



UNIVERSITA' DEGLI STUDI DELL'AQUILA

DIPARTIMENTO DI INGEGNERIA E SCIENZE DELL'INFORMAZIONE E MATEMATICA



Progetto del corso di Ingegneria del Software

a.a. 2017-2018

Lecturer: Henry Muccini, Ph.D.

Titolo: Industrial Robot dashboard

In un ambiente di produzione altamente automatizzato ci sono una serie di **ROBOT** (R1...RN) che performano delle operazioni di produzione. Questi robot sono raggruppati in **cluster**: i robot che appartengono allo stesso cluster sono interscambiabili in quanto effettuano le stesse operazioni ed hanno le stesse specifiche tecniche in termini di hardware. Il processo produttivo e' diviso in aree e quindi ciascuna area dispone di una serie di cluster di robot per eseguire i processi produttivi.



Ciascun robot ha un codice alfanumerico che lo identifica univocamente, così come i cluster e le aree.

Ciascun robot invia periodicamente alcuni segnali di stato per indicare se e' abile o meno a performare una certa operazione. In particolare ci sono 7 segnali possibili S1 ... S7. I segnali hanno valori 1 o 0.

Ad esempio, il segnale S1 indica se la precisione del suo laser e' nei parametri di specifica. Se il robot invia S1=1 significa che puo' eseguire tutti i processi produttivi che richiedono il laser in quanto la sua precisione

e' nei parametri di specifica. Se invece il robot invia il segnale S1=0 significa che il laser e' andato fuori specifica e che quindi non puo' piu' eseguire i processi produttivi che richiedono un laser.

I segnali, da S1 ad S7 indicano se i seguenti parametri di processo sono all'interno dei range di specifica per il particolare processo produttivo:

- S1: laser, calibrazione;
- S2: piastra di processo, livello di contaminazione;
- S3: camera di processo, temperatura;
- S4: camera di processo, livello di contaminazione;
- S5: camera di processo, pressione;
- S6: injector plasma, pressione;
- S7 injector plasma, temperatura.

Tali informazioni sono fondamentali per l'ottimizzazione del processo produttivo in quanto l'analisi real-time dei segnali permette al sistema di gestione della linea di decidere quando i robot devono andare in manutenzione o quando invece possono rimanere in produzione anche se non sono completamente utilizzabili. Ad esempio nel caso del robot con S1=0 si potra' decidere

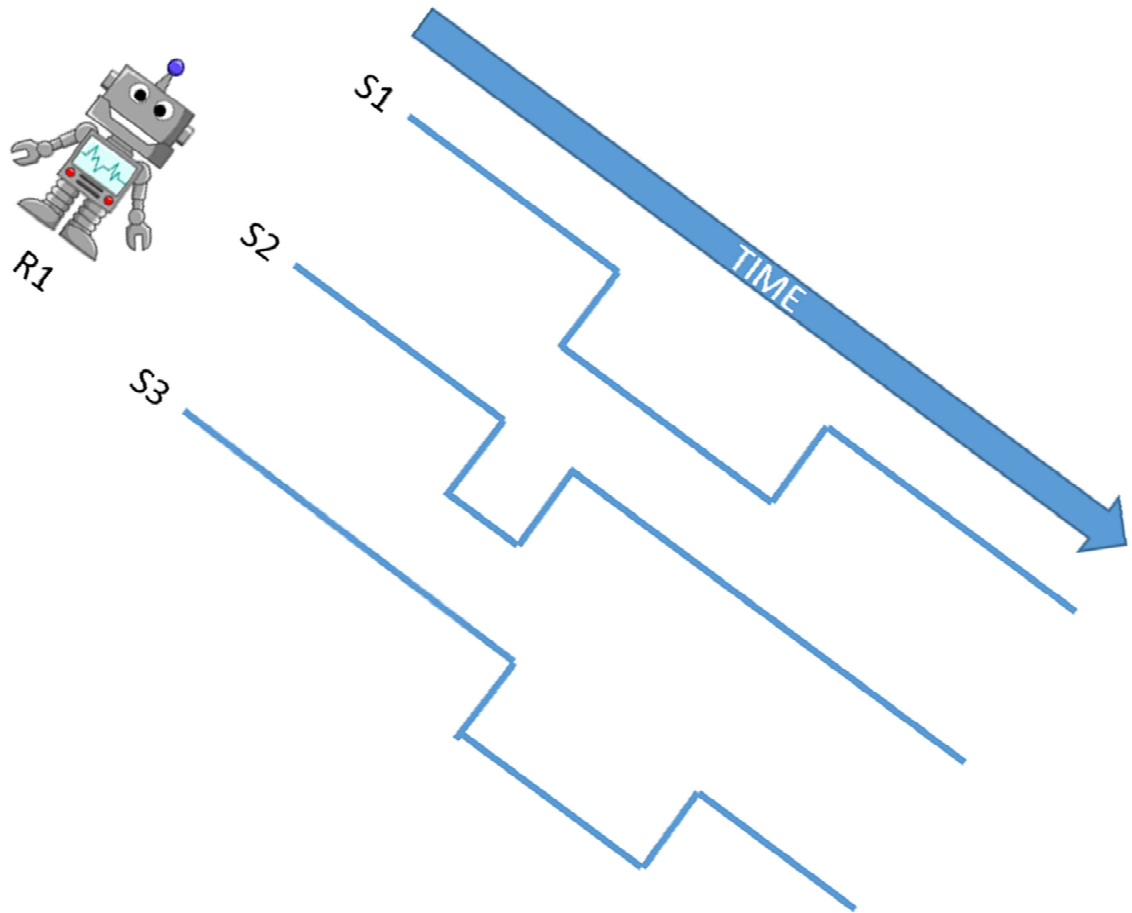
- di tenerlo in produzione a lavorare i processi che non richiedono laser in quanto ci sono sufficienti robot dello stesso cluster che possono soddisfare le richieste di linea
- di mandarlo in manutenzione per ricalibrare il laser.

Tali segnali vengono quindi processati da un sistema decisionale automatizzato che in base alla domanda di produzione **ottimizza le manutenzioni dei robot**. Tale sistema pero' deve essere configurato dagli ingegneri di linea per farlo lavorare al meglio. **Si richiede quindi di fornire al team di ingegneria di linea una dashboard che visualizzi l'inefficiency rate dei robot in linea.**

Il valore fondamentale che deve essere mostrato in questa dashboard e' la percentuale di DOWN (segnale = 0) dei robot sulla totalita' dei 7 segnali.

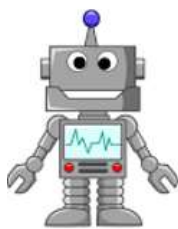
Esempio: per semplificare supponiamo che esistano solo 3 segnali S1, S2 ed S3 e supponiamo di voler analizzare solo il robot R1.

Questo e' lo **stream** dei segnali ricevuti dal sistema nell'ultima ora:

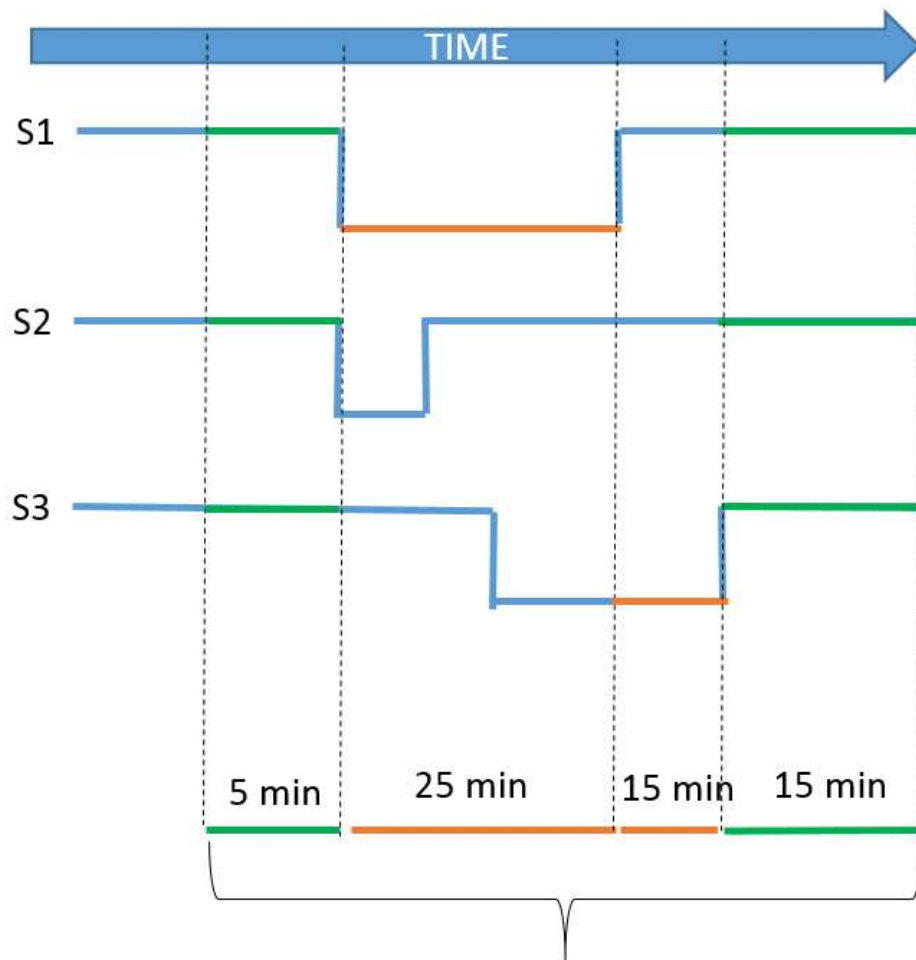


L'inefficiency rate (IR) viene calcolato come "DN=tempo di down"/"UP=tempo di up" cumulativo su tutti i segnali. Quindi, $IR = \text{DN cumulativo} / \text{UP}$.

E' importante chiarire che il robot si considera in stato DN se almeno uno dei segnali e' DN nel lasso temporale analizzato.



R1



Nel nostro esempio, i 3 segnali sono in DN cumulativamente per 40 minuti su di una finestra temporale di 60 minuti, quindi $IR = 40 \text{ (tempo di down)} / 60 \text{ (finestra temporale di 1 ora)} * 100 = 66\%$

La dashboard permettera' di selezionare un cluster di robot e mostrera' in rosso tutti i robot che hanno un inefficiency rate superiore al 40%:

Inefficiency Rate By Robot

R1
66%

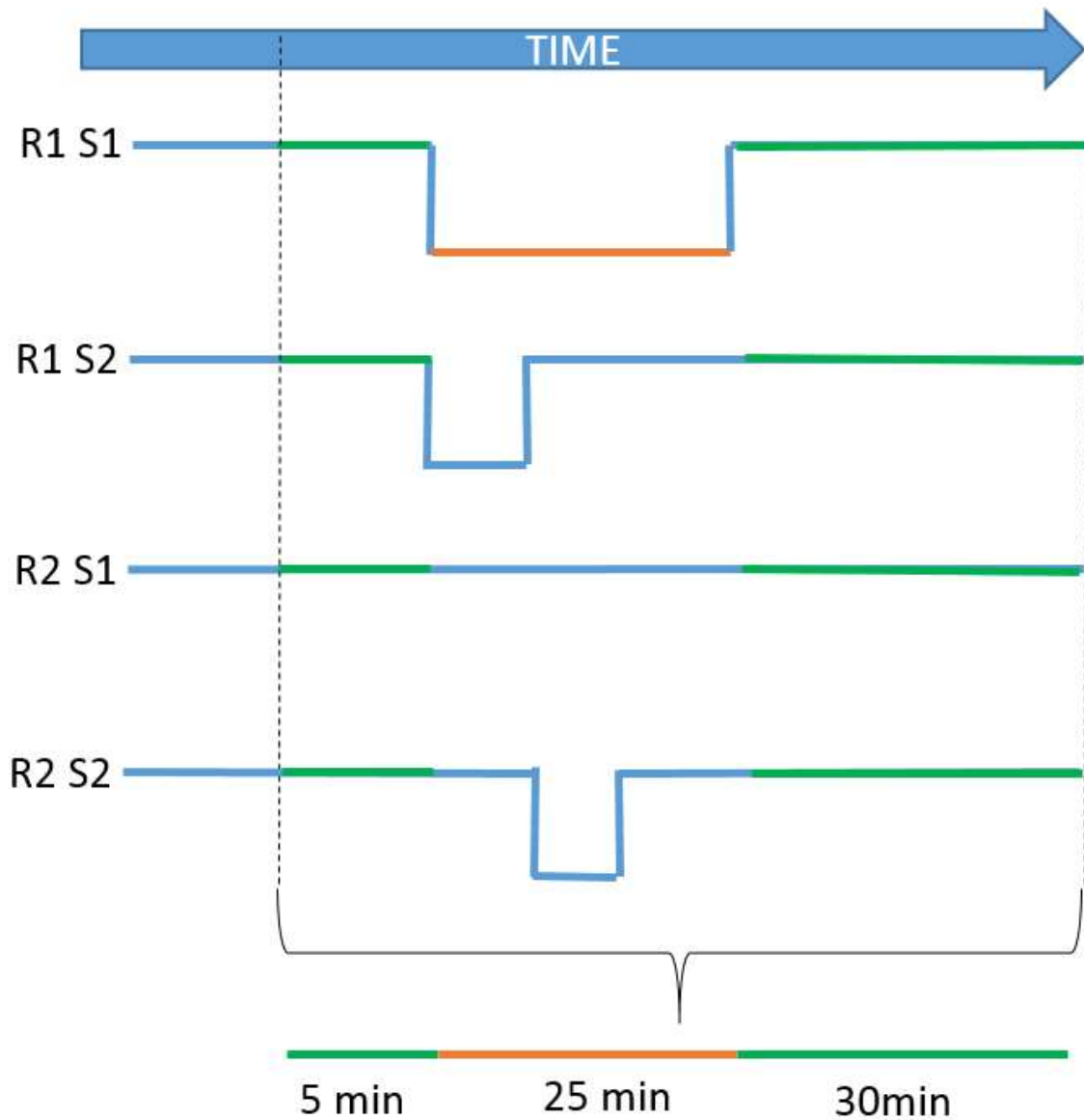
R2
30%

R3
20%

R4
15%

Inoltre, essendoci moltissimi clusters, e' necessaria una vista di piu' alto livello che mostra l'inefficiency rate all'interno del cluster.

Il calcolo by cluster deve essere fatto **considerando i singoli stream di tutti i robot appartenenti a quel cluster.**



In questo esempio, per semplicita' ipotizziamo che in un cluster ci siano solo 2 robot R1 ed R2 e che gli eventi (stream) siano solo due, S1 ed S2.

In questo caso l'inefficiency rate per il cluster e' $25/60 = 41.5\%$

In modo analogo quindi la dashboard by cluster visualizzera' in rosso i cluster che eccedono il limite impostato.

Inefficiency Rate By Cluster



Una nota importante sul sistema di eventi dei robot e' che **i robot non possono essere interrogati**, ma essi mandano solo un segnale broadcast di cambio stato. Il sistema dovra' intercettare tale segnale e memorizzarlo su opportuno storage. Quindi anche se la finestra temporale per il nostro calcolo e' "l'ultima ora" si dovranno comunque memorizzare i segnali per un piu' lungo periodo in quanto, per effettuare il calcolo dell'IR, **e' necessario avere sempre il dato dell'ultimo cambio stato** che potrebbe essere avvenuto 2 o piu' ore fa.

Il sistema deve essere in grado di gestire fino a **90.000** robot. Tipicamente un cluster e' formato da un range di robot che varia tra i 500 ed i 1000. Quindi si puo' considerare che il sistema sara' configurato per gestire circa **100 cluster** in cui vi saranno **900 robot** ciascuno (il valore riportato e' solo indicativo. Il sistema deve essere generico).

Un robot in media manda un messaggio di cambio stato al **minuto**. Infatti, ci sono stati come quello S1 del laser che cambiano molto raramente, ma ci sono altri legati alla temperatura e pressione di esercizio che possono cambiare molto di frequente. Ad esempio il segnale S7 di temperatura sara' 1 prima del processo produttivo, diventera' 0 per una decina di secondi dalla fine del processo, per poi tornare ad 1 una volta che il processo di raffreddamento abbia ripristinato la corretta temperatura di idle.

L'architettura di sistema dovra' quindi **supportare lo storage ed il processing di almeno 90.000 messaggi al minuto** e dovra' essere **scalabile** in modo da **supportare future espansioni**.

La **dashboard** sara' aperta sui PC di circa **100** ingegneri di linea e dovra' aggiornarsi automaticamente almeno ogni **5 minuti**. Soluzioni architetturali estremamente performanti che permettano di ridurre il ritardo di **refresh a meno di 1 minuto sono considerate preferenziali** purché possano essere realizzate senza aumentare "eccessivamente" i costi hardware del sistema e/o i tempi di sviluppo (come parte del progetto fornire una stima dei costi per una tale soluzione)