati dello studio sull'indipendenza a due a due svolta tra i 18 test relativi al Random Excursions Variant Test. Per ciascuna coppia di test è riportato il p-value relativo al test di indipendenza chi quadrato. Valori del p-value inferiori a 0,01 evidenziano una dipendenza tra la coppia considerata.

3.3.5 Suite NIST

I questo paragrafo è stata condotta un'analisi completa sulla dipendenza tra i test presenti nella Suite del NIST. I test di casualità distinti, che costituiscono la Suite, sono pari a 188. Tra essi vi sono i test riportati in Tabella 3.4, i 148 test relativi al Non Overlapping Template Matching Test (effettuato con parametro 9), gli 8 test che fanno parte del Random Excursions Test e i 18 che appartengono alla famiglia dei Random Excursions Variant Test. Analizzando le dipendenze tra i test presenti in questo insieme ci si aspetta di trovare almeno le relazioni evidenziate durante lo studio individuale di ciascun sottoinsieme della Suite considerato. Oltre alle dipendenze già note, è utile capire se vi siano ulteriori relazioni, in particolare, tra test i appartenenti alle famiglie del Random Excursions,

Random Excursions Variant, Non Overlapping Template Matching e i restanti test di casualità della Suite. Dal momento che nell'insieme di test di casualità considerato in questo paragrafo sono presenti anche i Random Excursions e Random Excursions Variant Test, per poter valutare correttamente la dipendenza tra questi test e i restanti, è necessario che tutte le sequenze considerate abbiano un numero di cicli maggiore di 500. L'insieme delle sequenze testate, in questo caso, risulta composto da 61.539 sequenze. Come negli studi precedenti, prima di applicare il test di indipendenza chi quadro tra tutte le coppie di test, è bene verificare che ciascun test di casualità venga superato da una proporzione di sequenze pari a $1 - \alpha$. Durante la verifica di questa condizione è stato notato un comportamento anomalo riguardo il numero di successi di alcuni test di casualità. Nella Sezione 3.3.1, il numero di successi del Frequency Test e del Cumulative Sums Test (sia forward che backward) era in accordo con la proporzione attesa. Rimuovendo dall'insieme di tutte le sequenze quelle con un numero di cicli inferiore a 500 e applicando questi due test alle restanti sequenze, si osserva, invece, una percentuale di successi maggiore rispetto a quella attesa. Dal momento che l'unica differenza tra gli esperimenti condotti nella Sezione 3.3.1 e questa risiede nelle sequenze considerate, si è ipotizzata una possibile relazione tra il numero di cicli in una sequenza e gli esiti del Frequency e Cumulative Sums (forward e backward) Test. Nell'Appendice A, in cui viene affrontato con maggior riguardo questo problema, sono state trovate le cause di questo comportamento. Per poter includere nell'analisi sull'indipendenza dei test della Suite del NIST anche il Frequency e il Cumulative Sums Test, è necessario calcolare la percentuale di insucessi sperimentale e utilizzare il valore trovato per ricavare le frequenze teoriche attese del test chi quadro, quando è presente uno dei due test nella coppia considerata.

Dal momento che l'insieme considerato è composto da 188 test, vi sono 17.578 possibili coppie tra cui studiare la dipendenza (il numero di coppie distinte è ottenuto con $(188 \cdot 187)/2$). Nella Tabella 3.8 sono riportate il numero di coppie indipendenti e il numero di

	Numero di coppie
Test di casualità indipendenti	16.938
Test di casualità dipendenti	640

Tabella 3.8. Risultati dello studio sull'indipendenza a due a due condotto tra tutti i test appartenenti alla Suite del NIST.

coppie tra cui è stata osservata una dipendenza. Rispetto alle coppie dipendenti trovate in precedenza, risultano circa 150 dipendenze in più. Tra queste, sono rilevanti delle associazioni trovate tra: i Random Excursions $1, 2, 3, 4$ e i Random Excursions Variant $1, 2, \ldots, 9$; i Random Excursions $5, 6, 7, 8$ e i Random Excursions Variant $10, 11, \ldots, 18$. Queste dipendenze, intuitivamente, derivano dal fatto che i primi due gruppi di test si basano sul numero di visite agli stati negativi del cumulative sum random walk, mentre gli altri due gruppi dipendono dal numero di visite agli stati positivi.

3.4 Indipendenza reciproca tra i test della Suite del NIST

Nei paragrafi precedenti è stata studiata l'indipendenza a due a due tra i test di casualità presenti nella suite del NIST. Tuttavia, come riportato nel Capitolo 2, esiste una definizione di indipendenza più forte: la mutua (o reciproca) indipendenza. In questo paragrafo verrà affrontato, seppur in modo parziale, lo studio sull'indipendenza reciproca tra i test del NIST.

Supponendo di avere una suite composta da N test di casualità, per verificare che comprenda test reciprocamente indipendenti, dovremmo verificare le seguenti condizioni:

- verificare che qualunque coppia di test sia composta da test indipendenti; ovvero la suite sia composta da test indipendenti a due a due.

- per k che varia da 3 al numero di test presenti nella suite, verificare che preso un qualsiasi insieme composto da k test, la porzione di sequenze che non superano tutti e k i test sia mediamente uguale a α^k, dove α è il livello di significatività di ogni test di casualità.

Questa metodologia all'aumentare del numero di test diventa sempre più complessa, siccome è necessario considerare ogni sottoinsieme possibile di test di casualità. Seppure molto costosa, la procedura appena descritta è l'unico metodo che consente in modo diretto di verificare la mutua indipendenza di un insieme di test. Esiste, tuttavia, un metodo indiretto che, in alcuni casi, ci consente di concludere che sicuramente una suite non è composta da test reciprocamente indipendenti.

Nel Capitolo 2, in cui è riportata la definizione di test reciprocamente indipendenti, è stato introdotto anche il concetto di coverage. La coverage di una suite è definita come il rapporto tra il numero di tutte le sequenze che falliscono almeno un test presente nella suite sul totale delle sequenze considerate. La coverage è legata al concetto di mutua indipendenza tramite il Teorema 1. Grazie a questo teorema si può affermare che se un insieme di test ha coverage diversa da quella riportata in formula 2.5, allora sicuramente non è composto da test reciprocamente indipendenti. È possibile, quindi, sfruttare la coverage per dedurre informazioni circa la mutua indipendenza di una suite, utilizzando la seguente strategia. Supponiamo siano disponibili gli esiti di successo/insuccesso di un certo numero di sequenze per ogni test presente all'interno della suite. Calcoliamo la coverage empirica della suite contando il numero di sequenze che falliscono almeno un test e dividendolo per il numero di sequenze totali. Nel caso in cui la coverage calcolata sia statisticamente differente dalla coverage attesa, ricavata tramite la formula 2.5, è possibile concludere che l'insieme di test considerato non sia composto da test reciprocamente indipendenti. Tuttavia, notare che nel caso in cui la coverge empirica sia statisticamente uguale a quella attesa, l'insieme potrebbe comunque essere composto da test che non sono reciprocamente indipendenti, dal momento che l'implicazione inversa del Teorema 1, in generale non è valida. Questa strategia, seppur non permetta di arrivare sempre ad una conclusione, in alcuni casi può risultare molto utile, soprattutto vista la semplicità del calcolo della coverage a partire dai dati.

Nella procedura appena descritta non è stato specificato come stabilire se la coverage empirica sia statisticamente differente da quella attesa. Per poter formalizzare un test statistico che consenta di determinare l'equivalenza o meno della coverage di un insieme di test rispetto alla coverage che essi avrebbero nel caso in cui fossero reciprocamente indipendenti, si procede nel seguente modo:

- Si dividono le sequenze in k sottoinsiemi
- Si applicano i test di casualità alle sequenze presenti in ciascun sottoinsieme e si ricavano gli esiti di successo/insuccesso per ciascun test, in ogni sottoinsieme;
- Per ciascun sottoinsieme, si calcola la coverage della suite, in modo da formare un campione di k coverage della suite analizzata.
- Nel caso in cui k sia maggiore di 40, è possibile supporre che il campione di coverage osservato si distribuisca secondo una normale. In tal caso, si può applicare un test statistisco per valutare se la media calcolata a partire dal campione osservato sia statisticamente equivalente alla coverage attesa, ricavata utilizzando la formula 2.5.

Il test statistico a cui si fa riferimento nell'ultimo punto della strategia è un test sulla media. Supponiamo di avere a disposizione un campione di k coverage, $Cov_1, \ldots, Cov_k$. Quando $k > 40$, la variabile

$$Z = \frac{\overline{Cov} - \mu}{S/\sqrt{k}},$$

in cui $\overline{Cov}$ rappresenta la media campionaria e S rappresenta la deviazione standard campionaria, ha approssimativamente una distribuzione normale standard. È possibile condurre un test statistico in cui l'ipotesi nulla H_0 è: $\mu = ExpectedCoverage$, dove $ExpectedCoverage$ è la coverage attesa che si dovrebbe avere nel caso in cui l'insieme di test di casualità sia composto da test reciprocamente indipendenti.

Prima di applicare la strategia descritta è importante considerare due osservazioni. In primo luogo, è utilie notare che siccome la mutua indipendenza è un concetto che include l'indipendenza a due a due, allora se all'interno dell'insieme di test considerato vi sono coppie di test che non sono indipendenti a due a due, sicuramente l'insieme non sarà composto da test reciprocamente indipendenti. Inoltre, dal momento che la formula 2.5 è valida solamente nel caso in cui la probabilità dell'intersezione delle aree di rifiuto di qualsiasi sottoinsieme di k test, dipenda unicamente dalla sua numerosità, k, è necessario che ogni test di casualità sia condotto con lo stesso livello di significatività α. Questa seconda osservazione implica che ciascun test presente nell'insieme di cui si vuole valutare la mutua indipendenza debba superare il test sulla proporzione di successi; in altre parole, è necessario che l'ampiezza della regione di accettazione, ovvero il numero di tutte le sequenze che superano il test di casualità sia statisticamente uguale a $(1 - \alpha)n$, dove n è il numero di sequenze considerate.

Sulla base degli esperimenti condotti nelle sezioni precedenti, siamo in grado di identificare delle suite che soddisfano i requisiti sopracitati. In Tabella 3.9 sono riportati due esempi di insiemi composti da test indipendenti a due a due e tali che dividono lo spazio delle sequenze testate in accordo con il livello di significatività α.

Insieme 1	Insieme 2
Frequency	Frequency within a Block
Frequency within a Block	Runs
Runs	Binary Matrix Rank
Binary Matrix Rank	Linear Complexity
Linear Complexity	Serial 2
Serial 1	Approximate Entropy
	Cumulative Sums (Forward)

Tabella 3.9. Esempi di suite di test di casualità, composte da test indipendenti a due a due e tali che ciascuno divide lo spazio delle sequenze testate in accordo con il livello di significatività α.

Ci chiediamo se questi insiemi, oltre ad essere composti da test indipendenti a due a due, siano o meno composti da test di casualità che sono anche reciprocamente indipendenti. Dividiamo l'insieme delle 100.000 sequenze generate tramite AES-DRBG, in 100 insiemi costituiti da 1000 sequenze ciascuno. Successivamente, si procede come descritto dalla procedura precedente: una volta ottenuto il campione di 100 coverage, viene verificato se la media campionaria sia statisticamente uguale alla coverage attesa. Nella Tabella 3.10 sono riportati i risultati ottenuti, considerando entrambi gli insiemi di test di

	Media campionaria coverage	Coverage attesa	P-value test coverage
Insieme 1	0,05827	0,05852	0,79632
Insieme 2	0,06727	0,06793	0,48696

Tabella 3.10. Risultati del test utilizzato per verificare che la coverage stimata dai dati sia statisticamente equivalente alla coverage attesa da una suite composta da test reciprocamente dipendenti. Il test è applicato ai due insiemi riportati nella Tabella 3.9.

casualità riportati nella Tabella 3.9. Le statistiche riportate in Tabella 3.10 sono relative alla coverage media osservata e al valore del p-value del test utilizzato per verificare che la coverage stimata dai dati sia statisticamente equivalente alla coverage attesa da una suite composta da test reciprocamente dipendenti. I due insiemi scelti possiedono una coverage equivalente alla coverage attesa, in quanto il valore del p-value del test condotto è maggiore di 0,01; livello di confidenza scelto per il test statistico relativo alla coverage media. Per i due esperimenti non è quindi possibile dedurre se siano effettivamente composti da test reciprocamente indipendenti, in quanto l'implicazione opposta del Teorema 1 in generale non è valida.

3.5 Conclusioni

In questo capitolo sono state svolte diverse analisi sull'indipendenza tra i test di casualità all'interno della Suite del NIST. Tramite l'analisi dell'indipendenza a due a due, il dato che emerge chiaramente è che all'interno della Suite del NIST vi siano numerose coppie di test tra loro dipendenti. Tra tutte le dipendenze osservate, di particolare rilevanza, sono quelle trovate nella Sezione 3.3.1 e riportate in Tabella 3.6. Tra queste coppie di test sono presenti dipendenze già riscontrate in lavori precedenti ad eccetto della coppia formata dal Runs Test e il Mauer's Universal Statistical Test, di cui non è stata trovata evidenza in studi precedenti. Inoltre, avendo svolto un'analisi completa per quanto riguarda il Random Excursions e il Random Excursions Variant Test, è stato possibile evidenziare una dipendenza tra alcuni sottoinsiemi di queste famiglie di test. In particolare, è stata osservata una stretta connessione tra i Random Excursions Test e i Random Excursions Variant Test condotti considerando stati vicini nel cumulative sum random walk associato alla sequenza. Infine, dalle riflessioni svolte sul sottoinsieme riguardante il Non Overlapping Temaplate Matching Test sono emerse numerose coppie di template tra loro dipendenti, evidenziando la ridondanza nell'applicare numerose volte il suddetto test.

Appendice A

Impatto del numero di cicli in una sequenza sui test della Suite del NIST

Durante le analisi svolte circa lo studio della relazione presente tra i test della Suite del NIST, nel Paragrafo 3.3.5, è stata osservata un'interazione tra il numero di cicli presenti in una sequenza e l'esito di alcuni test di casualità presenti nella suite del NIST. In particolare, è stato notato che il Frequency test e il Cumulative Sums test, se applicati a sequenze con un numero di cicli maggiore di 500, determinano più successi di quanti dovrebbero essercene nel caso in cui le sequenze siano casuali. Dal momento che gli esiti dei due test sopracitati sono coerenti con i risultati attesi nel caso in cui essi siano applicati ad un insieme di sequenze generico (dove non viene fatta una discriminazione tra sequenze con un numero di cicli inferiore a 500) è possibile supporre che esista una relazione tra il numero di cicli presenti in una sequenza e il Frequency e il Cumulative Sums Test.

Per poter indagare in maniera approfondita questo problema, è necessario introdurre alcune nozioni riguardo il numero di cicli di una sequenza binaria.

A.1 Numero di cicli in una sequenza casuale

Il concetto di ciclo all'interno di una sequenza binaria è strettamente legato alla rappresentazione della sequenza attraverso il corrispondente *cumulative sum random walk*. Il random walk associato ad una stringa è costruito utilizzando la seguente procedura. La sequenza binaria $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)$, con $\varepsilon_i \in \{0, 1\}$ è trasformata nella sequenza $\mathbf{x} = (x_1, \ldots, x_n)$, dove $x_i = 2\varepsilon_i - 1$. Il cumulative sum random walk corrispondente alla sequenza è individuato dagli elementi della successione $\{S\}_k = x_1 + x_2 + \ldots + x_k$. Nel caso di sequenze di lunghezza ridotta, è possibile visualizzare graficamente il cumulative sum random walk, tracciando un cammino che parte dall'origine e ad ogni step si sviluppa nello spazio bidimensionale facendo un passo su di 1 ogni qualvolta $\varepsilon_i = 1$ e un passo in giù di 1 ogni volta che $\varepsilon_i = 0$. Ad esempio, il cumulative sum random walk associato alla

sequenza $'0110110101'$ corrisponde al cammino riportato in Figura A.1.[1] In altre parole,

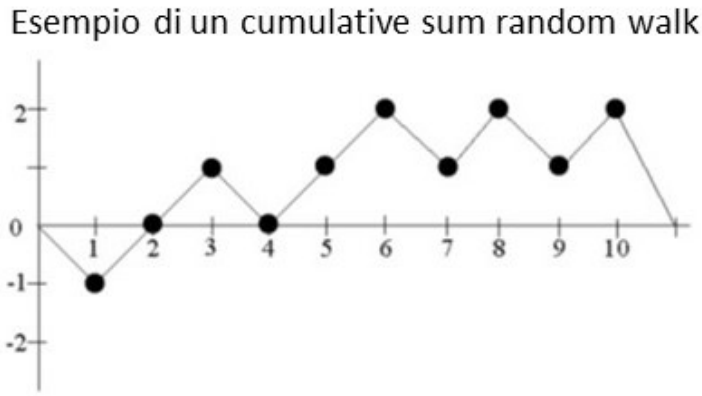

Figura A.1. Esempio di un cumulative sum random walk. Il random walk riportato in figura è ricavato a partire dalla sequenza $'0110110101'$.

un random walk consiste in una sequenza di passi di lunghezza unitaria che possono avvenire in due direzioni. La successione delle direzioni è determinata dagli elementi della stringa binaria considerata. Una volta che la sequenza è stata trasformata nel random walk equivalente, è possibile introdurre la definizione di *ciclo* o escursione in una sequenza binaria. Un ciclo in una sequenza binaria consiste in una sequenza di passi che iniziano e ritornano nell'origine, nel random walk ad essa associato. Considerando, ad esempio, la stringa precedente ($'110110101'$) si deduce dalla Figura A.1 che la sequenza possiede 3 cicli. Nel caso in cui non sia possibile visualizzare la sequenza, è possibile determinare il numero di cicli presenti in essa osservando la successione $\{S\}_k$. Dal momento che la successione $\{S\}_k$ individua gli stati del random walk, si avrà un ciclo ogni volta che un elemento della successione delle somme cumulative è nullo: $\{S\}_k = 0$.

Riassumendo, il numero di cicli in una sequenza è determinato dal numero di elementi nulli nella successione delle somme cumulative. Nel caso in cui sia possibile visualizzare il random walk associato, è possibile individuare il numero di cicli contando il numero di visite alla stato 0 del random walk, dove con stato di un random walk si intende l'altezza del cammino rispetto all'origine.

Apprese le definizioni introdotte, possiamo caratterizzare il numero di cicli presente in una sequenza binaria casuale. Il numero di cicli all'interno di una stinga casuale è un numero aleatorio. In letteratura esistono diversi studi circa la distribuzione del numero di cicli all'interno di una sequenza casuale [3]. Si dimostra che la distribuzione limite del numero di cicli, che indicheremo con J, per una sequenza casuale è pari a

$$\lim_{n\to\infty} P\left(\frac{J}{\sqrt{n}} < z\right) = \sqrt{\frac{2}{\pi}} \int_0^z e^{-u^2/2} du, \qquad z > 0. \tag{A.1}$$

[1] Immagine tratta da [1].

La distribuzione A.1 è valida nel caso in cui le sequenze casuali considerate abbiano lunghezza infinita. Tuttavia, dal punto di vista sperimentale è necessario utilizzare sequenze di lunghezza limitata. Ci si domanda come il numero di cicli all'interno di una sequenza si distribuisca al variare della lunghezza della stringa considerata. A tal proposito è stato condotto uno studio in cui è stata osservata la distribuzione del numero di cicli, al variare della lunghezza della sequenza. Sono stati considerati quattro insiemi di 1000 sequenze di lunghezza rispettivamente 1, 2, 5 e 10 milioni di bit. In Figura A.2 è riportata la distribuzione del numero di cicli al variare della lunghezza delle sequenze.

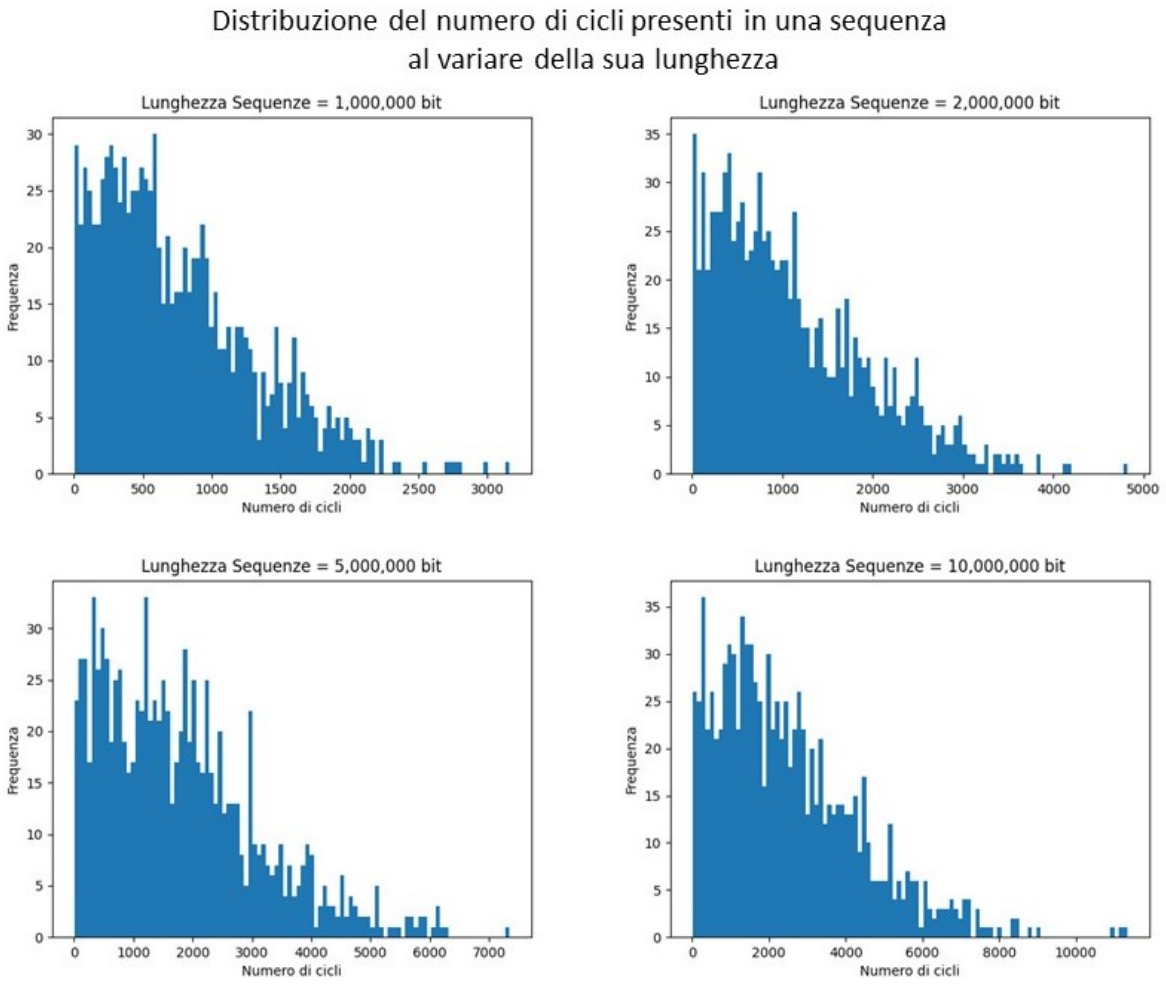

Figura A.2. Distribuzione del numero di cicli in una sequenza al variare della sua lunghezza. Le lunghezze considerate sono 1, 2, 5, 10 milioni di bit.

Si può notare come la distribuzione assomigli alla distribuzione di una variabile aleatoria chi quadrato e che essa sia spostata verso un numero di cicli maggiore all'aumentare della lunghezza delle sequenze.

Inoltre, dal momento che l'interazione tra i due test di casualità (Frequency Test e Cumulative Sums Test) e il numero di cicli è stata osservata in corrispondenza di sequenze con un numero di cicli maggiore di 500, è stato analizzato il numero di sequenze con un numero di cicli maggiore di 500 rispetto al variare della lunghezza delle sequenze. In particolare, sono stati considerati degli insiemi di 100 sequenze di diversa lunghezza e per ciascun insieme è stato osservato il numero di sequenze con un numero di cicli maggiore di 500. Complessivamente sono stati eseguiti 10 esperimenti, per cui, fissata una lunghezza delle sequenze vi sono 10 valori relativi al numero di sequenze che possiedono un numero di cicli maggiore di 500. Per ciascuna lunghezza considerata, è stato ricavato un valore medio di sequenze che presentano un numero di cicli maggiore di 500. Le lunghezze delle sequenze considerate sono pari a 500.000, 1.000.000, 2.000.000, 5.000.000, 10.000.000 bit. In Figura A.3 sono riportati i risultati ottenuti. In particolare, le linee tratteggiate si

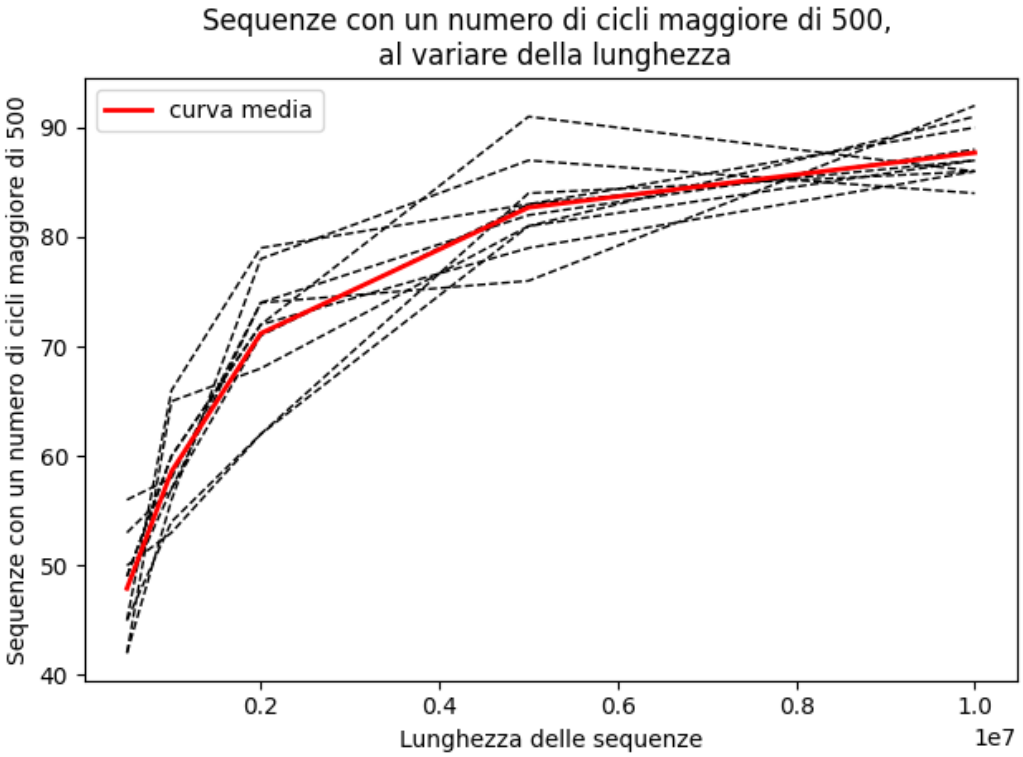

Figura A.3. Distribuzione del numero di sequenze con un numero di cicli maggiore di 500 in un insieme composto da 100 sequenze, al variare della lunghezza delle sequenze.

riferiscono ai 10 esperimenti svolti in maniera indipendente, da cui è stato ricavato il trend medio (curva rossa). Si noti che considerando un insieme di sequenze di lunghezza pari a un milione di bit, circa il 60% di queste avranno avranno un numero di cicli maggiore di 500. Questo dato è in accordo con quanto osservate nella Sezione 3.3.5; in quel caso, infatti, su un insieme di 100.000 sequenze, 61.539 avevano un numero di cicli maggiore di

500. Ritornando alla Figura A.3 è possibile notare, come ci si apetta, che all'aumentare della lunghezza delle sequenze la quantità di sequenze con un numero di cicli maggiore di 500 cresce. Nonostante ciò, è importante osservare che anche con sequenze lunghe 10 milioni di bit, in un campione di 100 sequenze, circa il 10% non raggiungono un numero di cicli maggiore di 500.

A.2 Interazione tra il numero di cicli e il Frequency Monobit Test

Le analisi svolte nella Sezione 3.3.5 hanno evidenziato una forte interazione tra il numero di cicli presenti in una sequenza e l'esito del Frequency Test. Applicando il test di casualità ad un insieme di sequenze con un numero di cicli maggiore di 500, il test sulla proporzione di successi non viene superato, in quanto la proporzione di sequenze che non passano il Frequency test è molto piccola rispetto a quella attesa. Questo comportamento, invece, non accade nel caso in cui non venga fatta nessuna nessuna discriminazione sull'insieme delle sequenze considerato. In particolare, dato un insieme generico di 100.000 sequenze, 99.056 superano il Frequency Test: si ha quindi una proporzione di successi pari a 0,9906; valore che supera il test sulla proporzione di successi. Quando da questo insieme vengono estratte le sequenze con un numero di cicli maggiore di 500, si ottiene un insieme composto da 61.539 sequenze. Il numero di sequenze che superano il Frequency test, in questo caso, è pari a 61.338, ovvero circa il 99,67%: valore che determina il rifiuto dell'ipotesi nulla del test sulla proporzione di successi. Il fatto che gli esiti dei test sulla proporzione di successi siano diversi nei due esperimenti suggerisce che vi sia una relazione tra il numero di cicli in una sequenza e il Frequency Test. In questo paragrafo è riportato uno studio volto ad indagare le possibili cause del comportamento osservato.

I motivi dell'interazione tra il numero di cicli e l'esito del Frequency Test vanno ricercati nelle statistiche calcolate durante il test di casualità. Il Frequency Monobit Test ha come obiettivo quello di valutare che il numero di 0 e 1 in una sequenza sia approssimativamente lo stesso che ci si aspetterebbe da una sequenza casuale. In una sequenza casuale il numero di 0 e 1 dovrebbe essere approssimativamente lo stesso; vi sarà, quindi, una frazione pari a 1/2 di 1 ed una frazione di 1/2 di 0. Il Frequency Test è condotto tramite la seguente procedura:

- La sequenza $\varepsilon \in \{0,1\}^n$, di lunghezza n, è convertita in una sequenza $\mathbf{x} \in \{-1,1\}^n$, dove $x_i = 2\varepsilon_i - 1$.
- Si calcola la somma cumulativa: $S_n = x_1 + x_2 + \ldots + x_n$.
- Si calcola la statistica del test: $s_{obs} = \frac{S_n}{\sqrt{n}}$.
- Si calcola il p-value della statistica del test utilizzando la distribuzione di riferimento sotto l'ipotesi nulla del test:

$$p - value = erfc\left(\frac{s_{obs}}{\sqrt{2}}\right),$$

dove $erfc$ è la funzione degli errori complementari ($erfc(z) = \frac{2}{\sqrt{n}} \int_z^\infty e^{-u^2} du$).

- Se il p-value ottenuto è minore di 0,01, si conclude che la sequenza non è casuale. Altrimenti la sequenza supera il test e viene considerata casuale.

È importante notare che valori piccoli del p-value, sono causati da valori di $|S_n|$ alti. Infatti, valori positivi grandi di S_n indicano che la sequenza possiede troppi 1, mentre valori negativi grandi di S_n indicano che la sequenza possiede troppi 0. Al fine di individuare una possibile causa dell'interazione tra il numero di cicli in una sequenza e l'esito del Frequency Test, potrebbe essere interessante osservare come cambia la variabile $|S_n|$ rispetto al numero di cicli presenti in una sequenza. Dato un campione di 5000 sequenze casuali di lunghezza pari a un milione di bit, è stato ricavato per ciascuna di esse il numero di cicli e il valore della variabile S_n. In Figura A.4 è riportata la distribuzione della somma

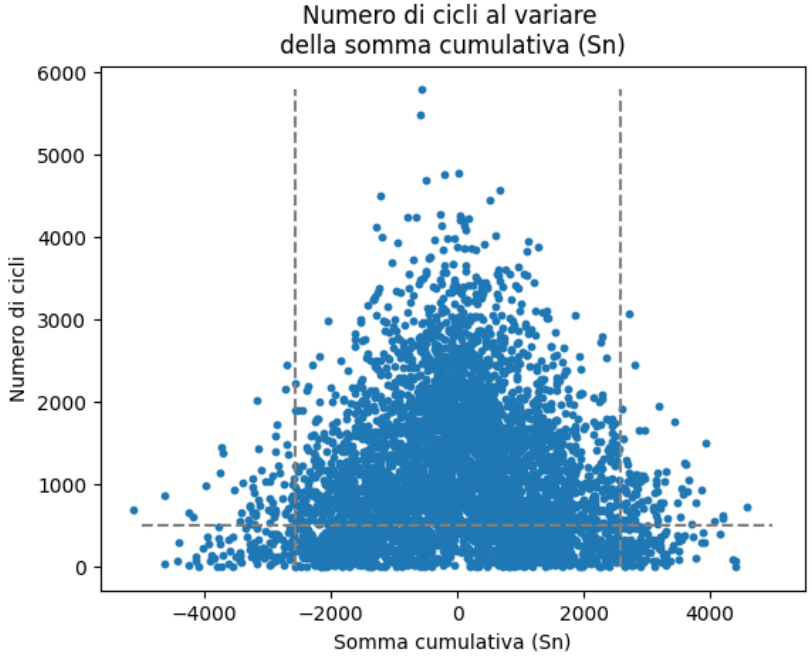

Figura A.4. Distribuzione della statistica S_n, calcolata all'interno del Frequency Test, rispetto al numero di cicli in una sequenza. La distribuzione è raffigurata utilizzando un campione di 5000 sequenze.

cumulativa S_n, rispetto al numero di cicli in una sequenza. La Figura A.4 è stata creata collocando ogni sequenza nello spazio individuato dal numero di cicli (asse y) e dalla somma cumulativa S_n (asse x). Si osserva che quando una sequenza possiede un numero di cicli molto alto, la variabile S_n tende ad essere più vicina al valore 0. Valori estremi di $|S_n|$ si ottengono prevalentemente quando il numero di cicli nella sequenza è basso. Nella Figura A.4 le due linee verticali, situate in corrispondenza di $S_n = -2576$ e $S_n = 2576$, individuano la regione dei valori di S_n che consentono di ottenere un p-value maggiore di 0,01, determinando quindi il successo del Frequency Test. Sequenze che invece possiedono una somma $S_n < -2576$ o $S_n > 2576$ corrispondono a sequenze che non supereranno il

Frequency Test. Come si nota dalla Figura A.4, le sequenze che rientrano nella regione di rifiuto del Frequency Test sono prevalentemente caratterizzate da un numero di cicli inferiore a 500. Si deduce, quindi, che il fatto di escludere dall'analisi sequenze con un numero di cicli inferiore a 500, influisce sull'esito del Frequency Test, in quanto sequenze con un numero di cicli elevato raggiungono difficilmente valori estremi di $|S_n|$; di conseguenza risulta maggiormente difficile rifiutare l'ipotesi nulla del test. Questa deduzione spiega perché si nota un numero di successi del Frequency Test maggiore rispetto a quanto atteso, nel caso in cui le sequenze considerate abbiano un numero di cicli maggiore di 500.

Appurato che l'influenza del numero di cicli sull'esito del Frequency Test sia dovuta alla variabile S_n, si può provare ad individuare, almeno intuitivamente, il motivo della connessione tra questa variabile e il numero di cicli. La relazione tra la variabile S_n e il numero di cicli non appare immediata. È importante osservare, infatti, che all'interno di un ciclo il numero di 0 e 1 sarà sempre lo stesso, per cui la variabile S, calcolata alla fine di un ciclo, assumerà valore nullo. Il valore di S_n è quindi dato solamente dallo squilibrio di 0 e 1 nella porzione terminale della sequenza che, in generale, sarà costituita da un ciclo chiuso artificialmente. Infatti, quando viene tracciato il cumulative sum random walk associato ad una sequenza, viene costruita la successione delle somme cumulative $\{S\}_k$ che determina la sequenza degli stati assunti dal random walk. Per poter contare il numero di cicli, viene posto al termine della sequenza $\{S\}_k$ un ultimo elemento pari a 0. In questo modo viene chiuso artificialmente l'ultimo tratto della sequenza e formato l'ultimo ciclo. Tuttavia questo ciclo è l'unico all'interno di tutta la sequenza che potrebbe avere un numero di 0 e 1 differente. Per comprendere il motivo della connessione tra il numero di cicli e la variabile S_n, bisognerebbe trovare la relazione tra lo squilibrio di 0 e 1 nell'ultimo tratto della sequenza e il numero di cicli che essa possiede. Tutte queste riflessioni, pongono le basi per possibili lavori futuri, volti a dimostrare analiticamente la relazione tra la variabile S_n e il numero di cicli in una sequenza.

A.3 Interazione tra il numero di cicli e il Cumulative Sums Test

Oltre all'interazione osservata tra il numero di cicli in una sequenza e l'esito del Frequency Test, è stato notato un comportamento simile per quanto riguarda l'esito del Cumulative Sums Test condotto in entrambe le modalità (forward e backward). A seconda che il test di casualità sia effettuato su un insieme generico di sequenze casuali oppure su un insieme composto solamente da sequenze casuali con un numero di cicli maggiore di 500, la proporzione attesa di successi viene in un caso rispettata mentre nell'altro no. Considerando solamente sequenze che possiedono un numero di cicli maggiore di 500 si riscontra un'irregolarità nella proporzione di successi attesa dal Cumulative Sums Test. In particolare, si osserva un numero di successi maggiore rispetto a quanto atteso. Seguendo gli stessi passi del caso precedente è stata indagata la connessione tra il numero di cicli e la test statistic utilizzata nel Cumulative Sums Test.

Il Cumulative Sums Test ha come obiettivo quello di determinare la massima escursione (da zero) nel cumulative sum random walk associato alla sequenza e valutare che essa sia in accordo con quanto ci si aspetta da una sequenza casuale. Con massima escursione, si

intende il valore più grande assunto dallo stato nel cumulative sum random walk associato alla sequenza. Per poter condurre il test di casualità, la sequenza composta da 0 e 1, $\varepsilon \in \{0,1\}^n$, viene mappata nella sequenza $\mathbf{x} \in \{-1,1\}^n$, dove $x_i = 2\varepsilon_i - 1$. Successivamente viene calcolata la somma parziale S_i, partendo dal termine x_1 e aggiungendo via via gli elementi successivi della sequenza casuale, nel caso in cui la modalità del test sia forward. Nel caso in cui la modalità sia backward si parte da x_n e si aggiungono via via gli elementi precedenti della sequenza casuale. La sequenza delle somme parziali $\{S\}_n$ è costituita dagli elementi S_k, ottenuti utilizzando la relazione ricorsiva $S_k = S_{k-1} + x_k$ se la modalità e forward oppure la relazione $S_k = S_{k-1} + x_{n-k+1}$ se la modalità è backward. Una volta costruita questa successione viene calcolata la statistica del test, z, attraverso:

$$z = \max_{1 \leq k \leq n} |S_k|$$

Utilizzando la distribuzione asintotica dalla variabile z per una sequenza casuale, viene estratto il p-value relativo alla statistica calcolata dai dati. Se il p-value è minore del livello di significatività, $\alpha = 0{,}01$, con cui è condotto il test di casualità, allora la sequenza non supera il test. In caso contrario la sequenza supera il test di casualità. In questo test, valori alti della statistica z indicano una deviazione dal comportamento di una sequenza casuale e determinano un p-value molto piccolo, comportando il rifiuto dell'ipotesi nulla.

Come nel caso precedente è utile osservare se, ed in caso affermativo in che modo, il numero di cicli influenzi la statistica del test z. In Figura A.5 è riportata la distribuzione del numero di cicli rispetto alla statistica z, utilizzando un campione di 5000 sequenze. È possibile notare come valori grandi della statistica z vengano assunti prevalentemente da sequenze con un numero di cicli basso. Se invece consideriamo sequenze con un numero elevato di cicli, la variabile z sarà relativamente piccola. Questo è in accordo con il comportamento osservato nella Sezione 3.3.5: rimuovendo le sequenze con un numero di cicli inferiore a 500 si ha un numero di successi minore rispetto a quanto atteso, perché la statistica z tende ad essere alta prevalentemente quando nella sequenza vi sono pochi cicli. Anche in questo caso, sarebbe un risultato degno di nota riuscire a dimostrare matematicamente le cause della relazione esistente tra la variabile z e il numero di cicli in una sequenza.

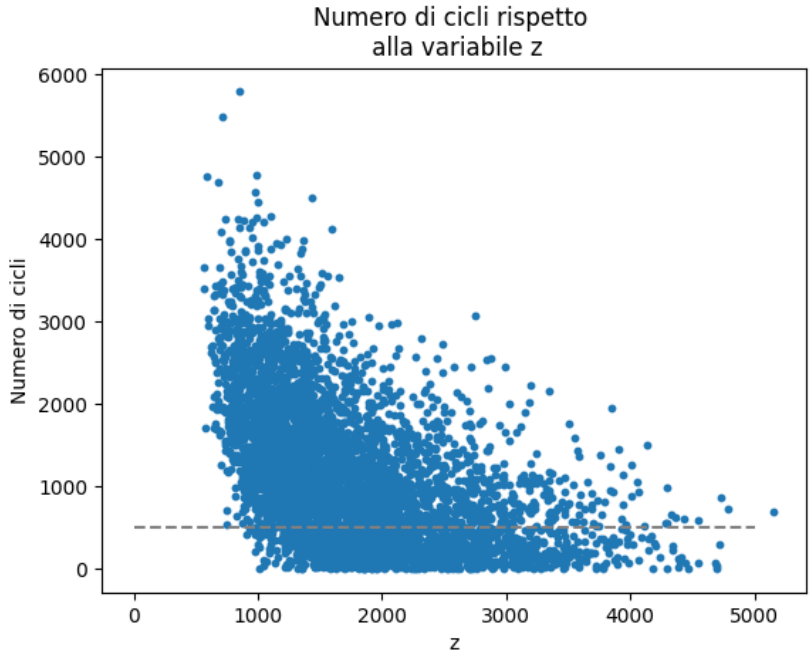

Figura A.5. Distribuzione della statistica z, calcolata nel Cumulative Sums Test (modalità forward), rispetto al numero di cicli in una sequenza. La distribuzione rappresentata in figura è ottenuta utilizzando 5000 sequenze.

Appendice B

Distribuzione p-value

Nella Sezione 1.4.2 sono riportate le due verifiche che il NIST suggerisce di effettuare per determinare se un generatore superi o meno un test di casualità. Tra di esse, vi è la richiesta che la distribuzione dei p-value, relativi all'insieme di sequenze testate, per ogni test, sia uniforme. Questa verifica, tuttavia, non è sempre giustificata. Infatti, la distribuzione dei p-value deve essere uniforme, nel caso in cui l'ipotesi nulla è corretta, soltanto se la distribuzione della statistica del test è invertibile. Nel caso in cui la statistica del test non abbia distribuzione invertibile, non vi è nessun risultato teorico che afferma che se l'ipotesi nulla è corretta allora la distribuzione dei p-value sarà uniforme. Il fatto che la statistica di un test di casualità segua una distribuzione invertibile non è in generale vero. Per diversi test di casualità, è stato osservato che i valori possibili della statistica del test e di conseguenza dei p-value, appartengono ad un insieme finito. Questi test di casualità, in generale non possiedono una distribuzione invertibile, per cui la richiesta che i p-value ottenuti siano distribuiti uniformemente tra 0 e 1 per poter affermare che l'ipotesi nulla sia corretta, non è del tutto giustificata. Per questo motivo, per valutare il superamento o meno di un test di casualità da parte di un generatore, è stato utilizzato soltanto il criterio sulla proporzione di successi.

Il fatto che un test di casualità abbia un numero finito di possibili valori del p-value calcolato, potrebbe influire anche sulla correttezza della proporzione di successi. Per comprendere meglio questa affermazione consideriamo un esempio ricavato da un fenomeno osservato durante l'applicazione dei test di casualità della suite del NIST.

B.1 Correttezza del test sulla proporzione di successi

Applicando un test di casualità ad un certo numero di sequenze, nel caso in cui esse siano casuali, ci si aspetta di trovare una proporzione di circa α sequenze che non superano il test, dove α è il livello di significatività con cui è condotto il test. Tuttavia, è corretto aspettarsi questo comportamento soltanto nel caso in cui i p-value abbiano una distribuzione continua e di conseguenza possano assumere qualsiasi valore nell'intervallo $[0, 1]$. Nel caso in cui i valori che i p-value possono assumere, appartengano ad un insieme finito, la proporzione di successi risulterà corretta sotto l'ipotesi nulla soltanto nel caso in cui l'α utilizzato nel

test di casualità sia pari ad uno dei valori che i p-value possono assumere. Nel caso in cui il valore α non sia uno tra i valori ammissibili del p-value, è possibile che al variare del livello di significatività considerato a volte il test sulla proporzione di successi venga superato e altre volte no. Dal momento che su alcuni test di casualità della suite del NIST si evidenzia una distribuzione dei p-value discreta, è stata condotta un'analisi volta ad indagare il comportamento del numero di successi rispetto al livello di significatività scelto. Tra i diversi test di casualità della Suite del NIST, un test che presenta un numero ridotto di valori distinti che possono essere assunti dalla statistica del test (e di conseguenza dal p-value) è il Discrete Fourier Transform Test. Infatti, nonostante siano state considerate 100.000 sequenze diverse, i p-value distinti ottenuti a seguito dell'applicazione del Discrete Fourier Transform Test sono solamente 419. Per questo motivo l'analisi è stata condotta considerando questo test di casualità. In Figura B.1 è riportata la distribuzione dei 100.000

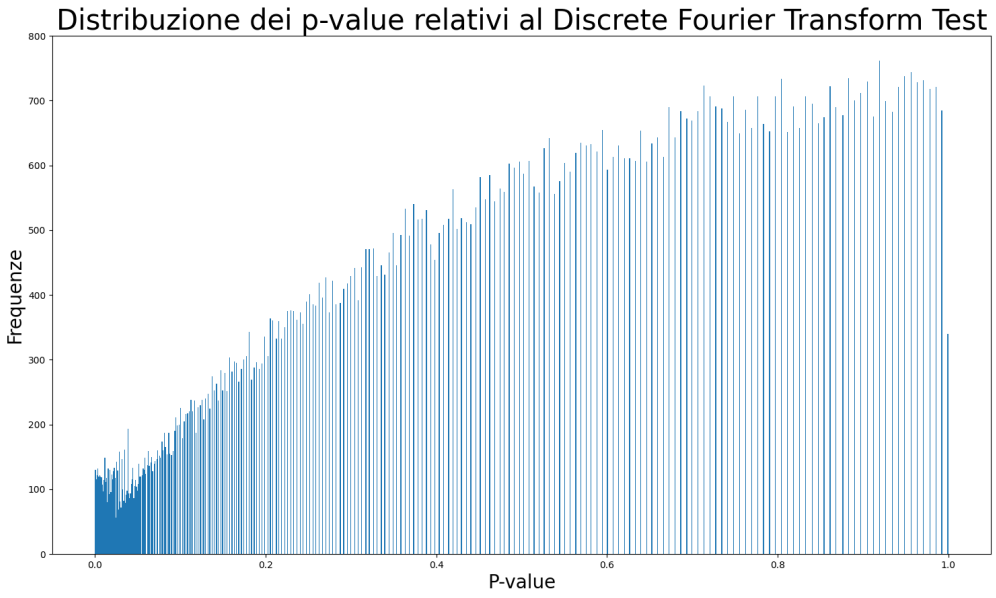

Figura B.1. Distribuzione dei p-value ottenuti applicando il Discrete Fourier Transform Test a 100.000 sequenze casuali.

p-value ottenuti applicando il Discrete Fourier Transform Test alle sequenze generate. Osservando la Figura B.1, si nota come la distribuzione dei p-value non sia uniforme: vi è un numero maggiore di p-value distinti per valori bassi, mentre per valori alti di p-value i valori che possono assumere sono più radi. Tuttavia, valori di p-value alti sono più frequenti rispetto a valori bassi.

Per evidenziare maggiormente il problema esposto in precedenza, consideriamo valori di α nell'intervallo $[0{,}698; 0{,}716]$. All'interno di questo range, sono presenti tre valori ammissibili per il p-value dell'FFT Test: $0{,}699927; 0{,}706737; 0{,}713570$. Per ciascun valore di α all'interno dell'intervallo $[0{,}698; 0{,}716]$ (utilizzando uno step di 0,001), è stato calcolato

il numero di sequenze che superano il Discrete Fourier Transform Test ed è stato condotto il test sulla proporzione di successi. Oltre ai valori sopracitati, sono stati anche inclusi i tre valori di p-value ammissibili.

In risultati ottenuti sono riassunti nella Tabella B.1. Notare che quando il test viene

Alpha	P-value
0,698	0,000140
0,699	0,001790
0,699927	0,012914
0,700	0,000000
0,701	0,000000
0,702	0,000000
0,703	0,000001
0,704	0,000016
0,705	0,000295
0,706	0,003399
0,706737	0,015549
0,707	0,000000
0,708	0,000000
0,709	0,000000
0,710	0,000001
0,711	0,000024
0,712	0,000410
0,713	0,004537
0,713570	0,014640
0,714	0,000000
0,715	0,000000

Tabella B.1. Risultati del test sulla proporzione di successi applicato agli esiti del Discrete Fourier Transform Test, utilizzando diversi livelli di significatività.

effettuato utilizzando un valore di α tra quelli ammissibili, il test sulla proporzione di successi viene rispettato. Questi risultati confermano le supposizioni espresse in precedenza, sul motivo per cui alcuni test di casualità non superano la proporzione di successi. Il problema non è dovuto alle sequenze utilizzate ma ad una formulazione scorretta del test di casualità. Infine, dai risultati riportati in Tabella B.1, è possibile notare che quanto più il livello di confidenza si avvicina da destra ad un valore ammissibile per il p-value del test di casualità, il p-value relativo al test sulla proporzione di successi cresce.

Bibliografia

[1] Juan Soto (NIST). Andrew Rukhin (NIST). Sp 800-22 rev. 1a: A statistical test suite for random and pseudorandom number generators for cryptographic applications, 2010. URL https://csrc.nist.gov/Projects/random-bit-generation/Documentation-and-Software.

[2] Jean-Philippe Aumasson. *Serious Cryptography, A Practical Introduction to Modern Encryption*. 2017.

[3] Michael Baron and Andrew L. Rukhin. Distribution of the number of visits of a random walk. *Communications in Statistics. Stochastic Models*, 15(3):593–597, 1999. doi: 10.1080/15326349908807552. URL https://doi.org/10.1080/15326349908807552.

[4] Dan Biebighauser. Testing random number generators, 2000.

[5] Paul Burciu and Emil Simion. A systematic approach of nist statistical tests dependencies. 2019.

[6] Samuele Cornell. Cryptographically secure aes drbg nist sp 800-90a, rev 1 pseudorandom number generator (prng) in pure python. URL https://github.com/popcornell/pyAES_DRBG.

[7] Haydar Demirhan and Nihan Bitirim. Statistical testing of cryptographic randomness. *İstatistikçiler Dergisi:İstatistik ve Aktüerya*, 9(1):1 – 11, 2016. ISSN 1308-0539.

[8] Jay L. Devore. *Probability & Statistics for Engineering and the Sciences*. 2012.

[9] Ali Doğanaksoy, Fatih SULAK, Muhiddin UĞUZ, Okan ŞEKER, and Ziya Akcengiz. Mutual correlation of nist statistical randomness tests and comparison of their sensitivities on transformed sequences. *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, 25:655–665, 01 2017. doi: 10.3906/elk-1503-214.

[10] John Kelsey (NIST) Elaine Barker (NIST). Sp 800-90a rev. 1: Recommendation for random number generation using deterministic random bit generators, 2015. URL https://csrc.nist.gov/publications/detail/sp/800-90a/rev-1/final.

[11] Jorge Karell-Albo, C.M. Legón, Evaristo Capo, Omar Rojas, and Guillermo Sosa Gómez. Measuring independence between statistical randomness tests by mutual information. *Entropy*, 22:741, 07 2020. doi: 10.3390/e22070741.

[12] Charmaine Kenny. Random number generators: An evaluation and comparison of random.org and some commonly used generators, 2005. URL https://www.random.org/analysis/.

[13] Donald E. Knuth. *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Addison-Wesley Longman Publishing Co., Inc., USA, 1997. ISBN 0201896842.

[14] David Lazar. Cryptographically secure prng for python. URL https://github.com/davidlazar/python-drbg.

[15] Pierre L'Ecuyer and Richard Simard. Testu01: A c library for empirical testing of random number generators. *ACM Trans. Math. Softw.*, 33(4), aug 2007. ISSN 0098-3500. doi: 10.1145/1268776.1268777. URL https://doi.org/10.1145/1268776.1268777.

[16] George Marsaglia. The marsaglia random number cdrom including the diehard battery of tests of randomness, 1995. URL https://web.archive.org/web/20160125103112/http://stat.fsu.edu/pub/diehard/.

[17] EMS Press. Inclusion-and-exclusion principle. URL http://encyclopediaofmath.org/index.php?title=Inclusion-and-exclusion_principle&oldid=47325.

[18] Python.org. os — miscellaneous operating system interfaces. URL https://docs.python.org/3/library/os.html.

[19] Andrea Röck. *Pseudorandom number generators for cryptographic applications*. na, 2005.

[20] O. B. Sheynin. The notion of randomness from Aristotle to Poincaré. *Mathématiques et Sciences humaines*, 114:41–55, 1991. URL http://www.numdam.org/item/MSH_1991__114__41_0/.

[21] Emil Simion and Paul Burciu. A note on the correlations between nist cryptographic statistical tests suite. *UPB Scientific Bulletin, Series A: Applied Mathematics and Physics*, 81:209–218, 01 2019.

[22] Fatih SULAK, Muhiddin UĞUZ, Onur Koçak, and Ali Doğanaksoy. On the independence of statistical randomness tests included in the nist test suite. *TURKISH JOURNAL OF ELECTRICAL ENGINEERING & COMPUTER SCIENCES*, 25: 3673–3683, 01 2017. doi: 10.3906/elk-1605-212.

[23] Meltem Turan, Ali Doğanaksoy, and Serdar Boztas. *On Independence and Sensitivity of Statistical Randomness Tests*, 09 2008.