Proposta Progetto IoT BS

Alessandro Di Stefano, Marco Grassia, Luca Musumeci

Introduzione

Contrastare le notevoli problematiche in merito all'approvvigionamento dell'acqua è diventata regolare incombenza di diversi enti locali nazionali e siciliani.
Si stima che in Sicilia oltre il 50% di acqua si perda nei collegamenti della rete, a fronte di una media del 35% a livello nazionale, e la causa principale è da cercarsi nelle condutture idriche obsolete e fatiscenti.
L'assenza di documentazione sulla topologia e/o topografia della rete idrica comporta enormi difficoltà anche negli interventi d'urgenza necessari a seguito di rotture dei condotti sotterranei. La problematica si inasprisce ulteriormente in quelle città che negli ultimi anni hanno subito una forte crescita demografica e che non hanno saputo rispondere alla stessa con una opportuna riqualificazione delle infrastrutture e dei servizi - nello specifico i servizi acquedotto - da fornire alla popolazione.
Sono diverse le realtà amministrative in cui un mancato controllo dell'erogazione comporterebbe difficoltà nell'approvvigionamento quotidiano dell'acqua per i cittadini. Nelle realtà siciliane, è frequente l'apertura e chiusura periodica, e manuale, del circuito in uscita verso le diverse zone della città al fine di limitare il deflusso dell'acqua dalle vasche (dovuto a perdite e/o consumi) e permetterne il rifornimento per il giorno successivo.

Il progetto proposto vuole promuovere una possibile soluzione basata sull'automazione del sistema idrico e sull'orchestrazione di sensori e attuatori posti nei punti cruciali (quali pozzi, vasche, e nodi locali nelle diverse zone) della rete al fine di ottimizzare il riempimento dei serbatoi per garantire qualità del servizio acquedotto. Poiché oltre lo scopo del progetto, trascuriamo la possibile mancanza di documentazione in merito alla topografia/topologia della rete idrica di una città, considerandola nota.

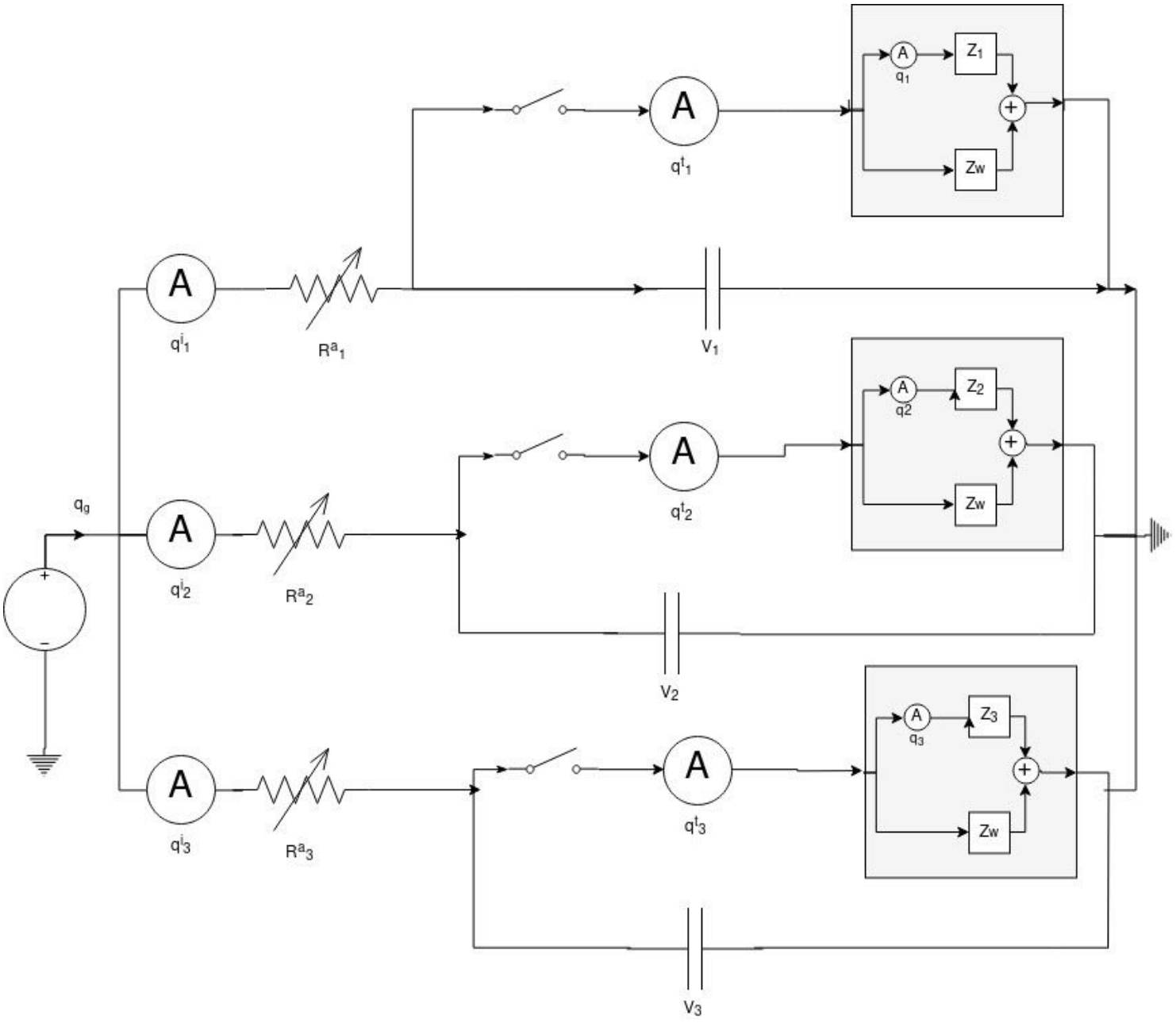
Soluzione proposta

La soluzione proposta si basa sulla tecnologia e sulle metodologie IoT che permettono di monitorare la quantità d'acqua residua nei serbatoi, le portate in ingresso ed in uscita, e di gestirne l'approvvigionamento e la erogazione.

È possibile modellare una rete idrica di un servizio acquedotto (A) in due reti:

- R1: costituita da M pozzi ed N vasche;
- R2: rete degli utilizzatori (abitazioni).

Sfruttando l'analogia idraulica-elettrica, il seguente circuito di esempio rappresenta l'intera rete in cui, per semplificazione e senza perdere di generalità, si è usato un solo pozzo.



Nel circuito sopra, i condensatori rappresentano le vasche, i blocchi in grigio i circuiti dei settori (vedi sotto), il generatore di tensione una pompa ideale su un pozzo.

Le reti presentano, rispettivamente:

- Rete 1 (pozzo e vasche):
    - Il pozzo genera una portata q_g;
    - I tre potenziometri lineari R^u_n sono gli attuatori (elettrovalvole) che variano la portata in ingresso ad una vasca in funzione dei messaggi ricevuti da un orchestrator;

- Le tre vasche sono rappresentate dai condensatori V_n e sono caratterizzate da una massima capacità (volume d'acqua), una portata (q^i_n) in ingresso (dai pozzi), una portata (q^o_n) in uscita verso le zone Z servite;

- Un controllo ON-OFF con isteresi è utilizzato per chiudere l'uscita dalle vasche nel caso in cui il volume d'acqua presente sia minore di una certa soglia di sicurezza.

- Rete degli utilizzatori (abitazioni, etc):
  Aggiungiamo un insieme di sensori lungo il percorso dalla vasca N alle abitazioni:
  - Suddivisa ulteriormente in settori;

  - Ogni settore presenta:
    - Un sensore di portata (q^i_{nn}) in ingresso;
    - Un sensore di portata per ogni abitazione, per esempio nel contatore;
    - Alcuni sensori, dislocati uniformemente all'interno;

  - Nell'ambiente virtuale simulato su Simulink, le portate in ingresso alle diverse abitazioni sono state considerate come un'unica portata, di cui misuriamo il valore con l'ausilio di un unico sensore.

Sulle precedenti vengono effettuati i seguenti controlli:

- Supponendo la disponibilità di acqua nei pozzi (generatore ideale di tensione/pressione), il problema sul lato della rete (1) diviene ottimizzare gli ingressi in modo tale da distribuire il flusso di acqua in entrata in maniera proporzionale al consumo nei settori di competenza delle vasche e alla loro capacità attuale.

- Nota la portata in ingresso ad un settore *s*, e la portata in ingresso ad ogni utilizzatore (abitazione) di *s* si ha una perdita se:

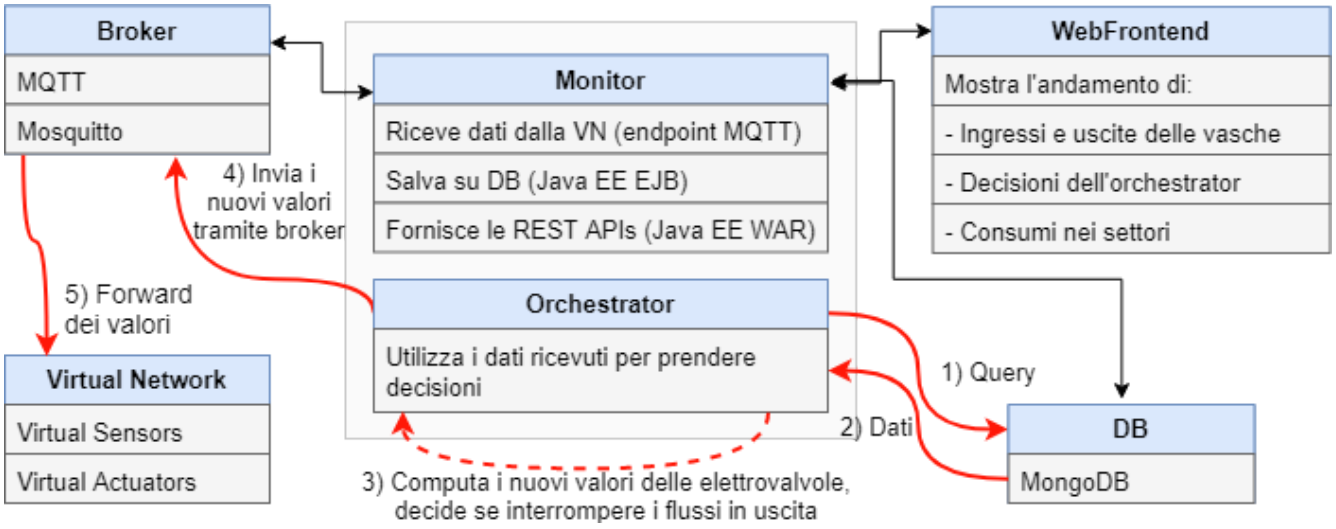
$$\sum_{a=1}^{N_s} (q_a) - q_s < err_{perdita}$$

  Dove N_s è il numero totale di abitazioni del settore *s*.

Implementazione

Architettura proposta

L'architettura proposta per la realizzazione del progetto è la seguente.



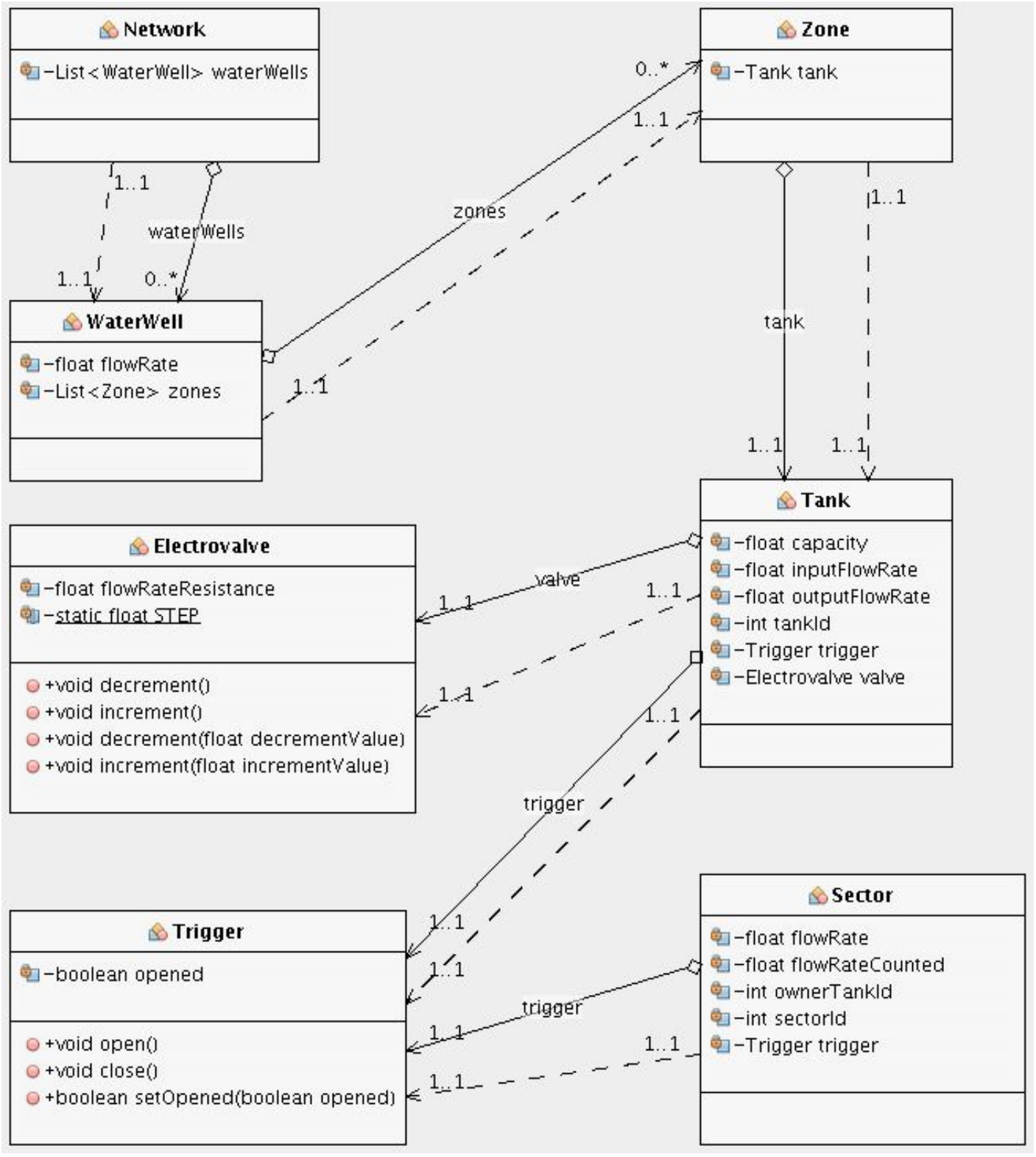
Per il sistema implementato, la rete (VirtualNetwork) è stata virtualizzata su Simulink, il resto è costituito da quattro container Docker che offrono un lato di frontend per REST API e visualizzazione delle statistiche ed un backend per le strategie di controllo e monitoraggio della rete.

Modello: Virtual-Network (sensori, attuatori, sistema fisico)

Il sistema fisico è stato modellato tramite l'ausilio di Simulink, nel quale i sensori e gli attuatori (e la relativa comunicazione tramite MQTT) vengono emulati da uno script MATLAB in parallelo. Per ottenere una simulazione in tempo reale, è stato utilizzato un blocco apposito 3rd-party.
Lo script invia le letture al Monitor e riceve dati dall'Orchestrator tramite protocollo MQTT, modificando i valori di resistenze che virtualizzano gli attuatori sulla base dei dati ricevuti dai sensori.
Nel primo caso, lo script agisce da *publisher* sui topic dei sensori, mentre nel secondo agisce da *subscriber* ai topic degli attuatori.

Modello software

Il sistema fisico emulato è stato modellato anche lato software, di cui segue il diagramma UML delle classi utilizzate per lo scopo, opportunamente semplificato.



Broker

Il broker utilizzato è Mosquitto, una implementazione open-source del broker del protocollo MQTT.

La granularità dei Topic rispecchia quella utilizzata nella suddivisione vista precedentemente. Nel dettaglio:

- Vasche/Zone: i topic di pubblicazione/sottoscrizione che vengono utilizzati per la regolazione della Vasca sono *"/actuators/zones/{$i}/"* e *"/sensors/zones/{$i}/"*, in cui *{$i}* rappresenta l'identificativo della vasca di interesse;

- Settori: i topic di pubblicazione/sottoscrizione che vengono utilizzati per la regolazione del Settore sono "*/actuators/zones/{$i}/sectors/{$j}/*" e "*/sensors/zones/{$i}/sectors/{$j}/*", in cui *{$i}* rappresenta l'identificativo della vasca e *{$j}* rappresenta l'identificativo del settore.

Orchestrator

L'Orchestrator si occupa della regolazione del sistema.
Periodicamente controlla l'attuale stato delle vasche e dei settori mediante i dati registrati e forniti dal Monitor, sulla base dei quali effettua le decisioni di controllo che propaga agli attuatori della Virtual Network tramite MQTT.
È responsabile, inoltre, del controllo delle perdite di acqua nel sistema. Nel caso in cui la differenza tra le rilevazioni del volume di acqua entrante in un settore e quello effettivamente utilizzato sia maggiore di una certa soglia di errore, è presente una perdita. Se questa supera un limite critico, l'Orchestrator invierà una mail di allarme.

Si nota che le strategie di controllo implementabili nell'Orchestrator sono differenti in funzione dei parametri della rete per la quale si vuole garantire un maggiore QoS (limitare lo svuotamento eccessivo della vasca, garantire una portata stabile in uscita, ecc..).
Seppure vadano oltre gli scopi del progetto, viene proposta una strategia semplificata, tenendo conto che in un caso reale il gen. di tensione (i pozzi con le relative pompe idrauliche) avrà una portata massima erogabile e risorse d'acqua limitate.

- Limitare lo svuotamento della vasca (caso ideale, portata erogabile sempre sufficiente): è stata utilizzata una strategia ispirata alla *congestion avoidance* del protocollo TCP:
    - Se la capacità residua della vasca è al di sotto di una soglia (flusso in uscita maggiore di quello in ingresso) viene decrementata la resistenza in ingresso al circuito (l'elettrovalvola) per permettere alla vasca di riempirsi più velocemente.
    - Se, viceversa, la capacità residua è al di sopra di una soglia massima, la resistenza in ingresso viene incrementata di un fattore minore.

- Ridistribuire il flusso in ingresso dando priorità alle vasche che hanno una portata in uscita maggiore;

- Se, al di sotto di una ulteriore soglia minima (caso di sistema sotto-dimensionato con alti utilizzi) un trigger blocca il circuito in uscito dalla vasca; gli utenti non vengono più serviti dopo un certo tempo di svuotamento delle condutture, contemporaneamente la vasca torna a riempirsi.

Le tre strategie sono state unite, semplicemente a scopo di esempio, e se ne fornisce uno snippet di seguito:

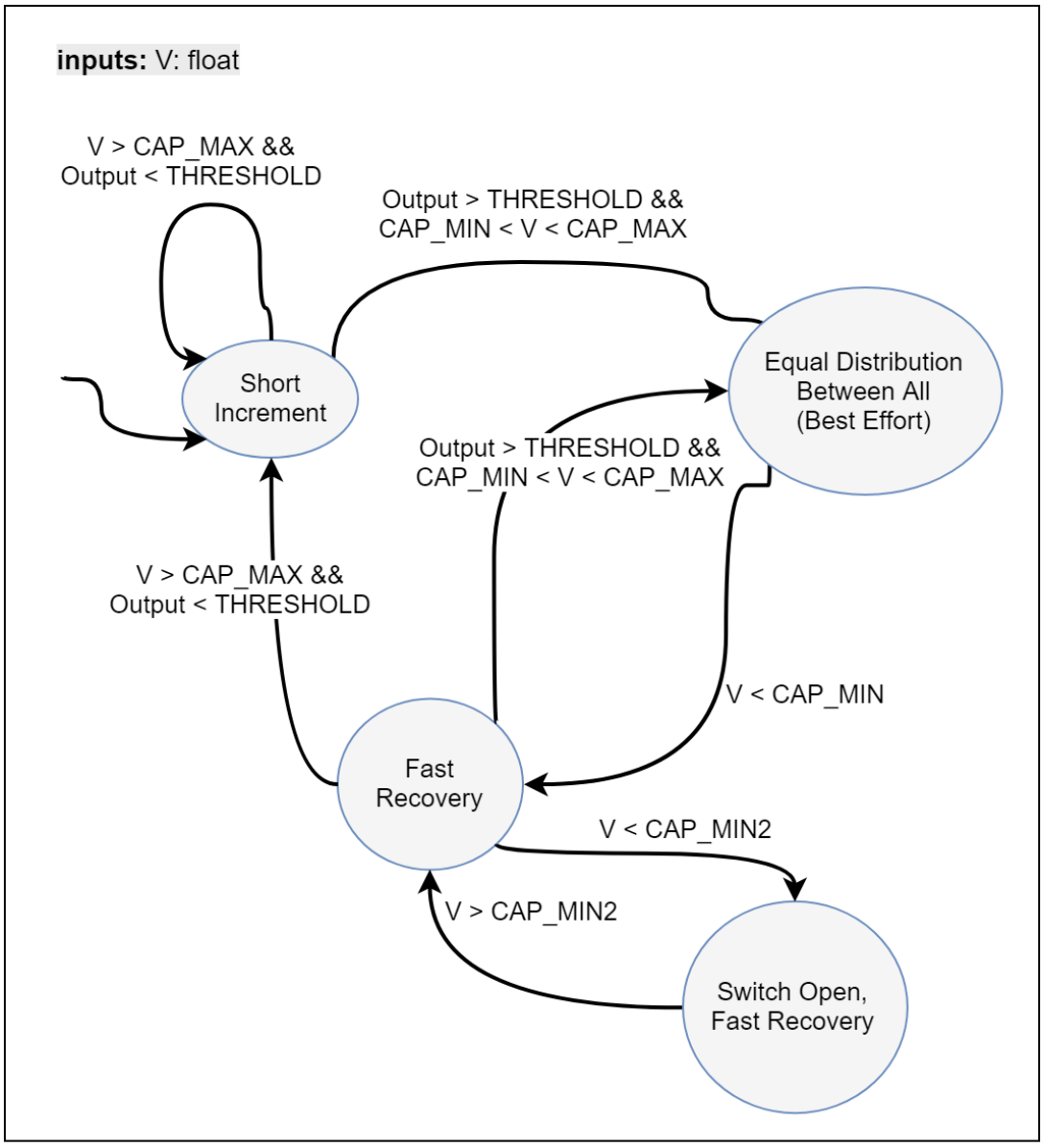
```
//Decide actuation strategy based on Tank capacity and Output
if (tank.getCapacity() > TANK_CAP_MIN &&
        tank.getCapacity() < TANK_CAP_MAX &&
        tank.getOutputFlowRate() > F_THRESHOLD) {
    rj = Ptot / (c * tank.getOutputFlowRate());
    tank.getValve().setFlowRateResistance(rj*100);
} else  if (tank.getCapacity() < TANK_CAP_MIN) {
    log += "Tank capacity under setPoint_min: DECREMENTING";
    tank.getValve().decrement(DECREMENT_STEP);
} else if (tank.getCapacity() > TANK_CAP_MAX) {
    log += "Tank capacity over setPoint_max: INCREMENTING";
    tank.getValve().increment(INCREMENT_STEP);
}

log += " - Setting input resistance to " + tank.getValve().getF

// OUTPUT SWITCH
if (tank.getCapacity() <= capacityError(true)) {
    log += " CLOSING TRIGGER";
    tank.getTrigger().close();
} else if (tank.getCapacity() > capacityError(false)) {
    log += " OPENING TRIGGER";
    tank.getTrigger().open();
}
mQTTClientSessionBean.publish(tank.getTankId() + "/", tank);
```

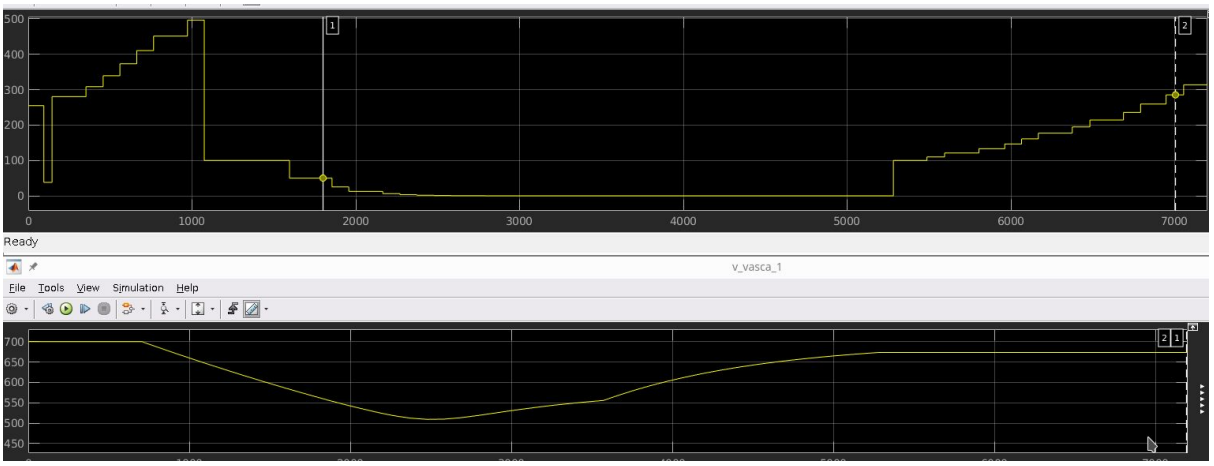
Come anticipato, la particolare strategia di controllo del processo va oltre gli scopi del progetto: in un caso reale andrebbe considerata la risorsa di flusso limitata (1) e il dimensionamento di un opportuno controllore discreto per un sistema dinamico continuo (2).

La strategia è di seguito esposta mediante FSM:

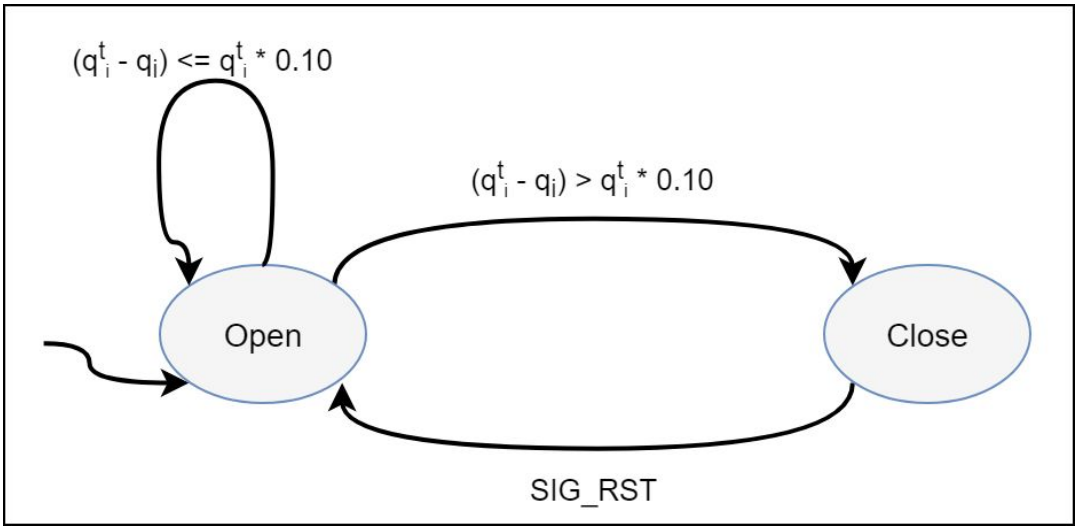


Un esempio di plot del comportamento su una vasca con andamento dell'utilizzo modellato come uno step di ampiezza elevata tra 1000s e 3500s, mostra il rallentamento dello svuotamento della vasca, dovuto prima allo stato "best-effort" di uguale distribuzione con le altre e dopo il "fast-recovery" (inverso) per riempirla (nell'ipotesi ideale di pompa ben dimensionata):

In alto: la resistenza in ingresso alla vasca. In basso: la tensione della vasca.



A seguire viene mostrata la macchina a stati utilizzate per la gestione dello switch in uscita di un settore:



Sono state eseguite altre simulazioni con diversi modelli di utilizzo e di perdita, di cui si riportano i dati sintetici, dai quali è possibile osservare un incremento del volume totale a seguito dell'introduzione del sistema di controllo.

Test Utilizzatori	**Test Perdite**	Step			
14	2				
Non Controllato		**Controllato**			
Volume finale	**% Vol. Finale**	**Volume finale**	**% Vol. Finale**	**Delta V**	**Delta V %**
503,57	27,80%	673,8	34,88%	170,23	141,04%
611,19	33,75%	663,67	34,35%	52,48	43,48%
696,43	38,45%	594,42	30,77%	-102,01	-84,52%
1811,19		1931,89		120,7	106,66%
Test Utilizzatori	**Test Perdite**	Nota: Rumore di Gauss (Sampled) con utilizzo basso			
1	2				
Non Controllato		**Controllato**			
Volume finale	**% Vol. Finale**	**Volume finale**	**% Vol. Finale**	**Delta V**	**Delta V %**
398,67	23,20%	673,7	35,45%	275,03	151,38%
662,8	38,56%	610,98	32,15%	-51,82	-28,52%
657,19	38,24%	615,667	32,40%	-41,523	-22,85%
1718,66		1900,347		181,687	110,57%

Monitor

Il Monitor si occupa di collezionare i dati provenienti dai sensori e salvarli in un database (nel caso di esempio MongoDB).
Viene utilizzato dall'Orchestrator che richiede le informazioni su Vasche e Settori, dal proxy locale/client MQTT per salvare le misurazioni provenienti dai sensori e dalle REST APIs per fornire i dati agli utenti.

REST APIs

Forniscono l'accesso pubblico ai dati relativi alle misurazioni e alle attuazioni effettuate.
È possibile accedervi all'URL
http://{$HOSTNAME}/rest/monitor-orchestrator-web/webresources/{$PATH}/
dove la variabile *{$PATH}* indica la risorsa di interesse.
Le risorse disponibili sono, in modo simile a quanto visto per i topic MQTT:

- Vasche:
  - /tank/ : mostra le statistiche di tutte le vasche;
  - /tank/{$i} : mostra le statistiche della vasca i-esima;
- Settori:
  - /sector/ : mostra le statistiche di tutti i settori;
  - /sector/{$i} : mostra le statistiche di tutti i settori della vasca i-esima;
  - /sector/{$i}/{$j} : mostra le statistiche del settore j-esimo della vasca i-esima.

Il formato dati utilizzato è JSON.

L'accesso ai dati è, quindi, OpenData-compliant, e si classifica con quattro stelle su cinque nella classificazione 5 ★ Open Data.

I dati delle abitazioni vanno considerati aggregati per settore secondo le normative in materia di privacy, così come virtualizzato su Simulink.

Front-end

Permette di visualizzare graficamente i dati ottenuti tramite le REST APIs.
È possibile eseguire il plot in funzione del tempo delle rilevazioni e delle attuazioni relativi alle vasche o ai settori associati.
È stato implementato sul framework AngularJS 2 e una libreria ad-hoc per il plot dei dati: C3Js.