

TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

**Relazione progetto
Performance Modeling of Computer
Systems and Networks
A.A. 2023/2024**

Donnini Francesco 0350960

Lupini Martina 0344256

Manenti Edoardo 0333574

Sommario

Questo studio modella le prestazioni del sistema di gestione dei contratti di prestito personale per Maac Finance Srl (MF), agente in attività finanziaria esclusivo di Santander Consumer Bank Spa (SCB). L'obiettivo è migliorare l'efficienza del processo, riducendo le istruttorie inutili e aumentando la percentuale di accettazioni del sistema automatico SCB. Il progetto include la costruzione di un modello concettuale, la simulazione delle prestazioni e l'implementazione di un modello migliorativo, la cui validità è stata confermata.

Indice

1 Introduzione	4
1.1 Processo di ottenimento di un prestito personale	5
1.2 Caratteristiche base inscindibili per richiedere un prestito	5
2 Problema e Obiettivi	7
3 Modello concettuale	7
3.1 Centri	7
3.2 Probabilità di routing	8
3.3 Eventi	9
4 Modello delle specifiche	9
4.1 Modellazione dei centri	10
4.1.1 Scelta delle distribuzioni	11
4.2 Equazioni di traffico	12
5 Modello computazionale	12
5.1 Simulation clock	14
5.2 Events e EventQueue	14
5.3 Applicant	14
5.4 Implementazione dei centri	16
5.5 getArrival() e getService()	18
5.6 Realizzazione di una LogNormale troncata	18
6 Verifica	19
6.1 Reparto istruttorie M/M/4	19
6.2 Sistema automatico M/M/1	20
6.3 Comitato credito M/M/ ∞	21
6.4 Reparto liquidazioni M/M/1	21
6.5 Tabella riassuntiva: medie analitiche, empiriche e intervalli di confidenza	22
7 Validazione	22
8 Design degli esperimenti	23
8.1 Intervalli di confidenza	23
8.2 Simulazione ad orizzonte finito	23
8.3 Simulazione ad orizzonte infinito	25
8.3.1 Autocorrelazione	26
9 Modello migliorativo	28
10 Modello concettuale	30
10.1 Eventi	30
11 Modello delle specifiche	30
11.1 Modellazione dei centri	30
11.2 Equazioni di traffico	31

12 Modello computazionale	31
13 Verifica	32
13.1 Sistema pre-scoring M/M/2	32
13.2 Reparto istruttorie M/M/2	33
13.3 Sistema automatico M/M/1	33
13.4 Comitato credito M/M/ ∞	34
13.5 Reparto liquidazioni M/M/1	34
13.6 Tabella riassuntiva: medie analitiche, empiriche e intervalli di confidenza	35
14 Validazione	35
15 Design degli esperimenti	35
15.1 Simulazione orizzonte finito	35
15.2 Simulazione orizzonte infinito	37
15.2.1 Autocorrelazione	38
15.3 Introduzione firma digitale per le liquidazioni	40
16 Conclusioni	42
A Appendice: Autorizzazione Acquisizione Dati MF	44

1 Introduzione

Il presente studio è incentrato sull'azienda **MAAC FINANCE SRL** (di seguito *MF*), una società iscritta all'OAM (*Organismo Agenti e Mediatori*) con il numero A8557, in qualità di agente in attività finanziaria per la **SANTANDER CONSUMER BANK SPA** (di seguito *SCB*), per la quale opera in esclusiva, rappresentandola sul territorio con quattro agenzie marchiate "Santander". La *MF* promuove tutti i prodotti offerti da *SCB* sul mercato italiano, tra cui: *Prestiti Personali*, *Cessione del Quinto*, *Credito al Consumo*, *Leasing Automobilistico* e *Noleggio a Lungo Termine*.

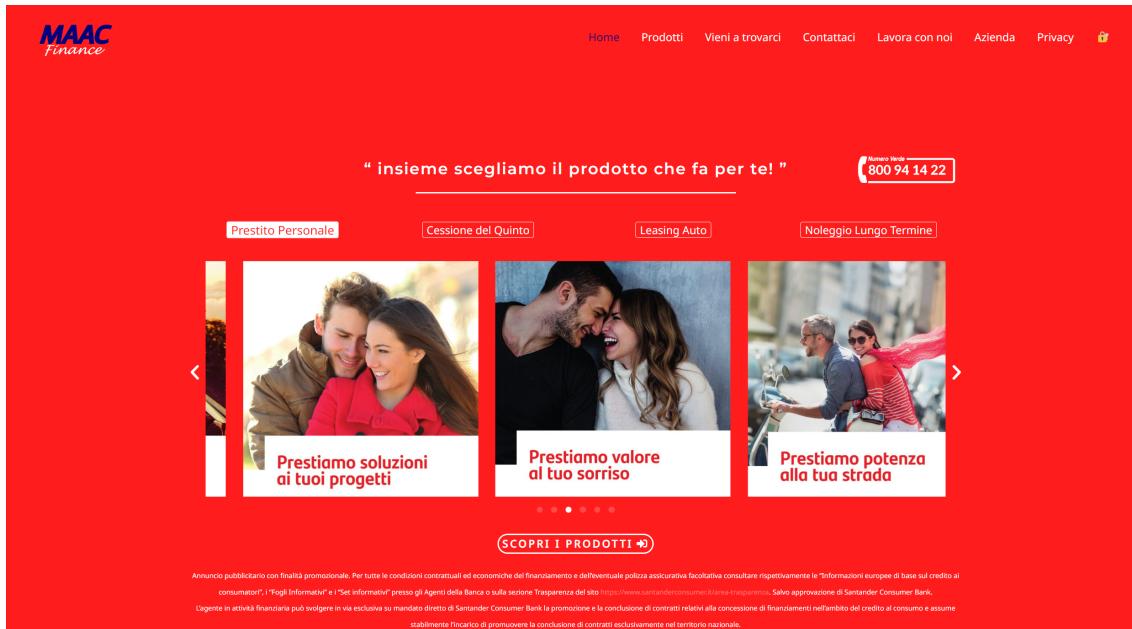


Figura 1: Sito web di MAAC FINANCE SRL

Nel corso di questa trattazione, verranno analizzati i dati forniti da *MF* (vedi autorizzazione in Appendice A) riguardanti i flussi di gestione del prodotto **Prestito Personale**. Questa scelta è stata dettata dal fatto che il prestito personale rappresenta l'unico prodotto che prevede un rapporto diretto tra il cliente e l'agenzia, senza che altre entità influenzino il processo, costituendo così un perfetto esempio di attività gestita integralmente dall'agenzia stessa.

Ad esempio:

- Nel *Credito al Consumo*, la richiesta viene presentata dal rivenditore;
- Nella *Cessione del Quinto*, è l'azienda in cui lavora il cliente a svolgere un ruolo determinante;
- Nel *Leasing*, il commercialista del cliente può influenzare l'operazione;
- Nel *Noleggio a Lungo Termine*, la disponibilità del modello desiderato o eventuali offerte specifiche possono incidere sulla decisione.

Il *Prestito Personale*, invece, permette di analizzare un processo decisionale gestito in maniera diretta dall'agenzia, senza interferenze esterne (questo fatto verrà approfondito al paragrafo successivo).

1.1 Processo di ottenimento di un prestito personale

Prima di proseguire con la trattazione, è opportuno spiegare i vari passaggi e le procedure effettuate a seguito della richiesta di un prestito personale da parte di un cliente.

1. **Istruttoria:** il primo passo consiste nella raccolta dei documenti necessari forniti dal cliente per completare la richiesta di prestito.
2. **Inserimento dati e controllo automatico:** successivamente, *MF* inserisce i dati extrapolati dai documenti e allega la digitalizzazione degli stessi in un sistema elettronico automatico di proprietà di *SCB*, il quale accetta o rifiuta le richieste basandosi su uno scoring automatico e sul controllo dell'assenza di anomalie nelle banche dati.
3. **Valutazione del comitato di credito:** se una richiesta viene accettata dal sistema automatico, passa al vaglio di un comitato di credito della *SCB*, composto da persone fisiche. Questa fase è la più lunga, in quanto prevede analisi e valutazioni approfondite sul conto del richiedente. Al termine, le richieste possono essere accettate, respinte, oppure può essere richiesta ulteriore documentazione o garanzie al cliente. Dai dati reali emerge che circa il 10% delle richieste ricade nell'ultimo caso; tuttavia, nel 4% di questi casi, il cliente decide di non procedere, per cui solo il 6% delle richieste riprende l'intero iter.
4. **Finalizzazione e liquidazione:** se le richieste vengono accettate dal comitato, *MF* procede con la finalizzazione tramite il cosiddetto *Reparto Liquidazione*, che prevede la firma del contratto da parte del cliente e infine l'esecuzione del bonifico della somma concordata sull'IBAN dello stesso.

Precisazione sulla scelta del Prestito Personale:

*Nel caso del prestito personale *MF* agisce come unico intermediario tra il cliente e il sistema decisionale. Santander interviene esclusivamente nelle fasi di valutazione del rischio e decisione finale tramite il sistema automatizzato e il comitato di credito, ma non influisce sul rapporto e sulla gestione operativa che *MF* ha con il cliente. Questo significa che, da un punto di vista gestionale e operativo, il cliente interagisce esclusivamente con *MF* durante tutto il processo, rendendo il prodotto unico nel suo genere rispetto ad altri, dove il coinvolgimento di terze parti è più invasivo e frequente.*

1.2 Caratteristiche base inscindibili per richiedere un prestito

Nella preparazione dell'istruttoria, è possibile rilevare informazioni importanti, alcune delle quali vengono acquisite dai documenti forniti dal cliente, mentre altre sono raccolte dal dialogo con il cliente.

Le caratteristiche base che vengono richieste durante il dialogo con il cliente sono le seguenti:

- **Presenza confermata del nominativo nelle banche dati interbancarie (es.: CRIF):**

- Il censimento del nominativo in una banca dati con un contratto aperto o chiuso, ma portato a termine, certifica l'affidabilità del cliente, avendo già rimborsato un prestito, anche se con un altro istituto di credito.
- Non devono esserci richieste o, peggio, rifiuti recenti da parte di altri istituti.
- La situazione delle banche dati riguardanti i debiti in corso non deve evidenziare un'esposizione elevata in rapporto al reddito del cliente.

- **Profilo lavorativo:**

- L'anzianità lavorativa dimostra continuità e stabilità (nel caso di liberi professionisti, l'anzianità lavorativa è ancora più indicativa).
- Il contratto deve essere a tempo indeterminato.

- **Situazione reddituale:**

- L'importo verrà confrontato con gli impegni mensili esistenti, sommati alla richiesta in corso.
- Devono sempre rimanere liberi almeno €526, considerati come reddito di sussistenza.

Dall'analisi dei dati forniti da *MF* abbiamo individuato, rispetto a questo tipo di informazioni, quattro macro-categorie responsabili dei rifiuti o accettazioni delle richieste, con le rispettive frequenze:

- Rifiuto o richieste recenti:** Questo è il problema più frequente (Si pensi che il 25% di chi effettua il colloquio con il reparto istruttorie rientra poi in questa casistica). Molti clienti effettuano preventivi online o presentano richieste a vari istituti di credito che vengono rifiutate o registrate nelle banche dati interbancarie come CRIF. Tali richieste multiple (anche se non ancora rifiutate), sommate, possono rappresentare un'esposizione debitoria eccessiva, rendendo improbabile l'accettazione di un ulteriore contratto.
- Tipologia di contratto:** Un'altra problematica comune riguarda la tipologia di contratto di lavoro. Spesso, i clienti credono di essere in regola ma, a causa di irregolarità nei versamenti contributivi (una criticità frequente nel sistema italiano), presentano contratti che offrono poche garanzie. Questo è un fattore che porta al rifiuto della richiesta (15.011%).
- Anzianità di lavoro:** Anche se meno frequente, un motivo ricorrente di rifiuto è l'anzianità lavorativa insufficiente. Molti clienti si presentano con il loro primo lavoro o con dichiarazioni come "*mi hanno messo in regola da poco*" (11.1%), il che non fornisce una stabilità sufficiente a garantire il prestito.
- Rapporto rata/reddito:** Questo è il fattore che causa meno problemi, in quanto i clienti che richiedono un prestito sono generalmente consapevoli delle proprie capacità di pagamento. Tuttavia, in rari casi (10%), può emergere un rapporto rata/reddito troppo alto rispetto al reddito disponibile, causando il rifiuto della richiesta.

Tuttavia, è importante considerare che tutto ciò che il cliente dichiara, ma che non è supportato da documenti, potrebbe non essere veritiero, soprattutto per quanto riguarda informazioni su debiti precedenti o in corso, o la presenza di richieste di prestito effettuate presso altri istituti di credito di recente. Queste informazioni possono essere confermate o confutate solo al termine dell'istruttoria, quando il sistema automatico di *SCB* accederà e verificherà le banche dati. In particolare, dallo storico fornito dai dati reali di *MF* risulta che circa il 18% di coloro che dichiaravano di avere le precedenti caratteristiche, sono stati poi confutati dal sistema automatico ricevendo un rifiuto. Riassumendo:

$$P\{\text{richiedente dice di non avere richieste o rifiuti recenti}\} = 0.75$$

$$P\{\text{richiedente dice di avere un contratto indeterminato}\} = 0.84989$$

$$P\{\text{richiedente dice di avere sufficiente anzianità di lavoro}\} = 0.889$$

$$P\{\text{richiedente dice di avere sufficiente rapporto tra rata e reddito}\} = 0.9$$

$$P\{\text{richiedente ha fornito dati riscontrati nelle banche dati}\} = 0.82$$

2 Problema e Obiettivi

Dai dati forniti da *MF* emerge che circa il 58% delle richieste vengono rifiutate dal sistema automatico. Questo significa che, per più della metà delle richieste, la procedura di istruttoria viene eseguita **“inutilmente”**. Il nostro obiettivo è quindi quello di ridurre questa percentuale, al fine di ottenere i seguenti benefici:

- **Maggior credibilità dell'azienda verso la mandante:**
 - Un maggior numero di richieste utili gestite indicherà una migliore qualità del portafoglio caricato da *MF* nei confronti della mandante *SCB*, la quale avrà così un indicatore economico più performante. Di conseguenza, *MF* verrà considerata un partner più affidabile e di valore dalla banca.
- **Risparmio di tempo e ottimizzazione delle risorse umane:**
 - Una riduzione della gestione delle istruttorie per i contratti che non verranno accettati consentirà di risparmiare tempo e lavoro, permettendo così una migliore ottimizzazione delle risorse umane disponibili.

Se fosse possibile determinare un metodo per scartare a priori le richieste non valide, si potrebbe risparmiare tempo nella fase di istruttoria e aumentare la percentuale di richieste approvate dal sistema automatico. L'obiettivo finale è quindi quello di trovare una strategia che permetta di aumentare la percentuale di accettazione, senza tuttavia ridurre il numero totale di contratti finalizzati e utilizzando le stesse risorse umane attualmente disponibili in *MF*.

3 Modello concettuale

3.1 Centri

Il processo è suddiviso nei seguenti reparti e fasi, modellati di conseguenza come dei centri del nostro modello:

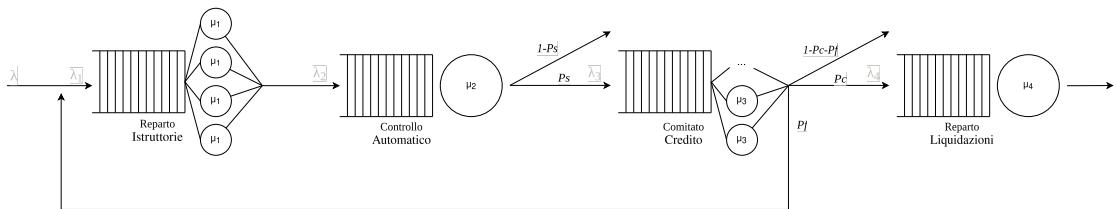
- **Reparto Istruttoria MF:** dove vengono espletate le funzioni di istruttoria:
 - Interazione col cliente per la raccolta dei documenti ($\sim 5min$);
 - Controllo completezza documentazione ($\sim 5min$);
 - Inserimento dei dati nel sistema *SCB* e digitalizzazione dei documenti ($\sim 15min$).
- **Sistema Automatico SCB:** dove le richieste vengono valutate in maniera automatica, con i seguenti esiti possibili:
 - Rifiutate;
 - Accettate.
- **Comitato Credito SCB:** le richieste approvate dal sistema vengono ulteriormente analizzate da un comitato di persone, che decide:
 - Accettazione;
 - Rifiuto;
 - Richiesta di approfondimento integrativo o maggiore garanzia.
- **Reparto Liquidazioni MF:** dove vengono finalizzati i contratti approvati dal comitato:
 - Interazione col cliente per la firma del contratto ($\sim 30min$);
 - Bonifico della somma contrattualizzata sull'IBAN del cliente.

3.2 Probabilità di routing

Dai dati forniti (numero di contratti accettati per ogni centro) risulta che:

- probabilità accettazione *Sistema scoring automatico* = 0.42
- probabilità di accettazione *Comitato crediti* = 0.65
- probabilità di feedback = 0.06

In base a quanto descritto il modello concettuale si può schematizzare nel seguente modo:



3.3 Eventi

Considerando una richiesta come un **job**, gli eventi considerati sono i seguenti:

- **Reparto istruttorie Maac Finance:**
 - ARRIVAL_REPARTOISTRUTTORIE è l'arrivo di una richiesta presso l'azienda o da parte del *Centro crediti Santander* (per aggiunta di documentazione).
 - COMPLETION_REPARTOISTRUTTORIE coincide con l'espletamento della procedura di istruttoria.
- **Sistema automatico Santander:**
 - ARRIVAL_SCORINGAUTOMATICO è l'arrivo di una richiesta proveniente dal centro *Reparto istruttorie Maac Finance*.
 - COMPLETION_SCORINGAUTOMATICO corrisponde al completamento della procedura di valutazione della richiesta.
- **Centro crediti Santander:**
 - ARRIVAL_COMITATO_CREDITO è l'arrivo di una richiesta proveniente dal centro *Sistema automatico Santander*.
 - COMPLETION_COMITATO_CREDITO corrisponde al completamento della procedura di valutazione della richiesta.
- **Reparto Liquidazioni Maac Finance:**
 - ARRIVAL_COMITATO_LIQUIDAZIONI è l'arrivo di una richiesta proveniente dal centro *Centro crediti Santander*.
 - COMPLETION_REPARTO_LIQUIDAZIONI corrisponde all'avvenuta finalizzazione della richiesta di prestito.

4 Modello delle specifiche

Per quanto riguarda gli arrivi al sistema, abbiamo deciso di utilizzare un processo di Poisson, in cui il tempo di inter-arrivo segue una distribuzione esponenziale. Questa scelta è motivata da tre fattori:

- I tempi di inter-arrivo sono indipendenti l'uno dall'altro;
- Possiedono la proprietà di memoryless (non dipendono dagli arrivi precedenti);
- Il processo di Poisson modella efficacemente la "casualità" degli arrivi.

Il tasso di arrivo medio, λ , è stato ricavato dai dati forniti dall'azienda. Da questi ultimi si evince che, in una giornata lavorativa di 8 ore, giungono circa 6 richieste di prestiti. Si ottiene dunque un tempo di inter-arrivo medio di:

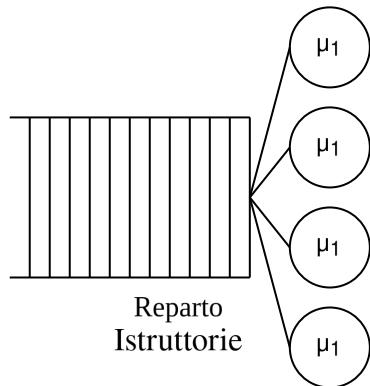
$$\frac{480 \text{ min}}{6} = 80 \text{ min}$$

ossia $\lambda = 0.0125 \frac{\text{job}}{\text{min}}$.

4.1 Modellazione dei centri

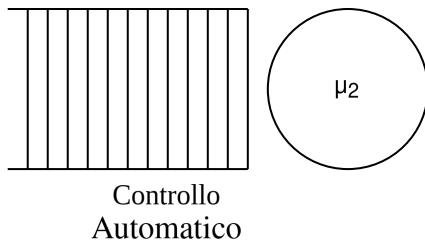
Passiamo ora alla modellazione dei singoli centri. Per scegliere le distribuzioni di servizio più appropriate ci siamo messi in contatto con i titolari dell'azienda Maac Finance Srl che ci ha fornito oltre che i tempi medi di servizio, anche dettagli sulla variabilità di questi.

- **Reparto istruttorie Maac Finance**



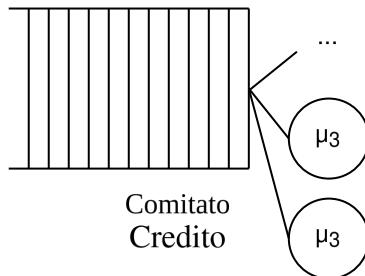
Questo centro è stato modellato come un multiserver con 4 serventi (che rappresentano i 4 dipendenti che si occupano dell'istruttoria dei contratti all'interno dell'azienda Maac Finance). Il servizio segue una distribuzione log normale troncata a 40min con $E[S] = 25\text{min}$ e $\sigma^2 = 10\text{min}^2$.

- **Sistema automatico Santander**



Il centro è stato modellato come un singolo server. Il tempo di servizio è modellato con una distribuzione esponenziale con $E[S] = 1\text{min}$.

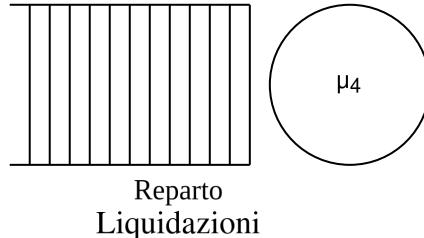
- **Centro crediti Santander**



Il comitato di credito è stato modellato come un sistema a capacità infinita ($M|G|\infty$), in quanto gestito da SCB, mandante di dimensioni notevoli, capaci di trattare un volume arbitrariamente elevato di richieste provenienti da

agenzie di tutta Italia, come MF. Questo approccio riflette la realtà operativa del comitato, che, grazie alla sua struttura robusta e ben organizzata, può processare tutte le richieste senza creare colli di bottiglia o ritardi significativi. Il tempo di servizio è modellato con una distribuzione log normale troncata a 1440min con $E[S] = 480$ min e $\sigma^2 = 180$ min.

- **Reparto Liquidazioni Maac Finance**



Il centro è stato modellato come un singolo server (che rappresenta il dipendente che si occupa della liquidazione dei contratti all'interno dell'azienda Maac Finance). Il tempo di servizio è modellato con una distribuzione log normale troncata a 60min con $E[S] = 30$ min, $\sigma^2 = 15$ min.

4.1.1 Scelta delle distribuzioni

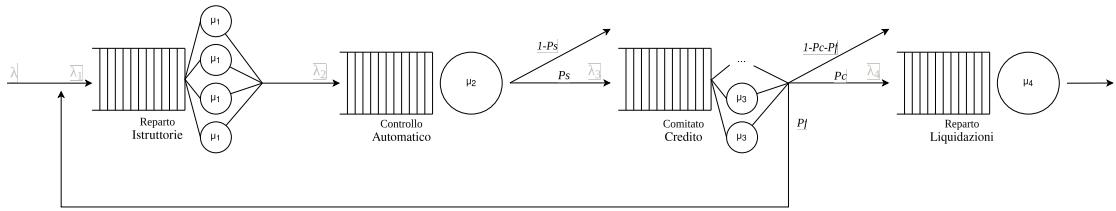
Per la modellazione dei tempi di servizio nei vari centri, abbiamo adottato un approccio deduttivo basato sulla natura dei centri stessi e sulle informazioni qualitative fornite dall'azienda. Non disponendo di dataset sufficientemente ampi per un fitting preciso delle distribuzioni, ci siamo concentrati su scelte che meglio riflettessero il comportamento atteso di ciascun centro.

Distribuzione Log Normale Troncata: La distribuzione log normale troncata è stata scelta per modellare i tempi di servizio in diversi centri, come il *Reparto Istruttorie*, il *Comitato di Credito* e il *Reparto Liquidazioni*. Questa scelta è stata motivata dalla necessità di rappresentare un processo di servizio che, per sua natura, non può durare un tempo troppo breve (tempo nullo o negativo) né estendersi indefinitamente.

Nel dettaglio, la distribuzione log normale cattura l'asimmetria dei tempi di servizio, dove la maggior parte delle operazioni si svolge intorno a un tempo medio, ma occasionalmente si verificano tempi più lunghi dovuti a casi complessi o imprevisti. La troncatura inferiore garantisce che i tempi di servizio non siano irrealisticamente bassi, mentre la troncatura superiore riflette i vincoli operativi o i tempi massimi previsti. Questa scelta è particolarmente appropriata per i centri che gestiscono attività con una certa variabilità o comunque complesse, come nel nostro caso.

Distribuzione Esponenziale: Per il *Sistema Automatico Santander*, abbiamo invece optato per una distribuzione esponenziale. Questo tipo di distribuzione è ampiamente utilizzato per modellare tempi di servizio in contesti dove le operazioni sono paragonabili a quelle di un "web server", cioè rapide, standardizzate e con bassa variabilità. L'assenza di memoria (memoryless property) della distribuzione esponenziale è ideale per un sistema automatico che processa richieste in maniera indipendente, senza che il tempo trascorso influenzi il tempo rimanente. Questo riflette bene la natura di un sistema automatico di scoring, dove ogni richiesta è trattata rapidamente e in modo uniforme, senza grandi variazioni nei tempi di elaborazione.

4.2 Equazioni di traffico



Come detto in 3.2, dai dati forniti dall'azienda emergono le seguenti probabilità di routing:

- $p_S=0.42$
- $p_C=0.65$
- $p_F=0.06$

Le equazioni di traffico sono:

$$\begin{cases} \lambda_1 = \lambda + \lambda_3 \cdot p_F \\ \lambda_2 = \lambda_1 \\ \lambda_3 = \lambda_2 \cdot p_S \\ \lambda_4 = \lambda_3 \cdot p_C \end{cases}$$

La tabella di routing è:

	Esterno	Reparto istruttorie	Sistema automatico	Comitato credito	Reparto liquidazione
Esterno	0	0	1	0	0
Reparto istruttorie	0	0	1	0	0
Sistema automatico	$1 - p_S$	0	0	p_S	0
Comitato credito	$1 - p_C - p_F$	p_F	0	0	p_C
Reparto liquidazione	1	0	0	0	0

Tabella 1: Tabella di routing modello base

5 Modello computazionale

Come linguaggio di programmazione per la simulazione è stato scelto Java per una serie di vantaggi quali, ad esempio, la gestione automatica della memoria, la maggiore flessibilità e il fatto che sia orientato agli oggetti. Inoltre possiede una libreria standard molto estesa e questo agevola il programmatore nello scrivere meno codice e introdurre così meno bug. Invece per creare i grafici è stato scelto Python perché fornisce una libreria (`matplotlib`) molto semplice da utilizzare nonché il concetto di notebook che permette un'analisi dei dati più approfondita.

```

.
.
.
Welcome to MAAC Finance Simulator!
Please select an option:
1. Start Simulation
2. Exit
Enter your choice >>> 1

```

Figura 2: Schermata iniziale del client a linea di comando per la simulazione del sistema in esame.

Per quanto riguarda l’organizzazione del codice esso è stato diviso in diversi *package*:

- **controller**: contiene il codice delle simulazioni, ossia della simulazione ad orizzonte finito ed infinito.
- **centers**: contiene le implementazioni dei vari centri. I centri derivano dalle classi astratte **SingleServer**, **Multiserver** e **InfiniteServer** che forniscono un’implementazione parziale delle operazioni comuni a tutti i centri dello stesso tipo - e.g. tutti i single server raccolgono le statistiche allo stesso modo e processano gli arrivi allo stesso modo - infatti la differenza tra le varie implementazioni (a parità di classe genitore) è data dai parametri (tempo di servizio medio, distribuzione del tempo di servizio, indici degli stream da cui generare i tempi di completamento o altro) e dagli eventi da generare.
- **model**: contiene le strutture dati principali nel nostro sistema come lo scheduler che si utilizza per prendere il prossimo evento da processare oppure per schedulare un evento generato e le varie classi per raccogliere le statistiche utilizzate ad esempio per calcolare le medie batch e gli intervalli di confidenza.
- **utils**: contiene classi ausiliarie per il elaborare le statistiche raccolte durante le varie simulazioni e presentarle a schermo.
- **libraries**: contiene il codice scritto da Steve Park e Dave Geyer. Si utilizzano le funzioni per generare le varie distribuzioni - e.g. **exponential** - e gli oggetti che consentono di generare i numeri casuali utilizzando il concetto di flussi.
- **configuration**: contiene una classe che permette di leggere un file di configurazione che contiene i parametri dei centri e delle varie simulazioni. Le informazioni sono codificate in sezioni e coppie chiave valore appartenenti ad una certa sezione, le chiavi devono essere univoche solamente all’interno della stessa sezione.

La simulazione adotta l’approccio del Next-Event Simulation, che consiste nel far avanzare il tempo processando gli eventi successivi in ordine cronologico. In altre parole, si individua l’evento più imminente (che sia un arrivo o un completamento) e si esegue l’azione associata ad esso, determinando così il cambiamento di stato del sistema e l’aggiornamento dei tempi. Andiamo ora ad analizzare alcune componenti importanti.

5.1 Simulation clock

Tra le principali strutture dati vi è sicuramente quella che registra il tempo nella simulazione che è `MsqTime`:

```
1 public class MsqTime {  
2     public double current;           /* current time */  
3     public double next;             /* next (most imminent) event  
4     time */  
5 }
```

In essa si tiene conto del tempo attuale e del tempo dell'evento successivo.

5.2 Events e EventQueue

L'evento è rappresentato dalla classe `MsqEvent` che è strutturata nel seguente modo:

```
1 public class MsqEvent {  
2     public double time;  
3     public final EventType type;  
4     public Applicant applicant;  
5     public int serverId;  
6     public double service;  
7 }
```

Tra gli attributi abbiamo:

- `time`: indica il tempo in cui si verifica l'evento
- `type`: indica il tipo di evento (sono quelli specificati al paragrafo 3.3)
- `applicant`: struttura dati che tiene traccia delle caratteristiche della richiesta in esame (ne parleremo in modo più approfondito nel paragrafo 5.3).
- `serverId`: è un attributo facoltativo. Viene utilizzato nel caso di un evento di completamento nel multiserver. Tiene traccia di quale server deve processare il completamento.
- `service`: se l'evento è di completamento mantiene la durata del servizio altrimenti è 0.

Per selezionare l'evento da processare si utilizza la classe `EventQueue` che implementa una coda FIFO. Sostanzialmente gli eventi vengono ordinati in base al loro attributo `time` e quando viene invocato il metodo `pop()` viene estratto dalla coda quello con tempo minore.

5.3 Applicant

Una grande sfida nella simulazione è stata modellare le richieste di prestito presentate e le loro caratteristiche. Basandoci sulle probabilità indicate al paragrafo 1.2, è stata creata ad hoc la classe `Applicant`:

```
1 public class Applicant {  
2     final int streamIndex;  
3     boolean haAnzianitaDiLavoro;  
4     boolean isRapportoRataRedditoOk;
```

```

5   boolean haContrattoIndeterminato;
6   boolean nonHaRichiesteORifiutiRecenti;
7   boolean haCorrispondenzaInBancaDati;
8
9   public static Applicant create(Rngs rngs) {
10    ConfigurationManager config = new ConfigurationManager();
11    int streamIndex = config.getInt("general", "applicantStreamIndex");
12    rngs.selectStream(streamIndex);
13    boolean nonHaRichiesteORifiutiRecenti = rngs.random() <=
14      0.75;
14    boolean haContrattoIndeterminato = rngs.random() <=
15      0.84989;
15    boolean haAnzianitaDiLavoro = rngs.random() <= 0.889;
16    boolean isRapportoRataRedditoOk = rngs.random() <= 0.9;
17    return new Applicant(streamIndex,
18      haAnzianitaDiLavoro,
19      isRapportoRataRedditoOk,
20      haContrattoIndeterminato,
21      nonHaRichiesteORifiutiRecenti);
22  }
23 }
```

Per ciascun attributo verrà assegnato il valore `false` oppure `true` a seconda delle probabilità discusse. Ad esempio, si avrà che il 25% delle richieste avrà l'attributo `nonHaRichiesteORifiutiRecenti` pari a `false`, ossia il cliente risulta avere richieste di prestito in sospeso o rifiuti nell'ultimo mese. La stessa logica viene applicata agli altri attributi. Per quanto riguarda la prima, essa viene determinata implicitamente dalla simulazione. Infatti, per decidere se una richiesta viene accettata o meno (quindi se un evento di completamento del centro *Sistema scoring automatico* debba essere propagato al centro successivo) si fa utilizzo degli attributi della classe `Applicant`.

```

1 public class Applicant {
2   public boolean isAcceptedBySysScoring() { /* ... */ }
3   public void setHasCorrispondenzaInBancaDati(Rngs rngs) { /*
4     ... */ }
```

- `isAcceptedBySysScoring()`: determina se il centro di *Scoring Automatico* debba accettare o meno la richiesta. In particolare, verifica che tutti gli attributi di `Applicant` siano `true`.
- `setHasCorrispondenzaInBancaDati()`: il centro di *Scoring Automatico* imposta seguendo la relativa distribuzione se i dati specificati nella richiesta sono effettivamente legittimi oppure non corretti.

Invece la probabilità di accettazione *Comitato crediti* e la probabilità di feedback vengono implementate utilizzando la funzione `getNextRoute()` che fa uso della classe `Rngs`:

```

1 private Route getNextRoute() {
2   rngs.selectStream(streamIndex + 1);
```

```

3     double x = rngs.random();
4     if (x < 0.06) {
5         return Route.FEEDBACK;
6     } else if (x >= 0.06 && x < 0.71) {
7         return Route.ACCEPTED;
8     } else {
9         return Route.REJECTED;
10    }
11 }
```

5.4 Implementazione dei centri

```

1 public abstract class SingleServer {
2     public void processArrival(MsqEvent arrival, MsqTime time,
3     EventQueue queue) { /* ... */ }
4     public void processCompletion(MsqEvent completion, MsqTime
5     time,
6     EventQueue queue) { /* ... */ }
7     protected abstract void spawnNextCenterEvent(MsqTime time,
8     EventQueue queue, MsqEvent current);
9     protected abstract void spawnCompletionEvent(MsqTime time,
10    EventQueue queue, MsqEvent current);
11    protected abstract double getService(int streamIndex);
```

`processArrival` e `processCompletion` sono utilizzati nell'*event loop* delle varie simulazioni e permettono ai centri di elaborare le richieste in entrata ed i relativi completamenti. Durante l'elaborazione degli eventi vengono raccolte le statistiche di interesse ($E[T_s]$, $E[N_s]$, $E[T_q]$, $E[N_q]$, $E[S]$, ρ , λ), generati gli eventi causati da quello appena elaborato - e.g. il processamento di un arrivo può causare la creazione e l'aggiunta in `queue` di un evento di completamento (se il servente/serventi sono liberi). In generale, diversi centri derivanti dalla stessa classe astratta processano gli eventi allo stesso modo, l'unica cosa che cambia sono gli eventi che possono essere generati a causa di quelli elaborati, per questo motivo non è stata fornita un'implementazione di `spawnNextCenterEvent` e `spawnCompletionEvent` in modo che ogni classe concreta possa decidere che tipologia di eventi generare. Anche la distribuzione dei tempi di servizio può cambiare da centro a centro e il compito di decidere come il tempo deve essere generato ricade sulla classe derivata.

```

1 public void processArrival(MsqEvent arrival, MsqTime time,
2     EventQueue queue){
3     numberOfJobsInNode++;
4     if(firstArrivalTime == Double.NEGATIVE_INFINITY) {
5         firstArrivalTime = arrival.time;
6     }
7     lastArrivalTime = arrival.time;
8
9     if (numberOfJobsInNode == 1) {
10        spawnCompletionEvent(time, queue, arrival);
11    } else {
12        fifoQueue.addFirst(arrival.applicant);
```

```

13      }
14  }
```

- 2 è la variabile di stato che memorizza il numero di richieste nel sistema: viene incrementata ad ogni nuovo arrivo.
- 10 se la richiesta appena arrivata è l'unica presente nel sistema allora viene subito elaborata dal servente, `spawnCompletionEvent` genera il tempo di registro tramite `getService` e aggiunge il relativo evento di completamento in `queue`.
- 12 mantiene in memoria l'oggetto `Applicant` associato ad un evento arrivato nel sistema ma che non può essere processato subito, questo oggetto viene ripreso dalla coda quando il relativo evento di arrivo può essere servito.

`MultiServer` e `InfiniteServer` funzionano in modo simile, ovviamente con le dovute differenze dovuto alla presenza di più serventi.

```

1 public void processCompletion(MsqEvent completion , MsqTime time ,
2     EventQueue queue) {
3     numberOfJobsInNode--;
4
5     sum.served++;
6     sum.service += completion.service ;
7
8     lastCompletionTime = completion.time ;
9
10    if (!warmup && jobServedPerBatch == batchSize ) {
11        saveBatchStats(time);
12    }
13    if (completion.applicant == null) {
14        completion.applicant = fifoQueue.poll();
15    }
16    spawnNextCenterEvent(time , queue , completion);
17    if (numberOfJobsInNode > 0) {
18        spawnCompletionEvent(time , queue , completion);
19    }
}
```

- 2 ad ogni completamente viene decrementato il numero di richieste nel sistema.
- 4, 5 memorizzano il numero di job e il busy time del server rispettivamente.
- 9 se è stata superata la fase di *warmup* e sono stati processati *B* richieste allora è possibile registrare le medie batch dei vari indici di prestazione.
- 12 permette di recuperare dalla coda il relativo oggetto `Applicant`.
- 15 è stato completato un job quindi è possibile generare un nuovo evento - e.g. viene processato un evento di completamento in *Reparto Istruttorie* e quindi si genera un evento di arrivo in *Controllo Automatico*.
- 17 una volta elaborato un job è possibile prendere un altro (se esiste) in coda per iniziare ad elaborarlo.

5.5 getArrival() e getService()

Per calcolare i tempi di arrivo, i tempi di servizio e le probabilità di routing si utilizza la classe `Rngs`. In particolare si utilizza un singolo generatore con un approccio multi-stream. In questo modo si garantisce l'indipendenza tra i numeri generati e si ha solo un'unica chiamata al metodo `rngs.plantSeed()` (approfondiremo nel paragrafo 8 come viene invocato tale metodo a seconda del tipo di simulazione).

Per quanto riguarda `getArrival()` tale metodo viene implementato nel primo centro del sistema in tal modo:

```

1 public double getArrival() {
2     rngs.selectStream(streamIndex + 1);
3     sarrival += exponential(interArrivalTime, rngs);
4     return sarrival;
5 }
```

`getService()`, come detto in precedenza, viene implementato in ogni centro in quanto differiscono per distribuzione del tempo di servizio e tempo medio. I centri *Reparto istruttorie*, *Reparto Liquidazioni* e *Comitato crediti* hanno un tempo di servizio che segue una distribuzione logNormale troncata per cui:

```

1 double getService(int streamIndex) {
2     rngs.selectStream(streamIndex);
3     double serviceTime;
4     if(approximateServiceAsExponential){
5         serviceTime = exponential(meanServiceTime, rngs);
6     } else {
7         serviceTime = truncatedLogNormal(meanServiceTime, sigma,
8                                         truncationPoint, rngs);
9     }
10    return serviceTime;
11 }
```

Il *Sistema scoring automatico* ha tempi di servizio che seguono una distribuzione esponenziale:

```

1 protected double getService(int streamIndex) {
2     rngs.selectStream(streamIndex);
3     double serviceTime;
4     serviceTime = exponential(meanServiceTime, rngs);
5     return serviceTime;
6 }
```

5.6 Realizzazione di una LogNormale troncata

Nella realizzazione della distribuzione LogNormale troncata, abbiamo seguito le linee guida descritte nel testo di riferimento, utilizzando le funzioni di libreria rilasciate da Steve Park, Dave Geyer e tradotte da Richard Dutton e Jun Wang. [3]

Per il calcolo dei parametri necessari alla generazione dei tempi di servizio, abbiamo utilizzato le seguenti formule per la media e la varianza della distribuzione lognormale (sempre presenti nel codice delle librerie suddette):

$$\text{mean} = \exp(a + 0.5 \cdot b^2)$$

$$\text{variance} = (\exp(b^2) - 1) \cdot \exp(2a + b^2)$$

Dalle inverse di tali formule è possibile ricavare i parametri a e b necessari nella funzione `idfLogNormal()`. Di seguito è riportato il codice Java che implementa questa distribuzione:

```

1 public static double truncatedLogNormal(double mu, double sigma,
2                                         double truncationPoint, Rngs rngs) {
3     Rvms rvms = new Rvms();
4     Rvgs rvgs = new Rvgs(rngs);
5
6     double variance = sigma * sigma;
7     double b = Math.sqrt(Math.log(1 + (variance / (mu * mu))));  

8     double a = Math.log(mu) - 0.5 * b * b;
9
10    double alpha = rvms.cdfLogNormal(a, b, 1e-20);
11    double beta = 1.0 - rvms.cdfLogNormal(a, b, truncationPoint);
12
13    double u = rvgs.uniform(alpha, 1.0 - beta);
14
15    return rvms.idfLogNormal(a, b, u);
16 }
```

6 Verifica

La fase di verifica si assicura che il modello computazionale sia implementato correttamente. Per far ciò si confrontano i risultati ottenuti dalla simulazione con i risultati analitici considerando un **intervallo di confidenza del 95%** (il calcolo degli intervalli di confidenza è trattato al paragrafo 8.1). Affinché il paragone possa essere fatto si considerano tempi di servizio di tipo **esponenziale**. Nella fase di verifica considereremo il caso stazionario con simulazione ad orizzonte infinito (di cui tratteremo più avanti). Per ogni centro sono elencati i calcoli analitici e viene allegato uno screenshot della verifica automatica del simulatore.

6.1 Reparto istruttorie M/M/4

$$\begin{aligned}\lambda &= 0.0125 \text{ job/min} \\ \lambda_1 &= \frac{\lambda}{1 - p_C \cdot p_F} = 0.0128231432 \text{ job/min} \\ E(S_i) &= \frac{1}{\mu_i} = 25 \text{ min}, i = 1, 2, 3, 4 \\ E(S) &= \frac{E(S_i)}{4} = 6.25 \text{ min} \\ \mu_i &= \frac{1}{E(S_i)} = 0.04 \text{ job/min} \\ \rho &= \frac{\lambda_1}{4 \cdot \mu_i} = 0.08014465 \\ N &= 4\end{aligned}$$

$$p_0 = \left[\sum_{i=0}^{N-1} \frac{(N \cdot \rho)^i}{i!} + \frac{(N \cdot \rho)^N}{N! \cdot (1 - \rho)} \right]^{-1} = 0.725596$$

$$P_q = \frac{(N \cdot \rho)^N \cdot p_0}{N! \cdot (1 - \rho)} = 3.479248 \cdot 10^{-4}$$

$$E(T_q) = \frac{P_q \cdot E(S)}{1 - \rho} = 0.0023591 \text{ min}$$

$$E(N_q) = E(T_q) \cdot \lambda_1 = 3.02506 \cdot 10^{-5}$$

$$E(T_s) = E(T_q) + E(S_i) = 25.00235907 \text{ min}$$

$$E(N_s) = E(T_s) \cdot \lambda_1 = 0.32060883$$

```
*****
FINAL RESULTS FOR REPARTOISTRUTTORIE_MAAC with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 25.016879005158188, diff 0.014519942392102081 is within the interval ±0.0724925815223798
E[Tq]: mean 0.002552850366797088, diff 1.9378760071058236E-4 is within the interval ±5.944165585319648E-4
E[Ns]: mean 0.3212020419282775, diff 5.932111070053314E-4 is within the interval ±0.0011708374332829808
E[Nq]: mean 3.293217290417787E-5, diff 2.6815732159532837E-6 is within the interval ±7.72773430218197E-6
ρ: mean 0.08030399202811887, diff 1.59346972722893E-4 is within the interval ±2.9229156777572815E-4
*****
```

6.2 Sistema automatico M/M/1

$$\lambda_2 = \lambda_1 = 0.0128231432 \text{ job/min}$$

$$E(S) = 1 \text{ min}$$

$$\mu = \frac{1}{E(S)} = 1 \text{ job/min}$$

$$\rho = \lambda_2 \cdot E[S] = 0.01283023$$

$$E(T_q) = \frac{\rho \cdot E(S)}{1 - \rho} = 0.0129970 \text{ min}$$

$$E(N_q) = E(T_q) \cdot \lambda_2 = 1.667548 \cdot 10^{-4}$$

$$E(T_s) = E(T_q) + E(S) = 1.0129970 \text{ min}$$

$$E(N_s) = E(T_s) \cdot \lambda_2 = 0.012997$$

```
*****
FINAL RESULTS FOR SYS_SCORING_AUTOMATICO_SANTANDER with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 1.0138667780783805, diff 8.770659304018213E-4 is within the interval ±0.0027080082799028843
E[Tq]: mean 0.013151941436362335, diff 1.622292883835339E-4 is within the interval ±4.051250173540709E-4
E[Ns]: mean 0.013017939538140389, diff 2.8227390161587526E-5 is within the interval ±4.926638019872607E-5
E[Nq]: mean 1.6888500451846554E-4, diff 2.316065403021306E-6 is within the interval ±5.228670516556224E-6
ρ: mean 0.012849054533621923, diff 2.5911324758565515E-5 is within the interval ±4.7714407429153584E-5
*****
```

6.3 Comitato credito M/M/ ∞

$$\begin{aligned}\lambda_3 &= \lambda_2 \cdot p_S = 0.0053857 \text{ job/min} \\ E(S) &= 480 \text{ min} \\ \mu &= \frac{1}{E(S)} = 0.002083 \text{ job/min} \\ \rho &= 1 - e^{-\frac{\lambda_3}{\mu}} = 0.9246149 \\ E(T_q) &= 0 \text{ min} \\ E(N_q) &= 0 \\ E(T_s) &= E(T_q) + E(S) = 480 \text{ min} \\ E(N_s) &= E(T_s) \cdot \lambda_3 = 2.5851456\end{aligned}$$

```
*****
FINAL RESULTS FOR COMITATO_CREDITO_SANTANDER with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 479.79116639662396, diff 0.20883360337603563 is within the interval ±1.3962828306559831
E[Tq]: mean 0.0, diff 0.0 is within the interval ±0.0
E[Ns]: mean 2.579470219100673, diff 0.005675451806180121 is within the interval ±0.010406329537432032
E[Nq]: mean 0.0, diff 0.0 is within the interval ±0.0
p: mean 0.9244090983083525, diff 2.058043058943415E-4 is within the interval ±0.0012183668344471696
*****
```

6.4 Reparto liquidazioni M/M/1

$$\begin{aligned}\lambda_4 &= \lambda_3 \cdot p_C = 0.0035007 \text{ job/min} \\ E(S) &= 30 \text{ min} \\ \mu &= \frac{1}{E(S)} = 0.0333333 \text{ job/min} \\ \rho &= \lambda_4 \cdot E[S] = 0.1050215 \\ E(T_q) &= \frac{\rho \cdot E(S)}{1 - \rho} = 3.5203599 \text{ min} \\ E(N_q) &= E(T_q) \cdot \lambda_4 = 0.0123237 \\ E(T_s) &= E(T_q) + E(S) = 33.5203599 \text{ min} \\ E(N_s) &= E(T_s) \cdot \lambda_4 = 0.1173453\end{aligned}$$

```
*****
FINAL RESULTS FOR REPARTO_LIQUIDAZIONI_MAAC with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 33.4753226932729, diff 0.04503722305228308 is within the interval ±0.10952888313329419
E[Tq]: mean 3.5071100486530185, diff 0.013249867672167515 is within the interval ±0.04878333670871706
E[Ns]: mean 0.11703457435239925, diff 3.1075619177360625E-4 is within the interval ±5.111142539440547E-4
E[Nq]: mean 0.012264540967753599, diff 5.924669582836606E-5 is within the interval ±1.8095159653668165E-4
p: mean 0.10477003338464565, diff 2.5150949594525407E-4 is within the interval ±3.811849175969086E-4
*****
```

6.5 Tabella riassuntiva: medie analitiche, empiriche e intervalli di confidenza

Indice	Valore Analitico	Valore Empirico	CI
$E[Ns]_{RIM}$	0.320609	0.321202	± 0.001171
$E[Ts]_{RIM}$	25.002359	25.016879	± 0.072493
$E[Nq]_{RIM}$	$3.02506 \cdot 10^{-5}$	$3.293217 \cdot 10^{-5}$	$\pm 7.727734 \cdot 10^{-6}$
$E[Tq]_{RIM}$	0.002359	0.002553	$\pm 5.944166 \cdot 10^{-4}$
ρ_{RIM}	0.080145	0.080304	$\pm 2.922916 \cdot 10^{-4}$
$E[Ns]_{SSAS}$	0.012997	0.013018	$\pm 4.051250 \cdot 10^{-4}$
$E[Ts]_{SSAS}$	1.012997	1.013867	± 0.002708
$E[Nq]_{SSAS}$	$1.667548 \cdot 10^{-4}$	$1.688850 \cdot 10^{-4}$	$\pm 5.228671 \cdot 10^{-6}$
$E[Tq]_{SSAS}$	0.012997	0.013152	$\pm 4.051250 \cdot 10^{-4}$
ρ_{SSAS}	0.012823	0.012849	$\pm 4.771441 \cdot 10^{-5}$
$E[Ns]_{CCS}$	2.585146	2.579470	± 0.010406
$E[Ts]_{CCS}$	480.000000	479.791166	± 1.396283
$E[Nq]_{CCS}$	0.0	0.0	-
$E[Tq]_{CCS}$	0.0	0.0	-
ρ_{CCS}	0.924615	0.924409	± 0.0012184
$E[Ns]_{RLM}$	0.117345	0.117035	$\pm 5.111143 \cdot 10^{-4}$
$E[Ts]_{RLM}$	33.520360	33.475323	± 0.10952
$E[Nq]_{RLM}$	0.012324	0.012265	$\pm 1.809516 \cdot 10^{-4}$
$E[Tq]_{RLM}$	3.520360	3.507110	± 0.0487833
ρ_{RLM}	0.0123237	0.104770	$\pm 3.811849 \cdot 10^{-4}$

Tabella 2: Tabella riassuntiva: medie analitiche, empiriche e intervalli di confidenza

7 Validazione

In questa fase ci assicuriamo che il modello computazionale già testato in precedenza sia conforme e consistente con il caso di studio preso in considerazione. Per far ciò si utilizzano nei tempi di servizio le distribuzioni effettive e non quelle esponenziali. In primo luogo è necessario controllare che la probabilità di accettazione di *Scoring automatico* sia coerente con quella osservata empiricamente. Dai dati aziendali emerge che circa il 42% delle richieste viene accettata dal sistema di scoring. La percentuale di accettazione ottenuta tramite simulazione è 41,857572%.

Un altro check effettuabile è quello sulle utilizzazioni. Infatti, come detto nell'introduzione, *MF* non si occupa solo di prestiti ma offre molti altri servizi. Di conseguenza, i dipendenti non occupano la totalità del loro turno lavorativo per espletare le pratiche relative ai contratti di prestito personale. Questo trova un riscontro positivo nella simulazione, osservando le utilizzazioni medie dei centri *Reparto istruttorie* e *Reparto liquidazioni*. Per quanto riguarda *Scoring automatico* anche in questo caso possiamo dire che non è occupato totalmente dalle richieste provenienti da *MF* dal momento che elabora richieste da molti altri centri di tutta Italia. Anche questo fatto si riflette nell'utilizzazione molto bassa riportata nella simulazione.

Infine, come ultimo check consideriamo il numero totale di richieste prestito portate a liquidazione. Il risultato ottenuto con la simulazione è coerente con quanto riportato dai dati aziendali. Il modello risulta quindi valido.

8 Design degli esperimenti

8.1 Intervalli di confidenza

All'interno della trattazione abbiamo fatto e faremo più volte riferimento agli intervalli di confidenza. In particolare, sono stati scelti intervalli di confidenza con un *level of confidence* del $95\% = (1-\alpha)\%$. Essi sono stati calcolati mediante la formula:

$$\text{Intervallo di Confidenza} = t^* \cdot \left(\frac{s}{\sqrt{n-1}} \right)$$

dove s è la deviazione standard campionaria, n è la dimensione del campione, t^* è il valore critico della distribuzione t di Student (per calcolarlo nel codice è stata utilizzata la funzione `idfStudent()` fornita dalla classe `Rvms`).

8.2 Simulazione ad orizzonte finito

La simulazione è stata condotta utilizzando un orizzonte temporale di una settimana lavorativa, corrispondente a un totale di 2,400 minuti (5×8 ore \times 60 minuti). L'adozione di un orizzonte di una settimana lavorativa permette di ottenere un quadro accurato delle performance del sistema durante la fase di transizione, senza tuttavia estendersi a periodi così lunghi da includere effetti stagionali o altre variabili esterne che potrebbero complicare l'analisi. Questo orizzonte è sufficientemente lungo da essere informativo, dato che il sistema processa in media solo 6 contratti al giorno per il prodotto di Prestito Personale, ma è ancora breve rispetto ai periodi che permetterebbero di osservare un comportamento completamente stazionario. In questo modo, possiamo focalizzarci sulle dinamiche transitorie del sistema.

In linea con le raccomandazioni metodologiche di Law [2], la simulazione è stata eseguita utilizzando il metodo delle replicazioni: sono state prodotte 150 esecuzioni indipendenti su questa finestra temporale di una settimana. Alla fine della simulazione, per ogni statistica di interesse, sono stati raccolti 150 campioni che rappresentano le medie calcolate durante la singola esecuzione. È di cruciale importanza che, tra due esecuzioni differenti, il sistema non abbia lo stesso stato iniziale. Per evitare questo scenario, si utilizza lo stato finale dell'istanza corrente del random generator stream (`Rngs`) (ottenibile tramite la chiamata a `rngs.getSeed()`) relativo alla replicazione appena terminata come stato iniziale per la replicazione successiva. Questo approccio garantisce l'indipendenza delle osservazioni, conforme alle best practices per la simulazione ad orizzonte finito.

Eseguendo la simulazione si osservano i seguenti risultati:

Centro	$E[N_s]$	$E[T_s]$	$E[N_q]$	$E[T_q]$	ρ
RIM	$0.3273 \pm 6.3250 \cdot 10^{-4}$	$25.0216 \pm 2.3263 \cdot 10^{-2}$	$0.0001 \pm 2.9703 \cdot 10^{-8}$	$0.0047 \pm 2.2996 \cdot 10^{-4}$	$0.0827 \pm 4.0482 \cdot 10^{-5}$
SSAS	$0.0129 \pm 1.9656 \cdot 10^{-6}$	$0.9859 \pm 6.1441 \cdot 10^{-3}$	$0.0002 \pm 2.1780 \cdot 10^{-8}$	$0.0133 \pm 1.2996 \cdot 10^{-4}$	$0.0127 \pm 1.8493 \cdot 10^{-6}$
CCS	$2.2648 \pm 6.5234 \cdot 10^{-2}$	479.6687 ± 2.2132	-	-	$0.8839 \pm 1.3109 \cdot 10^{-3}$
RLM	$0.0984 \pm 2.1833 \cdot 10^{-4}$	31.5648 ± 1.1917	$0.0053 \pm 1.5090 \cdot 10^{-5}$	$1.4698 \pm 7.6538 \cdot 10^{-1}$	$0.0932 \pm 1.5775 \cdot 10^{-4}$

Tabella 3: Statistiche riassuntive simulazione ad orizzonte finito del modello base

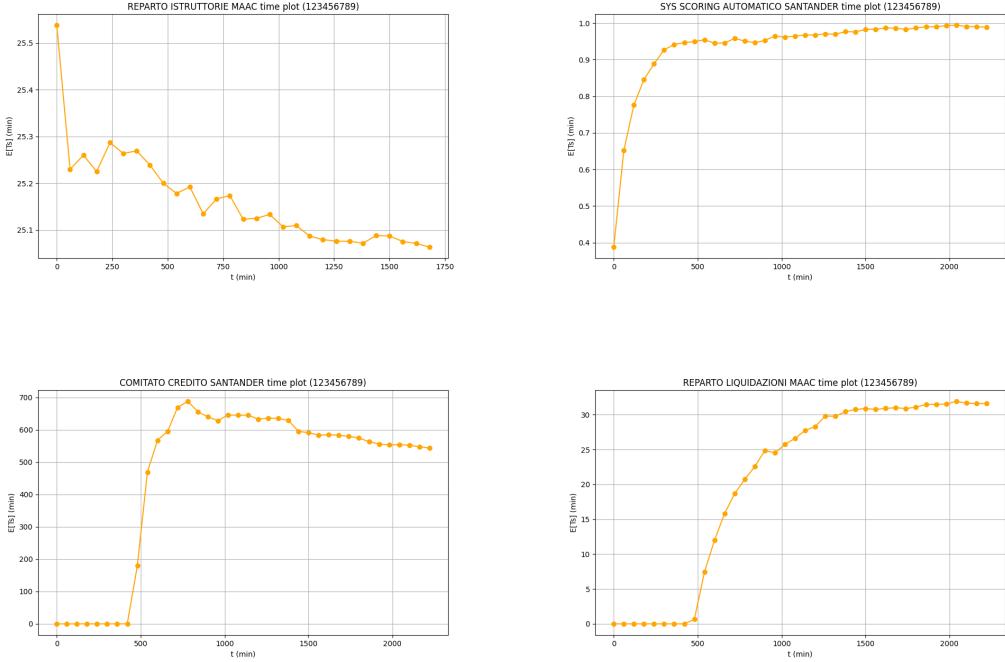


Tabella 4: Tempi di risposta dei vari centri. Alcuni dei valori iniziali sono nulli perché le statistiche sono state raccolte ogni 60 min e, quindi, durante l'istante di osservazione può succedere che un centro non abbia ancora ricevuto alcuna richiesta.

Nella simulazione, in cui sono state utilizzate le distribuzioni di servizio effettive, i risultati ottenuti non si discostano da quanto ci si aspettava. Le utilizzazioni sono basse ed i tempi di attesa praticamente nulli. Analizziamo le motivazioni alla base di ciò nei vari centri. Nel primo centro, i tempi di attesa e utilizzazioni minime sono dovute al basso tasso di arrivo se paragonato al tasso di servizio e alla presenza di quattro serventi. Il sistema di scoring automatico, pur essendo modellato come single server, detiene un tasso di servizio decisamente basso. Per quanto riguarda il reparto liquidazioni esso presenta un tempo di coda leggermente più alto ma questo è dovuto al fatto che si tratta di un single server con un tempo di servizio non troppo piccolo.

L'utilizzazione del comitato credito può risultare elevata. Tuttavia, l'utilizzazione in un sistema $M/M/\infty$ non è definita come in un sistema con un numero finito di serventi. In un sistema con infiniti serventi, l'utilizzazione viene definita come la probabilità che un qualsiasi servente sia occupato, il che è collegato alla probabilità che un job stia ricevendo servizio al momento. L'alta utilizzazione in un sistema $M/M/\infty$ è principalmente dovuta alla lunga durata del servizio. Anche se pochi job arrivano al sistema, ciascuno rimane in servizio per un tempo così lungo da garantire che ci sia sempre una frazione significativa di serventi occupati.

Centro	Job Serviti (Avg)	Busy Time
Reparto Istruttorie	31.340000	783.957865
Scoring Automatico	31.340000	30.42243
Comitato Credito	13.206667	6334.44399
Reparto Liquidazioni	8.360000	252.01390

Tabella 5: Misure di interesse per centro sul periodo di osservazione della simulazione del modello base

8.3 Simulazione ad orizzonte infinito

L'obiettivo di una simulazione ad orizzonte infinito è quello di stimare le caratteristiche del sistema quando si trova in regime stazionario. A differenza della simulazione ad orizzonte finito, che è progettata per analizzare il comportamento del sistema su un periodo di tempo specifico e limitato, la simulazione ad orizzonte infinito si concentra sul comportamento di lungo periodo del sistema, idealmente in uno stato dove le variabili di interesse hanno raggiunto una stabilità statistica, non influenzata dalle condizioni iniziali.

Di solito, le caratteristiche del sistema all'inizio della simulazione differiscono da quelle del sistema in regime stazionario, ad esempio quando il sistema viene avviato vuoto. È quindi necessario eseguire la simulazione per un certo periodo di tempo prima che il sistema raggiunga questo stato stazionario. Le osservazioni registrate durante questo periodo iniziale potrebbero influenzare negativamente la precisione delle stime degli indici di prestazione, specialmente se il periodo di transizione è particolarmente lungo. Questo problema è noto come *initialization bias* o *start-up problem* [4].

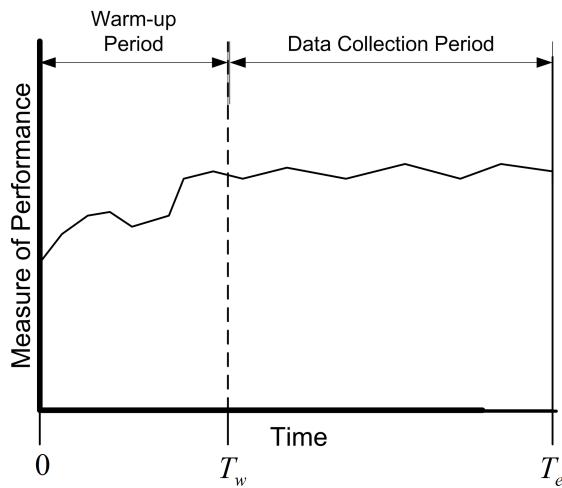


Figura 3: Immagine esemplificativa che illustra il motivo per cui è utile introdurre un tempo di warmup durante una simulazione a orizzonte infinito

Per mitigare questo problema, abbiamo deciso di scartare le misure relative ai primi $r \cdot b \cdot k$ job processati. Per la simulazione batch del modello iniziale, abbiamo scelto come parametro $r = 0.275$, mentre per il modello migliorato abbiamo selezionato $r = 0.3$. In generale, non esiste un metodo assoluto per determinare il valore di r , bisogna porre particolare attenzione a non scegliere un valore troppo alto di r perché potrebbe aumentare eccessivamente l'autocorrelazione tra i dati raccolti.

Dopodiché è stato utilizzato il metodo delle *Batch Means* per stimare le statistiche di interesse, come parametri abbiamo scelto $b = 4096$ (grandezza di un batch) e $k = 128$ (numero delle batch). Il valore di b scelto inizialmente era più basso ma è stato incrementato iterativamente fino a quando l'autocorrelazione dei valori medi di ogni batch (a parità di indice) non è diventata minore di 0.2 come proposto da Banks, Carson, Nelson, and Nicol (2001, page 438) [1] (vedere paragrafo 8.3.1).

Eseguendo la simulazione si osservano i seguenti risultati:

Centro	$E[N_s]$	$E[T_s]$	$E[N_q]$	$E[T_q]$	ρ
RIM	$0.3212 \pm 7.8071 \cdot 10^{-6}$	$25.0169 \pm 2.9928 \cdot 10^{-2}$	$0.0000 \pm 3.4009 \cdot 10^{-10}$	$0.0026 \pm 2.0122 \cdot 10^{-6}$	$0.0803 \pm 4.8655 \cdot 10^{-7}$
SSAS	$0.0130 \pm 1.3823 \cdot 10^{-8}$	$1.0139 \pm 4.1763 \cdot 10^{-5}$	$0.0000 \pm 1.5570 \cdot 10^{-10}$	$0.0132 \pm 9.3470 \cdot 10^{-7}$	$0.0128 \pm 1.2966 \cdot 10^{-8}$
CCS	$2.5795 \pm 6.1672 \cdot 10^{-4}$	479.7912 ± 11.1031	-	-	$0.9244 \pm 8.4538 \cdot 10^{-6}$
RLM	$0.1170 \pm 1.4878 \cdot 10^{-6}$	$33.4753 \pm 6.8321 \cdot 10^{-2}$	$0.0123 \pm 1.8647 \cdot 10^{-7}$	$3.5071 \pm 1.3553 \cdot 10^{-2}$	$0.1048 \pm 8.2750 \cdot 10^{-7}$

Tabella 6: Statistiche riassuntive simulazione ad orizzonte infinito del modello base

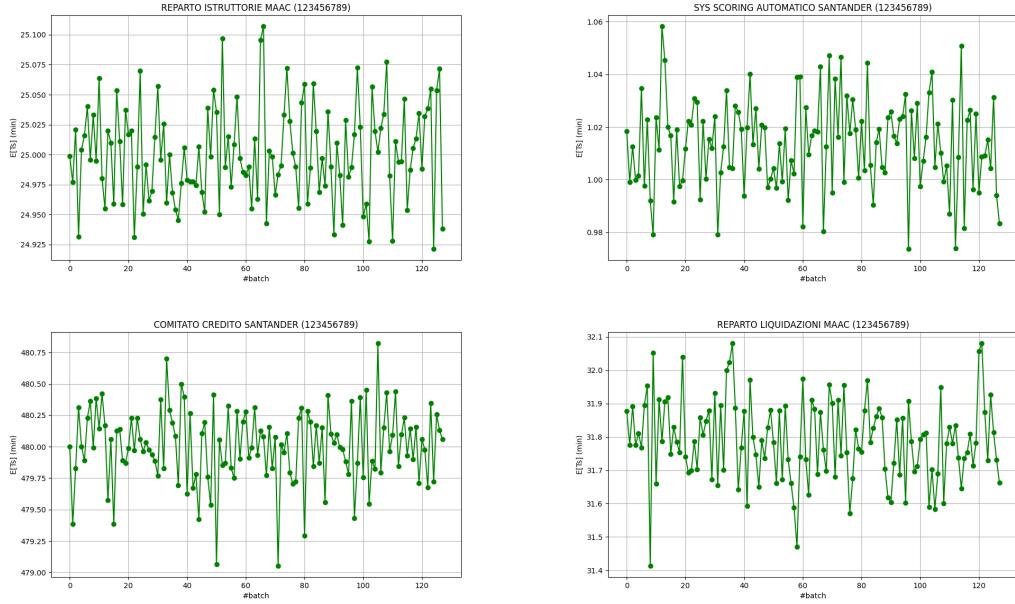


Tabella 7: Tempi di risposta dei vari centri

Anche nella simulazione ad orizzonte infinito, i risultati ottenuti confermano le aspettative precedenti. Le utilizzazioni si mantengono stabili e i tempi di attesa sono pressoché nulli. Nel primo centro, il lieve incremento nel numero medio di job in servizio non ha portato a un aumento significativo dei tempi di attesa, confermando che il sistema è ben bilanciato e capace di gestire il carico di lavoro a regime. Anche il sistema di scoring automatico mantiene un utilizzo molto basso e tempi di coda praticamente inesistenti. Per quanto riguarda il comitato di credito, l'alta utilizzazione osservata è dovuta alla lunga durata del servizio, in linea con quanto visto nella simulazione ad orizzonte finito. Questo riflette la capacità del sistema $M/M/\infty$ di gestire un elevato volume di richieste senza creare colli di bottiglia. Infine, il reparto liquidazioni continua a presentare tempi di coda e utilizzazioni leggermente superiori rispetto agli altri centri, ma questi restano comunque ben gestiti, grazie a un tasso di servizio adeguato alla domanda.

8.3.1 Autocorrelazione

Di seguito è possibile vedere i valori per l'autocorrelazione delle statistiche forniti dal simulatore che si appoggia alla classe di libreria *Acs.java*:

Reparto Istruttorie

```
*****
AUTOCORRELATION VALUES FOR REPARTO_ISTRUTTORIE_MAAC [B:4096|K:128]
*****
E[Ts]: 0.0527
E[Tq]: -0.0358
E[s]: 0.0521
E[Ns]: -0.1048
E[Nq]: -0.0364
ρ: -0.1059
λ: 0.0403
*****
```

Sys Scoring Automatico

```
*****
AUTOCORRELATION VALUES FOR SYS_SCORING_AUTOMATICO_SANTANDER [B:4096|K:128]
*****
E[Ts]: -0.0725
E[Tq]: 0.0342
E[s]: -0.0689
E[Ns]: -0.0733
E[Nq]: 0.0229
ρ: -0.0651
λ: 0.0406
*****
```

Comitato Credito

```
*****
AUTOCORRELATION VALUES FOR COMITATO_CREDITO_SANTANDER [B:4096|K:128]
*****
E[Ts]: 0.0186
E[Tq]: -
E[s]: 0.0207
E[Ns]: 0.0799
E[Nq]: -
ρ: 0.0610
λ: 0.0298
*****
```

Reparto Liquidazioni

```
*****
AUTOCORRELATION VALUES FOR REPARTO_LIQUIDAZIONI_MAAC [B:4096|K:128]
*****
E[Ts]: -0.0097
E[Tq]: 0.0105
E[s]: 0.0179
E[Ns]: 0.0918
E[Nq]: 0.0279
ρ: 0.1228
λ: 0.0401
*****
```

Tabella riassuntiva dei valori di autocorrelazione

Centro	$E[Ns]$	$E[Ts]$	$E[Nq]$	$E[Tq]$	$E[s]$	ρ	λ
RIM	-0.1048	0.0527	-0.0358	-0.0358	0.0521	-0.1059	0.0403
SSAS	-0.0733	-0.0725	0.0342	0.0342	-0.0689	-0.0651	0.0406
CCS	0.0799	0.0186	-	-	0.0207	0.0610	0.0298
RLM	0.0918	-0.0097	0.0105	0.0105	0.0179	0.1228	0.0401

Tabella 8: Tabella riassuntiva con i valori autocorrelazione

9 Modello migliorativo

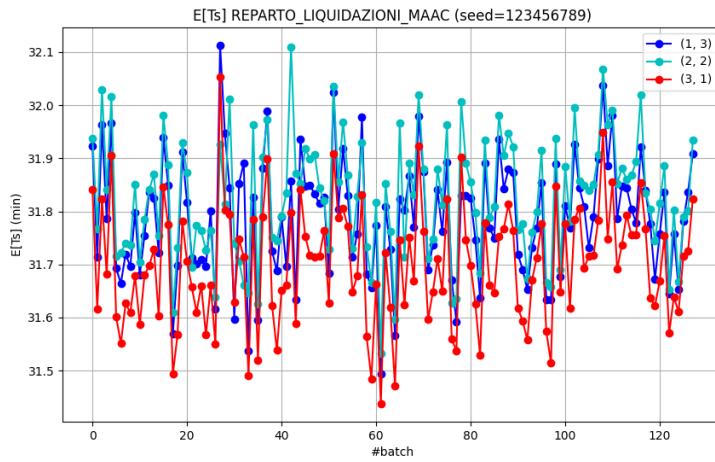
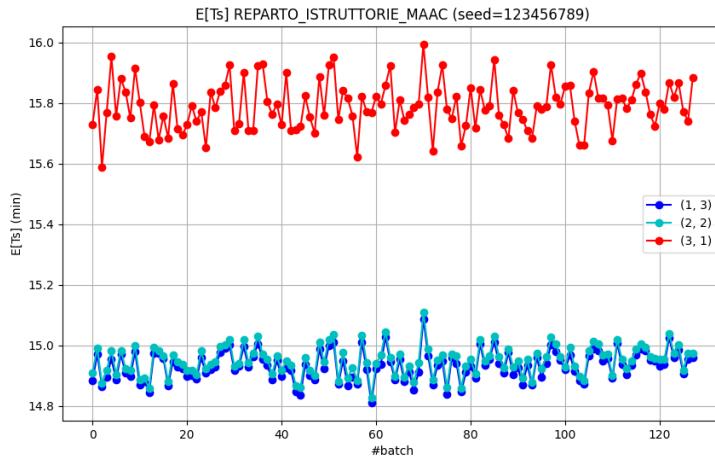
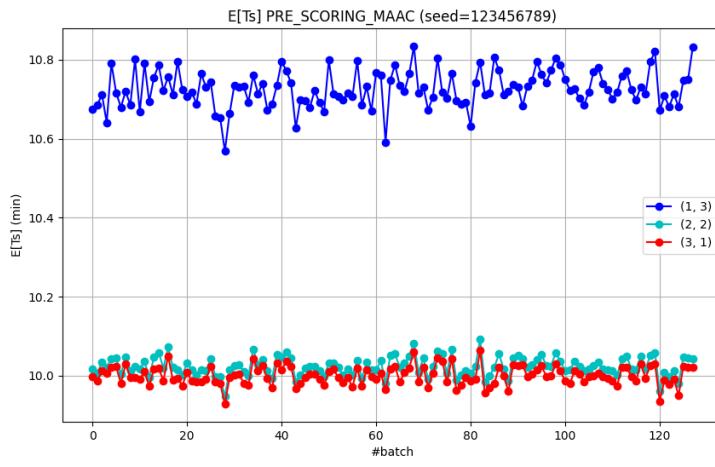
Considerando le problematiche menzionate in 1, il modello migliorativo proposto prevede l'introduzione di un controllo preliminare, denominato **pre-scoring**, che permetta di identificare e scartare a priori le richieste che presentano le suddette criticità, aumentando così l'efficienza del processo e riducendo il numero di istruttorie avviate inutilmente.

Per implementare il pre-scoring, si propone di suddividere i dipendenti del reparto istruttorie in modo che alcuni si occupino del pre-scoring e gli altri continuino con la procedura di istruttoria tradizionale, portando avanti i due processi senza aumentare le risorse umane impiegate. Questa modifica consentirà di ridurre il tempo totale dedicato all'istruttoria, portando il tempo medio di servizio a 15 minuti (tempo necessario per l'inputazione dei dati raccolti sul sistema SCB), rispetto ai 25 minuti attuali, grazie alla parziale esecuzione dei compiti come l'interazione col cliente e la raccolta di documenti già fatti in fase di pre-scoring, che richiederà solo 10 minuti ($\sim 5\text{min}$ per raccolta documenti e $\sim 5\text{min}$ per il colloquio col cliente dove raccogliere le altre informazioni).

Per decidere come distribuire in modo ottimale i dipendenti sono state eseguite simulazioni con configurazioni differenti. In particolare le configurazioni considerate sono:

- 1 servente in pre-scoring e 3 in reparto istruttorie
- 2 serventi in pre-scoring e 2 in reparto istruttorie
- 3 serventi in pre-scoring e 1 in reparto istruttorie

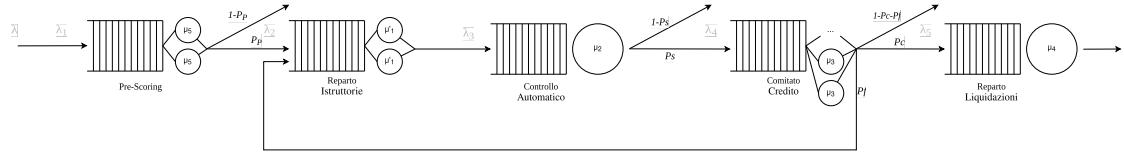
Le tre configurazioni sono state valutate sulla base del tempo di risposta all'interno dei vari centri che riguardano gli uffici della *MF*, scegliendo quella con tempo minore. I risultati ottenuti sono i seguenti.



Dall'osservazione dei grafici emerge che la configurazione migliore è quella con 2 serventi in *Prescoring* e 2 in *Reparto istruttorie*. Infatti sia le configurazioni (1,3) che (3,1) hanno tempi di risposta piuttosto alti nei centri rispettivamente *Prescoring* e *Reparto istruttorie*.

10 Modello concettuale

Con l'aggiunta del nuovo centro di *PreScoring*, il sistema si evolve nel seguente modo:



10.1 Eventi

Oltre agli eventi già considerati nel modello base vi sono:

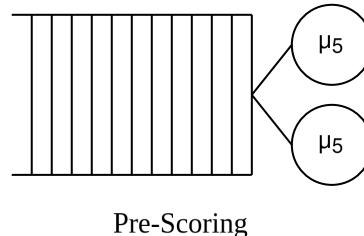
- ARRIVAL_PRE_SCORING è l'arrivo di una richiesta presso l'azienda.
- COMPLETION_PRE_SCORING coincide con l'espletamento della procedura di pre-scoring.

11 Modello delle specifiche

11.1 Modellazione dei centri

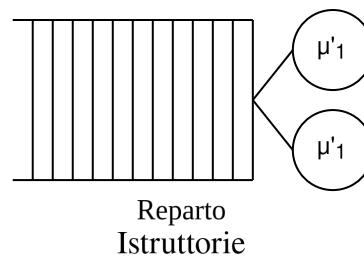
La modellazione dei centri rimane la stessa ad eccezione per i seguenti centri:

- Sistema Prescoring



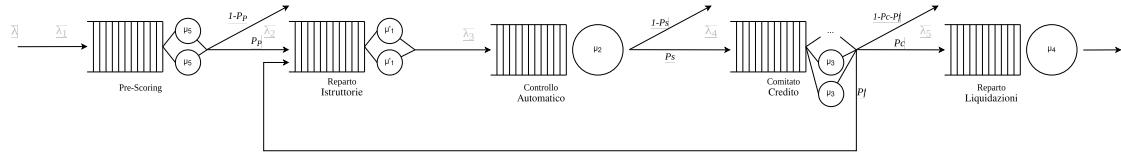
Questo centro è stato modellato come un multiserver con 2 serventi (che rappresentano i 2 dipendenti che si occupano del prescoring). Il servizio segue una distribuzione log normale troncata a 25min con $E[S] = 10\text{min}$ e $\sigma^2 = 2\text{min}$.

- Reparto istruttorie Maac Finance



Questo centro è stato modellato come un multiserver con 2 serventi (che rappresentano i 2 dipendenti che si occupano dell'istruttoria dei contratti all'interno dell'azienda Maac Finance). Il servizio segue una distribuzione log normale troncata a 25min con $E[S] = 15\text{min}$ e $\sigma^2 = 5\text{min}$.

11.2 Equazioni di traffico



Le equazioni di traffico sono:

$$\begin{cases} \lambda_1 = \lambda \\ \lambda_2 = \lambda_1 \cdot p_P + \lambda_4 \cdot p_F \\ \lambda_3 = \lambda_2 \\ \lambda_4 = \lambda_3 \cdot p_S \\ \lambda_5 = \lambda_4 \cdot p_C \end{cases}$$

La tabella di routing è:

	Esterno	Pre-Scoring	Reparto istruttorie	Sistema automatico	Comitato credito	Reparto liquidazione
Esterno	0	0	1	0	0	0
Pre-Scoring	$1 - p_P$	p_P	0	0	0	0
Reparto istruttorie	0	0	1	0	0	0
Sistema automatico	$1 - p_S$	0	0	p_S	0	0
Comitato credito	$1 - p_C - p_F$	p_F	0	0	0	p_C
Reparto liquidazione	1	0	0	0	0	0

Tabella 9: Tabella di routing

12 Modello computazionale

Il modello computazione non subisce variazioni nella sua struttura se non nell'aggiunta della classe relativa al centro di pre-scoring.

Inoltre, viene aggiunta un'ulteriore probabilità di routing che è la probabilità che una richiesta venga accettata dal sistema di *Pre-Scoring*. La classe **Applicant** viene quindi modificata nel seguente modo:

```

1 public class Applicant {
2     public boolean isAcceptedByPreScoring() { /* ... */ }
3     public boolean isAcceptedBySysScoring() { /* ... */ }
4     public void setHasCorrispondenzaInBancaDati(Rngs rngs) { /*
5         ... */ }
5 }
```

`isAcceptedByPreScoring()` determina se il centro *Pre-Scoring* debba accettare o meno la richiesta. In particolare, verifica che gli attributi `haAnzianitaDiLavoro`, `isRapportoRataRedditoOk`, `haContrattoIndeterminato`, `nonHaRichiesteORifiutiRecenti` siano tutti `true`.

13 Verifica

La verifica del modello migliorato viene condotta come nel modello di base: di seguito per ogni centro i calcoli analitici e gli screenshot del simulatore.

13.1 Sistema pre-scoring M/M/2

$$\begin{aligned}\lambda_1 &= 0.0125 \text{ job/min} \\ E(S_i) &= 10 \text{ min} \quad i = 1, 2 \\ E(S) &= \frac{E(S_i)}{2} = 5 \text{ min} \\ \mu_i &= \frac{1}{E(S_i)} = 0.1 \text{ job/min} \\ \rho &= \frac{\lambda_1}{2 \cdot \mu_i} = 0.0625 \\ N &= 2 \\ p_0 &= \left[\sum_{i=0}^{N-1} \frac{(N \cdot \rho)^i}{i!} + \frac{(N \cdot \rho)^N}{N! \cdot (1 - \rho)} \right]^{-1} = 0.8823529 \\ P_q &= \frac{(N \cdot \rho)^N \cdot p_0}{N! \cdot (1 - \rho)} = 0.0073529 \\ E(T_q) &= \frac{P_q \cdot E(S)}{1 - \rho} = 0.0392157 \text{ min} \\ E(N_q) &= E(T_q) \cdot \lambda_1 = 4.901961 \cdot 10^{-4} \\ E(T_s) &= E(T_q) + E(S_i) = 10.039216 \text{ min} \\ E(N_s) &= E(T_s) \cdot \lambda_1 = 0.1254902\end{aligned}$$

```
*****
FINAL RESULTS FOR PRE_SCORING_MAAC with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 10.050498999934092, diff 0.011283313659582461 is within the interval ±0.026054103662482297
E[Tq]: mean 0.03922150100227384, diff 5.814727764036354E-6 is within the interval ±0.00232655885603197
E[Ns]: mean 0.12539683451312547, diff 9.336156530589945E-5 is within the interval ±5.18642188905768E-4
E[Nq]: mean 4.894374959610201E-4, diff 7.585824703524495E-7 is within the interval ±2.905384313490202E-5
ρ: mean 0.062457352071912914, diff 4.264792808708617E-5 is within the interval ±2.5687737288362754E-4
*****
```

13.2 Reparto istruttorie M/M/2

$$\lambda_2 = \frac{\lambda_1 \cdot p_P}{1 - p_C \cdot p_F} = 0.00670498 \text{ job/min}$$

$$E(S_i) = 15 \text{ min}$$

$$E(S) = \frac{E(S_i)}{2} = 7.5 \text{ min}$$

$$\mu_i = \frac{1}{E(S_i)} = 0.066666667 \text{ job/min}$$

$$\rho = \frac{\lambda_2}{2 \cdot \mu_i} = 0.0502866$$

$$N = 2$$

$$p_0 = \left[\sum_{i=0}^{N-1} \frac{(N \cdot \rho)^i}{i!} + \frac{(N \cdot \rho)^N}{N! \cdot (1 - \rho)} \right]^{-1} = 0.9042408$$

$$P_q = \frac{(N \cdot \rho)^N \cdot p_0}{N! \cdot (1 - \rho)} = 0.0048155$$

$$E(T_q) = \frac{P_q \cdot E(S)}{1 - \rho} = 0.0380284 \text{ min}$$

$$E(N_q) = E(T_q) \cdot \lambda_2 = 2.549797 \cdot 10^{-4}$$

$$E(T_s) = E(T_q) + E(S_i) = 15.0380273 \text{ min}$$

$$E(N_s) = E(T_s) \cdot \lambda_2 = 0.1008282$$

```
*****
FINAL RESULTS FOR REPARTO_ISTRUTTORIE_MAAC with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 15.037469706618879, diff 5.575878417705127E-4 is within the interval ±0.03813121278595394
E[Tq]: mean 0.03607996956225179, diff 0.0019473248983969405 is within the interval ±0.002287035234666686
E[Ns]: mean 0.10079984811092252, diff 2.8321853514423623E-5 is within the interval ±3.698534661448619E-4
E[Nq]: mean 2.419609297694004E-4, diff 1.3007520153438977E-5 is within the interval ±1.547608920132633E-5
p: mean 0.05028120865052714, diff 5.392106729909529E-6 is within the interval ±1.8238991482868303E-4
*****
```

13.3 Sistema automatico M/M/1

$$\lambda_3 = 0.0067048 \text{ job/min}$$

$$E(S) = 1 \text{ min}$$

$$\mu = \frac{1}{E(S)} = 1 \text{ job/min}$$

$$\rho = \lambda_3 \cdot E[S] = 0.0067048$$

$$E(T_q) = \frac{\rho \cdot E(S)}{1 - \rho} = 0.0067501 \text{ min}$$

$$E(N_q) = E(T_q) \cdot \lambda_3 = 4.5258872 \cdot 10^{-5}$$

$$E(T_s) = E(T_q) + E(S) = 1.0067501 \text{ min}$$

$$E(N_s) = E(T_s) \cdot \lambda_3 = 0.0067501$$

```
*****
FINAL RESULTS FOR SYS_SCORING_AUTOMATICO_SANTANDER with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 1.0044307744100678, diff 0.002319364563381665 is within the interval ±0.0030638433301459767
E[Tq]: mean 0.006458384949524998, diff 2.9175402392445614E-4 is within the interval ±3.276901937227388E-4
E[Ns]: mean 0.0067332543735783915, diff 1.688459987106254E-5 is within the interval ±2.9615471671557833E-5
E[Nq]: mean 4.329008059763931E-5, diff 1.9687918842078436E-6 is within the interval ±2.2013289414747893E-6
p: mean 0.006689964292980752, diff 1.49158079868544E-5 is within the interval ±2.9290803133996952E-5
*****
```

13.4 Comitato credito M/M/∞

$$\begin{aligned}\lambda_4 &= \lambda_3 \cdot p_S = 0.005498 \text{ job/min} \\ E(S) &= 480 \text{ min} \\ \mu &= \frac{1}{E(S)} = 2.08333 \cdot 10^{-3} \text{ job/min} \\ \rho &= 1 - e^{-\frac{\lambda_4}{\mu}} = 0.9285702 \\ E(T_q) &= 0 \text{ min} \\ E(N_q) &= E(T_q) \cdot \lambda_4 = 0 \\ E(T_s) &= E(T_q) + E(S) = 480 \text{ min} \\ E(N_s) &= E(T_s) \cdot \lambda_4 = 2.6390408\end{aligned}$$

```
*****
FINAL RESULTS FOR COMITATO_CREDITO_SANTANDER with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 479.9978919978389, diff 0.0021080021610941913 is within the interval ±1.3399123772510797
E[Tq]: mean 0.0, diff 0.0 is within the interval ±0.0
E[Ns]: mean 2.640215717311574, diff 0.001174909570724303 is within the interval ±0.010965281835334096
E[Nq]: mean 0.0, diff 0.0 is within the interval ±0.0
p: mean 0.9295202270650703, diff 9.499786372771934E-4 is within the interval ±0.0012335059250880302
*****
```

13.5 Reparto liquidazioni M/M/1

$$\begin{aligned}\lambda_5 &= \lambda_4 \cdot p_C = 0.0035737 \text{ job/min} \\ E(S) &= 30 \text{ min} \\ \mu &= \frac{1}{E(S)} = 0.0333333 \text{ job/min} \\ \rho &= \lambda_5 \cdot E[S] = 0.107211 \\ E(T_q) &= \frac{\rho \cdot E(S)}{1 - \rho} = 3.6025657 \text{ min} \\ E(N_q) &= E(T_q) \cdot \lambda_5 = 0.0128745 \\ E(T_s) &= E(T_q) + E(S) = 33.6025657 \text{ min} \\ E(N_s) &= E(T_s) \cdot \lambda_5 = 0.1200855\end{aligned}$$

```
*****
FINAL RESULTS FOR REPARTO_LIQUIDAZIONI_MAAC with 95% confidence [B:4096|K:128]
*****
E[Ts]: mean 33.587348945749284, diff 0.015216837039766062 is within the interval ±0.11214882761909971
E[Tq]: mean 3.617256543543178, diff 0.014690760754125787 is within the interval ±0.05157481859839715
E[Ns]: mean 0.12037977433413431, diff 2.942482411659314E-4 is within the interval ±5.089744652277683E-4
E[Nq]: mean 0.01296573291393646, diff 9.12396354400876E-5 is within the interval ±1.9000901338782036E-4
ρ: mean 0.10741404142019785, diff 2.0300860572582125E-4 is within the interval ±3.830606941036326E-4
*****
```

13.6 Tabella riassuntiva: medie analitiche, empiriche e intervalli di confidenza

Indice	Valore Analitico	Valore Empirico	CI
$E[Ns]_{PSM}$	0.125490	0.125397	$\pm 5.186422 \cdot 10^{-4}$
$E[Ts]_{PSM}$	10.039216	10.050499	± 0.026054
$E[Nq]_{PSM}$	$4.901961 \cdot 10^{-4}$	$4.894375 \cdot 10^{-4}$	$\pm 2.905834 \cdot 10^{-5}$
$E[Tq]_{PSM}$	0.039216	0.039222	± 0.002327
ρ_{PSM}	0.062500	0.062457	$\pm 2.568774 \cdot 10^{-4}$
$E[Ns]_{RIM}$	0.100828	0.1007998	$\pm 3.968535 \cdot 10^{-4}$
$E[Ts]_{RIM}$	15.038027	15.0374697	± 0.038131
$E[Nq]_{RIM}$	$2.549797 \cdot 10^{-4}$	$2.419609 \cdot 10^{-4}$	$\pm 1.5476089 \cdot 10^{-5}$
$E[Tq]_{RIM}$	0.038028	0.03607997	± 0.002287
ρ_{RIM}	0.050287	0.050281	$\pm 1.823899 \cdot 10^{-4}$
$E[Ns]_{SSAS}$	0.006750	0.006733	$\pm 2.961547 \cdot 10^{-5}$
$E[Ts]_{SSAS}$	1.006750	1.004431	± 0.0030638
$E[Nq]_{SSAS}$	$4.5258872 \cdot 10^{-5}$	$4.329008 \cdot 10^{-5}$	$\pm 2.201329 \cdot 10^{-5}$
$E[Tq]_{SSAS}$	0.006750	0.0064584	$\pm 3.276902 \cdot 10^{-4}$
ρ_{SSAS}	0.006705	0.00668996	$\pm 2.9290803 \cdot 10^{-5}$
$E[Ns]_{CCS}$	2.639041	2.640216	± 0.0109653
$E[Ts]_{CCS}$	480.000000	479.997892	± 1.3399124
$E[Nq]_{CCS}$	0.0	0.0	-
$E[Tq]_{CCS}$	0.0	0.0	-
ρ_{CCS}	0.928570	0.929520	± 0.001233
$E[Ns]_{RLM}$	0.120086	0.1203798	$\pm 5.089745 \cdot 10^{-4}$
$E[Ts]_{RLM}$	33.602566	33.587349	± 0.112159
$E[Nq]_{RLM}$	0.012875	0.012966	$\pm 1.900901 \cdot 10^{-4}$
$E[Tq]_{RLM}$	3.602566	3.617256	± 0.0515748
ρ_{RLM}	0.107211	0.107414	$\pm 3.830607 \cdot 10^{-4}$

14 Validazione

Per quanto riguarda la validazione, possiamo fare considerazioni analoghe a quelle fatte per il modello base: per le stesse ragioni, tutti i centri presentano una bassa utilizzazione; anzi, dal momento che le richieste vengono filtrate a priori, in *Reparto istruttorie* e *Scoring automatico* l'utilizzazione risulta essere persino più bassa rispetto a quella del modello base. Inoltre, i numeri di contratti liquidati rimangono uguali a quelli del modello base (come ci aspettavamo) e coerenti con i dati aziendali.

15 Design degli esperimenti

15.1 Simulazione orizzonte finito

In questa sezione, presentiamo i risultati ottenuti dalla simulazione ad orizzonte finito del modello migliorato. L'obiettivo di questa simulazione era verificare l'efficacia delle modifiche introdotte, in particolare l'implementazione del pre-scoring e la

riorganizzazione del flusso di lavoro. Le statistiche riportate nella tabella seguente riassumono le principali metriche di performance per ciascun centro.

Eseguendo la simulazione si osservano i seguenti risultati:

Centro	$E[N_s]$	$E[T_s]$	$E[N_q]$	$E[T_q]$	ρ
PSM	$0.1311 \pm 9.0932 \cdot 10^{-5}$	$10.0338 \pm 1.3162 \cdot 10^{-2}$	$0.0003 \pm 1.2558 \cdot 10^{-7}$	$0.0219 \pm 6.3241 \cdot 10^{-4}$	$0.0657 \pm 2.2636 \cdot 10^{-5}$
RIM	$0.1057 \pm 1.2832 \cdot 10^{-4}$	$14.9758 \pm 8.1758 \cdot 10^{-2}$	$0.0002 \pm 1.1002 \cdot 10^{-7}$	$0.0225 \pm 1.3192 \cdot 10^{-3}$	$0.0531 \pm 3.1994 \cdot 10^{-5}$
SSAS	$0.0069 \pm 1.0137 \cdot 10^{-6}$	$0.9813 \pm 1.3013 \cdot 10^{-2}$	$0.0001 \pm 7.0699 \cdot 10^{-9}$	$0.0097 \pm 2.4378 \cdot 10^{-4}$	$0.0068 \pm 9.9669 \cdot 10^{-7}$
CCS	$2.3437 \pm 6.8011 \cdot 10^{-2}$	479.5846 ± 2.2005	-	-	$0.8913 \pm 1.2057 \cdot 10^{-3}$
RLM	$0.0998 \pm 2.0917 \cdot 10^{-4}$	31.5323 ± 1.0084	$0.0053 \pm 1.0790 \cdot 10^{-5}$	$1.4389 \pm 6.6736 \cdot 10^{-1}$	$0.0944 \pm 1.5461 \cdot 10^{-4}$

Tabella 10: Statistiche riassuntive simulazione a orizzonte finito del modello migliorato

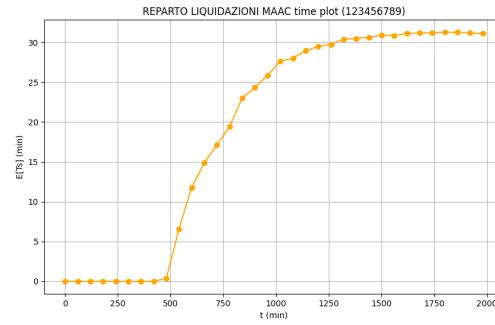
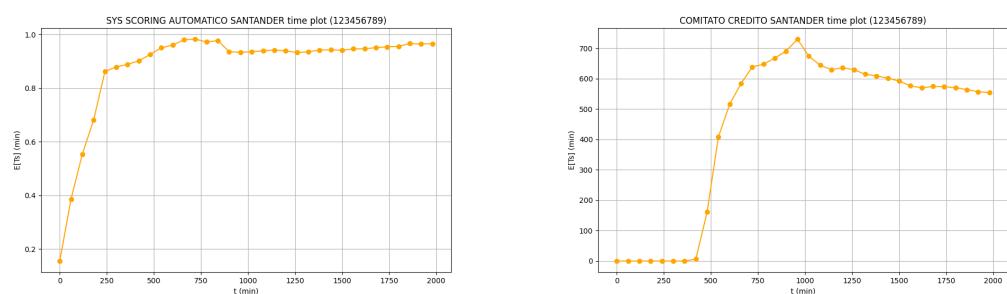
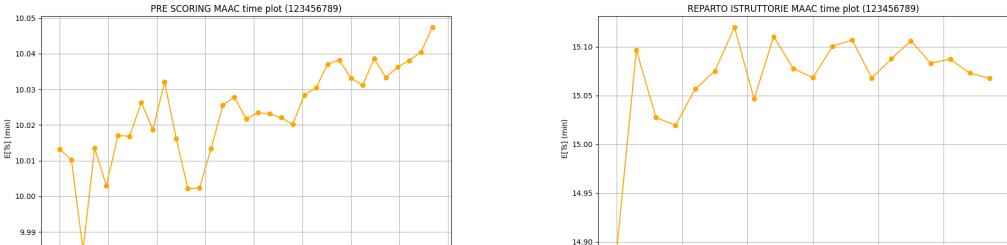


Tabella 11: Tempi di risposta dei vari centri

I risultati ottenuti nella simulazione ad orizzonte finito del modello migliorato presentano alcune differenze chiave rispetto al modello base.

Nel centro dedicato al pre-scoring, introdotto nel modello migliorato, il numero medio di job in servizio e in coda è anch'esso molto basso, riflettendo una dimensione adatta del centro rispetto al flusso di lavoro.

Nel reparto istruttorie, si nota un miglioramento in termini di utilizzazione e tempi

di coda, segno che il carico di lavoro è stato ridotto nonostante la diminuzione del personale. Questo dimostra l'efficacia dell'introduzione del pre-scoring nel ridurre il numero di richieste elaborate nel reparto istruttorie.

Per quanto riguarda il reparto liquidazioni, il modello migliorato mantiene un utilizzo leggermente superiore rispetto ad altri centri, ma continua a gestire il flusso di lavoro in modo efficace, con tempi di attesa contenuti.

In conclusione, possiamo dire che l'introduzione del centro di pre-scoring ha appor-tato benefici al sistema in termini di tempi di attesa e utilizzazione.

I volumi osservati sui vari centri e il busy time totale cambia come segue:

Centro	Job Serviti (Avg)	Busy Time
Pre-Scoring	30.446667	304.75469
Reparto Istruttorie	16.386667	245.38663
Scoring Automatico	16.386667	15.94205
Comitato Credito	13.593333	6517.59113
Reparto Liquidazioni	8.560000	257.51494

Tabella 12: Misure di interesse per centro sul periodo di osservazione della simula-zione del modello migliorato

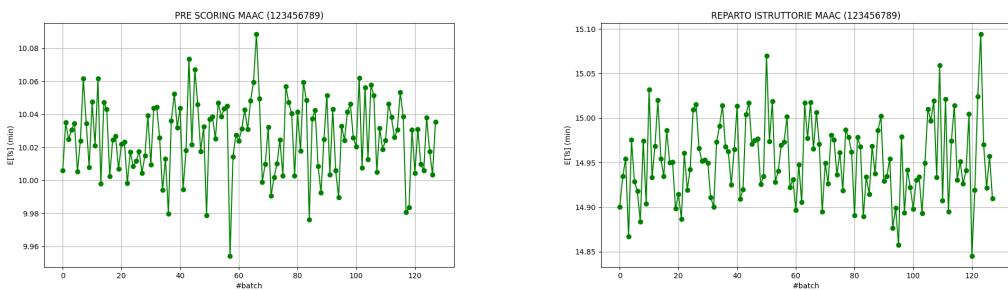
15.2 Simulazione orizzonte infinito

Anche in questo caso per trovare i valori di b e k ottimali è stata adottata la tecnica vista nel modello base. Si sono ottenuti gli stessi valori precedenti ossia $b = 4096$ e $k = 128$.

Eseguendo la simulazione si osservano i seguenti risultati:

Centro	$E[N_s]$	$E[T_s]$	$E[N_q]$	$E[T_q]$	ρ
PSM	$0.1254 \pm 1.5319 \cdot 10^{-6}$	$10.0505 \pm 3.8659 \cdot 10^{-3}$	$0.0005 \pm 4.8073 \cdot 10^{-9}$	$0.0392 \pm 3.0826 \cdot 10^{-5}$	$0.0625 \pm 3.7579 \cdot 10^{-7}$
RIM	$0.1008 \pm 7.7903 \cdot 10^{-7}$	$15.0375 \pm 8.2805 \cdot 10^{-3}$	$0.0002 \pm 1.3640 \cdot 10^{-9}$	$0.0361 \pm 2.9788 \cdot 10^{-5}$	$0.0503 \pm 1.8945 \cdot 10^{-7}$
SSAS	$0.0067 \pm 4.9950 \cdot 10^{-9}$	$1.0044 \pm 5.3460 \cdot 10^{-5}$	$0.0000 \pm 2.7597 \cdot 10^{-11}$	$0.0065 \pm 6.1154 \cdot 10^{-7}$	$0.0067 \pm 4.8861 \cdot 10^{-9}$
CCS	$2.6402 \pm 6.8476 \cdot 10^{-4}$	479.9979 ± 10.2247	-	-	$0.9295 \pm 8.6652 \cdot 10^{-6}$
RLM	$0.1204 \pm 1.4753 \cdot 10^{-6}$	$33.5873 \pm 7.1628 \cdot 10^{-2}$	$0.0130 \pm 2.0561 \cdot 10^{-7}$	$3.6173 \pm 1.5149 \cdot 10^{-2}$	$0.1074 \pm 8.3566 \cdot 10^{-7}$

Tabella 13: Statistiche riassuntive simulazione a orizzonte infinito del modello mi-gliorato



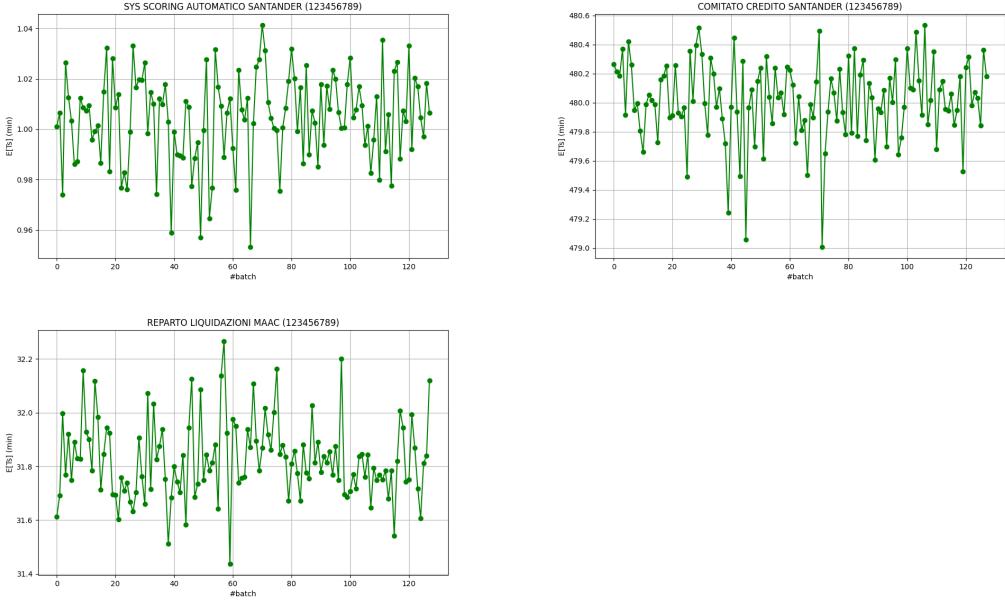


Tabella 14: Tempi di risposta dei vari centri

Anche nella simulazione ad orizzonte infinito del modello improved, i risultati ottenuti sono in linea con le aspettative e confermano la stabilità del sistema. Nel centro dedicato al pre-scoring (PSM), si osserva una leggera diminuzione nel numero medio di job in servizio e in coda rispetto alla simulazione ad orizzonte finito, segno che il sistema riesce a mantenere una gestione ottimale del flusso di lavoro anche a regime stazionario.

Nel reparto istruttorie (RIM), i risultati mostrano una stabilità nelle prestazioni con un utilizzo leggermente inferiore rispetto all'orizzonte finito, indicando che la distribuzione del carico continua ad essere ben bilanciata.

Il sistema di scoring automatico (SSAS) continua a mantenere un utilizzo molto basso e tempi di attesa trascurabili, riflettendo l'efficienza costante del centro. Per quanto riguarda il comitato di credito (CCS), l'utilizzazione rimane elevata, ma sempre per lo stesso motivo.

Infine, il reparto liquidazioni (RLM) mostra una lieve crescita del numero medio di job in coda rispetto all'orizzonte finito, anche se in entrambi i casi si tratta di valori prossimi allo 0.

15.2.1 Autocorrelazione

Possiamo quindi osservare ancora una volta i valori di acs forniti dal simulatore una volta avviata la run ad orizzonte infinito con i suddetti valori di b e k:

Pre-Scoring

```
*****
AUTOCORRELATION VALUES FOR PRE_SCORING_MAAC [B:4096|K:128]
*****
E[Ts]: 0.0397
E[Tq]: 0.0322
E[s]: 0.0316
E[Ns]: 0.0293
E[Nq]: 0.0243
ρ: 0.0308
λ: 0.0626
*****
```

Reparto Istruttorie

```
*****
AUTOCORRELATION VALUES FOR REPARTO_ISTRUTTORIE_MAAC [B:4096|K:128]
*****
E[Ts]: 0.0454
E[Tq]: 0.0847
E[s]: 0.0434
E[Ns]: 0.1055
E[Nq]: 0.0827
ρ: 0.1069
λ: -0.0776
*****
```

Sys Scoring Automatico

```
*****
AUTOCORRELATION VALUES FOR SYS_SCORING_AUTOMATICO_SANTANDER [B:4096|K:128]
*****
E[Ts]: 0.0268
E[Tq]: -0.1841
E[s]: 0.0465
E[Ns]: -0.0561
E[Nq]: -0.1904
ρ: -0.0417
λ: -0.0781
*****
```

Comitato Credito

```
*****
AUTOCORRELATION VALUES FOR COMITATO_CREDITO_SANTANDER [B:4096|K:128]
*****
E[Ts]: 0.0182
E[Tq]: -
E[s]: 0.0172
E[Ns]: 0.0085
E[Nq]: -
ρ: -0.0105
λ: -0.0307
*****
```

Reparto Liquidazioni

```
*****
AUTOCORRELATION VALUES FOR REPARTO_LIQUIDAZIONI_MAAC [B:4096|K:128]
*****
E[Ts]: -0.0204
E[Tq]: -0.0150
E[s]: -0.0466
E[Ns]: 0.0418
E[Nq]: -0.0018
ρ: 0.0421
λ: 0.0413
*****
```

Tabella riassuntiva dei valori di autocorrelazione

Centro	$E[Ns]$	$E[Ts]$	$E[Nq]$	$E[Tq]$	$E[s]$	ρ	λ
PSM	0.0293	0.0397	0.0322	0.0322	0.0316	0.0308	0.0626
RIM	0.1055	0.0454	0.0847	0.0847	0.0434	0.1069	-0.0776
SSAS	-0.0561	0.0268	-0.1841	-0.1841	0.0465	-0.0417	-0.0781
CCS	0.0085	0.0182	-	-	0.0172	-0.0105	-0.0307
RLM	0.0418	-0.0204	-0.0150	-0.0150	-0.0466	0.0421	0.0413

15.3 Introduzione firma digitale per le liquidazioni

Nel modello base, il *Reparto Liquidazione* gestisce la finalizzazione del prestito, che include la firma del contratto da parte del cliente e l'esecuzione del bonifico della somma concordata sull'IBAN del cliente. Questa procedura, nella sua forma tradizionale, prevede che i documenti vengano inviati al cliente via e-mail, firmati fisicamente, scannerizzati e poi rimandati indietro all'azienda (Per semplicità non si considerano casistiche in cui il cliente viene a firmare in presenza con un tempo arbitrario non quantificabile). Questo processo, sebbene funzionale, richiede tempo e diverse interazioni tra il cliente e l'azienda, rallentando così l'intera procedura di liquidazione.

Recentemente, la mandante *SCB* ha autorizzato la transizione alla firma digitale basata su OTP (*One-Time Password*), consentendo di eliminare i passaggi manuali e accelerando notevolmente il processo. La firma digitale, che viene effettuata in tempo reale tramite l'invio di un OTP sul cellulare del cliente, consente di ridurre i tempi necessari per la finalizzazione e liquidazione del prestito, migliorando l'esperienza del cliente e aumentando l'efficienza operativa di *MF*. Attualmente l'uso della firma digitale non è applicato in modo sistematico all'interno di *MF*. Tramite simulazione vogliamo analizzare i benefici della transizione totale a questa modalità di pagamento.

A questo scopo il tempo di servizio viene modellato con la stessa distribuzione log normale ma stavolta: troncata a 20min, con $E[S] = 10\text{min}$ e $\sigma^2 = 5\text{min}$ (Confrontandosi con i dipendenti dell'azienda si è capito che il nuovo tipo procedura avrebbe tempistiche pari circa ad un terzo di quelle della firma tradizionale).

Centro	Job Serviti (Avg)	Busy Time
Pre-Scoring	30.446667	304.75469
Reparto Istruttorie	16.386667	245.38663
Scoring Automatico	16.386667	15.94205
Comitato Credito	13.593333	6517.59112
Reparto Liquidazioni	8.560000	85.90429

Tabella 15: Misure di interesse per centro sul periodo di osservazione della simulazione del modello migliorato con introduzione di firma digitale

Confronto tra Residence Time nel Sistema Improved con e senza Firma Digitale OTP

Osserviamo graficamente come cambia il *residence time* di un job nel sistema improved considerando l'introduzione del sistema di firma digitale OTP nelle liquidazioni rispetto al sistema di firma tradizionale.

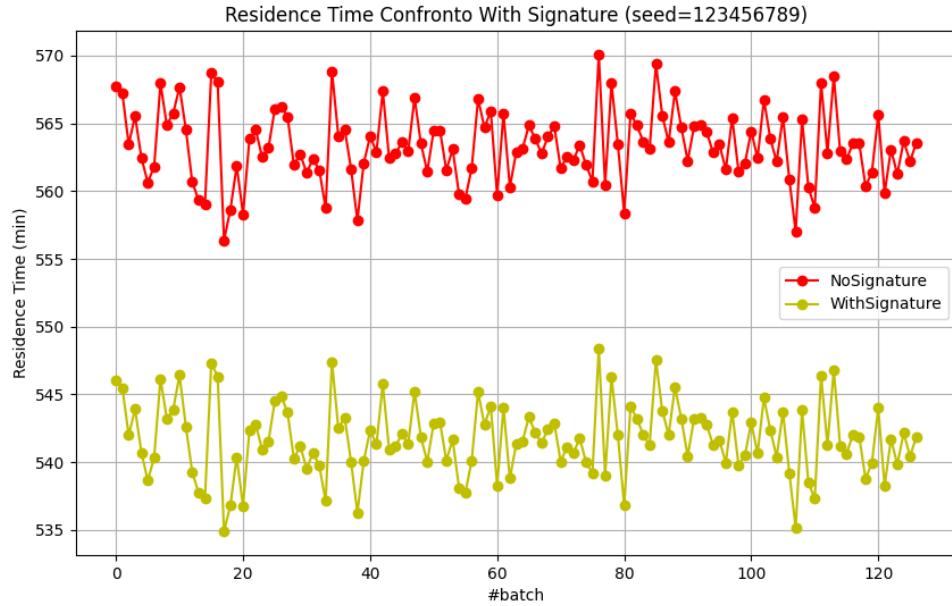


Figura 4: Residence Time nel Sistema Improved con e senza Firma Digitale OTP

L'adozione della firma digitale riduce significativamente il tempo impiegato nell'ultimo centro, portando a un residence time complessivo inferiore per il sistema improved e assicurando un servizio ancora più celere per il cliente medio (nonché amplificando i vantaggi di *MF* stessa).

16 Conclusioni

L'introduzione del pre-scoring ha indubbiamente portato a significativi miglioramenti rispetto al sistema analizzato all'inizio della trattazione, pienamente realizzando entrambi gli obiettivi prefissati.

Strategicamente, ha permesso di raddoppiare la percentuale di richieste accettate dal sistema automatico, passando dal 42% all'82%. Questo risultato si traduce in un significativo aumento della "qualità" dei contratti trasmessi da *MF* alla mandante *SCB*, rafforzando la posizione di *MF* come partner affidabile e di alto valore.

Operativamente, ha portato al dimezzamento del numero di contratti gestiti dal *Reparto Istruttorie* (e di conseguenza caricati sul *Sistema di Scoring Automatico*), passando da 31.34 a 16.39 job serviti, senza però compromettere il numero totale di contratti liquidati, garantendo quindi un guadagno comparabile con un effort notevolmente ridotto. Sebbene sia stato introdotto un tempo aggiuntivo per il pre-scoring (*277.629074 minuti*), la somma delle tempistiche dei centri relativi a *MF* è scesa da *1035.97176 minuti* a *807.65626 minuti*, con un risparmio netto di *228.3155 minuti*, pari a circa *3.81 ore*.

Estendendo questo risparmio su base annuale, considerando un anno lavorativo di 11 mesi attivi (si cumula approssimativamente un mese di inattività dovuto a ferie, chiusure e altre casistiche particolari) da 22 giorni ciascuno (per un totale di $11 \times 22 = 242$ giorni lavorativi), si otterrebbe un risparmio significativo. Il risparmio settimanale ottenuto è di circa *3.81 ore*, ovvero circa *45 minuti* al giorno. Per calcolare il risparmio annuale, moltiplichiamo questo valore per il numero di giorni lavorativi annui e poi dividiamo per 8 ore per ottenere il numero di giorni lavorativi annui risparmiati:

$$\text{Risparmio annuo} = \frac{0.76 \times 242 \text{ ore}}{8 \text{ ore/giorno}} = 23 \text{ giorni lavorativi}$$

Questo risparmio equivale a circa **23 giorni lavorativi**, che rappresenta **un mese di attività** dedicati a questo tipo di contratti. Questo risparmio di tempo non solo si traduce in una riduzione dei costi operativi, poiché lo stesso numero di contratti può essere elaborato con meno ore/uomo, aumentando così la redditività, ma offre anche vantaggi indiretti: il tempo liberato può essere infatti reinvestito in altre attività aziendali, come l'introduzione di nuovi prodotti o l'ampliamento dell'offerta esistente, incrementando ulteriormente la produttività globale di *MF*.

Inoltre, con l'introduzione della **firma digitale** nella fase di liquidazione, i miglioramenti operativi diventano ancora più marcati. I risultati della simulazione mostrano una riduzione drastica del *Busy Time* nel *Reparto Liquidazioni*, scendendo da 257.51 minuti a 85.90 minuti. Questo rappresenta un ulteriore risparmio di tempo significativo, che potrebbe amplificare ulteriormente i benefici già descritti, consentendo a *MF* di accelerare ulteriormente il processo di gestione dei prestiti.

Riferimenti bibliografici

- [1] Jerry Banks, John S Carson II, Brooks Automation, Barry L Nelson, and David M Nicol. Discrete-event system simulation fourth edition.
- [2] Averill M Law, W David Kelton, and W David Kelton. *Simulation modeling and analysis*, volume 3. McGraw-Hill New York, 2007.
- [3] Leemis. "Larry Leemis - Home Page." Wm.edu, 2017, www.math.wm.edu/~leemis/. Accessed 24 Aug. 2024. <https://www.math.wm.edu/~leemis/simtext.java>. [Accessed 24-08-2024].
- [4] P.S. Mahajan and R.G. Ingalls. Evaluation of methods used to detect warm-up period in steady state simulation. In *Proceedings of the 2004 Winter Simulation Conference, 2004.*, volume 1, page 671, 2004.

A Appendice: Autorizzazione Acquisizione Dati MF



Spett.li,
Edoardo Maneneti
Francesco Donnini
Martina Lupini
In qualità di Studenti
Università di Tor Vergata

Oggetto: Autorizzazione per l'Acquisizione di Dati Statistici Anonimizzati e Raccolta di Informazioni

Con la presente, la MAAC Finance S.r.l., con sede legale in Via di Casalotti 39 – 00166 ROMA, autorizza i signori Edoardo Manenti, Francesco Donnini e Martina Lupini, studenti universitari, a raccogliere e utilizzare dati statistici anonimizzati relativi ai processi aziendali per la realizzazione di un progetto di esame universitario.

La raccolta dei dati è finalizzata esclusivamente a scopi accademici, nell'ambito di un progetto di ricerca relativo al funzionamento del processo di trattamento dei contratti di prestito personale all'interno degli uffici della MAAC Finance S.r.l.

La presente autorizzazione è concessa nel rispetto delle normative vigenti in materia di privacy e protezione dei dati personali, in particolare del Regolamento (UE) 2016/679 (GDPR).

I dati raccolti saranno anonimizzati e utilizzati unicamente per l'elaborazione statistica richiesta dal progetto, garantendo che nessuna informazione sensibile o identificativa sia divulgata o utilizzata per scopi diversi da quelli indicati. Gli studenti hanno altresì ottenuto il permesso di raccogliere informazioni qualitative riguardanti il processo interno di trattamento dei contratti, sempre nel rispetto della riservatezza e delle disposizioni aziendali.

Per ulteriori informazioni o chiarimenti, restiamo a disposizione.

Distinti saluti,

PER PROCURA DI
SANTANDER CONSUMER BANK SPA
MAAC Finance S.r.l. ATTO NOTAIO L. MIGLIARDI 4/03/21
REP. 35584/18236

1/6/24

MAAC FINANCE SRL,

Sedi operative:

Roma Via Carrù 13c, 00166
Velletri - RM Viale Regina Margherita 7, 00049
Latina - Lt Vl Pieve 836, 04100
Civitavecchia - RM Via San V. Strambi 12/A, 00053

Tel. 800941422

Email: info@maacfinancesrl.it

Tel.: 06.95585429

Fax : 06.23328756

Tel.: 06.97654477

800941422

Numero Verde Gratuito
800 94 14 22

Sede legale:

Roma, Via di Casalotti 39 00166. Capitale sociale 10.000,00. Iscrizione registro imprese nr. 1238794 Cod Fiscale e Partita Iva 10540961009 Rea. Agente in attività finanziaria per Santanders Consumer Bank spa iscritto all'elenco Agenti in attività finanziaria tenuto dall'OAM al n. A8557