

Progetto PMCSN

Gestione dei mezzi in un'autorimessa AMA

Simone Festa
0320408

Michele Tosi
0327862

Università degli studi di Roma Tor Vergata

Abstract

Il presente studio vuole analizzare la gestione operativa di un'autorimessa AMA, composta da una parte di smaltimento dei rifiuti e una parte di riparazione dei mezzi circolanti. Lo studio è assecondato da **dati reali** del sistema, usati nella fase simulativa per analizzarne l'attuale comportamento. Si vuole inoltre fornire una strategia di miglioramenti che porteranno una gestione più efficiente dei mezzi, permettendo quindi più disponibilità di questi per la raccolta dei rifiuti (rispettando criteri reali), oltre che un maggior decoro per la città di Roma.

Introduzione

Il sistema preso in analisi è un centro di raccolta rifiuti dell'AMA. L'azienda dispone di una flotta di automezzi suddivisi per capienza. Dopo aver effettuato un turno di lavoro, i mezzi per la raccolta tornano al centro dove i rifiuti raccolti verranno smaltiti (inceneriti). Ogni mezzo, dopo aver completato il proprio servizio, entra nel centro. All'ingresso, se il mezzo non riporta problemi passa direttamente allo smaltimento. Se invece riporta qualche problema tecnico, il quale non ha permesso il completamento della raccolta, il mezzo viene prima fatto passare per l'accettazione, dove viene diagnosticato il problema, per poi essere reindirizzato in base a questo presso l'autofficina di competenza. Se, a seguito della diagnosi si rileva che il mezzo non è riparabile, o che il costo della riparazione superi l'effettivo valore del mezzo (es. combustione del mezzo) questo abbandona il sistema. Tipicamente, comunque, questa è una situazione che non si verifica spesso. Dopo la fase di smaltimento, se necessario i mezzi vengono sottoposti alla sanitizzazione oppure a rifornimento per poi uscire dal sistema, in una fase che compone il check-out; altrimenti escono direttamente.

1 Problematiche del sistema

L'AMA è obbligata a garantire un numero minimo di mezzi per ogni tipologia fuori dal sistema nelle varie fasce orarie. Spesso questo non è quello che succede, in quanto, comunemente, mezzi che entrano nel centro di raccolta per essere riparati incontrano grandi tempi di attesa anche per riparazioni minori, poiché devono attendere che vengano serviti tutti i mezzi arrivati prima. Si ha quindi una criticità dovuta al fatto che non c'è una gestione ottimale delle code dei mezzi, in quanto questa non si basa sulla necessità di avere un certo tipo di mezzo fuori dal sistema il prima possibile, in modo da rispettare le richieste riguardanti il numero di mezzi, ma solo sul loro ordine di arrivo. Non curare questi aspetti porta a malfunzionamenti nel servizio di raccolta che si ripercuotono sui cittadini e sul decoro pubblico oltre ad una penalità per l'azienda.

2 Scopi e Obiettivi

Lo studio condotto sul sistema si propone di raggiungere come obiettivo quello di minimizzare il numero di mezzi all'interno del sistema per ogni categoria in modo da rispettare per

complementarità i QoS richiesti (meno mezzi si hanno nel sistema, più mezzi sono effettivamente in uso per raccogliere rifiuti).

- Il numero di mezzi fermi di tipo CSL2 non deve superare il 27%.
- Il numero di mezzi fermi di tipo CSL3 non deve superare il 64%.

3 Modello concettuale

Il sistema si può modellare come una rete di code. Per iniziare, identifichiamo gli utenti del sistema in:

- **CSL2:** Mezzi leggeri, con una capienza di 50 tonnellate.
- **CSL3:** Mezzi pesanti, con una capienza di 100 tonnellate.

Il sistema è suddivisibile in due macroaree:

- **Smaltimento:** Zona adibita allo scarico dei rifiuti, permette lo scarico di 5 mezzi contemporaneamente.
- **Officina:** Zona dedicata alla riparazione dei mezzi. Dopo una diagnosi presso *l'accettazione*, il mezzo guasto viene reindirizzato presso l'autofficina di competenza. Ciascuna autofficina presenta un diverso numero di serventi.

Un operatore AMA, al rientro dal servizio, può presentare o meno un guasto al mezzo, non attualmente identificato.

- Se il mezzo è integro, l'operatore si reca presso la zona **smaltimento**, in cui attende il suo turno per smaltire i rifiuti. Il tempo richiesto per tale operazione è di 15 minuti, e non è particolarmente sensibile rispetto la capienza del mezzo. Il questo caso il servente è rappresentato dal *fosso dello scarico rifiuti*. Ciò

è modellabile come una coda singola con scheduling FIFO, avente 5 serventi.

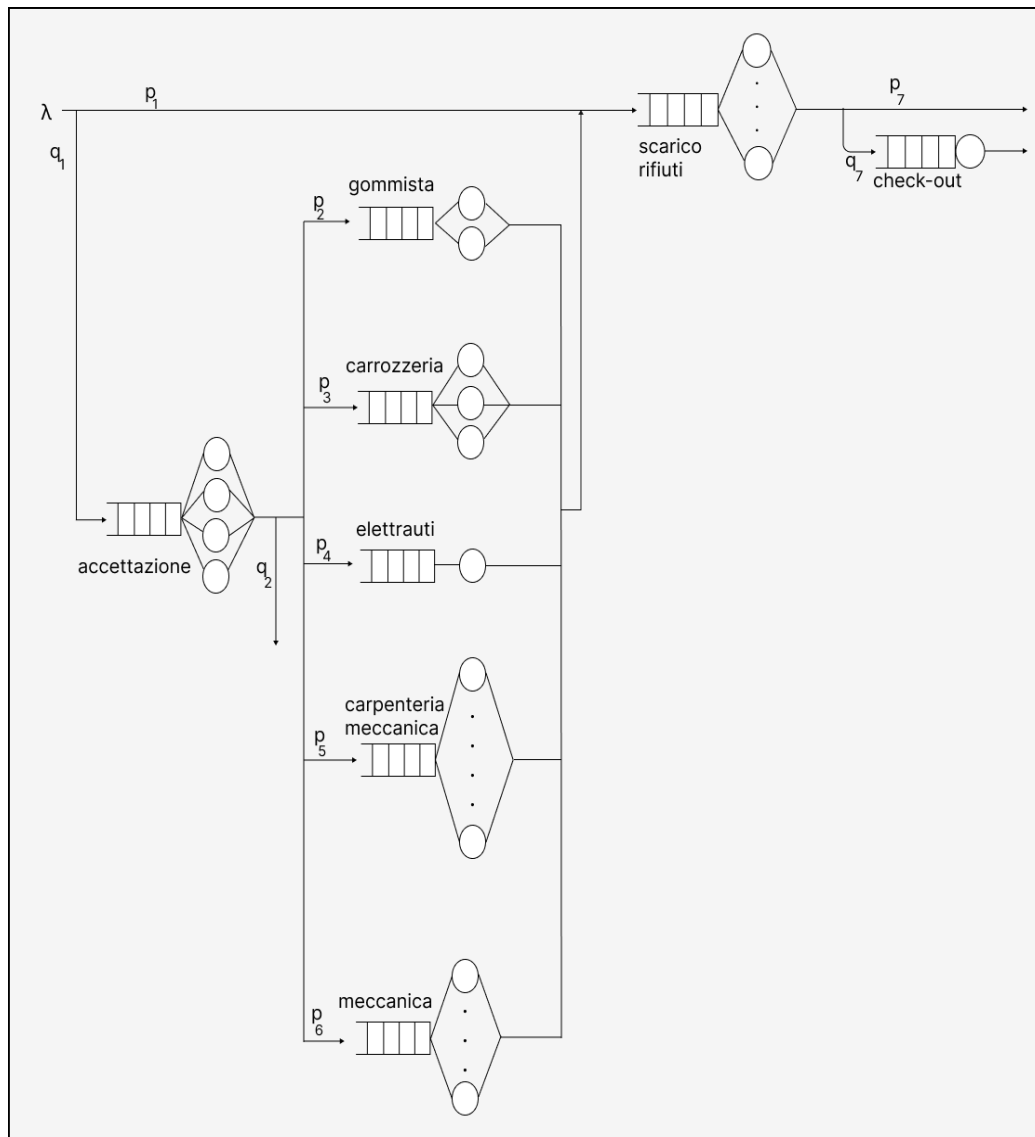
- Se il mezzo presenta anomalie, viene reindirizzato presso *l'accettazione*, che identifica il tipo di problema. Tipicamente, il mezzo stesso presenta un sistema di diagnostica per facilitare l'operazione, e ciò si traduce in tempi rapidi per l'identificazione dell'officina di competenza. Essa è modellabile come una coda singola a servente singolo con scheduling FIFO.

Con una determinata probabilità, analizzata nella sezione a venire sul modello delle specifiche, il responso dell'accettazione instrada il mezzo guasto in un determinato centro di riparazione. Nella nostra trattazione ne abbiamo 5, modellati nella maggior parte dei casi con scheduling astratto senza prelazione e multi-server. Riparato il guasto, il mezzo viene condotto nell'area dello smaltimento rifiuti per completare il lavoro. E' normale prassi, infatti, che un mezzo guasto non venga svuotato prima della riparazione, ma solo successivamente, in quanto lo scarico dei rifiuti non è un'operazione meccanicamente banale, e la sua attuazione in condizioni non idonee potrebbe portare a guasti di maggior entità.

Tutti i mezzi non guasti, e che hanno completato la fase di smaltimento, possono lasciare direttamente il sistema per un nuovo turno lavorativo, oppure entrare in una fase di *check-out*, che prevede una sanitizzazione del mezzo e un rifornimento per il carburante.

Dopo la presunta risoluzione del guasto relativo alla segnalazione, ci possono essere tre casi:

Quanto descritto finora nel modello concettuale si può schematizzare nell'immagine che segue.



Eventi

In questa trattazione, assumiamo come sinonimi il termine *mezzo* (inteso come camion adibito al trasporto di rifiuti) e **job**. Un evento considerabile nel del sistema di riferimento è:

- ❑ Arrivo di un un nuovo job nel sistema.

Un evento considerabile in **tutti i centri** del sistema è:

- ❑ Completamento del servizio di un job.

Gli eventi che possiamo considerare nel sotto-centro **“Smaltimento”** sono:

- ❑ Arrivo di un nuovo mezzo pronto per lo scarico dei rifiuti.

Gli eventi che possiamo considerare nell' **“Autorimessa”** sono:

- ❑ Arrivo di un nuovo job nella coda riservata alla tipologia di guasto **gommista**.
- ❑ Arrivo di un nuovo job nella coda riservata alla tipologia di guasto **carrozzeria**.
- ❑ Arrivo di un nuovo job nella coda riservata alla tipologia di guasto **carpenteria meccanica**.
- ❑ Arrivo di un nuovo job nella coda riservata alla tipologia di guasto **meccanica**.

Nell'area **“Checkout”** l'evento considerato è:

- Arrivo di un nuovo mezzo pronto per la sanitizzazione e il rifornimento.

Inoltre, nel centro "**Accettazione**" possiamo considerare come evento l'arrivo di un nuovo mezzo guasto.

Descrizione degli eventi

Un mezzo entrante nel sistema e **non guasto**, si reca presso l'area conforme allo scarico dei rifiuti. Questa prevede 5 zone (dette anche "buche") per lo scarico (che rappresentano i serventi). Un mezzo occupa una sola "buca". Se tutte le zone sono occupate, il mezzo viene messo in coda, ma non può abbandonare il sistema, in quanto non agibile per la raccolta di nuovi rifiuti.

Un mezzo entrante nel sistema e **guasto**, viene portato nell'area **accettazione**. Qui si esegue un lavoro di tipo *dispatcher*, in quanto il mezzo viene indirizzato verso il centro di riparazione più congeniale. Sono presenti 4 serventi dediti a tale fase. Se tutti i serventi sono occupati, il mezzo rimane in attesa.

Successivamente a questa fase, il mezzo guasto

viene preposto per la riparazione, classificata per la tipologia di guasto, a cui corrisponde una determinata officina. A seconda dell'officina, possono esserci più o meno serventi, i quali vanno da 1 (nel caso *elettrauti*) a 6 (nel caso *carpenteria meccanica*).

Un mezzo, in condizioni piuttosto rare, potrebbe essere non riparabile, e quindi abbandonare il sistema.

Fatta eccezione per questa possibilità, tutti i mezzi in attesa di riparazione non possono lasciare la coda né il sistema. Tutti i mezzi riparati sono reintroducibili nella coda principale per lo scarico dei rifiuti, di cui abbiamo già esaminato le caratteristiche.

Infine, tutti i mezzi che hanno espletato l'operazione di scarico dei rifiuti possono uscire dal sistema, o rimanervi in attesa della sanitizzazione e rifornimento.

Nello stato attuale dell'analisi, tutte le code sono definite con priorità FIFO astratte. L'unica differenziazione può avvenire nel numero di serventi, che confluiscono nell'uso di single-server o multi-server.

4 Analisi del contesto attuale

Trattandosi di un'azienda pubblica, non possono essere usati come parametri aspetti legati al denaro. Tuttavia, il decoro di una città presenta anche un impatto culturale sul turismo e sulla concezione del territorio. Ciò che si può analizzare è la qualità del servizio offerto, legato al numero di mezzi fermi nel sistema. Un mezzo fermo infatti, non partecipa alla pulizia delle strade, portando il malcontento tra i cittadini. Questi aspetti sono supportati da dei lower bound inerenti al minimo numero di mezzi circolanti per le strade.

5 Modello delle specifiche

Per rendere il più possibile realistica la modellazione del sistema, abbiamo chiesto ed ottenuto dei dati direttamente dall'AMA, come il numero di mezzi totali e quelli fermi in attesa di riparazione. Sono stati ottenuti anche report su alcune tempistiche di riparazione, oltre che dati sui tempi richiesti per lo smaltimento di un mezzo. Attualmente, i dati a nostra disposizione ci informano che:

- I mezzi attualmente presenti (sia attivi che in riparazione) presso l'autorimessa di Rocca Cencia sono 135, di cui:
 - TODO

Per quanto riguarda gli arrivi, che suddivideremo in tre fasce orarie, utilizzeremo, per la singola

fascia oraria, la *distribuzione esponenziale*, la quale presenta le seguenti proprietà:

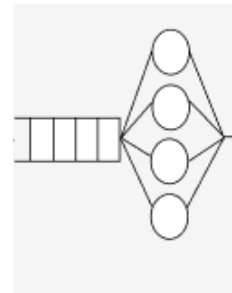
- ❑ Indipendenza per gli arrivi: l'arrivo di un mezzo nello stabilimento non influenza altri arrivi.
- ❑ Memoryless: Un arrivo futuro non è influenzato da arrivi passati.
- ❑ Arrivi Poissoniani: Trattasi di eventi casuali ed indipendenti, con un tasso di arrivo medio.

I servizi nell'area *autofficina* seguono una *distribuzione esponenziale*, data dalla casualità del guasto e del tempo di riparazione.

Nell'area *scarico rifiuti* si utilizza una *distribuzione BOH*.

Accettazione

Rappresentato da un *multi-server a coda singola*, poiché i mezzi guasti entranti nel sistema aspettano il loro turno per la diagnosi.

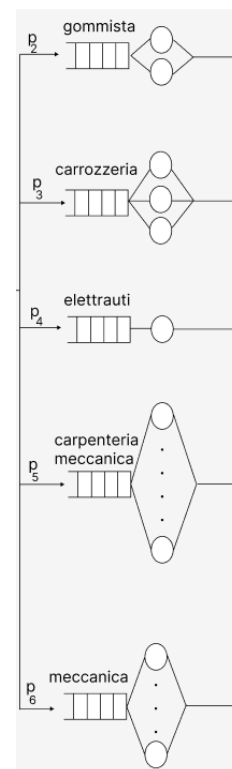


Autofficina

Dopo la diagnosi del guasto, il mezzo viene instradato presso l'autofficina di competenza. Troviamo 5 tipologie di autofficina, in particolare:

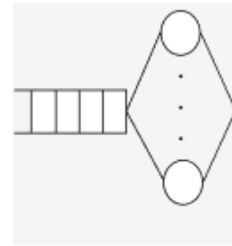
- ❑ **Gommista:** Trattasi di coda singola con due serventi, con arrivi esponenziali e tempi di servizio modellati tramite Poisson. Rappresentata come una M/M/2.
- ❑ **Carrozzeria:** Trattasi di coda singola con tre serventi, con arrivi esponenziali e tempi di servizio modellati tramite Poisson. Rappresentata come una M/M/3.
- ❑ **Elettrauti:** Trattasi di coda singola con un servente, con arrivi esponenziali e tempi di servizio modellati tramite Poisson. Rappresentata come una M/M/1.
- ❑ **Carpenteria Meccanica:** Trattasi di coda singola con sei serventi, con arrivi esponenziali e tempi di servizio modellati tramite Poisson. Rappresentata come una M/M/6.
- ❑ **Meccanica:** Trattasi di coda singola con quattro serventi, con arrivi esponenziali e tempi di servizio modellati tramite Poisson.

son. Rappresentata come una M/M/4.



Scarico rifiuti

Rappresentata mediante una M/M/5 (o forse no!?, magari uniforme?). I mezzi arrivano preso l'area di scarico rifiuti, ed occupano una delle cinque dislocazioni per effettuare l'operazione.



Checkout

Rappresentato mediante una M/M/1 (forse anche M/M/2). Un mezzo che ha completato lo smaltimento può essere sottoposto a lavaggio e rifornimento.

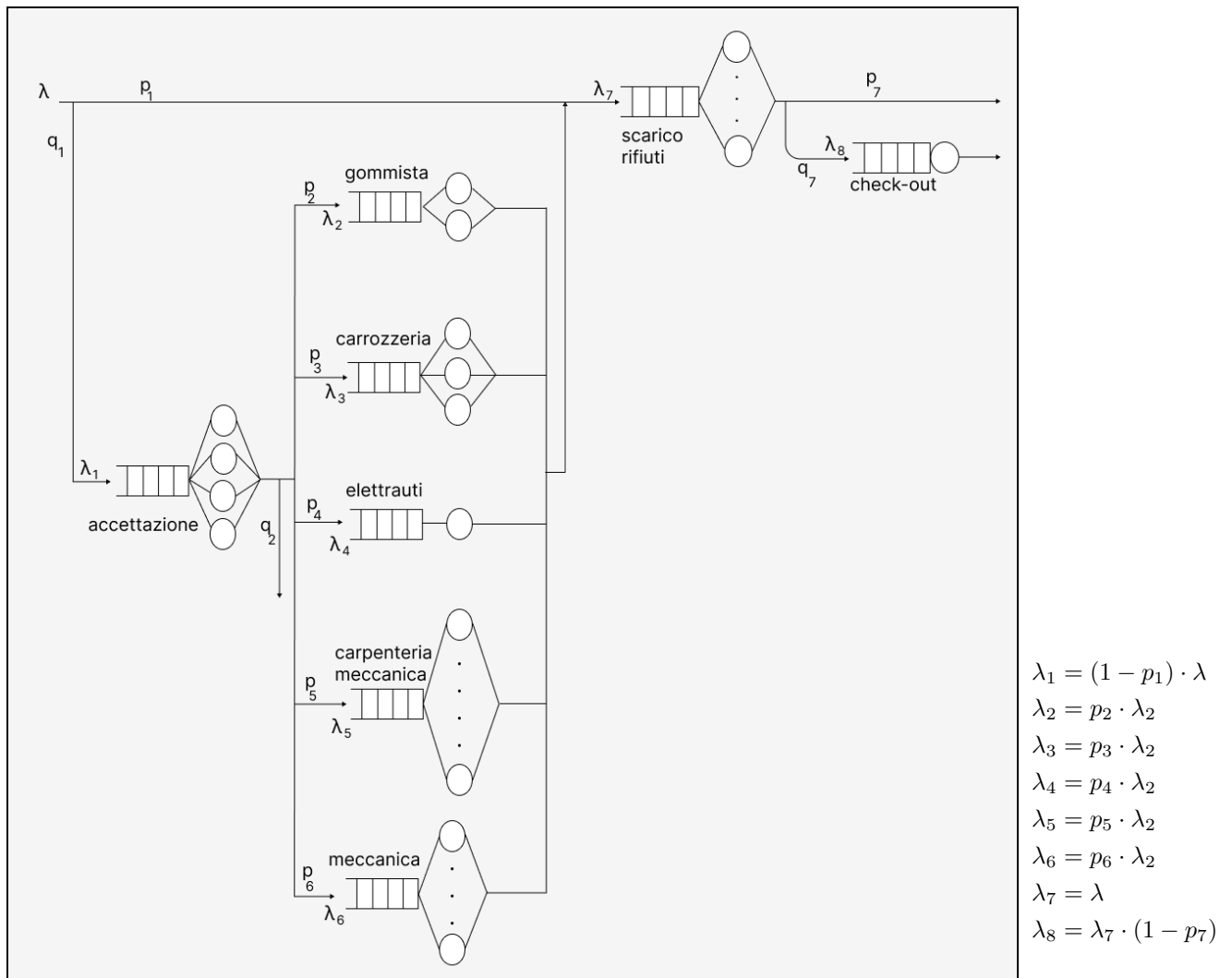


Descrizione degli arrivi

In totale sono disponibili 135 mezzi, suddivisi in due tipologie: X di tipo CSL2, Y di tipo CSL3. Anche se vi è una suddivisione in fasce lavorative, gli obiettivi aziendali sono quantificati in giornate. Inoltre, un eventuale servizio

non coperto in un turno, verrà coperto in un turno successivo. Ciò si trasforma in penalità sul servizio giornaliero, non sulla singola fascia oraria.

Equazioni di traffico



Di seguito, viene invece illustrata la matrice di routing.

Matrice di routing

	Esterno	Accettazione	Gommista	Carrozzeria	Elettrauti	Carpenteria	Meccanica	Scarico	Checkout
Esterno	0	$1 - p_1$	0	0	0	0	0	p_1	0
Accettazione	q_2	0	p_2	p_3	p_4	p_5	p_6	0	0
Gommista	0	0	0	0	0	0	0	1	0
Carrozzeria	0	0	0	0	0	0	0	1	0
Elettrauti	0	0	0	0	0	0	0	1	0
Carpenteria	0	0	0	0	0	0	0	1	0
Meccanica	0	0	0	0	0	0	0	1	0
Scarico	p_7	0	0	0	0	0	0	0	$1 - p_7$
Checkout	1	0	0	0	0	0	0	0	0

6 Implementazione

Architettura

Per la realizzazione del progetto, si è optato per l'uso di Java. I fattori che hanno indirizzato verso questa scelta sono:

- ❑ E' un linguaggio orientato agli oggetti, il che significa che è ottimo per rappresentare gli elementi di un sistema di code (come code, server, clienti) come oggetti.
- ❑ Offre un'ampia libreria standard che include molte classi e funzionalità utili per

la gestione delle strutture dati.

- ❑ Portabilità: Utile per la creazione di modelli di sistemi di code che possono essere eseguiti su vari ambienti senza dover essere riscritti.
- ❑ Familiarità con tale linguaggio di programmazione.

Struttura del progetto

Il progetto è composto da tre blocchi fondamentali:

- ❑ **Controllers:** Contiene la logica dei vari nodi formanti il sistema in esame, oltre che un EventHandler per la gestione centralizzata degli eventi, di cui disquisiremo più in avanti.
- ❑ **Models:** Contiene codice per l'implementazione di una Multi-Server-Queue, e costanti di uso comune nel

sistema. Presenta inoltre la definizione degli eventi del sistema, tempi di servizio, **tasso λ degli arrivi** e probabilità di essere indirizzati ad un centro rispetto che un altro.

- ❑ **Utils:** Contiene codice prodotto da **Steve Park** e **Dave Geyer**, il quale implementa facility per la generazione di numeri multi-stream, e produzione di statistiche di output

Descrizione dei controller dei vari nodi

Tale trattazione è applicabile per ogni controller facente parte il sistema, con delle piccole variazioni che analizzeremo in seguito. Ciascun controller ottiene l'istanza di EventHandler, per la gestione degli eventi mediante:

```
this.eventHandler=EventHandler.getInstance();
```

Si aggiunge alla lista di eventi un numero di entry pari al numero di server presenti nel sistema considerato. Nel metodo **baseSimulation** si aprono le porte per gli arrivi, si effettua un prelievo dell'indice del prossimo evento nella lista (che diventerà quello *corrente*, e si analizza il tipo di evento in questione. Se l'evento da gestire è di tipo arrivo, ovvero l'indice ($e==0$):

- ❑ `eventList.get(0).setT(this.time.getCurrent()+this.rnd.getJobArrival(1))`

Si genera un nuovo evento di tipo arrivo, avente come tempo la somma tra tempo corrente e tempo di inter-arrivo di parametro $\frac{1}{\lambda}$

- ❑ Si verifica se tale tempo, ottenuto per il next-event, eccede il tempo di orizzonte finito pari a **STOP**, il quale porterebbe alla chiusura delle porte per gli arrivi. Se ciò è vero, questo job non entrerà nel sistema. Successivamente, si incrementa il numero di job presenti nel centro.

- ❑ `if(this.number<=SERVERS_ACCETTAZIONE)`

Si verifica la presenza di server liberi attualmente, in caso positivo si ottiene un tempo di servizio, e si produce il tempo di completamento del servizio, con il server in questione occupato fino alla terminazione di questo tempo, considerando anche il tempo corrente di entrata in servizio.

Se l'evento è di tipo partenza, ovvero l'indice è $e \neq 0$, allora devo processare una partenza:

❑ Si decrementa il numero di job presenti nel centro.

❑ `if(this.number>=SERVERS_ACCETTAZIONE)`

Si verifica la presenza di altri job nel sistema, in caso affermativo si ottiene un nuovo tempo di servizio (che verrà inserito nella lista degli eventi, e verrà sfruttato successivamente da un job in coda quando verrà servito) e si aumenta il numero di job serviti.

Come accennato in precedenza, ogni centro del sistema presenta alcune caratteristiche che lo differenzia dagli altri:

Accettazione

Riceve arrivi dall'esterno, e ciò che produce in uscita fungerà da ingresso per i nodi dell'officina. Questo reindirizzando un mezzo guasto presso una certa officina, mediante probabilità P_i . Ciò viene realizzato mediante:

```
eventHandler.getInternalEventsOfficina(off).add(new EventListEntry(event.getT(),
                                                                    event.getX(),
                                                                    event.getVehicleType()));

if(eventHandler.getEventsSistema().get(off+2).getT()>eventList.get(e).getT()
    || eventHandler.getEventsSistema().get(off+2).getX()==0)
{
    eventHandler.getEventsSistema().get(off+2).setT(eventList.get(e).getT());
}
```

Ottenuto l'indice dell'officina 'off' (incrementato di 2 per coerenza con la numerazione delle probabilità), nel quale il mezzo andrà in riparazione, si aggiunge una nuova entry, contenente il tempo di arrivo nell'officina, lo stato dell'evento, il tipo di mezzo. Si effettua un controllo anche sugli eventi generali del sistema, in particolare controllando se questo evento di entrata in una certa officina corrisponda anche al prossimo evento da gestire, impostando di conseguenza il clock del sistema. E' possibile, seppur con probabilità ridotta, avere un abbandono dal sistema, ovvero un camion non più disponibile.