

Realizzazione di un ambiente di fault injection per applicazione ridondata [Manuale Utente]

Carlo Migliaccio¹, Federico Pretini¹, Alessandro Scavone¹, and Mattia Viglino¹

¹Laurea Magistrale in Ingegneria Informatica, Politecnico di Torino

Gennaio 2025

Contents

1	Requisiti	2
2	Come Aprire ed Eseguire il Programma	2
3	Guida al Menù	2
3.1	Come navigare nel menù	2
3.2	Scelte del menù	2
3.3	Diagramma della struttura del menù	5
3.4	Esempio di utilizzo	6

Introduzione

Questo manuale fornisce istruzioni rapide per utilizzare il programma scritto in Rust.

Un menù interattivo consente di personalizzare l'esecuzione della pipeline di fault injection, scegliendo i dati di input, l'algoritmo di esecuzione e il tipo di report generato.

1 Requisiti

- **Sistema operativo:** macOS, Linux o Windows.
- **Compilatore Rust:** rustc installato. Puoi installarlo da <https://rustup.rs>.

2 Come Aprire ed Eseguire il Programma

1. Accedi alla stessa directory che contiene il file `Cargo.toml` del progetto.
2. Esegui il programma con il comando `cargo run`.

3 Guida al Menù

3.1 Come navigare nel menù

Dopo l'avvio, il programma presenterà un menù interattivo.

La **scelta corrente** è evidenziata dall'indicatore visivo `>`.

Per navigare tra le opzioni del menù, utilizza i tasti freccia **Su** e **Giù**.

Premere il tasto **Invio** per confermare la selezione.

Una selezione predefinita è racchiusa tra parentesi quadre [**default option**], per cofermarla premere **Invio**.

3.2 Scelte del menù

Il menù del programma ti permetterà di eseguire la pipeline di fault injection in maniera personalizzata.

Di seguito vengono descritti gli step passo passo.

Passo 1: Inserisci il nome del file per il report.

All'avvio, il programma richiede di specificare il nome del file per il report, il documento pdf generato al termine dell'analisi con i risultati più importanti. Il nome del file può essere digitato da tastiera, non deve contenere l'estensione e consente solamente numeri, lettere, - e `_`. L'opzione `[report]` rappresenta il nome di default.

Esempio passo 1:

```
Realizzazione di un ambiente di Fault Injection per applicazione ridondata
```

```
Inserisci il nome del file per il report SENZA ESTENSIONE [report]:
```

Passo 2: Scegli la sorgente dei dati.

La sorgente dei dati permette di specificare il vettore sui cui verranno applicati gli algoritmi di ordinamento e le matrici che verranno moltiplicate tra loro.

- **Data file**

Il data file è un file di input personalizzabile con precaricati un vettore randomico, la matrice di Wilson e la sua inversa. Il file è disponibile al percorso `src/data/input.txt`.

- **Dataset**

Il dataset è una cartella sorgente composta da due file.

Il primo file contiene vettori casuali a dimensioni variabili.

Il secondo file contiene 64 matrici di rotazione 3x3.

Se viene eseguita un'analisi con algoritmo matrix multiplication (vedi il prossimo passo), una di queste matrici verrà estratta randomicamente e scalata con una matrice di scalamneto uniforme anch'essa randomica.

Esempio passo 2:

```
Seleziona la sorgente dei dati:  
> Data file  
Dataset
```

Passo 3: Seleziona il tipo di analisi.

In entrambi i casi (*Data file* o *Dataset*), è possibile scegliere tra:

- **Singolo algoritmo:** Esegue la pipeline di fault injection su un singolo algoritmo.
- **Tutti gli algoritmi:** Esegue la pipeline di fault injection sequenzialmente per tutti gli algoritmi disponibili.

Esempio passo 3:

```
Seleziona il tipo di analisi:  
> Esegui un singolo algoritmo  
Esegui un'analisi comparativa tra tutti gli algoritmi
```

Passo 4: Configura l'algoritmo (SOLO SE HAI SCELTO *Singolo algoritmo*).

Se è stata selezionata l'opzione *Singolo algoritmo*, sarà necessario scegliere un algoritmo tra:

- **Selection Sort:** algoritmo di ordinamento per un vettore.
- **Bubble Sort:** algoritmo di ordinamento per un vettore.
- **Matrix multiplication:** moltiplicazione tra matrici quadrate.

Esempio passo 4:

```
Scegli un algoritmo da utilizzare:  
> Selection Sort  
Bubble Sort  
Matrix Multiplication
```

Passo 5: Configura la modalità

A seconda della scelta effettuata precedentemente, ci sono due modalità di configurazione disponibili:

1) Modalità "Singolo algoritmo"

Dopo aver selezionato un algoritmo specifico, è possibile configurare la modalità scegliendo tra le seguenti opzioni:

- **Cardinalità a piacere della fault entry:**

L'analisi viene eseguita una sola volta con le entry della fault list selezionate manualmente. L'opzione predefinita è [2000].

- **Cardinalità 1000, 2000, 3000:**

L'analisi viene eseguita tre volte per le cardinalità della fault list pari a **1000**, **2000**, e **3000**.

Esempio passo 5.1:

Scegli una modalità di single analysis:

> Digita una cardinalità a piacere per la fault list entry

Tre esecuzioni con cardinalità della fault list entry che varia [1000, 2000, 3000]

2) Modalità "Tutti gli algoritmi"

Se è stata selezionata l'opzione **Tutti gli algoritmi**, la modalità di analisi è predefinita. Tuttavia, è possibile modificare la cardinalità della fault list:

- **Cardinalità a piacere della fault list entry:**

L'analisi viene eseguita sequenzialmente su tutti gli algoritmi, con le entry della fault list selezionate manualmente. L'opzione predefinita è [2000].

Esempio passo 5.2:

Inserisci il numero di fault entries desiderate [2000]:

Passo 6: Avvio dell'analisi.

Confermando l'ultima scelta la pipeline di fault injection inizierà la sua esecuzione con i parametri selezionati.

Una volta completata l'analisi, verrà mostrato il messaggio *"operazione completata"* e verrà salvato il report nella cartella *results*.

In base alla configurazione scelta, il report sarà di diverso tipo:

- Se è stata selezionata l'opzione *Singolo algoritmo* e *Cardinalità a piacere della fault entry*, verrà generato un report con nome [nomefile].pdf.
- Se è stata selezionata l'opzione *Singolo algoritmo* e *Cardinalità 1000, 2000, 3000*, verrà generato un report di tipo comparativo con nome [nomefile]_diffcard.pdf.
- Se è stata selezionata l'opzione *Tutti gli algoritmi*, verrà generato un report di tipo comparativo con nome [nomefile]_all.pdf.

3.3 Diagramma della struttura del menù

Di seguito viene riportato un diagramma esplicativo della struttura del menù.



3.4 Esempio di utilizzo

Di seguito viene riportato un esempio completo di esecuzione con sorgente da data file per tutti gli algoritmi con 2000 fault entries.

Scelte del menù:

Realizzazione di un ambiente di Fault Injection per applicazione ridondata

Inserisci il nome del file per il report SENZA ESTENSIONE: report
Seleziona la sorgente dei dati: Data file
Seleziona il tipo di analisi: Esegui un'analisi comparativa tra tutti gli algoritmi
Inserisci il numero di fault entries desiderate: 2000
Esecuzione Selection Sort
Esecuzione Bubble Sort
Esecuzione Matrix Multiplication
Operazione completata. Report salvato in: results/report_all.pdf

Report PDF di output:

Il report generato conterrà i risultati dell'analisi comparativa tra tutti gli algoritmi.

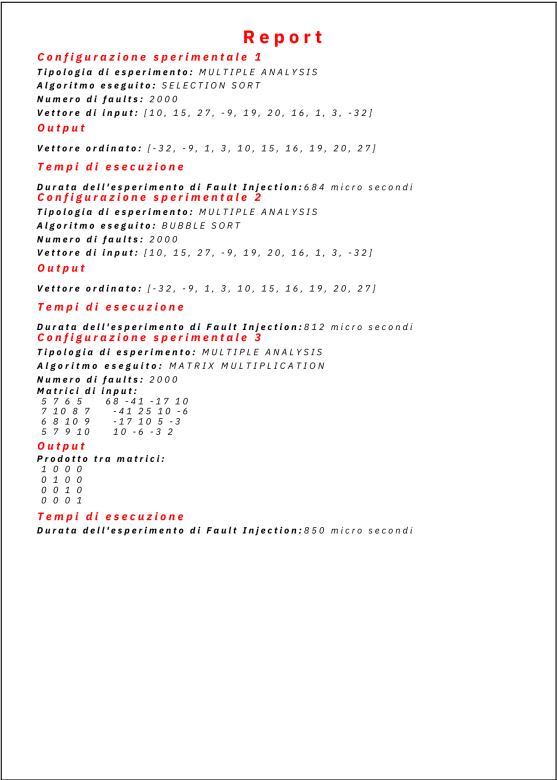


Figure 3.1: Pagina 1

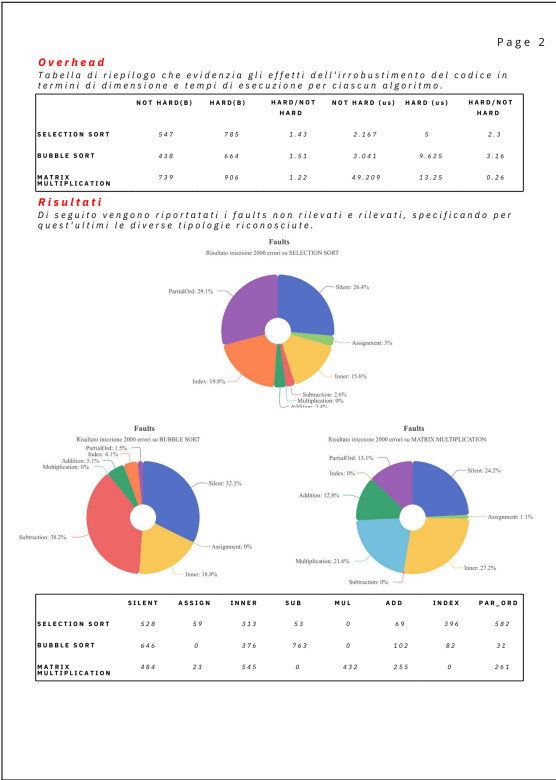


Figure 3.2: Pagina 2

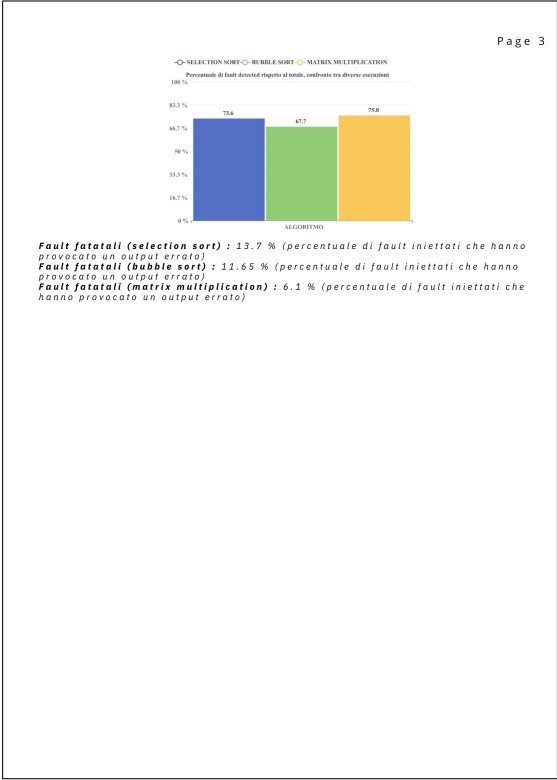


Figure 3.3: Pagina 3