



Consiglio Nazionale delle Ricerche

IV Accordo di Programma MSE/CNR

CONSUNTIVO PAR 2015

D.7: Smart cities e Smart communities

Relazione Tecnica

A cura del referente scientifico Ottorino Veneri

Dipartimento di Ingegneria,
ICT e Tecnologie per
l'Energia e i Trasporti

1 gennaio 2017
31 dicembre 2017

PAR 2015 – Relazione Tecnico-Scientifica

D.7 SMART CITIES E SMART COMMUNITIES

Progetto: Efficientamento dei processi di produzione e gestione integrata di utenze energivore con fonti rinnovabili e sistemi di accumulo mediante periferiche ICT in un contesto Smart District

Premessa

Il sistema elettrico nazionale sta attraversando una fase di profonda trasformazione in cui la classica architettura gerarchica adottata per la produzione, la distribuzione ed il consumo dell'energia elettrica viene ad essere soppiantata da modelli di tipo distribuito più efficienti e sostenibili volti a favorire la produzione diffusa da fonti rinnovabili, l'integrazione di sistemi di accumulo ed un utilizzo locale dell'energia prodotta. Questo è reso possibile anche grazie al diffondersi di utenti finali sempre più in grado di implementare in tempo reale azioni di Demand Response (DR) per ottimizzare la domanda di energia in base alla disponibilità da fonti non programmabili. Tali trasformazioni stanno facendo emergere con forza il concetto di distretti energetici intelligenti (o Smart Districts), in cui si punta a realizzare un elevato grado di integrazione tra la generazione distribuita dell'energia e differenti categorie di utenti finali, dagli edifici intelligenti ai moderni sistemi di produzione. Gli Smart Districts saranno uno dei componenti fondamentali delle Smart Cities del futuro, visto che contribuiranno a rendere le Smart Cities efficienti dal punto di vista energetico.

Quadro generale di riferimento programmatico del progetto

Dalle considerazioni riportate in premessa si evince che le nuove strategie di gestione energetica necessitano di appositi sistemi di monitoraggio integrato, basati il più possibile su tecnologie open source e plug&play. In quest'ottica, l'integrazione di tecnologie Internet of Things (IoT) è fondamentale, perché permette un monitoraggio continuo ed un'analisi dei profili di consumo energetico. Nell'ottica degli Smart District, le tecnologie IoT non si limitano al monitoraggio del profilo energetico, ma possono essere sfruttate anche per la realizzazione di soluzioni avanzate per la gestione della produzione distribuita dell'energia ed il suo consumo all'interno di uno stesso distretto. Pertanto, si rende necessaria una complessa gestione dei flussi energetici per l'integrazione tra diverse categorie di utenti finali con requisiti energetici e profili di consumo molto differenti, carichi attivi/passivi di diversa tipologia, sistemi di accumulo stazionario/mobile (i veicoli elettrici) dell'energia elettrica ed unità di generazione di piccola taglia dislocate su un territorio molto ampio e caratterizzate da potenza piuttosto ridotte. In questo contesto, emerge e si sviluppa la nuova figura dell'*aggregatore* che fornisce servizi di brokeraggio energetico, al fine di coordinare gli scambi di energia, sia all'interno che all'esterno del distretto energetico, e regolare le interazioni economiche di compravendita dell'energia sul mercato nazionale. Le sfide tecnologiche per la realizzazione di questo scenario vanno affrontate attraverso un approccio trasversale, rispetto ad altri temi di ricerca, ed interdisciplinare che, partendo da un'analisi previsionale nel breve e nel lungo periodo di diffusione di sistemi distribuiti di generazione, accumulo ed utilizzo dell'energia elettrica, e prevedendone diversi schemi di

aggregazione ad elevato livello gerarchico, integri competenze legate a sistemi innovativi di conversione dell'energia elettrica con competenze relative ad architetture di comunicazione e gestione che siano aperte, interoperabili, sicure ed affidabili. Infatti da un lato è necessario sviluppare piattaforme aperte ed interoperabili per comunicazione e la raccolta di dati da parte di dispositivi e macchinari (ad esempio presenti nell'ambiente industriale), sviluppando modelli distribuiti per l'estrazione di conoscenza dai dati raccolti. Dall'altro è necessario predisporre piattaforme di comunicazione che possano favorire l'interazione tra infrastrutture energetiche ed abilitare un mercato peer-to-peer dell'energia elettrica in cui le apparecchiature possono acquistare e vendere energia automaticamente, in base ai criteri definiti dall'utente in modo affidabile e sicuro. A tale scopo l'emergente tecnologia delle blockchain verrà presa in considerazione.

Particolare rilievo assume inoltre lo studio di nuove architetture di potenza che siano capaci di meglio gestire i flussi energetici bidirezionali che caratterizzano il funzionamento di uno Smart District. A riguardo, lo studio di convertitori elettronici di potenza per l'interconnessione di tali architetture alla rete elettrica è un argomento di crescente interesse. È infatti indispensabile lo sviluppo di convertitori e schemi di controllo che siano in grado di ridurre gli attuali requisiti dei filtri di ingresso e nel contempo il contenuto armonico delle correnti, ottimizzando i parametri di power quality.

Lo scopo principale del progetto è quindi lo studio e la realizzazione di sistemi di efficientamento energetico per Smart District all'interno di realtà eterogenee inclusive. In particolare, il progetto si articola su due direttrici principali: (i) lo sviluppo di una piattaforma ICT sicura per il brokeraggio energetico nello Smart District; (ii) lo sviluppo di nuove architetture di potenza per la gestione di flussi energetici bidirezionali in contesti Smart District, con elevati livelli di power quality.

Obiettivo generale del Progetto

L'obiettivo generale del progetto consiste nello sviluppo di piattaforme ICT, basate su tecnologie e standard aperti IoT, per la raccolta/gestione dei dati ed il supporto di servizi decentralizzati per il brokeraggio energetico. Nell'ambito di tale attività sarà sviluppata una architettura elettrica di conversione in corrente continua, integrata con fonti rinnovabili e sistemi di accumulo stazionari, a supporto di utenze energivore in una logica smart district. Sarà infine dato rilievo a soluzioni di interfacciamento di tale architettura con la rete elettrica principale che consentano di ottimizzarne i parametri di power quality.

Elenco dei Partecipanti al Progetto

Unità di Ricerca
Istituto Motori (CNR IM)
CNR Istituto di Informatica e Telematica (CNR-IIT)
Università di Bologna - Dipartimento d'Ingegneria dell'Energia Elettrica e dell'Informazione (UNIBO-DEI)

Partecipazione a progetti su tematiche affini

Correlato al presente PAR è il progetto Europeo H2020-FOF-2016 AUTOWARE (Wireless Autonomous, Reliable and Resilient Production Operation ARchitecture for Cognitive Manufacturing), con durata 30 mesi ed iniziato il 2016-10-01. Nel progetto AUTOWARE lo IIT-CNR sviluppa algoritmi per la gestione dei dati in ambito IoT industriale. Nel progetto di Ricerca di Sistema, lo IIT-CNR ha sviluppato un ambiente sperimentale in cui ha implementato e valutato gli algoritmi definiti, insieme a dei benchmark per il confronto delle prestazioni.

Costi

Costo totale del progetto: € 200.000,00

Il dettaglio dei costi è riportato nelle tabelle di rendicontazione.

Linea progettuale 1: SVILUPPO DI PIATTAFORME ICT PER LA RACCOLTA, GESTIONE ED ANALISI DISTRIBUITA DI DATI TRAMITE TECNOLOGIE IOT IN AMBIENTI SMART DISTRICTS, E PER IL SUPPORTO DI SERVIZI DI BROKERAGGIO ENERGETICO SICURI ED AFFIDABILI.

Elenco dei Partecipanti alle attività

Unità di Ricerca	Attività
Istituto di Informatica e Telematica (CNR-IIT)	1.1

Stato dell'arte

Il termine “Internet of Things” (IoT in breve) si riferisce al paradigma di rete per cui un numero potenzialmente illimitato di oggetti intelligenti, cioè dotati di tecnologie per comunicare, raccogliere ed elaborare informazioni, sono univocamente indirizzabili e possono essere interconnessi tra i loro a formare una rete di dispositivi che cooperano in modo autonomo per eseguire funzioni predefinite [1][2]. La realizzazione della visione dell'IoT necessita una significativa evoluzione delle esistenti infrastrutture di servizi e di rete. Proprio l'estrema diversità di dispositivi, tecnologie ed applicazioni rende particolarmente critica per i sistemi IoT la compatibilità ed interoperabilità tra piattaforme distinte. L'approccio tradizionale che è stato seguito nella progettazione di soluzioni per reti di sensori, precursori delle reti IoT, e cioè di progettare sistemi altamente specializzati per l'esecuzione di una specifica funzione ed utilizzando il più delle volte tecnologie proprietarie, chiaramente non è capace di soddisfare i requisiti dei sistemi IoT. Non si tratta solo di armonizzare dati che provengono da sorgenti diverse, ma anche di garantire che sensori dotati di funzionalità simili, ma diversi in termini di tecnologie trasmissive, sistemi operativi e capacità di elaborazione siano in grado di comunicare in modo trasparente ed autonomo. La soluzione più comunemente adottata per risolvere i problemi di interoperabilità e scalabilità è di seguire un approccio orizzontale nella progettazione del sistema IoT in cui un'unica piattaforma comune gestisce i servizi di rete ed i servizi necessari alle applicazioni per interagire con i dispositivi IoT. Quindi, per garantire l'interoperabilità tra reti IoT distinte è sufficiente sviluppare opportuni connettori tra le diverse tecnologie di comunicazione per reti IoT e la piattaforma orizzontale di rete. Questo è l'approccio che è stato seguito anche in questa linea progettuale per sviluppare i diversi componenti della piattaforma ICT proposta per implementare la raccolta, gestione ed analisi distribuita di dati IoT. Come riportato di seguito, la piattaforma di rete utilizzata come riferimento è quella standardizzata negli ultimi anni dall'IETF, che permette di utilizzare protocolli di comunicazione Internet in sistemi IoT [3].

Bibliografia

- [1] Eleonora Borgia, “The Internet of Things vision: Key features, applications and open issues”, in Computer Communications 54: 1-31 (2014).
- [2] L. Atzori, A. Iera, G. Morabito, The internet of things: a survey, Comp. Netw.: 2599 Int. J. Comp. Telecommun. Network. 54 (15) (2010) 2787–2805.

- [3] Maria Rita Palattella, Nicola Accettura, Xavier Vilajosana, Thomas Watteyne, Luigi Alfredo Grieco, Gennaro Boggia, Mischa Dohler, "Standardized Protocol Stack for the Internet of (Important) Things". IEEE Communications Surveys and Tutorials, 15(3): 1389-1406 (2013).

Obiettivi della linea

L'obiettivo generale del progetto consiste nello sviluppo di piattaforme ICT, basate su tecnologie e standard aperti IoT, per la raccolta/gestione dei dati ed il supporto di servizi decentralizzati per il brokeraggio energetico. In particolare, nel POA 2015 l'attenzione sarà focalizzata sulla progettazione di gateway IoT in grado di operare da proxy per le risorse ospitate dai dispositivi IoT, e di un Data Management Layer (DML), in grado di soddisfare stringenti requisiti di ritardo del trasferimento dati da sensori ad attuatori.

Le attività di questa linea proseguiranno nelle successive annualità del Piano Triennale (POA 2016 2017).

La linea progettuale 1 è stata suddivisa nelle seguenti attività:

1.1: Sviluppo di piattaforme ICT per la raccolta, gestione ed analisi distribuita di dati tramite tecnologie IoT in ambienti Smart Districts (CNR-IIT).

Unità di ricerca coinvolte:

- Istituto di Informatica e Telematica (CNR-IIT)

Attività 1.1 Sviluppo di piattaforme ICT per la raccolta, gestione ed analisi distribuita di dati tramite tecnologie IoT in ambienti Smart Districts.

Obiettivi

L'obiettivo di questa attività consiste nello sviluppo di alcune delle componenti necessarie per realizzare una piattaforma ICT per la raccolta, la gestione ed il supporto all'analisi distribuita di dati provenienti da sensori e dispositivi IoT. In particolare, l'obiettivo specifico delle attività nel POA 2015 è di definire e sviluppare quelle componenti per permettere alle applicazioni di scoprire l'insieme delle risorse (servizi) ospitate dai nodi della rete IoT e di ricevere aggiornamenti periodici sullo stato di tali risorse, anche con stringenti requisiti di ritardo del trasferimento dati. Prototipi dimostrativi sono sviluppati e testati in condizioni di laboratorio ed in ambienti sperimentali di larga scala, al fine di validare la fattibilità delle soluzioni tecnologiche proposte e valutare sperimentalmente le prestazioni.

Risultati conseguiti

I risultati conseguiti per questa attività, sono principalmente tre.

Il **primo risultato** è stato una analisi dettagliata ed esaustiva dei protocolli Internet che sono utilizzabili in ambito IoT al fine di valutare quali funzionalità sono presenti negli standard esistenti per gestire

servizi di raccolta, gestione ed analisi distribuita di dati, ed identificare le estensioni che sono necessarie per migliorarne l'efficienza e la flessibilità di utilizzo di tali servizi.

Il **secondo risultato** è stato la realizzazione di un prototipo di gateway IoT, che offre il supporto del service discovery sia a reti esterne che ai dispositivi IoT ad esso connessi utilizzando le funzionalità standard offerte da COAP.

Il **terzo risultato** è stato lo sviluppo di un ambiente sperimentale per la valutazione di algoritmi di gestione dei dati per applicazioni IoT critiche, e cioè con requisiti stringenti di latenza nel trasferimento dei dati. Un esempio di applicazioni IoT critiche sono rappresentate dal controllo di processi industriali o dei flussi energetici in una microgrid.

Si riporta di seguito una descrizione dettagliata dei risultati.

Lo Stack Protocollore IETF per le reti IoT

In questa sezione è fornita una breve descrizione dei principali protocolli di comunicazione che sono stati standardizzati dall'IETF per sistemi IoT, in quanto saranno gli standard di riferimento per la realizzazione del prototipo della piattaforma ICT, obiettivo dell'attività 1.1 del POA 2015, per la raccolta, la gestione ed il supporto all'analisi distribuita di dati IoT.

Architettura dello Stack Protocollore IETF per reti IoT

Nella seguente figura viene fornita una rappresentazione dell'architettura tipica dello stack protocollore IETF per reti IoT.

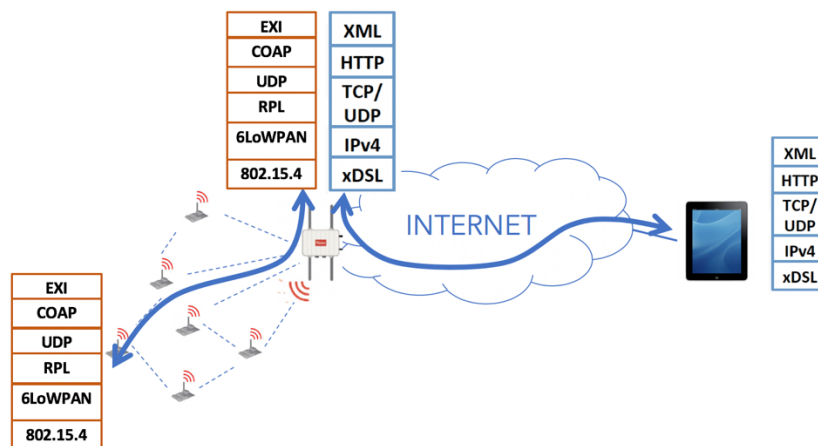


Figura 1: Stack protocollore IETF per reti IoT.

Storicamente la rete Internet ha adottato una struttura a livelli (*layer*) gerarchica per l'organizzazione della pila di protocolli: ogni livello esegue una specifica serie di operazioni che riguardano la trasmissione di dati e fornisce un ben definito servizio ai livelli più alti; ad ogni livello, ci si avvicina sempre più dall'interfaccia utente (quella con cui interagiamo) all'interfaccia di rete. Attualmente la struttura base della suite TCP/IP consiste di cinque distinti livelli, partendo dall'alto:

- i. Livello di applicazione: Esso è la sede delle applicazioni di rete (e.g., HTTP, SMTP, FTP, DNS) e specifica il significato dei messaggi applicativi che vengono scambiati tra le applicazioni
- ii. Livello di trasporto: Esso fornisce un servizio di trasferimento trasparente dei messaggi delle applicazioni sulla rete. Deve quindi effettuare la frammentazione dei dati, la correzione

degli errori e la prevenzione della congestione della rete. I pacchetti sono denominati segmenti.

iii. Livello di rete: Esso si occupa di gestire interconnessione dei nodi della rete Internet, cioè di trasferire i segmenti da un'origine ad una destinazione. Il protocollo principale di questo livello è l'Internet Protocol (IP) che fornisce il servizio di l'indirizzamento dei nodi di rete e l'instradamento dei pacchetti, denominati datagrammi.

iv. Livello di collegamento: Esso gestisce la trasmissione dei datagrammi da un nodo a quello successivo sui singoli link di collegamento fisico che costituiscono il percorso di rete.

v. Livello fisico: Esso gestisce il trasferimento dei singoli bit di informazione.

L'IETF ha quindi creato una serie di gruppi di lavoro (WGs) che si occupano di definire gli equivalenti dei protocolli principali della suite TCP/IP per reti di comunicazione costituite da dispositivi a bassa potenza (tipicamente alimentati a batteria). Essendo necessario ridurre al minimo l'energia consumata per la trasmissione di messaggi in genere le reti IoT si affidano a livelli fisici che garantiscono solo basse velocità di trasferimento dei dati (*low data rates*) e limitata affidabilità dei collegamenti fisici (*lossy links*). Per questo motivo, si parla comunemente di **Low Power and Lossy Networks (LLN)**, termine che ha progressivamente sostituito il più datato concetto di Wireless Personal Area Network, (WPAN), con cui si indicano le reti la cui copertura radio è limitata nello spazio nell'ordine di qualche metro. I protocolli più importanti sono [3]:

- **EXI** (*Efficient XML Interchange*): rappresentazione molto compatta delle informazioni XML tramite un formato binario di scambio.
- **COAP** (*COstrained Application Protocol*): protocollo leggero web di scambio messaggi.
- **RPL** (*Routing Protocol for Low power and lossy network*): protocollo di routing per reti di sensori e dispositivi IoT [12].
- **6LoWPAN** (*IPv6 over Low-power Wireless Personal Area Networks*): livello di adattamento di datagrammi IP su reti L-WPAN.

Come evidenziato in Figura 1, nel modello di comunicazione delle reti IoT svolge un ruolo fondamentale il gateway IoT, che agisce da intermediario fra la rete IoT e le altre reti e funge da punto di accesso alle reti esterne per i dispositivi IoT. In particolare il gateway IoT effettua le operazioni di traduzione e scambio di messaggi fra lo stack protocollare delle reti che interconnette e del relativo instradamento dei messaggi. Come riportato in seguito il gateway può fornire una serie di servizi aggiuntivi, quali ad esempio un supporto alla memorizzazione dei dati ricevuti dai dispositivi ad esso collegati, o alla scoperta dei servizi applicativi che sono offerti dai dispositivi IoT.

6LoWPAN (IPv6 over Low power WPAN)

Lo standard IEEE 802.15.4 è stato concepito per regolamentare il livello fisico ed il livello MAC in reti LLN. Ad esempio, le tecnologie Zigbee, WirelessHART, e MiWi adottano questo standard per trasmettere i pacchetti sui link fisici. Dato che il protocollo IPv6 richiede che le dimensioni dei pacchetti siano maggiori rispetto al più grande frame del protocollo IEEE 802.15.4, uno strato di adattamento è stato proposto dall'IETF per la convergenza tra reti di sensori ed IPv6, che prende il nome di **6LoWPAN**. Di seguito vengono riportate le principali discrepanze fra il protocollo IPv6 ed IEEE 802.15.4: i) il protocollo IPv6 richiede che il suo MTU (Maximum Transmission Unit) sia di almeno 1280 byte, mentre il protocollo 802.15.4 richiede che la massima dimensione del frame sia 127 byte; e ii) gli indirizzi IPv6 hanno una lunghezza fissa di 128 bit, mentre lo standard IEEE 802.15.4 definisce diversi tipi di indirizzamento e permette di utilizzare indirizzi a 64 bit oppure indirizzi a 16 bit unici all'interno di una PAN.

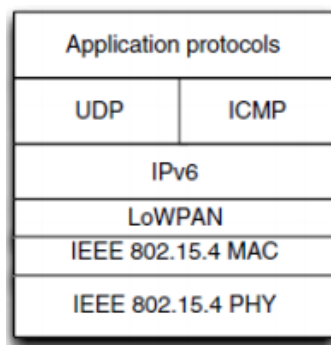


Figura 2: Stack protocollare 6LoWPAN.

6LoWPAN, il cui stack protocollare è presente in Figura 2, definisce i seguenti meccanismi per adattare il protocollo IPv6 al protocollo IEEE 802.15.4: i) la compressione efficiente dell'header per rendere fattibile la trasmissione di pacchetti IPv6 sul protocollo IEEE 802.15.4, ii) la frammentazione del pacchetto IPv6 per IEEE 802.15.4, e iii) una procedura ottimizzata per il neighbor discovery.

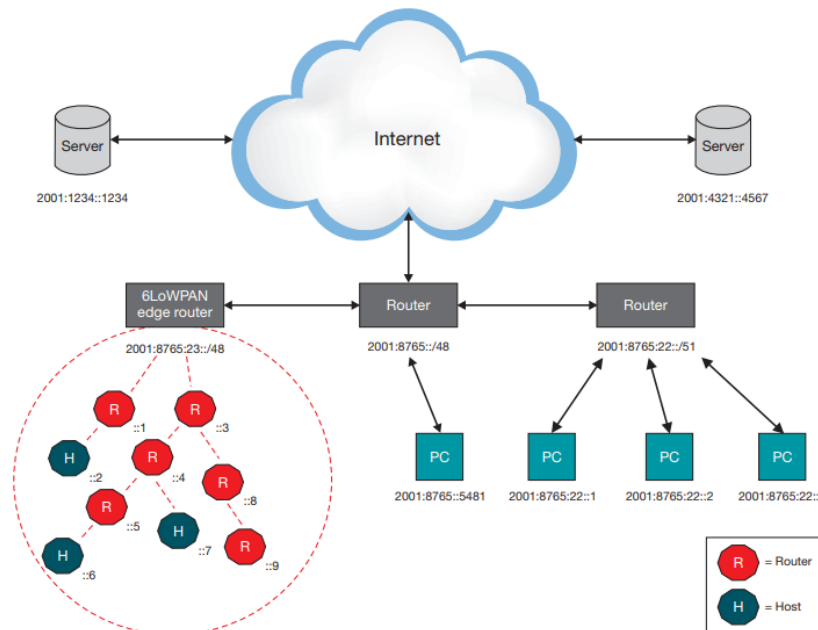


Figura 3: Architettura di una rete 6LoWPAN.

L'architettura di una rete 6LoWPAN è mostrata in Figura 3. Si può notare che diversi tipologie di nodi sono definiti in questa architettura: a) host (H), cioè nodi che inviano pacchetti (ad esempio un nodo sensore che invia periodicamente misure delle ambientali), b) 6LoWPAN Router (R), cioè nodi che effettuano solamente operazioni di routing, e c) il 6LoWPAN **Edge Router**, che consente gli host della rete 6LoWPAN di comunicare con i nodi IPv6 fuori dalla rete 6LoWPAN. In questo modo, i nodi IPv6 fuori dalla rete 6LoWPAN possono inviare e ricevere pacchetti a nodi presenti all'interno della rete Internet.

Uno dei meccanismi più importanti di 6LoWPAN è la compressione dell'header di IPv6. L'header del protocollo IPv6 ha una dimensione fissa pari a 40 byte e alcuni dei campi, mostrati in Figura 4 (parte alta), come versione, traffico della classe e così via, hanno valore tipicamente statico o ridondante. Inoltre, grossa parte dell'header è costituita dagli indirizzi di sorgente e destinazione i quali possono essere, in buona parte, dedotti al livello più basso nello stack protocollare. Per tale motivo, è stato definito il cosiddetto **IP Header Compression (IPHC)** che permette di ridurre drasticamente la dimensione dell'header di IPv6.

L'IPHC si basa sul contesto condiviso fra i nodi: essendo i nodi facenti parte della stessa 6LoWPAN, è possibile comprimere l'indirizzo IPv6 di destinazione di dimensione 128 bit non usando addirittura nessun bit, perché grazie al prefisso della 6LoWPAN e all'indirizzo MAC del dispositivo, è possibile ricostruire l'intero indirizzo IPv6. In questo modo viene anche risolto il problema dell'assegnazione degli indirizzi IPv6.

In Figura 4 (parte bassa), è riportato l'header di un pacchetto 6LoWPAN dopo l'IPHC sia nel caso il cui in destinatario è un indirizzo link-local (la dimensione dell'header è solamente 2 byte), sia nel caso in cui l'indirizzo non è link local e quindi il pacchetto necessita di più hop per essere consegnato.

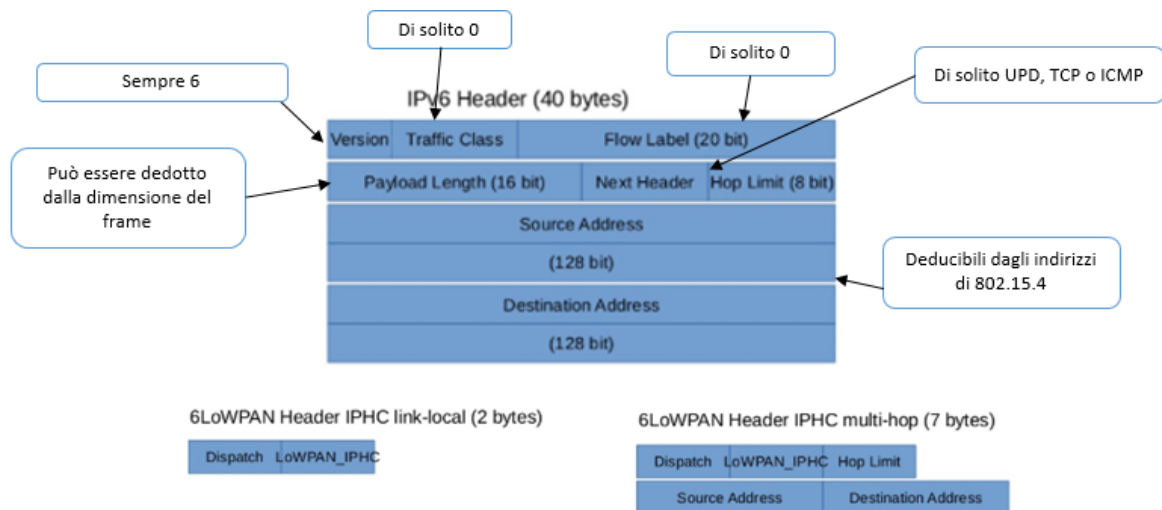


Figura 4: Header IPv6 dopo IPHC

Per quanto riguarda il routing, lo standard IEEE 802.15.4 definisce una strategia semplificata, che prevede che ogni pacchetto perso venga rinviato da tutti i nodi che ne attraversano il percorso. Nel caso di reti di dimensioni medie-grandi un elevato numero di pacchetti può essere ritrasmesso per ogni pacchetto perso. Inoltre, in reti LLN i link sono instabili e possono essere persi molteplici pacchetti. Se il numero dei pacchetti persi aumenta, aumenta in maniera vertiginosa anche il numero di rinvii, spingendo la rete ad una situazione di saturazione. Per tale motivo, nelle reti LLN si preferisce adottare un protocollo di routing a livello di rete, quale RPL (vedi paragrafo 3.3 per ulteriori dettagli).

Visto che il payload di un pacchetto IPv6 potrebbe essere più grande del massimo payload trasportabile da un frame 802.15.4, allora anche la frammentazione e il riassettaggio dei pacchetti è necessario, visto che, nel caso peggiore, solamente 88 byte possono essere inviati come payload del messaggio. Quindi all'interno del frame 802.15.4, oltre all'header compresso sarà quindi anche

presente l'informazione relativa alla frammentazione del pacchetto. Nella *Figura 5* è mostrato quanto precedentemente descritto.

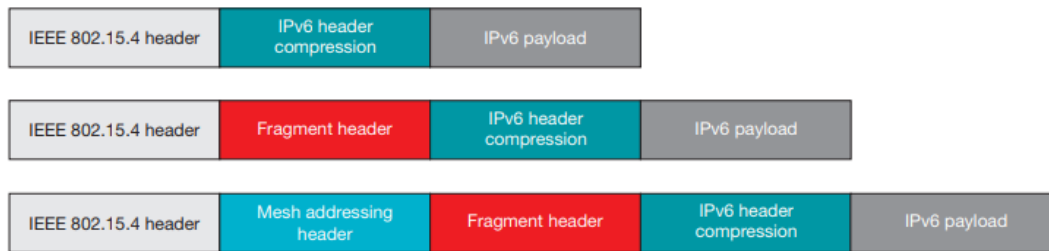


Figura 5: Frammentazione pacchetto IPv6 in 6LoWPAN

Altro meccanismo presente in 6LoWPAN per l'adattamento a 802.15.4 è la procedura di neighbor discovery. Di seguito vengono elencate tutte le ottimizzazioni introdotte rispetto al protocollo di neighbor discovery dello stack protocollare IPv6:

1. non viene effettuata la cosiddetta **DAD (Duplicate Address Detection)**, ovvero la verifica di indirizzi duplicati all'interno della rete perché in una rete 6LoWPAN si può ottimisticamente supporre che ai nodi è (auto)assegnato un indirizzo diverso l'uno dall'altro.
2. Non vengono inviati messaggi di **Router Advertisement (RA)** in maniera periodica dai router agli host: questo risulterebbe in una continua sollecitazione ai nodi che potrebbero essere in stato di sleeping e quindi non sono andando a consumare loro batteria, ma aumentando il traffico all'interno della rete.
3. Un nuovo meccanismo per la propagazione dei prefissi ma allo stesso tempo per il controllo del flusso degli stessi messaggi: è utilizzato un campo specifico chiamato **Authoritative Border router Option (ABRO)** all'interno dei messaggi di RA. In questo modo, una volta che il BR annuncia il proprio prefisso, i restanti router presenti nella 6LoWPAN non possono fare altro che propagare l'indirizzo ricevuto agli altri nodi. Dopo che tutti i nodi avranno ricevuto tale messaggio, possiederanno lo stesso prefisso.

Routing Over Low power and Lossy networks (ROLL)

Come è stato osservato nelle sezioni precedenti, i link fisici di una rete LLN sono caratterizzati da un alto tasso di perdita di pacchetti, velocità di trasmissione dei dati limitata (nell'ordine della decina di Kilobyte al secondo), ed instabilità dovuta alle caratteristiche non ideali del canale di comunicazione. Un'altra caratteristica delle LLN è che il traffico nella maggior parte dei casi risulta essere da un singolo punto a più punti e viceversa (*many-to-one* e *one-to-many*).

Per ovviare ai problemi presenti in reti LLN, è stato progettato e realizzato un protocollo di routing capace di inviare e inoltrare i pacchetti indipendentemente dallo stato della LLN. Questo protocollo di routing prende il nome di **RPL (Routing Protocol for Low Power And Lossy Network)** e utilizza esclusivamente indirizzi IPv6 [12].

I criteri secondo cui RPL inoltra e invia i pacchetti (chiamato **Objective Function (OF)**) possono essere diversi e anche contrastanti fra di essi: utilizzare il link con minore latenza, non utilizzare nodi la cui batteria è al di sotto di una certa soglia, utilizzare un link con il minore tasso di perdita di pacchetti e così via. Quindi, una singola LLN può avere contemporaneamente diverse istanze di RPL all'interno della stessa, tante quante sono le OF scelte per instradare i diversi tipi di traffico presenti nella LLN. La scelta della OF dipende tipicamente dagli scopi dell'applicazione finale: ad esempio, in un sistema in cui si vuole mitigare il consumo energetico dei nodi, verrà scelta un OF in cui pacchetti seguiranno un percorso che tende ad evitare i nodi la cui batteria ha poca carica residua.

La formazione della topologia di una LLN attraverso RPL è regolata attraverso l'algoritmo **Trickle** che permette la disseminazione automatica e controllata all'interno della rete LLN di pacchetti per la formazione della stessa. Questo algoritmo consente, in linea generale, di formare la cosiddetta **DODAG (Destination Oriented Direct Acyclic Graph)**, ovvero un grafo aciclico i cui pacchetti convergono in un unico punto, il **Border Router (BR)**.

Come mostrato in Figura 6, la formazione di una DODAG è divisa nelle seguenti fasi:

- a) Il BR (nodo 1 in Figura 6) annuncia la propria presenza agli altri nodi inviando in multicast un messaggio **DIO (DODAG Information Object)**. In questo pacchetto è presente, fra tutte le informazioni, anche quelle relative al prefisso del BR che viene utilizzando dai restanti nodi per l'assegnazione di un indirizzo IPv6.
- b) I nodi che hanno ricevuto il messaggio DIO dal BR (i nodi 2 e 3 nella figura) selezionano il BR come il nodo parente nel grafo, e continuano a propagare le informazioni ricevute ai nodi circostanti.
- c) Per creare il *reverse path*, cioè il percorso di instradamento di pacchetti dati destinati dal BR ai vari nodi della rete, i nodi inviano un messaggio **DAO (Destination Advertisement Object)** al BR che viene instradato dai vari nodi parente che sono sul percorso.

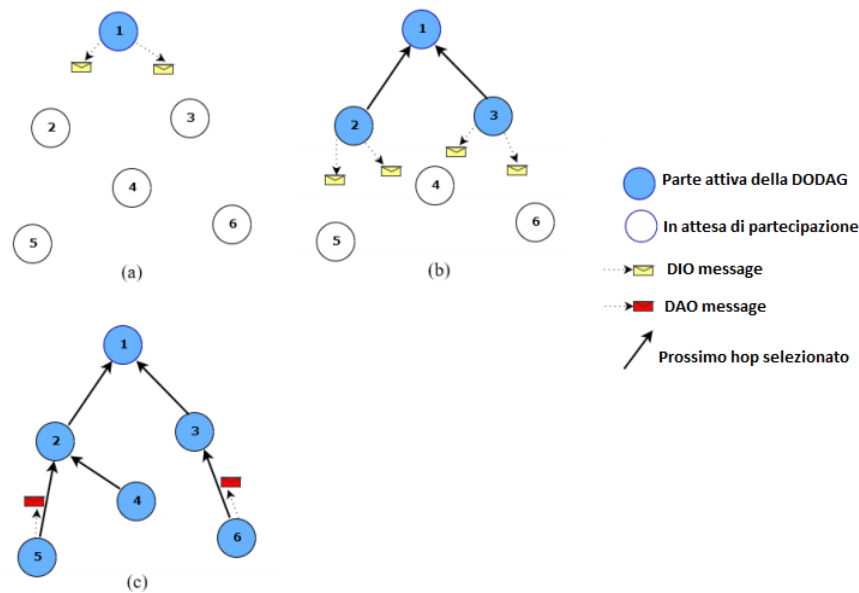


Figura 6: Formazione di una DODAG

Per ridurre il livello di congestione nella rete dovuta alla trasmissione di messaggio DIO e DAO, l’algoritmo Trickle limita il numero di messaggi che vengono instradati assicurando che vengano ritrasmessi solo messaggi che contengono nuove informazioni sulla topologia (ad esempio cambi di qualità di un link, nodo che si è spostato, etc.) La durata della formazione della DODAG dipende da diversi fattori: numero di nodi presenti all’interno della LLN, parametri di Trickle e disposizione fisica dei nodi stessi. In uno scenario semplice in cui sono presenti un numero limitato di nodi, il tempo di formazione è quasi immediato. Se invece, la rete contiene un numero elevato di nodi la formazione della DODAG richiederebbe più tempo. Nella fase della costruzione della DODAG ogni nodo utilizza i messaggi DIO per crearsi una lista dei nodi vicini e dei nodi che possono fungere da parent, cioè da next hop per raggiungere il BR. Nella lista dei preferred parent viene selezionato un solo **preferred parent**, cioè il nodo vicino da utilizzare per spedire traffico verso il BR. Normalmente un nodo invia i pacchetti al preferred parent. Se questo non fosse disponibile, allora sceglierà uno degli altri parent.

Dopo che la DODAG è stata formata, ogni nodo possiede un **rank**, calcolato dalla OF, ovvero uno scalare che rappresenta la posizione all’interno della DODAG. Minore è il rank del nodo, più vicino è questo, da un punto di vista del routing, al BR della LLN e migliore risulta essere, secondo la OF, quel nodo. Secondo le specifiche del protocollo RPL, il rank dei nodi deve essere monotonicamente decrescente. In questo modo si evitano teoricamente dei loop, dato che i pacchetti vengono inviati ed inoltrati al nodo il cui rank è monotonicamente minore. Questo, però, non garantisce la totale assenza di loop: causa, ad esempio, un’alta dinamicità dei nodi della LLN, è possibile che un nodo riceva un pacchetto da un altro nodo il cui rank è minore. Questo viene rilevato semplicemente confrontando il rank del mittente e del destinatario: se il destinatario ha un rank che è maggiore rispetto al mittente, allora si è in presenza di un loop. Col fine di interrompere il loop, RPL ricalcola la DODAG attivando una procedura di riparazione della topologia, che può essere locale (solo i nodi vicini sono interessati) o globale.

Nelle reti dinamiche in cui i nodi possono essere mobili le informazioni riguardanti lo stato dei collegamenti diventa fondamentale: l’avvicinarsi o l’allontanarsi di un nodo rispetto ad un altro fa sì che la qualità del collegamento fra questi possa migliorare o peggiorare. In questo modo, i nodi circostanti restano aggiornati sullo stato della rete di cui fanno parte potendo così cambiare, se necessario, destinazione a cui inviare i pacchetti. In questo modo cambia la versione della DODAG, seppur utilizzando la stessa OF come mostrato in Figura 7.

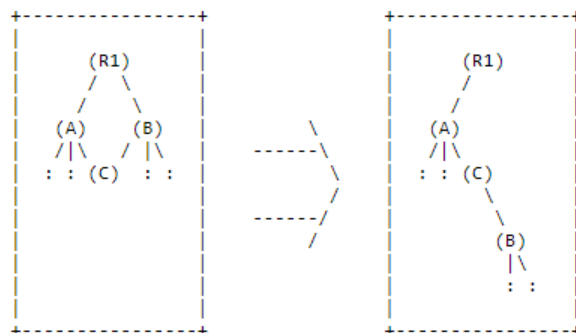


Figura 7- Cambio versione DODAG

Infine un DODAG può operare in due modalità: **storing mode** e **non-storing mode**. Nel primo caso ogni nodo conosce la lista di tutti gli hop necessari per consegnare il pacchetto a destinazione. Nel secondo caso, invece, il nodo conosce solamente il prossimo hop a cui inviare il pacchetto e la conoscenza completa della topologia della rete è centralizzata nel BR. Inoltre, nello storing mode ogni nodo ha come prefisso quello del BR, mentre nello non-storing mode ogni nodo ha un proprio prefisso.

La scelta di operare in storing mode o non-storing mode dipende dalla capacità di memoria dei nodi: se questi non hanno a sufficienza per memorizzare l'intero percorso fino al BR, allora risulterebbe controproducente utilizzare lo storing mode.

Constrained Application Protocol (CoAP)

CoAP è un protocollo per scambio di messaggi applicativi utilizzato nelle LLN [10]. In particolare, CoAP fornisce un **modello in interazione fra le applicazioni web di tipo request/response**, basato sul modello **RESTful**, oltre a supportare la funzionalità di discovery dei servizi e delle risorse (vedi il Capitolo 0 per ulteriori dettagli). Nel modello RESTful, le entità fisica o logica espongono delle risorse con cui è possibile interagire in modo standard. Ad esempio, in un sistema di controllo dell'illuminazione (tipico scenario applicativo dell'IoT), in cui un controllore deve inviare comandi per l'accensione e regolazione di una lampada smart, sarà sufficiente che la lampada smart esponga al controllore una risorsa CoAP per il controllo dell'accensione dell'illuminazione attraverso comandi standard.

CoAP può essere considerato la versione leggera di http in quanto utilizza gli stessi metodi GET, PUT, POST e DELETE rispettivamente per ottenere, modificare, creare e cancellare una risorsa che risiede ad uno specifico **URI (Uniform Resource Identifier)**. È importante sottolineare che il concetto di risorsa è assolutamente generale ed astratto. In tal senso, una richiesta CoAP può essere considerata equivalente ad una richiesta HTTP. CoAP, però, a differenza di HTTP, non si basa su un protocollo di trasporto affidabile (TCP), ma bensì sul protocollo UDP. Per questa ragione CoAP, a differenza di HTTP, prevede anche dei meccanismi di ritrasmissione di alcuni tipi di messaggi. In particolare, CoAP definisce quattro diversi tipi di messaggi:

- **CON** (Confirmable): messaggi che necessitano la conferma di ricezione
- **ACK** (Acknowledgement): utilizzato per confermare la ricezione dell'avvenuta consegna
 - **NON** (non confirmable): messaggio che non necessita la conferma di ricezione
 - **RST** (messaggio di reset): messaggio utilizzato quando come risposta non è disponibile nemmeno il messaggio di errore

CoAP, quindi, utilizza un approccio a due strati per gestire la trasmissione dei messaggi come mostrato in Figura 8. Lo strato inferiore gestisce la trasmissione asincrona dei messaggi attraverso i datagrammi UDP, ed uno strato superiore che gestisce le interazioni di tipo request/response ed assegna una semantica ad ogni messaggio sulla base di specifici codici di identificazione inseriti nell'header CoAP.

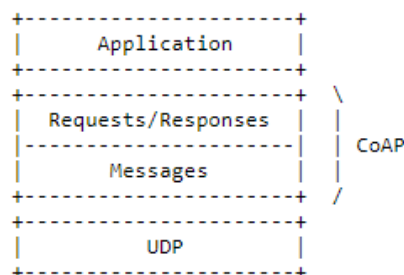


Figura 8 - Stratificazione dei messaggi CoAP

Un tipico esempio dell'utilizzo dei messaggi CON è quello mostrato in Figura 9. Un CoAP Client effettua una richiesta di GET all'URI */temperature* del CoAP Server. La risposta è un ACK che conterrà il valore richiesto (parte sinistra della Figura 9) oppure un codice ed un messaggio di errore (parte destra della Figura 8). In questo modo, viene implementato da CoAP stesso il meccanismo di consegna affidabile, non presente in UDP. Col fine di associare un messaggio CON con il relativo ACK è utilizzato il campo **messageID**, presente all'interno dell'header del messaggio. Questo è utilizzato anche per rilevare possibili messaggi duplicati all'interno della rete.

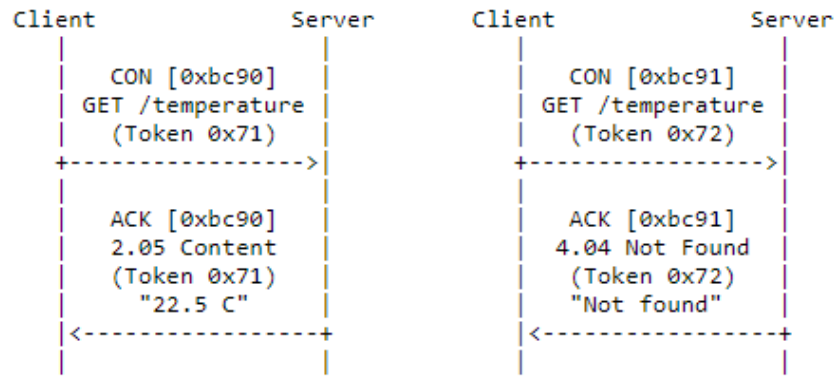


Figura 9 - Invio messaggio CON e relativo ACK

In alcuni casi, però, la risposta del server potrebbe richiedere una certa latenza per essere elaborata. Per tale motivo, il server può anche inviare solo la conferma di avvenuta ricezione della richiesta attraverso un ACK e successivamente, un messaggio CON contenente il contenuto dopo l'avvenuta elaborazione. Nel caso mostrato nella *Figura 10*, la risposta sarà un messaggio CON che dovrà essere confermato con il relativo ACK. Differentemente dal *messageID*, per associare la richiesta con una o più risposte, viene utilizzato il cosiddetto **Token**.

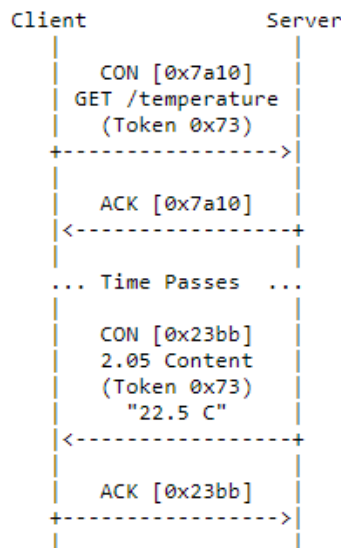


Figura 10 - Richiesta CoAP che richiede una certa latenza

Oltre ai CON e agli ACK, CoAP definisce altri due tipi di messaggi: i messaggi NON ed i messaggi RST. Il primo è un messaggio che non necessita di ACK per confermare l'avvenuta ricezione (*Figura 11*), mentre il secondo viene utilizzato per notificare che il server non è capace di fornire nemmeno un messaggio di errore ad un messaggio CON.

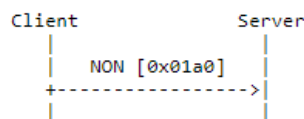


Figura 11 - Invio messaggio NON

Una particolare richiesta CoAP può avere più di una risposta, come nel caso delle richieste multicast o nelle richieste di Observe [11]. L'**Observe** è un meccanismo di sottoscrizione dei messaggi presente all'interno del protocollo CoAP ed è utilizzato per trasmettere in modo automatico al client aggiornamenti dello stato di una specifica risorsa. Senza l'Observe, il client dovrebbe effettuare richieste continue al server sullo stato delle risorse, andando sia ad aumentare il traffico all'interno della rete che a ridurre la durata di vita della batteria del nodo stesso a causa delle numerose richieste inviate. Per tale ragione, è stato progettato e implementato il meccanismo di Observe all'interno di CoAP.

Come mostrato in Figura 12, il client prima di tutto effettua una richiesta di Observe ad uno specifico Server, e successivamente questo invierà le notifiche al client quando lo stato della risorsa è cambiato: è il server a decidere quando considerare che lo stato della risorsa sia cambiato: ad esempio, può decidere se inviare la misura della temperatura ogni qualvolta supera una certa soglia oppure inviarla periodicamente.

Il flusso di notifiche inviato dal server al client può essere interrotto se il client richiede esplicitamente di non ricevere più notifiche attraverso un messaggio RST, oppure quando il client rifiuta una notifica CON inviata dal server. In questo, caso il server rimuove il client dalla sua lista degli observer.

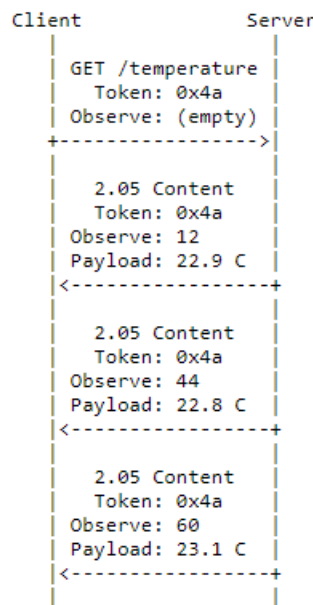


Figura 12 - Meccanismo di Observe in CoAP

CoAP Proxying and Caching

Un'interessante caratteristica del protocollo CoAP è il supporto nativo all'uso dei proxy, ossia entità intermedie tra un endpoint sorgente ed uno destinatario. Si distinguono due categorie di proxy CoAP che prendono il nome di proxy diretto (**forward-proxy**) e proxy inverso (**reverse-proxy**). Il primo viene direttamente interpellato da un CoAP client e ha lo scopo di inviare richieste CoAP al posto di quest'ultimo; il secondo, invece, risponde a richieste CoAP al posto di uno o più CoAP server. L'approccio più comune per implementare un proxy inverso è quello di consentire al proxy di esporre le risorse disponibili su un server, come se fossero risorse proprie. In questo modo, attraverso opportuni meccanismi di traduzione degli URI, è possibile fare in modo che i client non siano a conoscenza del fatto che stanno comunicando attraverso un proxy, piuttosto che direttamente con il server. In questo modo il proxy può essere utilizzato per tradurre richiesta HTTP in richieste COAP e viceversa.

Essendo un intermediario, il proxy, inoltre, rappresenta un elemento ideale su cui implementare meccanismi di caching delle informazioni contenute sui sensori. Infatti, se uno o più dispositivi effettuano la stessa richiesta allo stesso server, il proxy può utilizzare una risposta recente come risposta a questi nodi senza dover inviare richiesta diretta al server stesso.

Strategie per il Link binding in CoAP

In CoAP, per *Link Binding* si intende l'insieme di regole che definiscono il meccanismo con cui due endpoint mantengono sincronizzato il contenuto tra due risorse, definite risorsa sorgente e risorsa destinataria. In particolare, CoAP supporta tre metodi di binding: Polling, Observe e Push.

Il meccanismo di *polling* consiste nell'inviare richieste GET periodiche verso la risorsa sorgente, per poi copiare nella risorsa destinataria la rappresentazione della risorsa sorgente ricevuta nel pacchetto di risposta. Il meccanismo di *observing* (introdotto nel capitolo 0) rappresenta un metodo di sincronizzazione più raffinato, in cui, la risorsa destinataria registra il suo interesse nell'essere aggiornata con la rappresentazione della risorsa sorgente, quando questa varia. In questo caso è il CoAP endpoint che gestisce la risorsa sorgente ad inviare, tramite appositi messaggi detti notifiche, la copia più aggiornata della risorsa all' endpoint destinatario. Il *polling* e l'*observing*, rappresentano, quindi, dei meccanismi per attuare il *sensing* di una risorsa, ossia per recuperarne lo stato attuale.

Il meccanismo di *push* si basa sull'invio di richieste di PUT da parte dell'endpoint sorgente verso la risorsa destinataria. Esso, a differenza dei precedenti, si può configurare come un meccanismo di *attuazione*, in quanto la modifica dello stato di una risorsa remota può essere associata all'esecuzione di un comando fisico. Infatti, prendendo nuovamente ad esempio un sistema di illuminazione, il controllore può tramite una PUT settare su "OFF" il valore della risorsa esposta dalla lampada smart, a cui seguirà l'azione fisica di quest'ultima di spegnimento della lampada.

Il meccanismo di binding può essere reso più fine attraverso l'uso di attributi che rispecchiano delle condizioni che devono verificarsi affinché l'azione oggetto del binding abbia effetto. L'elenco degli attributi, così come il loro effetto è tutt'ora oggetto attivo di ricerca. Nella seguente tabella vengono riportati alcuni esempi di parametri che (ad esclusione del primo) vengono suggeriti in [13].

Attributo	Simbolo	Interpretazione
Periodo	p	Rappresenta un periodo predefinito, in secondi, che deve intercorrere tra due eventi di sincronizzazione consecutivi (equivalente a $p_{min}=p_{max}$).
Periodo minimo (s)	p_{min}	Rappresenta il periodo minimo, in secondi, che deve intercorrere tra due eventi di sincronizzazione consecutivi.
Periodo massimo (s)	p_{max}	Rappresenta il periodo massimo che può intercorrere tra due eventi di sincronizzazione consecutivi.
Step di cambiamento	st	Se presente, indica di quante unità dovrebbe variare lo stato della risorsa prima di generare un evento di sincronizzazione.
Maggiore di	gt	Se presente, indica quale soglia superiore dovrebbe superare lo stato della risorsa prima di generare un evento di sincronizzazione.
Minore di	lt	Se presente, indica quale soglia inferiore dovrebbe superare lo stato della risorsa prima di generare un evento di sincronizzazione.

L'interazione tra gli attributi prevede che i parametri di temporizzazione abbiano priorità, ad esempio al superamento del periodo massimo la sincronizzazione avviene anche se lo stato della risorsa non è mutato. Come già accennato, l'attività di ricerca relativa alla definizione di tali attributi è ancora attiva, in quanto, l'interpretazione degli attributi può variare sia in funzione del metodo di binding che in funzione del tipo (o della rappresentazione) della risorsa. Inoltre, il supporto al meccanismo di observing parametrico in CoAP non è ancora ampiamente diffuso, in quanto oneroso

in termini computazionali. In tal senso, una soluzione adottabile è quella di fare uso di un reverse proxy, installato su un dispositivo dalle capacità computazionali più elevate di un semplice nodo sensore, il quale da una parte maschera una serie di nodi sensori e dall'altra espone l'observing condizionale ai client esterni. Maggiori dettagli circa questa soluzione sono presentati nella prossima sezione.

Service Discovery in reti IoT: Architettura, Protocolli e Prototipo

In ambiente IoT è necessario fornire una serie di servizi di supporto in grado di consentire l'integrazione di sistemi tra loro eterogenei. Uno dei servizi più importanti che deve essere messo a disposizione è il *service discovery* che fornisce supporto alla localizzazione, alla configurazione e all'accesso a risorse e funzionalità messe a disposizione da un dispositivo e/o disponibili in una determinata località geografica [14]. Una possibile architettura per tale servizio prevede l'utilizzo di un elemento, detto *resource directory*, che funge da database di descrittori di risorse e servizi, di una data rete IoT. La configurazione più semplice prevede una singola entità centralizzata che si fa carico di tutte le informazioni. Un setup più complesso è invece rappresentato da un sistema federato di *resource directory* distribuite. In entrambi i casi, a livello concettuale, l'elenco dei servizi viene immagazzinato in una entità terza che appare come un sistema centralizzato, sempre disponibile. L'uso di una tale soluzione risulta preferibile negli ambienti dinamici dell'IoT, in cui, è irrealizzabile un sistema di Discovery dei servizi che prevede l'interrogazione su richiesta dei nodi sensori di una rete, in quanto potrebbero non essere sempre attivi (i nodi possono spegnersi per risparmiare batteria), o spostarsi da un dominio ad un altro (per via della mobilità o per mutate condizioni del canale wireless) variando il proprio indirizzo IPv6 di contatto.

Un sistema tipico di *resource directory* prevede la presenza di due interfacce di interazione, una che consente la registrazione di risorse e servizi ed una che consente la ricerca delle stesse. Assunta la presenza di tale directory in una rete e assunto che i nodi IoT sappiano come raggiungerla, il primo passo per la pubblicazione dei servizi è quello di *registrazione*. Un nodo si registra specificando, possibilmente, una serie di informazioni di contatto (dominio, indirizzi etc.) e/o tecniche (tipo di piattaforma, etc.). Dopodiché, il dispositivo può trasmettere una lista di servizi da esso esposti e richiederne la memorizzazione nella *resource directory*. Come abbiamo detto l'ambiente IoT è molto dinamico per cui i nodi sono chiamati ad aggiornare periodicamente le informazioni pubblicate, pena la cancellazione della registrazione. Un tale approccio protegge, inoltre, la directory dal contenere informazioni obsolete o riferite a nodi non più raggiungibili.

Dall'altro ramo di interazione, l'interfaccia di lookup viene utilizzata dai client per poter avere accesso al repository di servizi. Si noti che per client si può intendere sia un applicativo esterno alla rete IoT, sia un nodo stesso della IoT che ha necessità di effettuare una comunicazione macchina a macchina (M2M). Un'interfaccia di lookup è solitamente strutturata in modo da accettare query parametriche, ad esempio è possibile richiedere la lista dei nodi che pilotano un sensore di temperatura, oppure, la lista dei nodi che sono autoalimentati etc. La risposta della directory prevedrà la lista dei servizi e le relative informazioni di contatto e quelle aggiuntive, sulla base dei criteri presenti nella query. Ricevuta la lista dei servizi disponibili, il cliente potrà iniziare ad interagire con essi, sia in modo diretto che attraverso entità intermediare, ad esempio rivolgendosi ad un sistema proxy.

In linea di principio un sistema di *resource directory* è la soluzione ideale per gestire il *service discovery* in ambienti altamente dinamici, densi e distribuiti come IoT, tuttavia, l'implementazione di un tale modulo software deve tenere presente i requisiti tecnici stringenti tipici dei dispositivi di queste reti, al fine di realizzare un sistema capace di garantire scalabilità, offrire context-awareness e minimizzare la comunicazione in rete. L'obiettivo di questa attività è di realizzare un prototipo di gateway IoT che offra il supporto del *service discovery* sia a reti esterne che ai dispositivi IoT ad esso connessi utilizzando le funzionalità standard offerte da COAP. Sarà inoltre presentata una valutazione tramite modello simulativo della scalabilità delle diverse opzioni di progetto considerate nell'implementazione del prototipo.

Architettura della Resource Directory (RD) in COAP

Il protocollo CoAP fornisce un meccanismo elementare per il discovery delle risorse ospitate da un nodo sensore. Tale meccanismo prevede che ogni server CoAP disponga di una risorsa raggiungibile

all'URI relativo *"/.well-known/core"*, il cui compito è quello di rispondere a richieste di tipo GET con la lista dei descrittori (in formato CoRE link) delle risorse disponibili sul nodo. Per le motivazioni introdotte nel paragrafo precedente, esso risulta un approccio limitante in grandi reti IoT rispetto all'uso di una resource directory. Questo ha spinto il gruppo di lavoro CoRE a proporre un'architettura CoAP per implementare un servizio di resource directory [15].

In linea con i principi di un servizio di discovery basato su un repository, la resource directory (o **RD** in breve) in CoAP fornisce un'interfaccia ReST per la registrazione di nodi e servizi (a sinistra in Figura 13) ed una per il lookup (a destra in Figura 13).

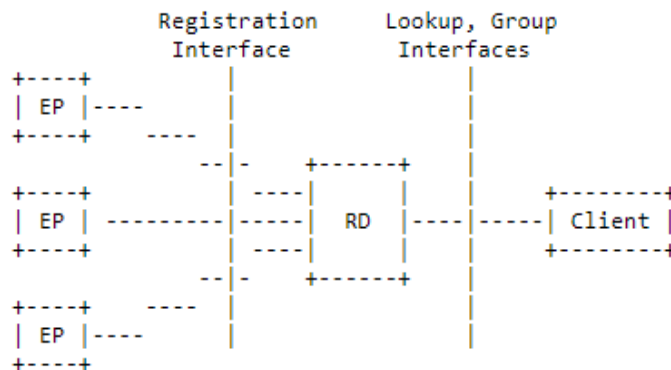


Figura 13 – Architettura della RD in CoAP

I CoAP server, altresì identificati come **EndPoint** CoAP (EP) usano l'interfaccia di registrazione per registrare la propria presenza all'interno della rete di competenza della RD. Tale processo di registrazione può avvenire secondo una delle seguenti modalità:

- *Registrazione semplice*: Prevede che l'endpoint invii una richiesta POST, detta di *"simple registration"*, alla risorsa *"/.well-known/core"* della RD. Alla ricezione di tale messaggio la RD crea una entry relativa al nodo sensore configurando le informazioni di contatto in modo implicito (l'indirizzo IPv6 di contatto viene estratto dal sorgente della richiesta CoAP). Dopodiché, l'RD si fa carico di inviare una richiesta GET alla risorsa di discovery dell'endpoint al fine di acquisire la lista di risorse esposte dallo stesso. Tali risorse vengono memorizzate nella RD e associate all'endpoint. Tale meccanismo è utile in presenza di nodi sensori con capacità molto ridotte, in quanto il processo di registrazione e refresh richiede semplicemente l'invio di una richiesta POST con payload vuoto.
- *Registrazione standard*: Prevede che il server CoAP invii una richiesta POST all'interfaccia di registrazione endpoint (raggiungibile di default a *coap://RD_IP:RD_PORT/rd*). Solitamente, in questo tipo di registrazione il nodo specifica informazioni aggiuntive quali ad esempio tipo di piattaforma, posizione, proxy di accesso, intervallo validità ecc. Se la richiesta va a buon fine, la RD registra il nodo e le sue informazioni e restituisce un'interfaccia ReST dedicata che l'endpoint può usare per estendere la validità e gestire l'elenco dei descrittori delle risorse CoAP.
- *Registrazione esterna*: Prevede che a registrare un nodo sia un elemento terzo indentificato come **Commissioning Tool (CT)**. In questo caso è necessario che la richiesta POST di registrazione contenga i dettagli di contatto (IP, porta, proxy etc....) in modo esplicito, in quanto non derivabili dal pacchetto di richiesta. Tale sistema di registrazione può essere usato, per esempio, da un amministratore di rete o da un software di auto-configurazione e può risultare utile in quelle reti IoT già messe in campo prima dell'introduzione della RD in cui non è praticabile un aggiornamento di massa dei firmware per abilitare i processi di registrazione semplice o standard.

Una volta effettuata la registrazione, è possibile effettuare il caricamento dei descrittori delle risorse CoAP ospitate dall'endpoint. A seconda del processo di registrazione utilizzato ciò può avvenire secondo diverse modalità. In particolare, nel caso di registrazione semplice il caricamento dei descrittori delle risorse è contestuale alla registrazione e ad opera della resource directory. Negli altri due casi il caricamento può essere contestuale, se la richiesta POST contiene nel suo payload la lista dei descrittori, altrimenti può avvenire in un secondo momento. Infatti, alla fine del processo di registrazione vengono predisposte delle interfacce ReST per ciascun endpoint utilizzabili per caricare, recuperare, modificare e/o eliminare descrittori di risorse CoAP. Si noti come nel caso di utilizzo di un CT, il caricamento avviene per mano dello stesso il quale oltre a registrare gli endpoint, usa le relative interfacce di gestione per gestire i descrittori delle risorse ad esso associate. Si noti, inoltre, che per descrittore di risorsa si intende un **web link** in formato **CoRE link** che contiene una serie di attributi specifici come tipo di risorsa, formato di rappresentazione, supporto all'observing ecc. I web link memorizzati nella RD sono del tutto equivalenti a quelli che si otterrebbero se si effettuasse una richiesta di discovery direttamente su un server CoAP tramite la sua risorsa `.well-known/core`.

Il descrittore di un endpoint e tutti i quelli delle risorse ad esso associate possono essere eliminati nei seguenti casi:

- Per richiesta esplicita dell'EP o del Commissioning Tool. Questo può accadere per esempio quando un nodo mobile lascia la rete di competenza della RD, o perché si pone in modalità sleep prolungata per preservare la batteria, oppure, perché viene rimosso manualmente dalla rete.
- Se l'endpoint non aggiorna la sua registrazione entro l'intervallo di validità. Ciò può accadere a seguito del malfunzionamento di un nodo.

All'interno della resource directory oltre che descrittori degli endpoint e delle risorse è possibile registrare oggetti avanzati quali *gruppi* e *domini*. Per gruppo si intende un insieme di endpoint. Per esempio, un gruppo può essere costituito da tutti gli endpoint che hanno un sensore temperatura, oppure, da tutti i destinatari di un certo flusso di traffico Multicast (in tal caso è possibile associare al gruppo un indirizzo Multicast). Un *dominio* è anch'esso un insieme di endpoint, ma il significato di gruppo è solitamente associato ad un confinamento geografico (ad esempio `edificio_A_stanza_31`) oppure ad una sottorete. La registrazione e gestione di gruppi e domini è possibile grazie a due interfacce ReST dedicate, ma, si consiglia limitare il loro accesso al solo Commissioning Tool.

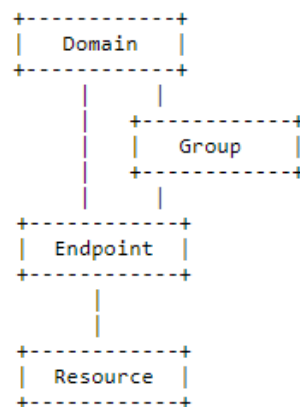


Figura 14 – Gerarchia entità registrate sulla RD

L'interfaccia di lookup della resource directory consente a client (nella fattispecie ci si riferisce sia a CoAP client esterni alla rete IoT che a nodi IoT che fungono da CoAP client per comunicazioni M2M) di recuperare le informazioni in essa memorizzate. Nel dettaglio sono predisposte 4 interfacce ReST ciascuna dedicata relativamente alla ricerca di endpoint, risorse, gruppi e domini. Attraverso

opportune query è possibile applicare un filtering sui risultati, ad esempio: “restituisce tutti i descrittori degli endpoint che ospitano almeno un sensore di temperatura”.

In linea generale, il processo di discovery si articola nelle seguenti fasi:

- Gli EP si registrano e caricano i descrittori delle proprie risorse (ad esempio, sensori ambientali) all'interno della RD;
- Un client effettua una query di lookup della RD al fine di scoprire le risorse disponibili in rete;
- Il client, apprese le informazioni di contatto dell'endpoint può interagire con le risorse da esso esposte.

Proxyng delle risorse tramite RD

A causa delle richieste computazionali (memoria e calcolo) ed energetiche (sistema sempre attivo) necessarie ad una RD non è pensabile la sua installazione su uno dei nodi sensori a basso consumo attualmente disponibili. Bensì, il candidato ideale risulta essere il gateway, un elemento, solitamente, connesso alla rete elettrica e dalle capacità computazionali superiori a tutti i nodi della rete di sensori essendo l'elemento di congiunzione tra la rete IoT di sensori e le reti esterne. Un vantaggio di tale setup riguarda il fatto che una resource directory posta sul gateway ben si presta ad agire da Proxy per tutte le risorse presenti nella rete. L'idea è quella di limitare il traffico nel ramo della rete 6LoWPAN attraverso i meccanismi di proxyng e caching nativi di CoAP. Con tale meccanismo viene alterato il processo di discovery ed interazione esposto alla fine del paragrafo precedente, come segue:

- Gli EP si registrano e caricano i descrittori delle proprie risorse (ad esempio, sensori ambientali) all'interno della RD;
- Un client effettua una query di lookup della RD al fine di scoprire le risorse disponibili in rete;
- Il client dichiara il suo interesse per una risorsa (o per un set di risorse) alla resource directory, la quale si interpone tra client ed endpoint fungendo da proxy di richieste CoAP.

I meccanismi con i quali la resource directory agisce da proxy, le tecniche di caching e come i client possono registrare il proprio interesse saranno chiariti contestualmente alla descrizione dei moduli JAVA che compongono il prototipo di resource directory che è stato implementato.

Prototipo del modulo RD in JAVA

In questo paragrafo viene presentata l'implementazione del modulo resource directory in Java. L'implementazione si poggia sull'utilizzo di *Californium*, un potente framework scritto in Java per lo sviluppo di applicativi cloud-IoT basati su CoAP. In particolare, il progetto RD-13 sfrutta un'implementazione di riferimento di una CoRE resource directory (versione 04), disponibile nel repository *californium.tools*. Risulta, per cui, ereditata la struttura cardine per organizzare e rappresentare le risorse CoAP. Come spiegato in precedenza, *il prototipo ideato oltre a fungere da resource directory, integra un meccanismo di reverse proxyng delle risorse che vi vengono registrate*, maggiori dettagli su tale funzionalità sono presenti nel paragrafo in cui viene dettagliata la struttura della classe *RDProxyManagement.java*.

I principali elementi costitutivi della nostra soluzione sono le seguenti classi Java:

- *ResourceDirectory13.java*
- *RDDiscover.java*
- *RDDiscoveryResource.java*
- *RDResource.java*

- RDNodeResource.java
- RDDomain.java
- RDGroup.java
- RDProxyResource.java
- RDProxyManagement.java
- RDLookupEpResource.java
- RDLookupResResource.java
- RDLookupDomainResource.java
- RDLookupGroupResource.java

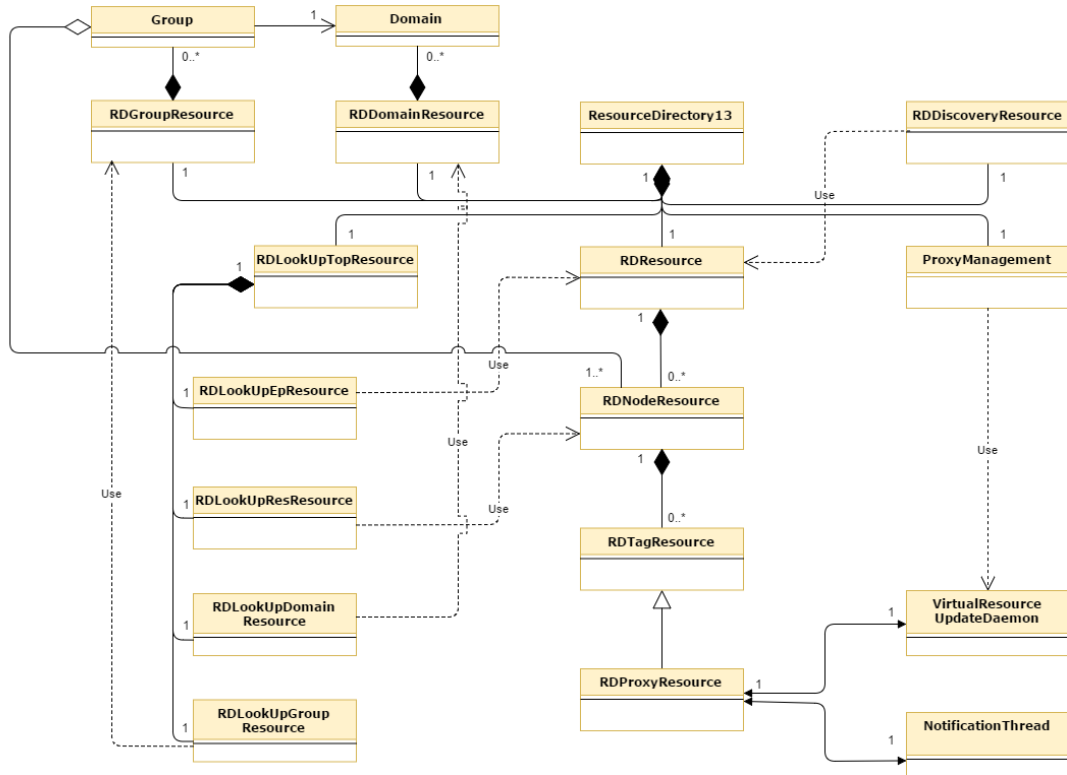


Figura 15 – Diagramma delle classi del prototipo sviluppato

Nei seguenti paragrafi verrà brevemente illustrato il funzionamento di ciascuna di esse. La Figura 15 illustra il diagramma UML delle classi con le loro relazioni.

ResourceDirectory13.java

Rappresenta la classe principale in cui è presente il metodo **main**. Di base, tale classe estende la classe **CoAPServer**, fornita dal framework **Californium**. Un oggetto **CoAPServer** ospita una serie di risorse, esposte ai **CoAP** client attraverso uno o più endpoint. Un endpoint è un oggetto che si lega ad un'interfaccia di rete (ed una sola) utilizzando un socket, rappresenta, quindi, il punto di ingresso ed uscita di pacchetti **CoAP**. Un **CoAPServer** può essere fatto partire e arrestare. Quando un **CoAPServer** è in funzione gli endpoint ad esso associato sono in listening sulla porta e sull'interfaccia di rete, specificati in fase di allocazione dell'endpoint. Quando un server viene arrestato, gli endpoint rilasciano la porta ad essi associata.

Le risorse **CoAP** ospitate da un **CoAPServer** sono oggetti della classe **CoAPResource**. Essi possono rappresentare sia delle risorse reali che risorse virtuali. Tali risorse sono organizzate seguendo una gerarchia ad albero, ossia ogni **CoAPResource** può avere come discendente una o più **CoAPResource**.

La radice di tale albero è, quindi, rappresentata dal CoAPServer. A ciascun livello della discendenza ogni risorsa deve essere associata con un identificatore univoco, in quanto l'accesso alla risorsa avviene specificando il path, composto dai nomi delle risorse, che si trovano dalla radice ad essa.

Ad esempio, il seguente URI-path "coap://192.168.1.1:5683/sensors/temperature/temp1" si riferisce ad un CoAP Server con un endpoint in listening sull'interfaccia il cui indirizzo IP è 192.168.1.1, sulla porta 5683 e la cui gerarchia delle risorse si presenta così composta:

1. Un oggetto CoAPResource di nome *sensors* discendente della radice. Il cui nome univoco suggerisce che su quel ramo dell'albero si trovano tutti i sensori del server CoAP;
2. Un oggetto CoAPResource di nome *temperature* discendente di *sensors*. Il cui nome suggerisce la presenza di sensori di temperatura;
3. Un oggetto CoAPResource di nome *temp1* che rappresenta un sensore.

Gli oggetti al punto 1 e 2 possono essere classificati come *virtuali* in quanto non corrispondono ad un sensore concreto a differenza dell'oggetto al punto 3 il quale predisporrà degli handler dei metodi GET, POST, PUT e DELETE.

Il metodo main della classe ResourceDirectory ha il compito di acquisire ed analizzare i parametri di input all'avvio del modulo RD-13, creare l'oggetto ResourceDirectory13 e configurarlo in base alla configurazione di input, inizializzare le interfacce di discovery, registrazione, management e lookup, quindi far partire il CoAPServer. Quando viene creato un server CoAP in Californium viene automaticamente creata l'interfaccia standard per il discovery in CoAP all'indirizzo *.well-known/core*, purtroppo, la sua implementazione non è conforme ai requisiti della resource directory. Infatti, per poter supportare il meccanismo del *simple publishing* è necessario che tale risorsa sia in grado di elaborare richieste POST. Per tale ragione si rende necessario estendere la classe DiscoveryResource nella classe RDDiscoveryResource e sostituirla alla versione di default in fase di start-up del server.

Una volta configurato l'oggetto ResourceDirectory13, prima di avviare le funzionalità server, il metodo main deve configurare la porta e le/a interfacce/ia di rete su cui ascoltare richieste. Le impostazioni di default fanno uso della porta CoAP standard (5683) e di tutte le interfacce di rete disponibili sulla macchina all'avvio del server. In alternativa, è possibile usare il parametro "-i" seguito dall'indirizzo IP dell'interfaccia di rete e dalla porta, per forzare l'avvio del CoAP server su una singola interfaccia.

[RDDiscoveryResource.java](#)

Come precedentemente menzionato, tale classe estende la classe DiscoveryResource fornendo il metodo accessorio in grado di accettare ed elaborare richieste di tipo POST. La ricezione di una richiesta POST viene interpretata come una richiesta di registrazione passiva da parte di nodo CoAP. Il payload di tale richiesta deve essere vuoto, mentre tra le opzioni presenti nell'URI deve figurare il parametro *ep* (nome dell'endpoint). In seguito alla verifica della correttezza dei parametri, la resource directory si fa carico di recuperare il contenuto della risorsa *.well-known/core* del nodo CoAP, acquisendo così la lista delle risorse disponibili su tale nodo e successivamente di registrare il nodo, passando i parametri di registrazione e i web-link ottenuto al metodo handlePOST della classe RDResource.java.

[RDResource.java](#)

Estende la classe CoAPResource, risulta raggiungibile usando il path standard *coap://IP:PORT/rd* e consente ad un nodo CoAP di registrarsi alla resource directory attraverso il metodo *handlePost*.

Alla ricezione di una richiesta POST, vengono eseguiti i controlli di validità richiesti dal draft RD, in particolare una richiesta è accettata se il nome dell'endpoint è presente ed unico all'interno del dominio specificato. Se il dominio non è specificato si assume il dominio di default. Se la richiesta passa

i controlli viene determinato se un nodo con l'endpoint name specificato esiste all'interno del repository. Se tale nodo non esiste viene creata un'istanza della classe `RDNodeResource`, avente per nome il nome dell'endpoint e viene aggiunta come risorsa figlia dell'oggetto `RDResource`. Il path che si verrà a creare sarà del tipo `coap://IP:PORT/rd/nomeEndpoint`. Se, invece, il nodo risulta già registrato, viene recuperato il riferimento all'oggetto di tipo `RDResource` che lo rappresenta. In entrambi i casi il payload della richiesta POST, contenente il set delle risorse esposte dal nodo, viene utilizzato per aggiornare le informazioni immagazzinate nella resource directory. Al fine di rendere la registrazione idempotente, la resource directory rifletterà lo stato del nodo relativo all'ultima richiesta di registrazione, ossia, una doppia registrazione sovrascrive le informazioni già presenti. Se la procedura di registrazione avviene con successo, al nodo viene ritornato il path URI della risorsa che lo rappresenta, la quale può essere usata per aggiornare e/o fare l'aggiornamento della stessa.

Nota: Il draft della resource directory, specifica che il nome dell'endpoint debba essere univoco all'interno di un dominio. Tuttavia, gli endpoint essendo aggiunti tra l'elenco dei figli della risorsa `RDResource`, devono avere nomi univoci all'interno della resource directory. LA Figura 16 illustra il diagramma delle operazioni di registrazione.

[RDNodeResource.java](#)

Un oggetto di tale classe modella un endpoint CoAP registrato nella RD. Anche questa classe estende la classe `CoAPResource`, in modo da fornire gli handler dei metodi REST CoAP usati per interagire con la RD. Secondo le specifiche del draft RD, ad ogni nodo è associato un parametro di validità temporale detto *lifetime*, il cui valore di default è di 24 ore o che può essere specificato in fase di registrazione. Se un nodo non effettua il refresh della registrazione entro tale termine, le informazioni relative al nodo salvate sulla resource directory vengono cancellate.

Segue una descrizione delle principali operazioni gestite dalla classe.

- Nuova registrazione: A seguito della creazione di un nuovo oggetto, il metodo `setParameter` viene invocato al fine di aggiornare i parametri specificati nella Uri-query della richiesta di registrazione (contesto, lifetime, tipo dell'endpoint, etc). Dopodiché, il payload della richiesta, contenente la lista dei core-link rappresentanti le risorse esposte dal nodo CoAP, viene elaborato attraverso la funzione `updateEndpointResources`.
- Gestione richiesta POST: Una richiesta POST all'indirizzo `coap://IP:PORT/rd/nomeEndpoint` è usata, dal nodo a cui quella registrazione si riferisce, per segnalare la sua presenza all'interno della rete (messaggio di keep-alive) oppure per aggiornare le informazioni quali parametri dell'endpoint e/o lista delle risorse esposte. Le funzioni `setParameter` e `updateEndpointResources` sono nuovamente usate a tale scopo.
- Gestione richiesta DELETE: Questa funzione gestisce la cancellazione dell'endpoint e di tutte le risorse ad esso collegate.
- Gestione richiesta GET: Questa funzione restituisce la lista dei link registrati dall'endpoint sulla RD. È possibile specificare opzioni di filtering nell'URI-query della richiesta.
- Gestione delle richieste PUT/PATCH/iPATCH: Tali richieste vengono interpretate secondo quanto definito nel draft core-rd-11. Nel dettaglio, usando un payload di tipo `merge-patch+json` è possibile modificare il set dei link registrati da un nodo.
- Metodo `updateEndpointResources` : Tale metodo viene invocato per analizzare la lista dei core-link che un endpoint fornisce alla resource directory. Ogni core-link rappresenta una risorsa ospitata dal nodo CoAP ed è caratterizzata da un path relativo e da una serie di parametri. Il path relativo (ad es. `/sensors/temperature/temp1`) viene elaborato e per ciascun elemento intermedio (*sensors, temperature*) viene creato un oggetto di tipo `RDTagResource`, tali oggetti

sono poi collegati in modo da rispettare l'ordine gerarchico del path (*temperature* figlio di *sensors*, *sensors* figlio di *endpoint*). L'ultimo elemento del path (*temp1*) rappresenta l'identificativo della risorsa fisica sul nodo CoAP, e viene, invece, rappresentato da un oggetto di tipo *RDProxyResource* e poi aggiunto alla gerarchia di risorse dell'endpoint.

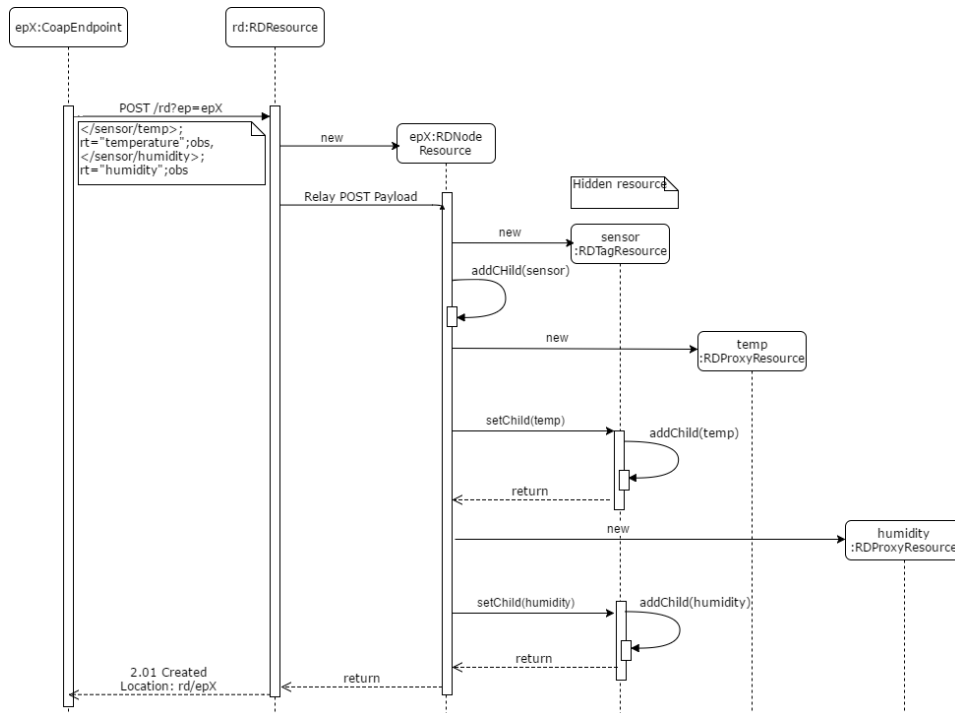


Figura 16 – Diagramma di sequenza delle operazioni di registrazione

RDTagResource.java

Tale classe estende la classe *CoAPResource*, al fine di poter essere inserita nella gerarchia delle risorse della resource directory. Essa si caratterizza per essere un contenitore di attributi per web-link, attraverso l'utilizzo di una hashmap di tipo stringa-stringa.

RDProxyResource.java

La classe *RDProxyResource* rappresenta l'elemento chiave per l'implementazione delle funzionalità di proxying e caching delle risorse CoAP. Nel dettaglio, un oggetto di tipo *RDProxyResource* può essere considerato come una risorsa CoAP virtuale che ha il compito di rappresentare lo stato di una risorsa CoAP reale (ossia ospitata da un nodo sensore di una rete 6LoWPAN). L'oggetto *RDProxyResource* è a tutti gli effetti una risorsa CoAP, per cui, raggiungibile da client esterni attraverso i metodi REST. Un client interessato a ricevere aggiornamenti (o semplicemente l'ultima rappresentazione) da una risorsa CoAP, anziché interrogare direttamente il sensore, dirige le sue richieste alla risorsa virtuale esposta dalla resource directory. Tutto ciò è possibile grazie a due meccanismi di sincronizzazione, il primo che esiste tra la risorsa virtuale e la risorsa reale che dipende dal tipo stesso di risorsa e dall'interesse da parte dei client esterni nel riceverne la rappresentazione più recente, ed il secondo che esiste tra la risorsa virtuale e i client esterni.

Tali meccanismi sfruttano un campo della classe *RDProxyResource*, chiamato *resourceValue*, adibito a contenere l'ultima rappresentazione della risorsa reale. L'aggiornamento di tale campo è affidato, in maniera ThreadSafe, ad un oggetto di tipo *VirtualResourceUpdateDaemon*, mentre, la trasmissione ai client è affidata ad un oggetto di tipo *NotificationThread*. Maggiori dettagli sul funzionamento di tali

elementi saranno presentati nei seguenti paragrafi. La *Figura 17* illustra il diagramma di sequenza delle operazioni di sincronizzazione delle risorse virtuali del proxy.

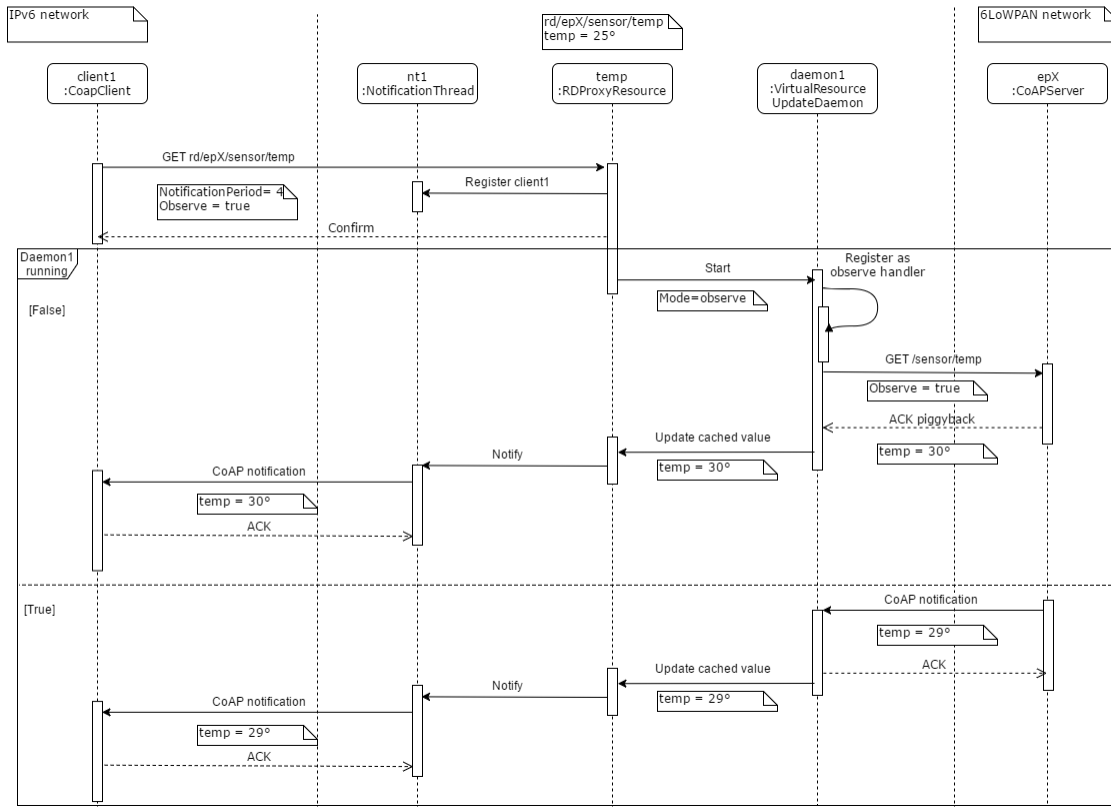


Figura 17 – Diagramma di sequenza delle operazioni di aggiornamento di sincronizzazione delle risorse virtuali del proxy

Classificazione delle risorse

Prima di addentrarci nella discussione del meccanismo di sincronizzazione è importante puntualizzare cosa si intende per *tipo di risorsa*. Infatti, sono state individuate tre tipologie di risorse CoAP che possono essere ospitate da nodi sensori:

- **NoObs**: Risorsa CoAP senza il supporto all’observing. È una condizione che può verificarsi in caso di nodi sensori con limitate capacità computazionali. Tale condizione è individuabile analizzando il descrittore della risorsa trasmesso dal nodo sensore in fase di registrazione alla resource directory, sarà, infatti, assente il parametro *obs*;
- **Obs**: Risorsa CoAP con supporto all’observing di tipo periodico. Nel descrittore della risorsa è presente il parametro *obs*. Non essendo né completamente definito né diffuso, l’observing periodico configurabile, si assume che il periodo di generazione delle notifiche sia fisso e definito dal firmware del nodo sensore in modo statico;
- **EventObs**: Risorsa CoAP con il supporto all’observing di tipo ad eventi. Alcuni sensori hanno una rappresentazione significativa nel momento in cui si verifica un determinato evento a cui corrisponde la generazione di una notifica CoAP. Ad esempio, un PIR può essere usato per generare un evento quando viene rilevato del movimento, oppure, un sensore di allarme può inviare una notifica al superamento di una certa soglia. Allo stato attuale non esiste un attributo che differenzi questo tipo di observing dal precedente. Nella progettazione del lato proxy della resource directory si **assume** che sensori di questo tipo vengano contrassegnati dall’attributo **event**, oltre che dall’attributo *obs*, all’interno del descrittore della risorsa.

Meccanismi di binding

Classificare le risorse è importante per definire il metodo di binding da usare per gestire la sincronizzazione tra risorsa reale e virtuale. In particolare, si delineano le seguenti possibilità

- **NoObs**: L'unico modo per sincronizzarsi è quello di usare il **Polling** come metodo di binding. Per scegliere il periodo di Polling possono essere usate diverse policy che verranno illustrate più avanti;
- **Obs**: In questo caso è possibile usare sia l'**Observing** (opzione attiva di default) che il **Polling**. Usare l'observing sarebbe la scelta naturale, ma, il polling consentirebbe un maggiore controllo sulla frequenza di notifica: se, infatti, i client interessati nella risorsa hanno un requisito in termini di frequenza inferiore alla frequenza con cui l'observing produrrebbe le notifiche, il polling consentirebbe di ridurre il numero di messaggi prodotti nella rete 6LoWPAN. Le strategie di selezione della frequenza del polling verranno illustrate più avanti.
- **EventObs**: Il binding utilizzato è l'**Observing** al fine di preservare la rappresentazione della risorsa.

VirtualResourceUpdatingDaemon.java

L' oggetto di tipo VirtualResourceUpdatingDaemon, come anticipato precedentemente, è l'elemento adibito a sincronizzare lo stato della risorsa reale con quello della risorsa virtuale. Il suo funzionamento dipende dal tipo di binding:

- **Observing**: all'avvio del demone, questo si registra come observer della risorsa reale e si mette in ascolto delle notifiche (a tale scopo implementa l'interfaccia CoapHandler). Alla ricezione di una notifica aggiorna lo stato della risorsa virtuale (*resourceValue*) a cui esso è associato. Al termine del demone, viene inviata una richiesta di tipo *observe cancellation* per rimuovere l'associazione di observing col nodo sensore.
- **Polling**: quando il demone è attivo, esso periodicamente invia una richiesta GET al nodo sensore, attende la ricezione della risposta, aggiorna lo stato della risorsa virtuale ed infine, sfruttando uno *ScheduledExecutorService*, programma la sua prossima esecuzione.

Al fine di ridurre il consumo di risorse nel ramo 6LoWPAN, il demone è attivo se c'è almeno un client con una relazione di observing attiva nei confronti della risorsa virtuale.

Interazione con i client

I client possono interagire con la risorsa virtuale attraverso richieste di tipo GET semplici o con l'opzione *observe* abilitata. Se la RDProxyResource riceve una richiesta GET semplice viene analizzata la copia in cache del valore della risorsa reale, se risulta esser ancora valida (il max-age non è stato superato) viene mandata in risposta alla richiesta del client, altrimenti, la risorsa agendo da proxy, invia a sua volta una richiesta al sensore reale, acquisisce la copia più recente dello stato della risorsa ed infine lo trasmette in risposta al client. Se un client invia una richiesta di observing esso viene rappresentato con un oggetto *ObservingClient* e aggiunto alla lista degli observer. Se esso è il primo client a stabilire una relazione di observing viene avviato il demone di aggiornamento. In caso contrario, sulla base delle policy di sincronizzazione, potrebbe essere necessario aggiornare il periodo di aggiornamento del demone.

Se un client invia una richiesta di cancellazione di observing, esso viene rimosso dalla lista degli observer. Se tale lista diventa vuota il demone di aggiornamento viene arrestato.

L'interazione client – proxy supporta l'observing condizionale attraverso i parametri di binding introdotti nella sezione "Prototipo del modulo RD in JAVA". Tali parametri vengono estratti dall'URI query della richiesta di observing e memorizzati nell'oggetto *ObservingClient*.

NotificationThread.java

L'oggetto *NotificationThread* implementa la logica necessaria per inviare le notifiche ai client che hanno stabilito una relazione di observing, rispettando i vincoli temporali e sui valori eventualmente specificati in fase di registrazione. Esso può essere considerato come un thread ricorrente, la cui esecuzione è scandita dal verificarsi di eventi, che:

- Si sospende a tempo indefinito, in attesa che lo stato della risorsa virtuale cambi, se nessun client ha richieste temporali (pmin, pmax o p);
- Si sospende a tempo determinato (tramite un timeout), in attesa che lo stato della risorsa virtuale cambi, se almeno un client ha una richiesta temporale.

La sospensione del thread avviene grazie all'uso di una *java condition* condivisa con la *RDProxyResource*, mentre il timeout viene calcolato nella funzione *setTimeout* analizzando tutti i requisiti temporali dei client attivi (maggiori dettagli si trovano nella sezione commenti del file *NotificationThread.java*).

Durante la fase di sospensione il thread può essere risvegliato da una delle seguenti condizioni:

- Lo stato della risorsa virtuale è stato aggiornato. In questo caso è necessario notificare i client i cui requisiti temporali e non sono soddisfatti;
- Il timeout di sospensione viene raggiunto. In questo caso è necessario notificare il client il cui vincolo temporale è prossimo alla scadenza;
- La lista dei client è cambiata. In questo caso potrebbe essere necessario aggiornare il valore del timeout.

Politiche per la gestione del periodo di polling

Secondo le specifiche descritte nella sezione "*VirtualResourceUpdatingDaemon.java*", se il demone di aggiornamento della risorsa virtuale è configurato per operare con il binding di tipo *polling* è necessario definire il periodo con cui effettuare la sincronizzazione. A tale scopo si individuano le seguenti politiche:

- Periodo statico configurato in fase di start-up della resource directory (default a 60 secondi).
- Periodo dinamico:
 - Basato sul max-age: Il periodo viene settato ad un valore prossimo al max-age dell'ultima risposta CoAP ricevuta dal server. In tale modalità, il controllo è lasciato al programmatore del firmware del server. Se il max-age è assente viene applicata la politica statica al punto precedente.
 - Basato sui requisiti del client: Viene usato come periodo, il periodo tale a soddisfare i requisiti temporali del client più esigente.
- Periodo statico configurato dall'interfaccia di controllo *ProxyManagement*.

Le prime due modalità sono selezionabili in fase di avvio, mentre l'accesso all'interfaccia di controllo avviene a tempo di esecuzione. In caso quest'ultima politica sia attiva, essa ha la priorità sulle altre. E' comunque possibile ripristinare la configurazione iniziale da tale interfaccia.

RDGroupResource.java e RDDomainResource.java

Rappresentano gli oggetti adibiti a fornire un'interfaccia CoAP per la gestione dei gruppi e dei domini rispettivamente. Risultano raggiungibili usando i rispettivi indirizzi *coap://IP:PORT/rd-group* e *coap://IP:PORT/rd-domain*. Attraverso il metodo POST è possibile creare un nuovo elemento, mentre il metodo DELETE ne consente l'eliminazione se opportune condizioni sono soddisfatte (es. non è possibile eliminare un gruppo non vuoto etc).

Interfacce di look-up

In linea con le specifiche della resource directory versione 13 sono state implementate quattro interfacce di look-up i cui dettagli vengono riassunti nella tabella sottostante.

Tipo di lookup	URL	Classe java
Risorsa CoAP	coap://IP:PORT/rd-lookup/res	RDLookupResResource.java
Endpoint CoAP	coap://IP:PORT/rd-lookup/ep	RDLookupEpResource.java
Gruppo	coap://IP:PORT/rd-lookup/gp	RDLookupGroupResource.java
Dominio	coap://IP:PORT/rd-lookup/d	RDLookupDomainResource.java

Un lookup per risorse CoAP fornisce come risultato dei link in formato core-link semanticamente equivalenti ai descrittori di risorse che i nodi CoAP hanno inviato alla resource directory in fase di registrazione. Un lookup per endpoint fornisce una lista di core-link rappresentanti l'istanza di registrazione di un endpoint. In ciascun link sono contenute informazioni aggiuntive dell'endpoint come il suo contesto, dominio, tempo di validità etc. Un lookup per gruppo o dominio fornisce una lista dei gruppi o dominio presenti (e relativi parametri) rispettivamente. L'operazione di lookup può essere filtrata attraverso criteri di inclusione che il client può specificare nella URL query string del pacchetto di richiesta. È supportato l'uso di più parametri contemporaneamente (in tal caso si ha match se e solo se tutti i criteri sono soddisfatti) così come l'uso del carattere jolly "*". La lista completa dei filtri disponibili è presente nel draft della resource directory.

RDProxyManagement.java

Rappresenta un'interfaccia CoAP per la gestione delle politiche di binding attuate dalle risorse virtuali. Attraverso il metodo POST usando un payload in formato JSON è possibile specificare per ogni risorsa quale binding usare e/o quale periodo usare in caso di **polling** periodico.

Testing e analisi delle prestazioni della RD

Al fine di testare le prestazioni della resource directory si è fatto utilizzo del simulatore Cooja parte del progetto Contiki OS [16]. Con Cooja è possibile creare reti 6LoWPAN di nodi sensori commerciali il cui comportamento (sia hardware che software) viene simulato. Inoltre, esso consente di iniettare nella rete simulata traffico IP generato da dispositivi esterni o applicazioni attraverso un sistema di *bridging* grazie all'utility *Tunslip*. Il setup simulato ha previsto l'uso di una topologia di rete a griglia 6x6 (vedi *Figura 18*), con il border router RPL implementato su un nodo al bordo (il nodo verde nella *Figura 18*), mentre tutti gli altri nodi fungono da CoAP server. A livello fisico (PHY) e MAC i nodi usano il protocollo IEEE 802.15.4, con una velocità di trasmissione di 250kbps e banda radio a 2.4Ghz. La resource directory è eseguita come un processo JAVA sulla stessa macchina che esegue Cooja (HP ProDesk 400 g4), così come i CoAP client sono stati implementati in Java. La comunicazione tra la resource directory e la rete Cooja avviene grazie all'utility *Tunslip*, mentre i CoAP client e la resource directory dialogano tramite comunicazione interprocesso. Si assume, per semplicità, che ciascun CoAP Server espone una sola risorsa CoAP.

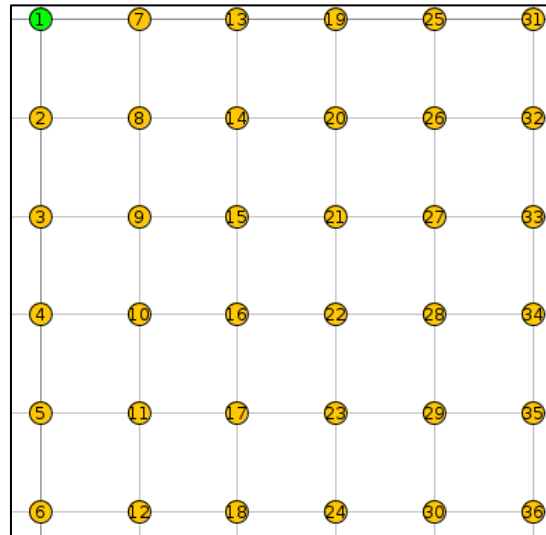


Figura 18 – Topologia della rete usata nel modello simulativo.

Il test ha preso in esame le seguenti tre configurazioni:

- **OBS-C**: è una configurazione in cui i client CoAP non fanno uso delle funzionalità proxy della resource directory. I client, infatti, usano la RD per ottenere i descrittori delle risorse CoAP della rete, dopodiché inviano una richiesta di Observing direttamente al nodo sensore. La richiesta di observing prevede la specifica del periodo desiderato di notifica come attributo di binding. Ogni volta che il periodo specificato dal client termina il server CoAP genera una notifica e gliela invia. Si noti che il supporto all'observing periodico è stato sviluppato in quanto assente in Contiki.
- **OBS-P**: in tale configurazione la resource directory agisce anche da proxy. I client, registrano il proprio interesse, inviando richieste di Observing, alle risorse virtuali esposte dalla RD. Anche in questo caso i client specificano il periodo di notifica desiderato. Quindi, l'RD ha il compito di stabilire una relazione di observing con le risorse reali, dove il periodo di notifica specificato nel contattare una risorsa è dato dal periodo più piccolo tra quelli specificati dal set di client interessati alla risorsa stessa. Ogni volta che una notifica viene ricevuta dalla RD questa viene inoltrata ai client il cui periodo di notifica è terminato.
- **POLL**: è una variante dello schema precedente in cui l'unica differenza riguarda il metodo di binding, ossia, viene utilizzato il *polling* anziché l'*observing*. Anche la scelta del periodo è identica al caso precedente.

Lo scenario sotto indagine ha previsto che tutti client specificassero un periodo di notifica comune (test omogeneo), mentre il numero di client per risorsa è stato fatto variare. La metrica di performance usata è stato il *goodput*, espresso come il numero medio di notifiche correttamente ricevute da ciascun client.

Si noti che la differenza sostanziale tra OBS-C e OBS-P/POLL riguarda il numero di relazioni di binding per ciascuna risorsa CoAP. Definendo come n_i il numero di CoAP client per risorsa reale r_i e v_i la risorsa virtuale ad essa associata sulla RD, lo schema OBS-C richiede n_i relazioni di binding direttamente verso r_i , mentre gli schemi OBS-P e POLL richiedono n_i relazioni di binding tra i client e v_i ed una **singola** relazione di binding fra v_i e r_i . Infatti, l'obiettivo primario di questo test è quello di quantificare il vantaggio di introdurre un intermediario (la funzionalità Proxy della RD) tra le risorse IoT e le applicazioni CoAP. In secondo luogo, lo stesso test ci consente di investigare se esistono differenze

sostanziali tra un proxy che stabilisce una singola relazione di observing (OBS-P) specificando un periodo di notifica ed uno che usa un approccio di polling con lo stesso periodo (POLL).

La Figura 19 mostra il goodput come funzione del carico della rete (*observing load* - anch'esso espresso come media per singolo client) per due scenari: *i*) quattro client per risorsa ($n_i = 4$), e *ii*) dieci client per risorsa ($n_i = 10$). I risultati mostrano come il goodput migliori in maniera significativa quando l'observing viene mediato dall'uso di un proxy. Considerando il caso $n_i = 4$ (Figura 19(a)), un periodo equivalente a una notifica ogni 30 secondi (ossia un observing load pari a 0.033 richieste al secondo) è sufficiente a saturare la banda dell'approccio *OBS-C*. Usando il proxy, invece, è stato possibile raggiungere frequenze di notifica molto più alte, soddisfacendo i requisiti di tutti i client. È interessante notare come *OBS-P* e *POLL* abbiano un comportamento molto simile mostrando piccole differenze solo a tassi di notifica molto elevati. Questo comportamento può essere riconducibile alle dimensioni dei pacchetti CoAP, ossia inviare una richiesta GET e ricevere una risposta di tipo *piggyback* nell'ACK di ritorno, come in *POLL*, sembra consumare leggermente più banda rispetto a *OBS-P*, in cui si ha la trasmissione di un pacchetto di notifica e il relativo ACK.

Osservando il caso $n_i = 10$. (Figura 19(b)), un periodo equivalente a una notifica ogni 2 minuti è sufficiente a causare il sovraccarico nella rete con la strategia *OBS-C*, mentre *OBS-P* e *POLL* mostrano un comportamento simile allo scenario precedente. Ciò si riconduce al fatto che, in *OBS-P* e *POLL*, data una risorsa r_i l'incremento del numero dei client interessati a riceverne aggiornamenti n_i , non si traduce in un aumento del numero di relazioni di binding verso r_i . Sulla base dei risultati dei test effettuati è stata selezionata la modalità di funzionamento *OBS-C* in quanto garantisce di minimizzare il sovraccarico della rete da parte di applicazioni di monitoraggio che usano sensori in una rete 6LoWPAN.

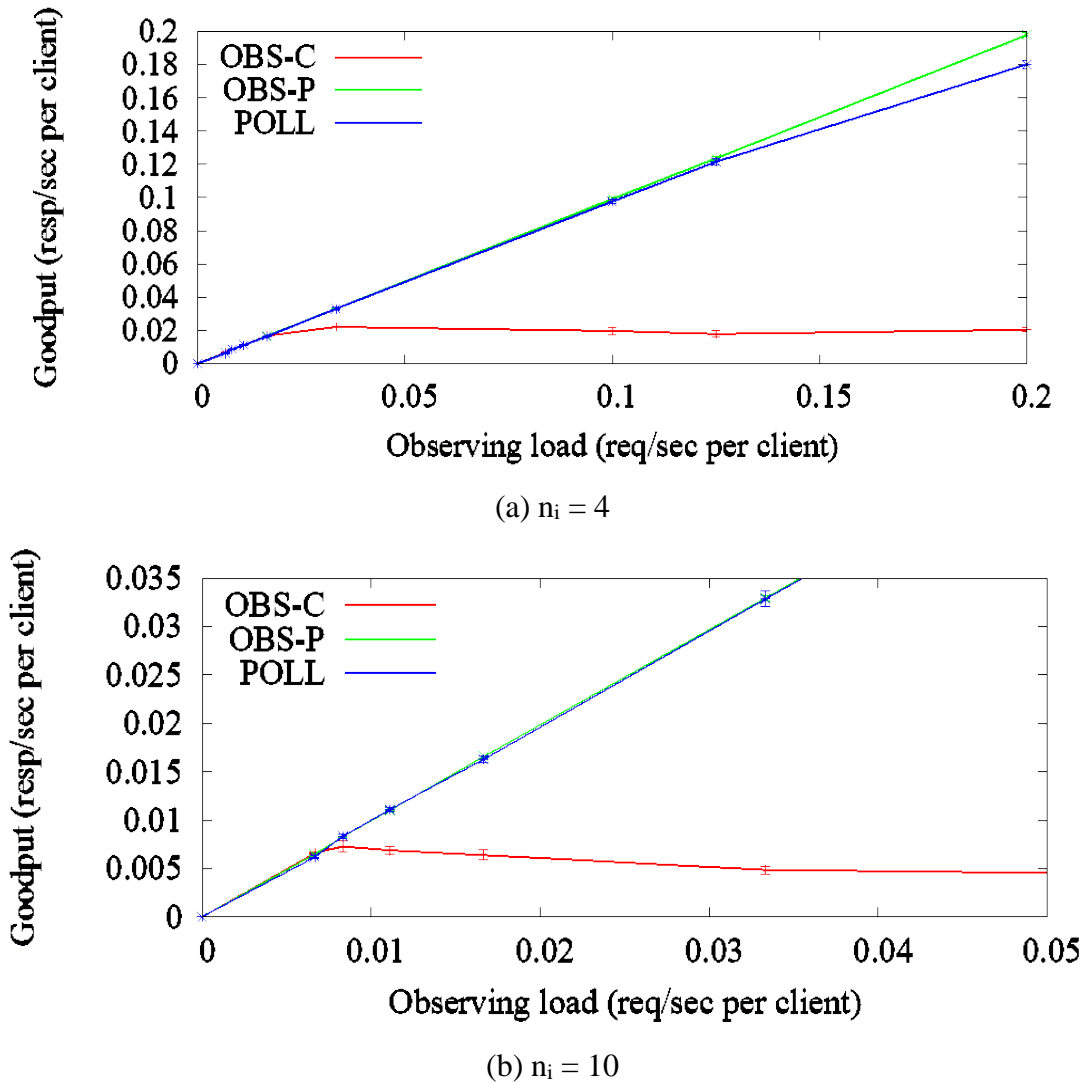


Figura 19: Goodput quando tutti i client richiedono lo stesso periodo di notifica dello stato aggiornato della risorsa.

Prototipo hardware della RD

In questo paragrafo si illustrerà il prototipo di sistema IoT integrante una resource directory che è stato predisposto nei laboratori dell'IIT.

Nello specifico, è stato realizzato un gateway IoT usando una Raspberry model Pi 3 B+¹ (Figura 20), un popolare sistema embedded, a basso costo e dal consumo energetico limitato, in grado di eseguire una distribuzione completa del sistema operativo Linux. Per quanto concerne i nodi della rete 6LoWPAN sono stati utilizzati dei sensori Zolertia RE-Mote² (Figura 21). Un nodo RE-mote è piattaforma hardware IoT popolare dal basso consumo e compatibile col sistema operativo ContikiOS³. Ciascun dispositivo è stato equipaggiato con un sensore di temperatura ed uno di luminosità ambientale. Le tabelle seguenti riassumono le caratteristiche tecniche principali delle due piattaforme.

¹ <https://www.raspberrypi.org/products/raspberry-pi-3-model-b-plus/>

² <https://zolertia.io/>

³ <http://www.contiki-os.org/>

Tabella 1: Caratteristiche hardware del componente Raspberry P3 B+

SoC	BCM2837B0 quad-core A53 (ARMv8) 64-bit @1,4 GHz
GPU	Broadcome Videocore-IV
RAM	1GB LPDDR2 SDRAM
Networking	- Gigabit Ethernet - 2.4 Ghz and 5 Ghz 802.11 b/g/n/ac Wi-Fi
Storage	Micro-SD 8 GB
GPIO	40-pin GPIO
Porte	HDMI, 4x USB 2.0
Corrente (Potenza) assorbita	800 mA (4,0 W)
Dimensioni	82mm x 56mm x19,5mm – 50g



Figura 20 – Raspberry P3 B+.

Tabella 2: Caratteristiche hardware del componente Zolertia RE-Mote

SoC	ARM Cortex-M3 32 Mhz
ROM	512 KB flash
RAM	32 KB
Networking	CC2538 radio: - ISM 2.4-GHz IEEE 802.15.4 compliant radio - ISM 863-950 MHz ISM/SRD band IEEE 802.15.4 compliant radio
Storage	Micro-SD
GPIO	-3.3 V ADC port 3-pin 2.54mm -5.1 V ADC port 3-pin 2.54mm -I2C/SPI Digital port 5-pin 2.54 mm
Porte	2x Micro USB (Debug and genal purpose)
Corrente assorbita	Da un minimo di 150 nA in shutdown mode
Dimensioni	73 mm x 40 mm



Figura 21 – Zolertia RE-Mote.

Dal punto di vista software, il firmware scritto per i nodi 6LoWPAN è basato su ContikiOS e sfruttando l'implementazione di CoAP (Erbium) disponibile per tale sistema operativo, i sensori di temperatura e luminosità sono stati rappresentati come risorse CoAP. Periodicamente, durante il ciclo di vita del nodo sensore, i descrittori di tali risorse vengono inviati alla resource directory.

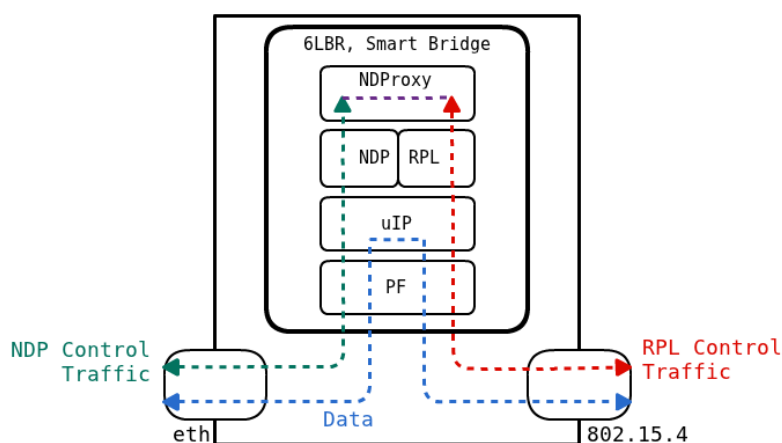


Figura 22 – Configurazione smart-bridge del modulo CETIC 6LBR.

Sul nodo Raspberry, invece, si è utilizzato il sistema operativo raspbian, una distribuzione di Linux derivata da Debian. Ciò consente l'installazione di una JVM e l'installazione del modulo java della resource directory sul gateway stesso. Inoltre, si è sfruttato il software CETIC 6LBR⁴ al fine di far agire il gateway come border router RPL. Tale applicativo, infatti, usato nella configurazione *smart-bridge* (illustrata in Figura 22), consente di interconnettere una rete standard IPv6 ad una rete mesh 6LoWPAN basata su RPL. L'architettura complessiva del prototipo hardware e software del gateway IoT è illustrata in Figura 23. In particolare, il ramo ethernet del nodo Raspberry viene connesso alla rete locale

⁴ <https://github.com/cetic/6lbr/wiki>

IPv6, mentre, un dongle USB viene utilizzato per connettersi alla rete 802.15.4 (è possibile sfruttare un nodo RE-Mote a tal proposito). A questo punto il 6LBR in modalità *smart-bridge* agisce da proxy per il protocollo IPv6 di neighbour discovery (NDP) [17] sul ramo Ethernet, e usa i parametri appresi da tale interfaccia per configurare la rete mesh RPL (ad esempio per la determinazione del prefisso da utilizzare).

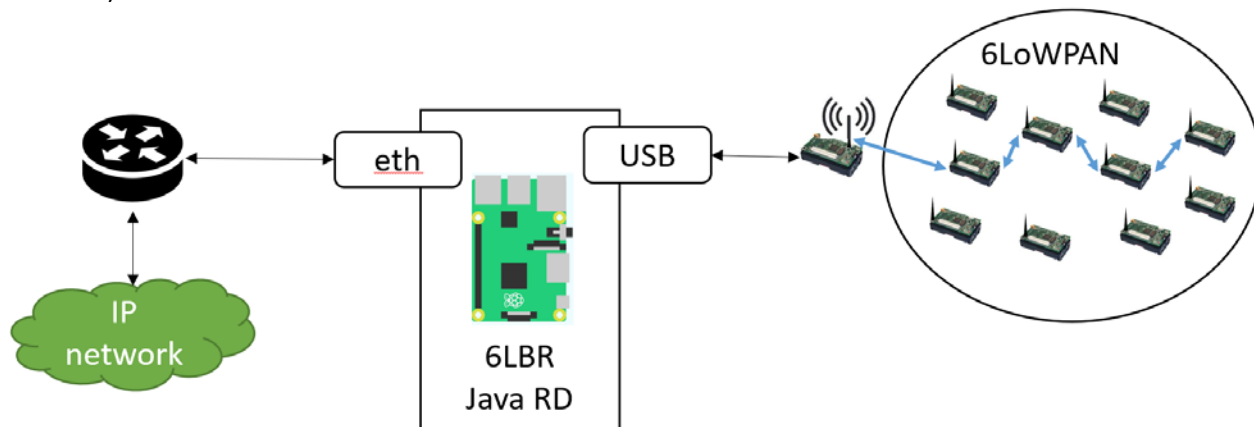


Figura 23 – Architettura del prototipo hardware e software del gateway IoT.

Gestione Distribuita dei Dati in reti IoT: Concetti, Algoritmi e Prototipo

Sinergia tra i progetti Ricerca di Sistema e H2020 AUTOWARE

Come indicato nella proposta, le attività relative alla gestione distribuita dei dati in reti IoT sono state svolte in sinergia con il progetto Europeo H2020 AUTOWARE, di cui IIT-CNR fa parte (<http://autoware-eu.org/>). In particolare, il contributo sviluppato tramite il progetto Ricerca di Sistema si è focalizzato sullo sviluppo di un ambiente sperimentale per la valutazione di algoritmi di gestione dei dati in ambito IoT industriale. All'interno del progetto AUTOWARE sono stati sviluppati gli algoritmi per la gestione dei dati, da testare nell'ambiente sperimentale sviluppato in RDS. Inoltre, AUTOWARE ha anche definito i benchmark per il confronto delle prestazioni ottenute utilizzando l'ambiente sperimentale, mentre in RDS sono state implementate le soluzioni complete degli algoritmi di data management (sia gli algoritmi nuovi sviluppati in AUTOWARE che i benchmark individuati) all'interno dell'ambiente sperimentale sviluppato.

Si sottolinea che l'ambiente sperimentale realizzato in RDS è un componente fondamentale di questa attività congiunta con AUTOWARE, ma il suo contributo è più generale. In particolare, tale ambiente sperimentale può essere utilizzato per testare in generale algoritmi di gestione dei dati in ambiente IoT diversi da quelli proposti in AUTOWARE, in maniera flessibile e *plug&play*.

Infine, si sottolinea anche che i componenti implementati nell'ambiente sperimentale sono conformi all'architettura IoT IETF illustrata nelle precedenti sezioni di questo rapporto tecnico, e quindi è possibile integrare i componenti fin qui descritti all'interno di tale ambiente sperimentale.

Per chiarezza espositiva, nel seguito della sezione viene prima illustrata l'architettura logica dei sistemi per la gestione dei dati considerati come riferimento per lo sviluppo dell'ambiente sperimentale. Vengono quindi illustrati i componenti dell'ambiente sperimentale sviluppato in RDS, ed alcuni risultati ottenuti confrontando le prestazioni degli algoritmi sviluppati in AUTOWARE con i benchmark di riferimento per quel progetto. In particolare, l'ambiente sperimentale è stato sviluppato su IoT-LAB (<https://www.iot-lab.info/>), un ambiente di testbed IoT pubblico, aperto alla comunità scientifica internazionale. I risultati ottenuti dimostrano la superiorità degli algoritmi sviluppati rispetto ai benchmark considerati.

Algoritmi di Data Management per reti IoT industriali

Le reti IoT industriali sono generalmente utilizzate per il monitoraggio di sistemi e il supporto di circuiti di controllo, nonché per sistemi di rilevamento del movimento, controllo di processo e automazione industriale. A tal fine, i dati generati dal monitoraggio dei dispositivi IoT vengono raccolti, elaborati e inviati a controllori e attuatori. Il concetto di una tipica rete IoT in scenari industriali è mostrato nella Figura 24. L'instradamento dei dati dai sensori IoT agli attuatori è parte integrante di qualsiasi rete industriale su larga scala per rispettare requisiti sul ritardo nella consegna dei dati stessi, che spesso sono critici per le applicazioni citate. Per mantenere la stabilità e controllare le prestazioni, le applicazioni di monitoraggio e controllo industriale impongono requisiti di ritardo *end-to-end* rigorosi sulla comunicazione dei dati tra sensori e attuatori. Mancare la consegna o ritardare i dati di processo può peggiorare gravemente la qualità del controllo, fino a causare gravi danni alle cose e alle persone, in casi estremi. Negli ultimi anni sono stati emessi da organismi internazionali numerosi standard per supportare lo sviluppo di reti industriali in diversi domini applicativi, come IEEE 802.15.4e (già illustrato in questo documento) e WirelessHART.

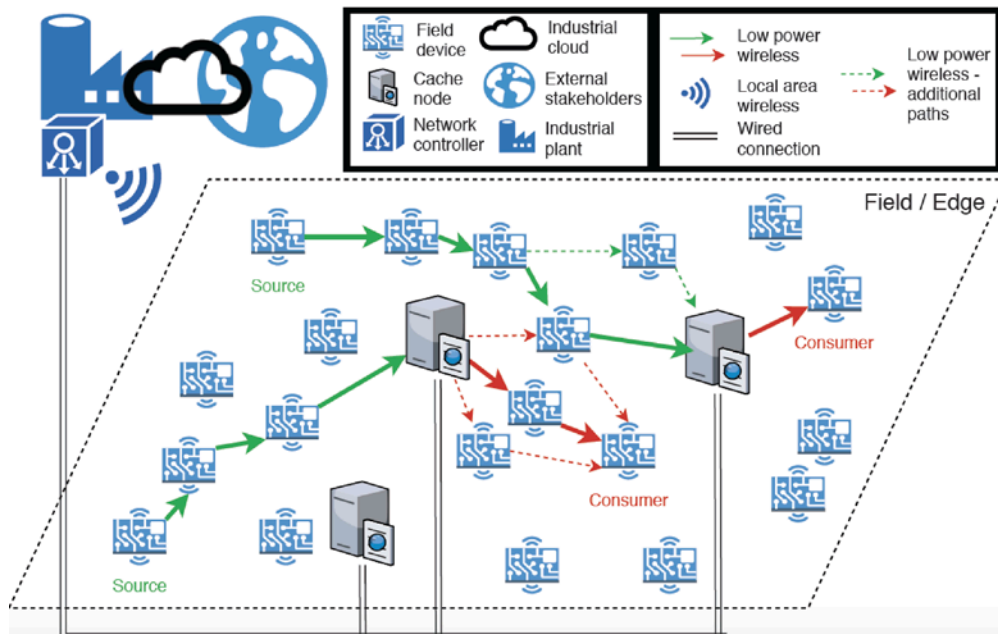


Figura 24. Esempio di rete e flusso di dati per applicazioni IoT industriali.

Negli algoritmi tradizionali di gestione dei dati per applicazioni industriali, vengono generalmente utilizzati schemi centralizzati, in base ai quali i dati vengono trasferiti ad un controller centrale, dal quale sono accessibili da qualsiasi altro nodo che li richiede. Il controller analizza le informazioni ricevute e, se necessario, modifica il comportamento dell'ambiente fisico tramite dispositivi attuatori. Tuttavia, l'instradamento (*routing*) dei dati basato solo su tale nodo centrale e l'imposizione di trasferimenti di dati avanti e indietro nella rete possono portare a percorsi gravemente non ottimali, che a loro volta influiscono negativamente sulla latenza complessiva del trasferimento dati in rete. Allo stesso tempo, questi schemi di trasferimento portano a prestazioni di rete scadenti, poiché i dispositivi spesso devono tollerare tempi di risposta più lunghi del necessario. Questo problema sta diventando ancora più serio con il decentramento delle decisioni di controllo, grazie al quale non solo un controller centrale, ma anche altri nodi della rete possono prendere decisioni su come configurare i dispositivi e gli attuatori IoT.

In questo lavoro, è stato definito un *Data Management Layer* (DML) per superare il suddetto problema, che viene realizzato sfruttando il set completo di dispositivi IoT disponibili nella rete, e non il solo nodo centrale, per la gestione dei dati stessi. Il DML è disaccoppiato ma interagisce con il piano di rete sottostante. Specificamente, considerando un generico insieme di dati, i gruppi di nodi che li generano e li richiedono e un ritardo massimo di accesso L_{max} che i nodi richiedenti possono tollerare, il DML identifica in modo efficiente i nodi *proxy* nella rete che dovranno memorizzare localmente i dati dei nodi produttori, fornendoli quando richiesti ai nodi consumatori. Dati i vincoli menzionati, il DML trova i nodi di rete da selezionare come *proxy* in modo ottimale, cioè quelli che, garantendo il rispetto dei vincoli di latenza massima, garantiscono anche il consumo minimo delle risorse energetiche dei nodi stessi. Anche questo aspetto è fondamentale, visto che le reti IoT industriali sono sempre più costituite da un alto numero di dispositivi miniaturizzati alimentati da batterie, ed è molto importante garantire che tali nodi durino il più a lungo possibile per limitare interventi di manutenzione continui.

Grazie all'ambiente sperimentale sviluppato nel progetto è stato possibile dimostrare che il metodo proposto basato su DML (i) garantisce che la latenza di accesso ai dati rimane al di sotto della soglia data e (ii) supera gli approcci centralizzati e persino distribuiti, sia in termini di latenza di accesso che in termini di massima garanzia di latenza.

Il Data Management Layer (DML)

Per gestire il processo di distribuzione dei dati e ridurre la latenza media nella rete, è stato introdotto il Data Management Layer (DML), illustrato in *Figura 25*. L'idea principale alla base del DML è il disaccoppiamento del piano di rete dal piano di gestione dati. La figura seguente mostra la struttura del DML. La funzione di base del DML è la selezione di alcuni nodi, che fungeranno da nodi *proxy*, e la definizione di tecniche efficienti per la distribuzione e la consegna dei dati. Si ricorda che il ruolo di *proxy* consiste nel memorizzare i dati provenienti da uno o più nodi che generano dati, in attesa che i nodi che useranno tali dati ne facciano richiesta.

Più specificamente, il ruolo del DML è duplice:

- Fornire un metodo efficiente di distribuzione e consegna dei dati: il DML impone alcune regole sulla circolazione dei dati alla rete, in base ai *proxy* disponibili, utilizzando i principi del meccanismo di *routing* sottostante nel livello rete.
- Identificare i nodi di rete che opereranno come *proxy*: il DML definisce un insieme di nodi *proxy* selezionati. Il numero dei *proxy* può variare da 1 fino al numero dei nodi nella rete che può fungere da *proxy*. Il caso di un *proxy* equivale ad avere solo il controller di rete che opera come un singolo punto di distribuzione dei dati. In questo caso, la distribuzione dei dati funziona come negli ambienti IoT industriali tradizionali.

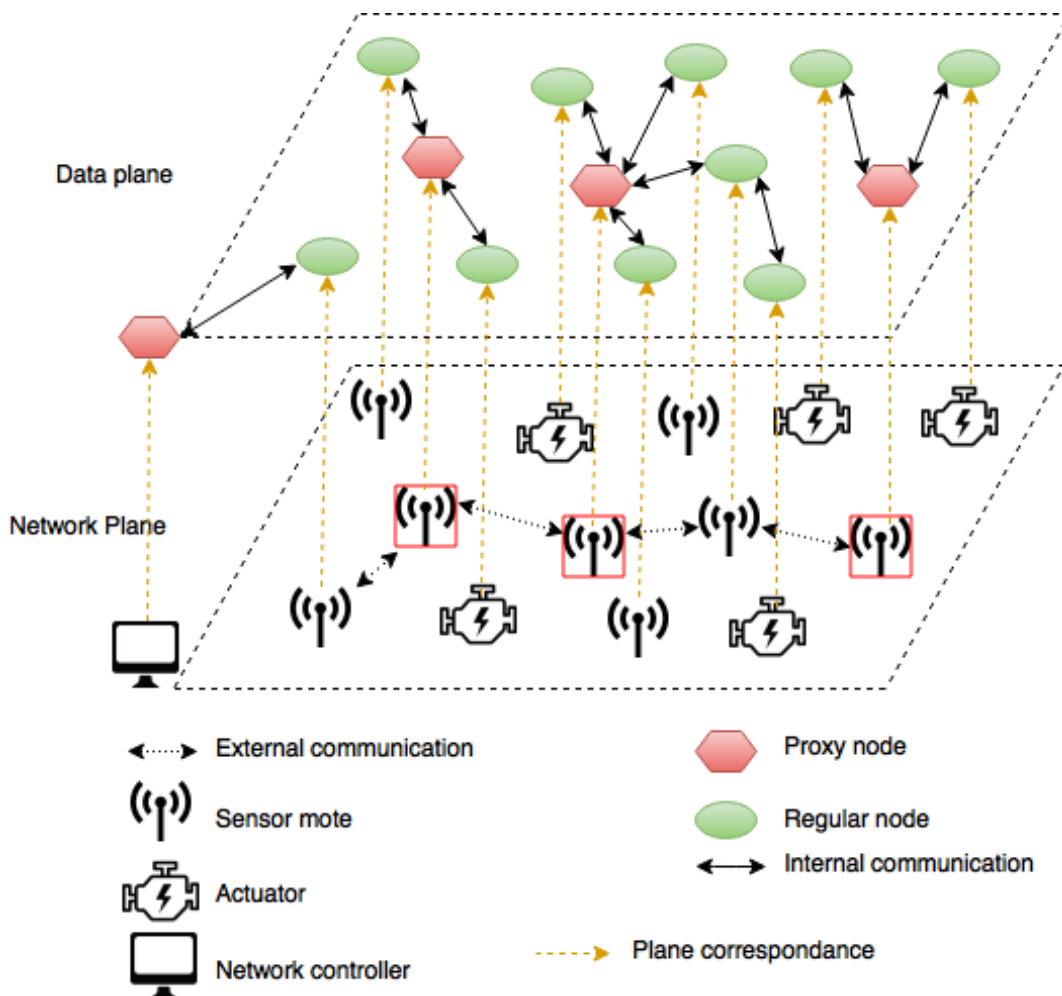


Figura 25. Schema logico del Data Management Layer.

Il processo di consegna dei dati è suddiviso in due parti, ognuna delle quali utilizza il meccanismo di *routing* standard fornito dal *Network Plane* (come, ad esempio, i protocolli dello standard IETF illustrati

nelle precedenti sezioni di questo rapporto tecnico). Una prima parte (logica) si occupa della comunicazione “interna”, tramite cui un nodo può richiedere dati di interesse al *proxy* corrispondente. Dopo aver ricevuto la richiesta da un tale nodo, il *proxy* è in grado di inoltrare direttamente i dati al consumatore, se questi sono già disponibili. Ciò significa che i dati in generale saranno arrivati al *proxy* dal nodo di origine corrispondente in precedenza. Una seconda parte (logica) si occupa della comunicazione “esterna”, tramite cui un *proxy* è in grado di ricevere e memorizzare i dati da un nodo di origine.

Questo modello di scambi di dati è formulato come un’istanza del generico paradigma pubblicazione/sottoscrizione (*publish/subscribe*, o *pub/sub*). In un modello *pub/sub*, i nodi interessati ai dati si iscrivono ad essi, cioè, rendono evidente il loro interesse all’interno della rete generando una “sottoscrizione” per quei dati, mentre i nodi che offrono i dati “pubblicano” annunci in rete sulla disponibilità di tali dati. All’interno della rete, i *proxy* sono responsabili di far incontrare le sottoscrizioni con le pubblicazioni, ovvero forniscono una funzione di *rendez-vous* e memorizzano i dati disponibili in base alle sottoscrizioni corrispondenti. Il modello *pub/sub* che usiamo è mostrato in Figura 26.

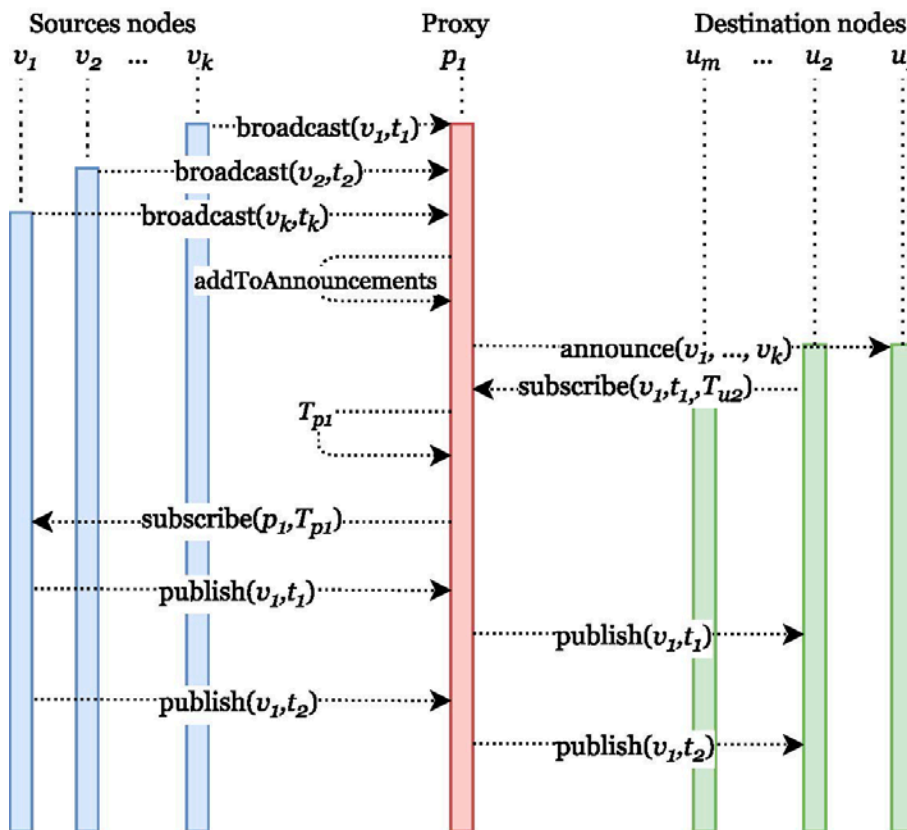


Figura 26. Modello *pub/sub* nel Data Management Layer.

La forza del modello di comunicazione *pub/sub* utilizzato deriva dal fatto che le operazioni di pubblicazione e sottoscrizione sono disaccoppiate nel tempo e nello spazio. La comunicazione tra un nodo che pubblica un dato ed un sottoscrittore non deve essere sincronizzata nel tempo, cioè il nodo che pubblica può pubblicare i dati prima che i sottoscrittori lo abbiano richiesto e possano avviare richieste per tali dati (ai *proxy*) dopo gli annunci di pubblicazione. I nodi che pubblicano i dati di solito non hanno riferimenti diretti ai nodi sottoscrittori, né sanno quanti sottoscrittori ricevono un determinato dato.

Le caratteristiche principali che fornisce l'approccio con i *proxy* secondo il modello *pub/sub* sono:

- i. evitare che dei dati vengano trasmessi più volte dalla sorgente verso le destinazioni servite dallo stesso *proxy*;

- ii. la consegna proattiva dei dati generati dalle sorgenti ai *proxy*, in modo tale che quando una richiesta di dati proviene da un nodo di destinazione, può essere servita in un tempo breve dal *proxy* in cui è memorizzato.

Una rappresentazione visiva della caratteristica (i) è illustrata nella Figura 27, dove è mostrato un esempio puramente illustrativo con due casi estremi di comunicazione esterna in una rete di 9 nodi. Nel primo caso, tutti i nodi di destinazione necessitano di dati da un singolo nodo di origine mentre nel secondo caso, tutti i nodi di destinazione necessitano di dati provenienti da diversi nodi di origine. È evidente che la quantità di comunicazione può essere significativamente ridotta nel primo caso, se viene utilizzato un metodo di raggruppamento appropriato per le consegne di dati basato su un nodo *proxy*.

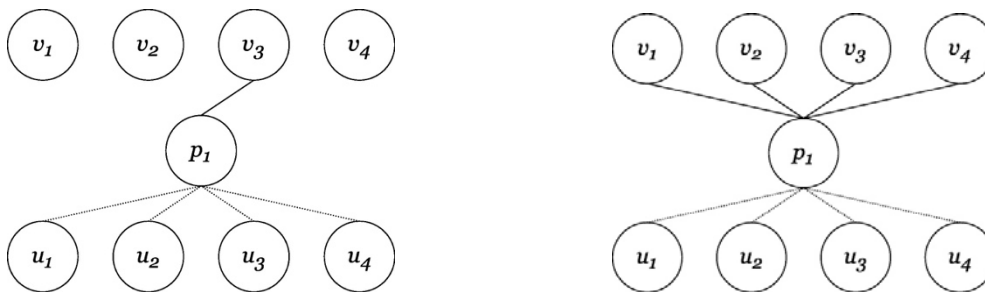


Figura 27. Vantaggi dell'approccio pub/sub e dell'uso di proxy.

Ambiente sperimentale

Al fine di misurare i benefici del DML, abbiamo sviluppato un ambiente sperimentale all'interno del progetto RDS. Lo scopo strategico della valutazione sperimentale con dispositivi reali è fornire una dimostrazione realistica di come un'efficace gestione dei dati utilizzando i metodi DML può migliorare significativamente la latenza di accesso ai dati nelle reti IoT industriali, utilizzando un numero limitato di *proxy*. Come anticipato più volte, l'ambiente sperimentale è generale ed allineato con gli standard IETF descritti in precedenza, e quindi può essere utilizzato sia per la valutazione del DML, ma più in generale per la valutazione di diversi algoritmi di gestione dati in reti IoT industriali.

Nel seguito della sezione vengono descritti i componenti più significativi dell'ambiente sperimentale sviluppato.

Il testbed Euratech

Per lo sviluppo dell'ambiente sperimentale, abbiamo utilizzato la piattaforma IoT-LAB, una raccolta su larga scala di *testbed* per reti IoT, gestiti dal CNRS francese e dagli istituti di ricerca INRIA. I *testbed* IoT-LAB sono situati in sei diversi siti in tutta la Francia, che forniscono l'accesso a 2728 nodi IoT wireless. Ai fini di questo lavoro, abbiamo utilizzato il *testbed* Euratech di IoT-LAB situato a Lille, in Francia. Il *testbed* Euratech è utilizzato nello showroom Inria Lille - Nord Europa e 224 nodi sono distribuiti come segue: due strati orizzontali in griglia di 5x19 nodi ciascuno e 34 nodi collegati a una parete, a una distanza di 0,60 m l'uno dall'altro. La Figura 28 mostra una fotografia del testbed, dove sono visibili i due strati orizzontali ed i nodi attaccati alla parete.

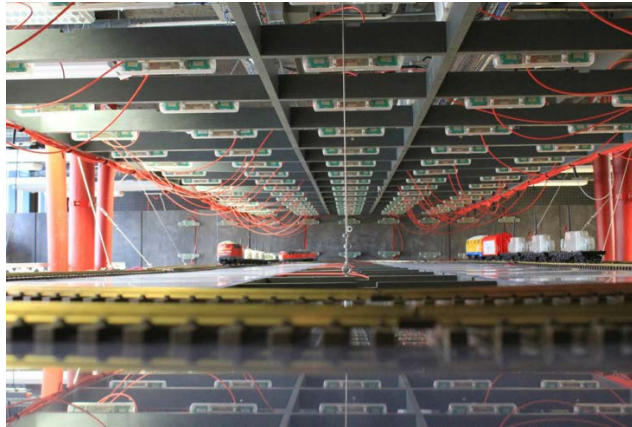


Figura 28. Euratech testbed.

I nodi IoT

I nodi utilizzati per lo sviluppo dell'ambiente sperimentale sono WSN430 Open Node. Il nodo WSN430 è basato su tecnologia Mote, a sua volta basata su una piattaforma MSP430 a bassa potenza, con un'interfaccia radio ISM completamente funzionale e un set di sensori standard. Per quanto riguarda la tecnologia radio, sono state sviluppate due versioni: la versione 1.3b presenta un'interfaccia radio aperta a 868 MHz, mentre la versione 1.4 ha un'interfaccia radio IEEE 802.15.4 a 2,4 GHz, ed utilizza un'antenna CC2420. Dato che ci stiamo concentrando su applicazioni IoT industriali, utilizziamo naturalmente la versione 1.4, che supporta le impostazioni IEEE 802.15.4e (descritta in precedenza nel documento) e WirelessHART, altro standard di riferimento per reti IoT industriali. Il design dei nodi è visualizzato nella *Figura 29*.

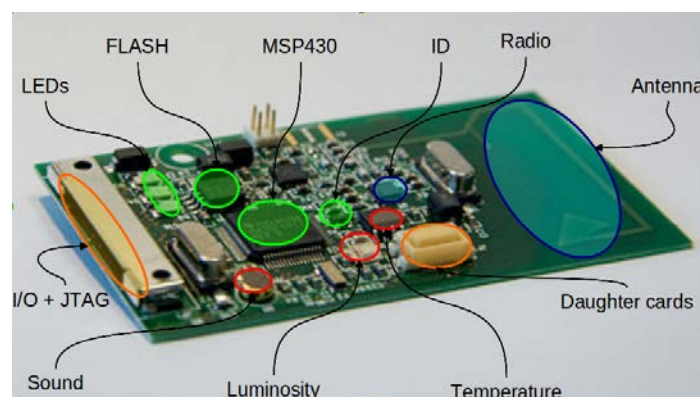


Figura 29. Nodi IoT del testbed.

Per gli scopi del progetto, è sufficiente considerare, senza perdita di generalità, una distribuzione 2D dei nodi IoT. Quindi, abbiamo utilizzato solo un sottoinsieme dei nodi disponibili nel *testbed* Euratech e in particolare uno dei due livelli orizzontali di nodi. Per le valutazioni sperimentali in particolare, abbiamo riservato 95 nodi. Chiaramente, l'ambiente sperimentale sviluppato può essere esteso senza necessità di modifiche ad un numero arbitrario di nodi del *testbed*. La *Figura 30* mostra un esempio dei nodi utilizzati in fase di valutazione, come viene visualizzata nella piattaforma IoT-LAB. I punti viola corrispondono ai nodi riservati e i punti verdi corrispondono al resto dei nodi del *testbed*. Si noti che, occasionalmente, ci sono anche punti rossi che corrispondono a nodi temporaneamente non funzionanti, ovvero nodi che hanno esaurito l'energia disponibile e non sono in grado di funzionare. Questa occasionale indisponibilità di un sottoinsieme di nodi rende gli esperimenti ancora

più realistici, poiché le situazioni di nodi non funzionanti si verificano frequentemente nelle reti IoT industriali reali.

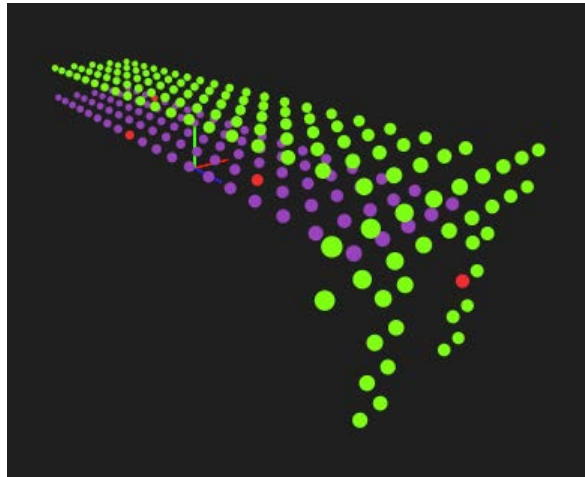


Figura 30. Esempio di topologia durante gli esperimenti.

Note sul sistema operativo dei nodi IoT

Abbiamo programmato e gestito i nodi tramite il sistema operativo TinyOS 2.1.1 (<http://www.tinyos.net/>), un sistema operativo di riferimento per reti IoT. TinyOS è un sistema operativo integrato basato su componenti per dispositivi wireless a bassa potenza, come quelli utilizzati nelle applicazioni IoT industriali. Una comunità mondiale di università e industrie utilizza, sviluppa e supporta il sistema operativo e gli strumenti associati, con una media di 35.000 download all'anno. TinyOS è scritto nel linguaggio di programmazione nesC, un'estensione del linguaggio di programmazione C progettato per incorporare i concetti di strutturazione e il modello di esecuzione di TinyOS, come un insieme di attività e processi cooperanti.

I programmi TinyOS sono costituiti da componenti software, alcuni dei quali presentano astrazioni hardware. I componenti sono collegati tra loro tramite interfacce. TinyOS fornisce interfacce e componenti per astrazioni comuni come comunicazione a pacchetti, *routing*, rilevamento, attuazione e archiviazione dati.

TinyOS è completamente non bloccante, visto che è costituito da un unico *stack* di chiamate. Pertanto, tutte le operazioni di input/output (I/O) che durano più di qualche centinaio di microsecondi sono asincrone e sono implementate tramite il meccanismo delle *callback*. Per consentire al compilatore nativo di ottimizzare al meglio a cavallo di diverse invocazioni di funzione, TinyOS utilizza le funzionalità di nesC per collegare questi callback, chiamati eventi, staticamente. Pur essendo non bloccante, TinyOS consente di mantenere un'elevata concorrenza, forzando i programmatori a scrivere logica complessa tramite l'interazione di un insieme di gestori di eventi di base. Per supportare calcoli più articolati, TinyOS offre l'astrazione dei *task*, simili ad una chiamata di procedura differita. Un componente TinyOS può pubblicare un *task*, che il sistema operativo programmerà per essere eseguito successivamente. I *task* non sono *pre-emptive* (cioè non possono bloccare l'esecuzione di *task* precedentemente in esecuzione) e vengono eseguiti secondo un ordine FIFO. Questo semplice modello di concorrenza è in genere sufficiente per le applicazioni con preponderanza di attività di I/O. Tuttavia, le difficoltà di un tale modello per applicazioni con carichi computazionali significativi hanno portato allo sviluppo di una libreria di *thread* per il sistema operativo, denominata TOSThreads.

Il codice TinyOS è collegato staticamente al codice del programma ed è compilato in un piccolo file binario, utilizzando una toolchain GNU personalizzata. Le utility associate sono fornite per completare una piattaforma di sviluppo per lavorare con TinyOS.

Note sul linguaggio di programmazione nesC

I programmi nesC sono costruiti a componenti, che sono assemblati per formare interi programmi. I componenti hanno una concorrenza interna sotto forma di *tasks*. I thread di controllo possono passare in un componente attraverso le sue interfacce. Questi thread originano o da un task o da un interrupt hardware. Le interfacce possono essere fornite o utilizzate dai componenti. Le interfacce fornite rappresentano la funzionalità che il componente fornisce all'utente, le interfacce utente rappresentano la funzionalità necessaria al componente per svolgere il proprio lavoro.

In nesC, le interfacce sono bidirezionali: specificano un insieme di funzioni che devono essere implementate dal provider dell'interfaccia (comandi) e un set che deve essere implementato dall'utente (eventi) dell'interfaccia. Ciò consente a una singola interfaccia di rappresentare una complessa interazione tra componenti (ad esempio, la registrazione di interesse in alcuni eventi, seguita da una chiamata ad una funzione di *callback* quando si verifica quell'evento). Questo è fondamentale perché in TinyOS tutti i comandi di una certa complessità funzionale (ad esempio, il pacchetto di invio) non sono bloccanti; il loro completamento è segnalato attraverso un evento (invio fatto). Specificando le interfacce, un componente non può chiamare il comando *send()* a meno che non fornisca un'implementazione dell'evento *sendDone*. In genere, i comandi chiamano “verso il basso”, vale a dire, dai componenti dell'applicazione a quelli più vicini all'hardware, mentre gli eventi causano esecuzione di funzioni “verso l'alto”. Alcuni eventi primitivi sono legati agli interrupt hardware. I componenti sono collegati in modo statico l'uno all'altro tramite le loro interfacce. Ciò aumenta l'efficienza di runtime, incoraggia la progettazione robusta e consente una migliore analisi statica dei programmi.

Sulla base di questi paradigmi, sono stati implementati gli algoritmi di gestione dei dati che implementano il DML, progettati in AUTOWARE. L'ambiente sviluppato permette comunque di implementare e testare generici meccanismi di data management, che possono essere installati nel testbed secondo la procedura descritta nella seguente sezione.

Installazione degli algoritmi nei nodi testbed

Abbiamo implementato in nesC gli algoritmi DML ed i benchmark descritti in sezione “Soluzioni di confronto sperimentale”. Abbiamo compilato il codice nesC separatamente per ogni nodo WSN40, seguendo l'identificazione standard dei nodi di Euratech testbed usando il comando *make wsn430v14 id,X command*, dove *X* è l'identificativo del nodo corrispondente. Abbiamo “flashedo” i nodi del testbed (cioè, abbiamo installato sui nodi il programma così compilato) tramite il portale web della piattaforma e abbiamo utilizzato le API REST offerte dal testbed e gli strumenti di misurazione a riga di comando. Un esempio di questo processo è mostrato nella *Figura 31* (per i nodi M3 di IoT-LAB).

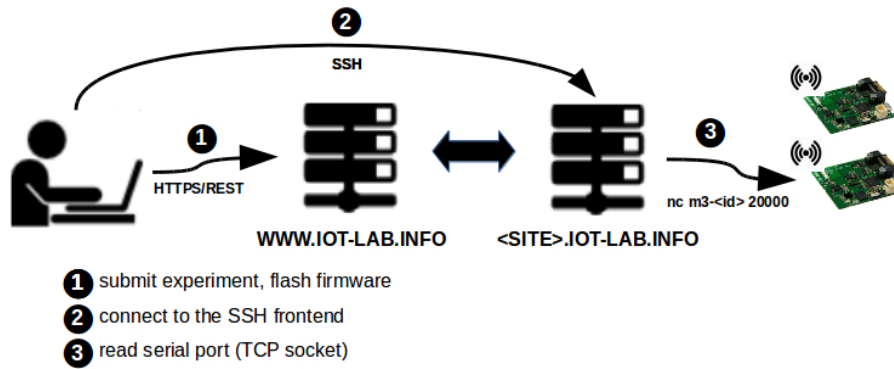


Figura 31. Processo concettuale di configurazione dei nodi Euratech.

Poiché i nodi del testbed sono posizionati a breve distanza l'uno dall'altro, abbiamo regolato il loro raggio di trasmissione, in modo da ottenere una topologia multi-hop realistica. La potenza di uscita RF dell'antenna CC2420 di WSN430 è programmabile ed è controllata da TinyOS attraverso un registro specifico. Date le distanze dei nodi di rete e il fatto che idealmente desideriamo una topologia di rete multi-hop, abbiamo settato la direttiva preprocessore *CC2420_DEF_RFPOWER* a 3, in modo da ottenere, in base al datasheet CC2420, una potenza in uscita di - 25dBm (come mostrato in Figura 32), che a sua volta genera un raggio di trasmissione di circa 3m. Tuttavia, dato che questo valore è stato misurato in condizioni ideali, senza tener conto di fattori esterni come ostacoli e interferenze, assumiamo ulteriormente che il raggio di trasmissione possa essere sia inferiore a questo valore.

PA_LEVEL	TXCTRL register	Output Power [dBm]	Current Consumption [mA]
31	0xA0FF	0	17.4
27	0xA0FB	-1	16.5
23	0xA0F7	-3	15.2
19	0xA0F3	-5	13.9
15	0xA0EF	-7	12.5
11	0xA0EB	-10	11.2
7	0xA0E7	-15	9.9
3	0xA0E3	-25	8.5

Figura 32. Parametri di configurazione dell'antenna CC2420.

Data questa configurazione, si ottiene una topologia che è rappresentata nella Figura 33. Si noti che i tre "spazi vuoti" nella topologia sono il risultato dei nodi non funzionanti della distribuzione (punti rossi nella Figura 30) che non sono in grado di comunicare con altri nodi.

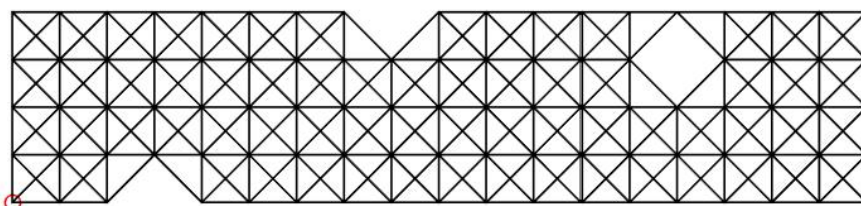


Figura 33. Topologia di rete.

Data questa topologia, i proxy assegnati tramite DML e i nodi destinatari dei dati nella rete sono indicati nella Figura 34. I punti rossi rappresentano i nodi proxy e i punti verdi i nodi destinatari.

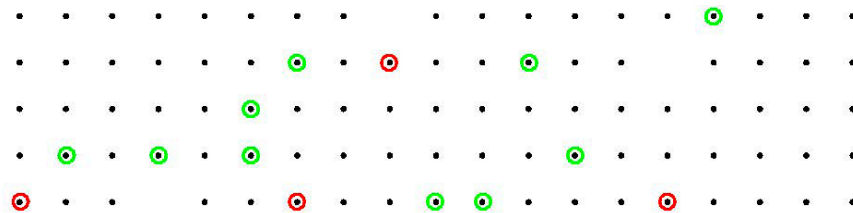


Figura 34. Configurazione del DML: nodi proxy (rossi) dati i nodi destinatari dei dati (verdi).

Si fa presente che tale topologia e configurazione è un esempio specifico di quello che si può ottenere nel nostro ambiente sperimentale. A seconda dei requisiti sperimentali, è sufficiente modificare il numero di nodi sorgenti/destinatari, il set di destinatari, i requisiti in termini di latenza massima per la consegna dei dati, per ottenere la corrispondente configurazione del DML.

Valutazioni sperimentali

In questa sezione sono presentati i dettagli della metodologia di misurazione implementata nell'ambiente sperimentale considerato. Sulla base di questa metodologia, come esempio concreto, sono mostrati alcuni risultati ottenuti misurando le prestazioni dei sistemi di gestione dei dati progettati in AUTOWARE.

Calibrazione: Latenza single-hop

In generale, la latenza nella consegna dei dati si può misurare o in numero di hop, o, più realisticamente, in tempo reale. Misurare la latenza di consegna dei dati end-to-end in numero di hop è un approccio significativo, poiché questa unità di misura esprime le prestazioni dei metodi, indipendentemente dall'implementazione e dalle restrizioni della tecnologia sottostante dei nodi e dei dispositivi che stiamo utilizzando, che può variare in modo significativo in base al produttore. Tuttavia, dato che con il testbed Euratech è possibile sperimentare le prestazioni in scenari reali, è interessante misurare la latenza ottenuta in unità di tempo reale e più specificamente in millisecondi (ms). Dal punto di vista del testbed e dei nodi componenti, questa misura viene ottenuta calcolando e riportando i tempi di richiesta dei dati e consegna dei singoli nodi. Tuttavia, per comprendere correttamente l'effetto dei nostri metodi nelle impostazioni del mondo reale, dobbiamo tradurre in tempo reale anche il valore di soglia L_{max} (nel caso in cui esso sia specificato in numero di hop) entro cui i dati devono essere consegnati.

A causa del fatto che i tempi di trasmissione e ricezione dei dati su un singolo nodo possono variare in modo significativo anche se ripetiamo lo stesso esperimento, la traduzione della latenza dal numero di hop a unità di tempo non è semplice. Al fine di acquisire una base per questa mappatura, abbiamo studiato i valori di alcune latenze rappresentative di una trasmissione tra due nodi soltanto (*single hop*). Abbiamo ottenuto le misurazioni mostrate nella seguente tabella (valori minimi, massimi e medi), dopo aver misurato i tempi rilevanti utilizzando direttamente le funzioni disponibili sui nodi WSN430 (tramite il supporto dei servizi del sistema operativo TinyOS) e facendo riferimento alla scheda tecnica CC2420.

	Valore	σ
Massima latenza single hop	23 ms	
Latenza media	17.4 ms	3.2
Minima latenza single hop	13 ms	
Tempo per la generazione di un campione di dato da parte del sensore	1 ms	
Ritardo base di comunicazione (meccanismo SFD)	3 μ s	

Inizialmente, è stato misurato il tempo necessario per l'acquisizione della lettura del sensore che usiamo anche nella valutazione delle prestazioni dei metodi. È stato scelto di utilizzare il sensore predefinito di WSN430 implementato in TinyOS dal componente *DemoSensorC*. *DemoSensorC* gestisce un dispositivo sensore generico che fornisce un valore a 16 bit. L'autore della piattaforma sceglie quale sensore si trova effettivamente dietro a *DemoSensorC*, e di solito è voltaggio, luce o temperatura. Questo particolare *DemoSensorC* sulla piattaforma WSN430 fornisce una lettura della temperatura, utilizzando il componente *TemperatureC*.

È stato misurato il tempo necessario per la lettura di un sensore misurando il tempo trascorso tra l'inizio del comando *read()* e la fine del comando *readDone()* di TinyOS, utilizzando un timer. Alla prima riga della tabella, si può notare che la lettura del sensore richiede sempre un tempo non superiore a 1 ms. Si noti che non è necessaria una misurazione a grana più fine, poiché l'ordine di grandezza dominante per le misurazioni del tempo sotto esame è dell'ordine del millisecondo.

Successivamente, è stato misurato il tempo necessario per la lettura del dato ottenuto dal sensore, da inviare e ricevere da un nodo a un altro. Questo ritardo include il tempo trascorso nella fase di *back-off* del protocollo MAC di trasmissione (che non può essere previsto), il tempo trascorso nell'invio del segnale via radio e il tempo trascorso durante la propagazione del segnale. Il ritardo di propagazione è stato considerato trascurabile, dal momento che è sicuramente molto minore dell'unità di misura dei tempi in questione (ms). Più specificamente, è stato misurato il tempo necessario per inviare una lettura del sensore misurando il tempo trascorso tra l'inizio del comando *send()* e la fine del comando *sendDone()* di TinyOS, entrambi implementati utilizzando l'interfaccia *AMSend*, utilizzando un timer.

Per ottenere risultati affidabili, sono state ripetute le misure di propagazione per diverse coppie di nodi di trasmissione e ricezione del testbed Euratech, per 30 volte per ciascun nodo. Nella tabella precedente, possiamo vedere che i valori di ritardo della propagazione dei dati da un nodo all'altro variano in modo significativo. Mentre il ritardo minimo può essere 13ms, abbiamo misurato istanze di ritardo di propagazione fino a 23ms. Il ritardo medio dei valori raccolti dalla ripetizione di questo esperimento è di 17,4ms.

Infine, per tenere conto di eventuali ulteriori ritardi che potrebbero influire sulla trasmissione della lettura del sensore, abbiamo consultato la scheda tecnica CC2420. Uno di questi casi è il ritardo di elaborazione nel ricevitore, che comprende il tempo trascorso da una trasmissione completa dell'SFD (*Start of Frame Delimiter*) fino alla ricezione completa dell'SFD, cioè da quando l'SFD si attiva sul trasmettitore fino a quando si attiva sul ricevitore. Come mostrato nella tabella, questo ritardo è pari a 3μs e quindi non influenza in modo significativo il ritardo di trasmissione complessivo.

Soluzioni di confronto sperimentale

Al fine di misurare le prestazioni del sistema di DML considerato rispetto alle tradizionali alternative IoT industriali, sono state implementate due ulteriori strategie di consegna dei dati. Il primo metodo è la strategia di consegna dei dati più tradizionale negli attuali ambienti IoT industriali e impone che tutte le richieste di dati vengano indirizzate al controllore centrale *C*, che è quindi responsabile di gestire tutte le richieste di accesso ai dati. In particolare, la richiesta viene instradata dal generico nodo consumatore *u* a *C* e quindi da *C* all'appropriato produttore *v*. Al passaggio successivo, i dati vengono instradati da *v* nuovamente a *C* e quindi da *C* a *u*. Tale modalità operativa è indicata come modalità "senza archiviazione", ed è ovvio che è completamente centralizzata e non cooperativa (non esiste nessuna collaborazione tra i nodi). Si noti che questo è l'approccio di gestione dei dati più semplice che può essere implementato in meccanismi di *routing* rilevanti come RPL (descritto in precedenza nel documento), in cui i nodi intermedi non sono autorizzati a memorizzare i dati nella cache (quindi, la modalità di "non-storing mode" della terminologia RPL).

Il secondo metodo considerato è meno comunemente usato nelle applicazioni IoT industriali, ma rappresenta comunque un'alternativa utile. Impone che tutte le richieste di dati e le consegne di dati vengano instradate attraverso il cosiddetto nodo LC (*Lowest Common Ancestor*) dell'albero di *routing* la cui *root* è sempre il controller C. L'LCA di due nodi *u* e *v* nell'albero di *routing* è il nodo più basso (ovvero, la più profondo, rispetto alla radice) che ha sia *u* che *v* come discendenti. Questa modalità è indicata come “*storing mode*”, perché i nodi LCA dovrebbero memorizzare ulteriori informazioni sui loro discendenti, ed è ovvio che si tratta di un'alternativa distribuita e collaborativa. Ancora una volta, questo è il metodo più semplice che si potrebbe implementare con meccanismi di *routing* come RPL in modalità “*storing mode*”, vale a dire quando i nodi intermedi tra gli endpoint di comunicazione sono autorizzati a memorizzare localmente il contenuto. Questa modalità fornisce quindi un metodo distribuito.

Le scelte fatte per quanto riguarda le soluzioni di confronto da utilizzare sono state fatte dopo un'attenta considerazione dell'attuale status-quo delle reti IoT industriali. I protocolli selezionati sono componenti standardizzati di uno stack di riferimento e ben consolidato per la gestione delle comunicazioni e dei dati. In effetti, sono considerati lo stato dell'arte, per le applicazioni industriali IoT attuali e future, come ampiamente discusso nella letteratura pertinente. Nello specifico, lo stack protocollare usato è considerato lo standard di fatto per le reti IoT industriali. Un esempio delle tre alternative (la soluzione completamente centralizzata attraverso il controllore di rete, la soluzione distribuita tramite nodi LCA, e la soluzione distribuita tramite il DML e quindi tramite nodi proxy) è mostrato in Figura 35.

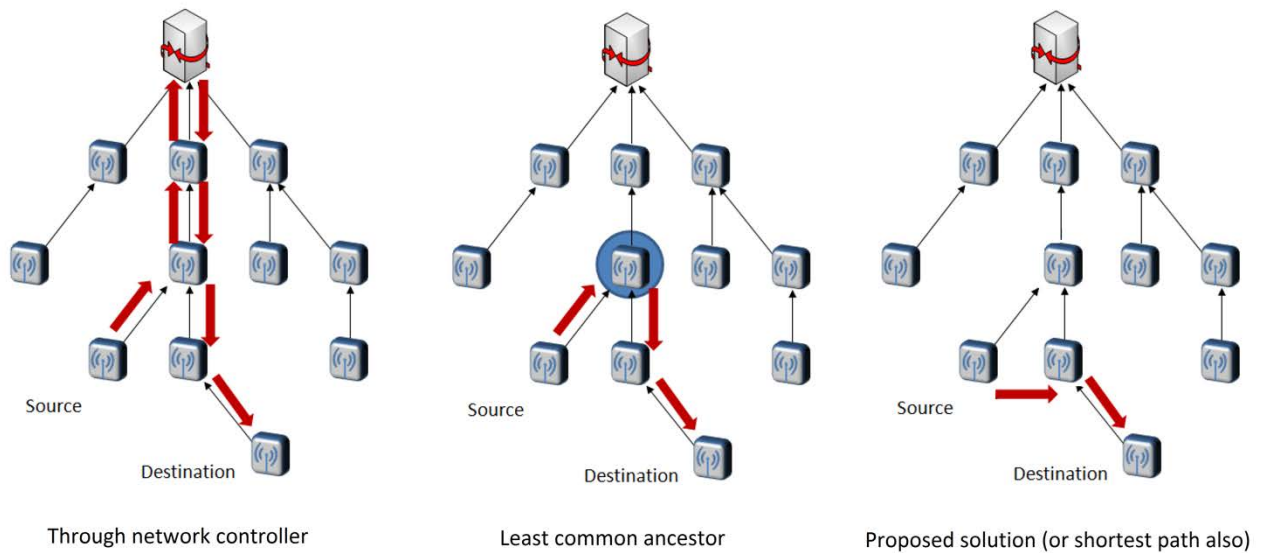


Figura 35. Visualizzazione schematica delle soluzioni considerate.

Un parametro fondamentale per gli esperimenti è il valore di *Lmax* considerato, cioè il valore massimo tollerabile nella latenza tra il nodo dove un certo dato è memorizzato, ed il nodo che tale dato ha richiesto. Per eseguire gli esperimenti nel modo più realistico, è importante che il valore di *Lmax* sia allineato con i requisiti di comunicazione di riferimento per applicazioni IoT industriali (in particolare, basandosi sugli scenari *Industry 4.0*). Sia il WG1 della Plattform Industrie 4.0 (architetture di riferimento, standards e norme) che il Comitato di esperti 7.2 di ITG (sistemi radio) stabiliscono come riferimento i requisiti di latenza per le applicazioni di monitoraggio industriale una latenza massima nell'ordine di 100 ms. Questo è il valore che è stato considerato negli esperimenti presentati nel seguito.

Risultati sperimentali

Come anticipato, in questa sezione si riportano alcuni dei risultati ottenuti confrontando le prestazioni dell'algoritmo di DML (sviluppato in AUTOWARE) all'interno dell'ambiente sperimentale sviluppato in RDS. Nel contesto di questo rapporto tecnico, tali risultati esemplificano l'utilizzo dell'output di RDS per la valutazione delle prestazioni di algoritmi di gestione dei dati per reti IoT industriali.

La Figura 36 mostra i risultati per quanto riguarda latenza di accesso ai dati per l'algoritmo basato su DML e i due metodi di confronto alternativi implementati nell'ambiente sperimentale sviluppato in RDS e descritti nella sezione "Soluzioni di confronto sperimentale" del rapporto tecnico. La barra gialla per il metodo DML è il valore di latenza quando consideriamo il caso peggiore di latenza single-hop misurato nella rete (la latenza più elevata riportata, in base alla tabella presentata in sezione "Calibrazione: Latenza single-hop"). Questo è un punto importante da considerare, poiché la figura mostra che il DML è in grado di garantire sempre il vincolo di massima latenza L_{max} (linea rossa orizzontale), indipendentemente dal fatto che la latenza sul singolo hop sia considerata dall'algoritmo come la media delle latenze medie (barra blu) o come una media delle latenze del caso peggiore (barra gialla). Si può osservare che la gestione efficiente dei dati fornita dal DML tramite la scelta oculata dei nodi *proxy* offre prestazioni migliori rispetto alle altre due alternative.

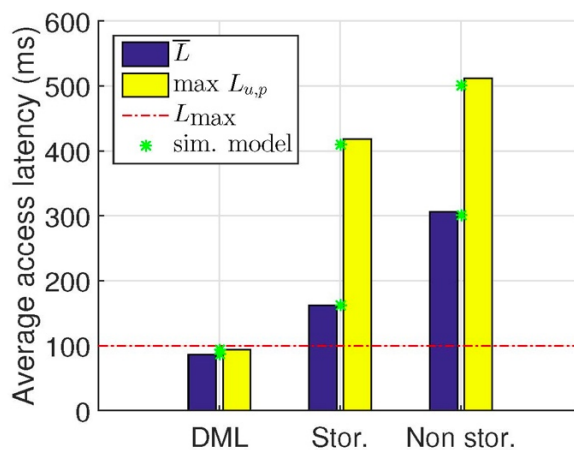


Figura 36. Prestazioni in termini di latenza.

Di seguito, confrontiamo anche i tre metodi rispetto al numero di "nodi speciali" che usano. Per il DML i nodi speciali sono i proxy; la modalità centralizzata utilizza solo il controller C; la modalità "storing mode" utilizza i nodi LCA. Questa metrica è importante perché l'uso di nodi speciali ha un impatto diretto sul consumo delle risorse di rete, e sulle risorse energetiche dei nodi (e, quindi, sulla durata del funzionamento dell'intera rete). Ad esempio, i *proxy* archiviano i dati richiesti, e la corrispondenza tra le richieste arrivate dai nodi consumatori ed i dati generati dai nodi produttori, mentre i nodi LCA conservano informazioni di routing sui nodi loro discendenti nell'albero di routing. Nella Figura 37 possiamo vedere che il DML funziona molto meglio rispetto alla modalità "storing mode", visto che utilizza molti meno nodi speciali. Ovviamente, la modalità centralizzata utilizza solo un nodo speciale, ma ciò ha un impatto grave sulla latenza ottenuta, come mostrato in Figura 36. Anche quando il DML utilizza più proxy per garantire latenze nel caso peggiore (cioè, quando viene considerato il peggiore ritardo possibile sui singoli hop, barra gialla), il loro numero è paragonabile al caso "storing mode". Tuttavia, in questo caso il DML riduce drasticamente la latenza (si veda ancora la Figura 36), ottenendo così un utilizzo molto più efficiente dei proxy e quindi della rete nel suo complesso.

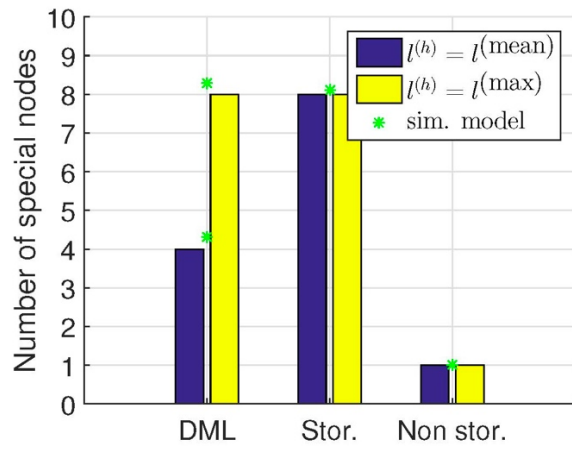


Figura 37. Prestazioni in termini di consumo di risorse.

Conclusioni

L'attività 1.1 ha riguardato la progettazione, sviluppo e testing di due prototipi hardware e software. Il primo prototipo è relativo ad un gateway IoT, che supporta le funzionalità di resource discovery e proxying di una rete 6LoWPAN, cioè una rete IoT che utilizza lo stack protocollare IETF. Il secondo prototipo, è un ambiente sperimentale per la valutazione di algoritmi distribuiti di gestione dei dati in reti IoT con requisiti sui tempi di consegna.

Le suddette attività proseguiranno nell'ambito della Linea Progettuale 1 del PAR 2016/2017. In linea con le attività descritte in questo documento si proseguirà con lo sviluppo del prototipo di gateway IoT seguendo due direzioni principali. In particolare, le funzionalità di proxy del gateway sviluppato verranno estese per permettere di gestire in modo efficiente le richieste simultanee di acquisizione dei dati IoT che sono generate da molteplici applicazioni. Inoltre sarà sviluppato un connettore per permettere al gateway IoT di connettersi a piattaforme middleware di pubblicazione/sottoscrizione basate sul protocollo MQTT. Infine, per validare l'efficacia del prototipo di piattaforma di comunicazione sviluppato in questa linea progettuale, il prototipo sarà utilizzato per realizzare un servizio di monitoraggio dell' "energy footprint" di un edificio commerciale.

Riferimenti bibliografici

- [4] CISCO, “Cisco visual networking index: Global mobile data traffic forecast update 2015–2020 white paper,” CISCO, Tech. Rep., 2016.
- [5] P. G. Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere, “Edge-centric computing: Vision and challenges,” *ACM Sigcomm Computer Communication Review*, 2015.
- [6] Eleonora Borgia, “The Internet of Things vision: Key features, applications and open issues”, in *Computer Communications* 54: 1-31 (2014).
- [7] Marco Conti, Sajal K. Das, Chatschik Bisdikian, Mohan Kumar, Lionel M. Ni, Andrea Passarella, George Roussos, Gerhard Tröster, Gene Tsudik, Franco Zambonelli, “Looking ahead in pervasive computing: Challenges and opportunities in the era of cyber–physical convergence”, in *Pervasive and Mobile Computing*, Vol. 8, N. 1, 2012, Pages 2-21.
- [8] L. Atzori, A. Iera, G. Morabito, The internet of things: a survey, *Comp. Netw.: 2599 Int. J. Comp. Telecommun. Network.* 54 (15) (2010) 2787–2805.
- [9] Maria Rita Palattella, Nicola Accettura, Xavier Vilajosana, Thomas Watteyne, Luigi Alfredo Grieco, Gennaro Boggia, Mischa Dohler, “Standardized Protocol Stack for the Internet of (Important) Things”. *IEEE Communications Surveys and Tutorials*, 15(3): 1389-1406 (2013).
- [10] Z. Shelby, K. Hartke, and C. Bormann, “The Constrained Application Protocol (CoAP),” IETF RFC 7252, June 2014. [Online]. Available: <http://www.ietf.org/rfc/rfc7252.txt>.
- [11] K. Hartke, “Observing Resources in the Constrained Application Protocol (CoAP),” IETF RFC 7641, September 2015. [Online]. Available: <http://www.ietf.org/rfc/rfc7641.txt>.
- [12] T. Winter, P. Thubert, A. Brandt, T. Clausen, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, RPL: IPv6 routing protocol for low-power and lossy networks, IETF RFC 6550, March 2012. Available: <http://www.ietf.org/rfc/rfc6550.txt>.
- [13] Z. Shelby, M. Koster, C. Groves, J. Zhu, B. Silverajan, “Dynamic Resource Linking for Constrained RESTful Environments”, draft-ietf-core-dynlink-06, July 03, 2018. Available: <https://tools.ietf.org/id/draft-ietf-core-dynlink-06.txt>.
- [14] G. Tanganelli, C. Vallati and E. Mingozzi, "Edge-Centric Distributed Discovery and Access in the Internet of Things," in *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 425-438, Feb. 2018.
- [15] Z. Shelby, M. Koster, C. Bormann, P. van der Stok, C. Amsuess, “CoRE Resource Directory”, draft-ietf-core-resource-directory-17, October 23, 2018, Available: <https://www.ietf.org/id/draft-ietf-core-resource-directory-17.txt>.
- [16] F. Osterlind, A. Dunkels, J. Eriksson, N. Finne, and T. Voigt, “Cross-Level Sensor Network Simulation with COOJA,” in *Proc. of IEEE LCN’06*, November 2006, pp. 641–648.
- [17] T. Narten, E. Nordmark, W. Simpson, H. Soliman, “Neighbor Discovery for IP version 6 (IPv6)”, IETF RFC 4861, September 2007. [Online]. Available: <https://tools.ietf.org/rfc/rfc4861.txt>.

Confronto con i risultati attesi ed eventuali scostamenti rilevati

I risultati conseguiti sono in linea con le attività preventivate. Sono stati sviluppati prototipi software della maggior parte delle componenti definite, e sono stati utilizzati ambienti sperimentali su larga scala per la verifica delle funzionalità e delle prestazioni di tali prototipi. Nel POA 2016/2017 continuerà lo sviluppo dei prototipi software con la definizione di ulteriori funzionalità. Inoltre la piattaforma di raccolta e gestione di dati IoT sarà utilizzata nella realizzazione di prototipo di un sistema di monitoraggio e controllo dell'efficienza energetica di un edificio.

Per tale attività non si riscontano particolari scostamenti con i risultati attesi.

Deliverables

Unità di ricerca	DESCRIZIONE DELIVERABLE	RISULTATI QUANTITATIVI
CNR-IIT	<ul style="list-style-type: none"> Report D_1.1 “Piattaforma per la raccolta, la gestione ed il supporto all’analisi distribuita di dati provenienti da sensori e dispositivi IoT: architettura, protocolli e validazione preliminare” 	<p>Sviluppo e testing di 2 prototipi HW&SW.</p> <p>Il primo è relativo ad un gateway IoT, che supporta le funzionalità di resource discovery e proxying di una rete 6LoWPAN.</p> <p>Il secondo è un ambiente sperimentale per la valutazione di algoritmi distribuiti di gestione dei dati in reti IoT con requisiti sui tempi di consegna.</p>

Linea 1: Benefici per gli utenti finali

Le tecnologie IoT hanno assunto un ruolo sempre più fondamentale per la gestione di uno smart district. È sufficiente pensare al fatto che le tecnologie IoT per il monitoraggio e controllo di processi possono rendere possibile l’implementazione in tempo reale di azioni di Demand Response (DR) per ottimizzare la domanda di energia degli utenti finali in base alla disponibilità da fonti non programmabili. L’architettura classica dei sistemi IoT prevede che i dispositivi alla periferia della rete rappresentano semplicemente degli strumenti per raccogliere dati, mentre le risorse di calcolo necessarie a elaborare quei dati ed analizzarli per trarne informazioni utili sono concentrati nei corporate data center oppure nel cloud. In questa architettura, i gateway IoT hanno quindi l’unica funzione di collettori dei dati e di supporto a quelle funzionalità di rete necessarie per l’instradamento di questi dati verso server remoti in cui vengono immagazzinati e successivamente elaborati. Tuttavia, architettura centralizzate per la gestione dei dati, aumentano le preoccupazioni sulla privacy dei dati, creando un ostacolo al diffondersi su larga scala dei servizi di demand response. Ad esempio, mentre gli utenti sarebbero interessati ad estrarre conoscenza da dati generati localmente, potrebbero non essere disposti a condividere questi dati con la piattaforma cloud gestita dal fornitore del servizio. In molti casi, il trasferimento di tutti i dati grezzi su una piattaforma cloud remota potrebbe non essere necessario in quando spesso i dati sono di valore locale e sono pertinenti solo per gli utenti vicini a dove sono stati generati. Il trasferimento di dati aggregati potrebbe essere sufficiente (od anche preferibile) per l’esecuzione della maggior parte delle applicazioni IoT non critiche. Infine è importante sottolineare che per gestire ambienti IoT complessi è necessario evitare gli alti tempi di latenza legati al trasferimento dei dati verso server remoti. La piattaforma ICT sviluppata in questa attività va nella direzione di permettere di realizzare sistemi decentralizzati per il monitoraggio ed il controllo di processi complessi, garantendo agli utenti finali una maggiore riservatezza dei propri dati e la possibilità di gestire lo smart district in modo cooperativo, senza la presenza di un Aggregatore centrale.

Diffusione dei risultati (indicare anche le modalità di diffusione)

R. Bruno and S. Bolettieri, "Design and Implementation of a COAP-Based Broker for Heterogeneous M2M Applications," 2018 IEEE International Congress on Internet of Things (ICIOT), San Francisco, CA, 2018, pp. 1-8. doi: 10.1109/ICIOT.2018.00008

Apparecchiature disponibili e acquisite

Per l'attività di ricerca sono state utilizzate apparecchiature già di proprietà degli Istituti CNR affidatari e del gruppo universitario partecipante di cui è riportato di seguito una descrizione delle principali:

- Raspberry model Pi 3 B+, un sistema embedded a basso costo e dal consumo energetico limitato, in grado di eseguire una distribuzione completa del sistema operativo Linux.
- sensori Zolertia RE-Mote, un piattaforma hardware IoT dal basso consumo e compatibile col sistema operativo ContikiOS.

Non sono state acquisite nuove attrezzature per lo svolgimento delle attività del progetto.

Linea progettuale 2: Sviluppo di un dimostratore di smart micro-grid, basato su un'architettura in corrente continua che integra fonti rinnovabili - sistemi di accumulo stazionari e carichi energivori, con ottimizzazione della power quality mediante controllo di convertitori active front end

Elenco dei Partecipanti alle attività

Unità di Ricerca	Attività
Università di Bologna - Dipartimento d'Ingegneria dell'Energia Elettrica e dell'Informazione (UNIBO-DEI)	2.1
Istituto Motori CNR IM	2.2

Stato dell'arte

Recenti statistiche mostrano che, ad oggi, il 50% della popolazione mondiale vive in città o negli immediati dintorni. Tali città sono pertanto considerate responsabili dei tre quarti dei consumi energetici e delle emissioni di CO₂ globali. Gli incrementi di popolazione e dei relativi consumi attesi entro il 2050 suggeriscono l'impiego di nuove soluzioni, basate su tecnologie ICT, a supporto dell'efficientamento energetico dei principali processi e servizi cittadini, in scenari identificati come smart cities and smart communities, o, su scala più ridotta, come smart district [1][2]. Infatti uno smart-district è un'area urbana in cui le risorse (nel senso più ampio del termine) sono utilizzate in modo efficiente, in quanto le decisioni sono prese in tempo reale sulla base dell'analisi di dati provenienti da una miriade di sensori, che monitorano lo stato della città e la sua evoluzione.

In tale contesto, anche il sistema elettrico nazionale è interessato da una importante fase di transizione, che va da una configurazione a potenza concentrata, basata sull'impiego di singole centrali termiche tradizionali ad elevata potenza, verso una configurazione di tipo distribuito, caratterizzata da una maggiore diffusione di sistemi di generazione di energia elettrica da fonti rinnovabili, sistemi di accumulo stazionari e carichi energivori distribuiti su una vasta area geografica [3]. Un'ulteriore particolarità che caratterizza il nuovo contesto energetico è la presenza di flussi di potenza bidirezionali tra gestore dei servizi di rete ed utenti del sistema elettrico. Infatti, i singoli utenti diventano sia consumatori che produttori/venditori (*prosumers*) di energia elettrica, grazie all'impiego di sistemi di generazione da fonti rinnovabili e/o alla possibilità di fornire servizi ancillari di supporto alla rete (es. Vehicle to Grid). La gestione di tali applicazioni richiede l'impiego di contatori intelligenti (*smart metering systems*) in grado di misurare l'energia elettrica in entrambe le direzioni e di specifiche architetture di comunicazione, per lo scambio di informazioni tra singole utenze e gestore di rete [4]. In tale contesto, i sistemi di accumulo svolgono un ruolo di importanza primaria, in quanto consentono di immagazzinare l'energia in eccesso prodotta localmente da sistemi di generazione da fonti non programmabili (es. vento, sole, ecc...), per poi rilasciarla durante le fasi di richiesta di picco o per offrire servizi di back-up alla rete. Nel nuovo scenario energetico, il sistema elettrico nazionale punta quindi a diventare più efficiente ed affidabile ed a favorire una consistente diffusione di fonti energetiche rinnovabili.

La riorganizzazione del sistema elettrico può convenientemente avvenire mediante architetture di tipo micro-grid. L'impiego di tali architetture è infatti caratterizzato da costi inferiori e maggiore efficienza, rispetto ad interventi correttivi sulle reti convenzionali in AC. In particolare, attraverso schemi di controllo di tipo multi-agent, le micro-grid possono essere considerate dal gestore di rete come singole unità di elevata potenza, in grado di lavorare sia come carico che come sistema di generazione dell'energia elettrica (*virtual power plant*), fornendo servizi ausiliari di supporto alla rete (*peak shaving, back up, load levelling, ecc..*), con vantaggi in termini di costi di gestione e di investimenti infrastrutturali. Inoltre, l'integrazione di sistemi distribuiti di produzione dell'energia elettrica da fonti non programmabili, come i sistemi ad energia solare ed eolica, può essere efficientemente gestita all'interno della micro-grid, mediante sistemi di controllo che operano a livello locale. Infine, in base alle condizioni operative richieste, le micro-grid hanno la possibilità di lavorare sia in modalità *grid-connected*, con scambio di potenza continuo tra micro-grid e rete elettrica, sia in modalità *islanding*, in cui la potenza elettrica è fornita solo dalle unità di generazione incluse nella micro-grid. Quest'ultima modalità operativa è particolarmente adatta a zone rurali ristrette, dove il collegamento con la rete elettrica non risulta particolarmente conveniente dal punto di vista economico.

Una micro-grid è generalmente basata su un bus comune, in corrente continua o alternata, che consente lo scambio energetico tra i diversi sistemi di generazione, accumulo ed utilizzo dell'energia elettrica. A questo proposito, le architetture con bus in corrente alternata presentano diversi vantaggi, in termini di costi e complessità, in quanto le loro modalità operative si basano su standard preesistenti relativi al funzionamento della rete elettrica in AC in termini di qualità dell'alimentazione, livelli di frequenza/tensione e protezioni. Di conseguenza, le micro-grid in AC possono essere convenientemente utilizzate per applicazioni tradizionali relative a carichi industriali, funzionanti in corrente alternata, e sistemi di generazione distribuiti in AC (es. turbine eoliche e sistemi di micro-cogenerazione).

D'altra parte, lo scenario smart-cities e smart communities è caratterizzato dalla diffusione su larga scala, prevista per i prossimi anni, di unità di accumulo stazionario a supporto di sistemi di generazione

fotovoltaici e da carichi elettronici, che in gran parte lavorano in corrente continua. Questo nuovo scenario giustifica pertanto il crescente interesse della letteratura scientifica e del settore industriale verso l'impiego di micro-grid in corrente continua, che trovano diffusione sempre maggiore in applicazioni geograficamente limitate (distretti) per l'alimentazione di carichi critici, come ad esempio edifici commerciali e/o apparecchiature elettroniche ospedaliere [5][6], con ottimi risultati in termini di qualità della potenza e affidabilità.

Per quanto riguarda le micro-grid in corrente continua, i convertitori statici sono una tecnologia chiave sia per alimentare i carichi ad essa collegati sia per interconnettere la micro-grid alla rete AC (*active front end converter oppure grid-tied converter*). A tal riguardo, è interessante approfondire lo studio del convertitore elettronico di potenza, entrando nei dettagli del sistema controllo e della progettazione circuitale. Infatti, alcune recenti ricerche internazionali hanno dimostrato che le prestazioni dei convertitori active front end possono sensibilmente migliorare con appropriate tecniche di controllo dei parametri elettrici caratterizzanti loro funzionamento. L'uso di tali tecniche, e delle tecnologie ad esse associate, ha trovato ampio riscontro internazionale negli ultimi anni, ma risulta ancora da affinare e approfondire per le diverse applicazioni [7]-[20].

In definitiva, nonostante il crescente interesse della letteratura scientifica verso tali tipologie di architetture, il know how sperimentale sulla gestione di flussi energetici all'interno di micro-grid DC e sul comportamento dei singoli convertitori elettronici di potenza in condizioni operative reali è ancora piuttosto limitato a causa della loro ridotta diffusione. Pertanto lo sviluppo di dimostratori da laboratorio assume un'importanza cruciale per la validazione ed ottimizzazione di strategie di gestione dei flussi energetici da applicare in contesti smart-district su piccola e larga scala.

Bibliografia

- [1] T. Guelzim, M.S. Obaidat, B. Sadoun, Chapter 1 - Introduction and overview of key enabling technologies for smart cities and homes. Eds: Mohammad S. Obaidat, Petros Nicosopolitidis, Smart Cities and Homes. Morgan Kaufmann 2016.
- [2] Obaidat, M. S. "Key enabling ICT systems for smart homes and cities: the opportunities and challenges. Keynote speech." Proceedings of the 2014 IEEE international conference on network infrastructure and digital content (IC-NIDC 2015), Beijing, China. 2015.
- [3] Xu, Guobin, et al. "Toward integrating distributed energy resources and storage devices in smart grid." IEEE internet of things journal 4.1 (2017): 192-204.
- [4] Malik, Arif S., et al. "Smart grid scenarios and their impact on strategic plan—A case study of Omani power sector." Sustainable Cities and Society 37 (2018): 213-221.
- [5] Meagher, Kevin, David Loisselle, and Rodger Koopman. "Real time microgrid power analytics portal for mission critical power systems." U.S. Patent No. 8,321,194. 27 Nov. 2012.
- [6] Kwasinski, Alexis. "Technology planning for electric power supply in critical events considering a bulk grid, backup power plants, and micro-grids." IEEE Systems Journal 4.2 (2010): 167-178.
- [7] H. Akagi, "Control strategy and site selection of a shunt active filter for damping of harmonic propagation in power distribution systems," IEEE Trans. Power Del., vol. 12, no. 1, pp. 354–362, Jan. 1997.
- [8] C. D. Schauder and S. A. Moran, "Multiple reference frame controller for active filters and power line conditioners," U.S. Patent 5 309 353, May 3, 1994.
- [9] S. Bhattacharya, A. Veltman, D.M. Divan, and R. D. Lorenz, "Flux-based active filter controller," IEEE Trans. Ind. Appl., vol. 32, no. 3, pp. 491– 502, May/Jun. 1996.
- [10] P.-T. Cheng, S. Bhattacharya, and D. M. Divan, "Experimental verification of dominant harmonic active filter for high-power applications," IEEE Trans. Ind. Appl., vol. 36, no. 2, pp. 567–577, Mar./Apr. 2000.
- [11] L.Malesani, P. Mattavelli, and S. Buso, "Robust dead-beat current control for PWM rectifiers and active filters," IEEE Trans. Ind. Appl., vol. 35, no. 3, pp. 613–620, May/Jun. 1999.
- [12] S. Buso, L. Malesani, and P. Mattavelli, "Comparison of current control techniques for active filters applications," IEEE Trans. Ind. Electron., vol. 45, no. 5, pp. 722–729, Oct. 1998.
- [13] S. Buso, S. Fasolo, L. Malesani, and P. Mattavelli, "A dead-beat adaptive hysteresis current control," IEEE Trans. Ind. Appl., vol. 36, no. 4, pp. 1174–1180, Jul./Aug. 2000.
- [14] M. J. Newman, D. N. Zmood, and D. G. Holmes, "Stationary frame harmonic reference generation for active filter systems," IEEE Trans. Ind. Appl., vol. 38, no. 6, pp. 1591–1599, Nov./Dec. 2002.
- [15] X. Yuan, W. Merk, H. Stemmler, and J. Allmeling, "Stationary-frame generalized integrators for current control of active power filters with zero steady-state error for current harmonics of concern under unbalanced and distorted operating conditions," IEEE Trans. Ind. Appl., vol. 38, no. 2, pp. 523–532, Mar./Apr. 2002.
- [16] M. Bojrup, P. Karlsson, M. Alakula, and L. Gertmar, "A multiple rotating integrator controller for active filters," in Proc. EPE Conf.,

1999. CD-ROM.

- [17] C. Lascu, L. Asiminoaei, I. Boldea, and F. Blaabjerg, "High performance current controller for selective harmonic compensation in active power filters," *IEEE Trans. Power Electron.*, vol. 22, no. 5, pp. 1826–1835, Sep. 2007.
- [18] C. Lascu, L. Asiminoaei, I. Boldea, F. Blaabjerg, "Frequency Response Analysis of Current Controllers for Selective Harmonic Compensation in Active Power Filters," *IEEE Trans. Ind. Elect.*, vol. 56, no. 2, pp. 337-347, Feb. 2009.
- [19] P. Mattavelli, "A closed-loop selective harmonic compensation for active filters," *IEEE Trans. Ind. Appl.*, vol. 37, no. 1, pp. 81–89, Jan./Feb. 2001.
- [20] P. Mattavelli and F. P. Marafao, "Repetitive-based control for selective harmonic compensation in active power filters," *IEEE Trans. Ind. Electron.*, vol. 51, no. 5, pp. 1018–1024, Oct. 2004.

Obiettivi della linea

L'obiettivo di questa linea progettuale consiste nella progettazione ed allestimento di un dimostratore di micro-grid, basato su un'architettura in corrente continua, che integra fonti rinnovabili e sistemi di accumulo stazionari a supporto di utenze energivore in contesti di tipo smart district. Per i convertitori che interfacciano tale architettura con la rete elettrica sono stati sviluppati sistemi di controllo ad alte prestazioni della corrente di linea al fine di ottimizzarne i parametri di power quality.

Le attività di questa linea proseguiranno nelle successive annualità del Piano Triennale (POA 2016-2017).

La linea progettuale 2 è stata suddivisa nelle seguenti attività:

2.1: Controllo ad elevate prestazioni di convertitori per applicazioni grid-connected o smart-grid (UNIBO-DEI)

Unità di ricerca coinvolte:

- Università di Bologna - Dipartimento d'Ingegneria dell'Energia Elettrica e dell'Informazione (UNIBO-DEI)

2.2: Sviluppo di un dimostratore di smart micro-grid, basato su un'architettura in corrente continua che integra fonti rinnovabili - sistemi di accumulo stazionari e carichi energivori (CNR-IM)

Unità di ricerca coinvolte:

- Consiglio Nazionale Delle Ricerche – Istituto Motori (CNR IM)

Attività 2.1 Controllo ad elevate prestazioni di convertitori per applicazioni grid-connected o smart-grid (UNIBO-DEI)

Obiettivi

Questa attività ha come obiettivo lo sviluppo di sistemi di controllo della corrente di linea di convertitori per applicazioni *grid-connected* o *smart-grid*, basati su regolatori ripetitivi, che consentano il miglioramento della *power quality*, ottenendo valori di *Total Harmonic Distortion* della corrente di rete (THDi) inferiori a 2-3 %. Inoltre, è prevista la realizzazione combinata di tecniche di *islanding detection*, per rilevare la condizione di funzionamento in isola dei sistemi di generazione distribuita.

Nel corso del POA 2015, l'obiettivo della ricerca è di portare tutte le attività da TRL 3 a TRL 4 e raggiungere per alcune di esse TRL5.

Risultati conseguiti

Nel seguito sono brevemente descritti i risultati conseguiti da UNIBO-DEI nel corso del PAR 2015.

a) Realizzazione del modello di un convertitore active-front end/filtro attivo con sistema anti-islanding

La topologia di convertitore oggetto di studio è di tipo Voltage Source Inverter, in quanto è la più comune e facilmente utilizzabile. In accordo con le più recenti tecniche di controllo, la corrente del convertitore è regolata da due anelli: il primo prevede l'utilizzo di regolatori risonanti per l'armonica fondamentale, il secondo può impiegare un banco di regolatori risonanti o un unico regolatore ripetitivo e si occupa del controllo delle armoniche a frequenza superiore. La Fig. 38 mostra la gerarchia di livello più alto dello schema a blocchi realizzato in ambiente Simulink.

Il modello comprende le seguenti parti:

- un modello della rete, del filtro in ingresso, del carico distorto;
- un modello dell'inverter in commutazione;
- il sistema di controllo, comprensivo di tutti i regolatori e dell'anello di aggancio di fase (Phase Locked Loop, PLL) necessario per la sincronizzazione con la tensione di rete.

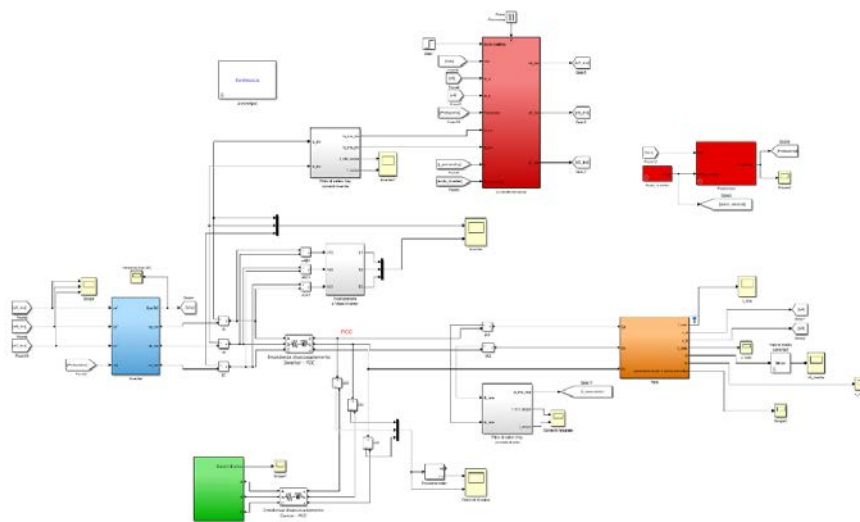


Fig. 38. Modello Simulink del sistema (gerarchia di livello più alto).

Sebbene il sistema di controllo sia stato implementato in ambiente Matlab-Simulink, sono state considerate le limitazioni imposte dalla piattaforma dSpace, utilizzata successivamente per lo svolgimento delle attività sperimentali.

La simulazione è, per quanto possibile, una replica del sistema reale. Si tengono in considerazione i seguenti fenomeni: il funzionamento *switching* dell'inverter (comprensivo di tempi morti di commutazione), tempo di calcolo finito dei regolatori, discretizzazione del sistema di controllo e saturazione dei regolatori.

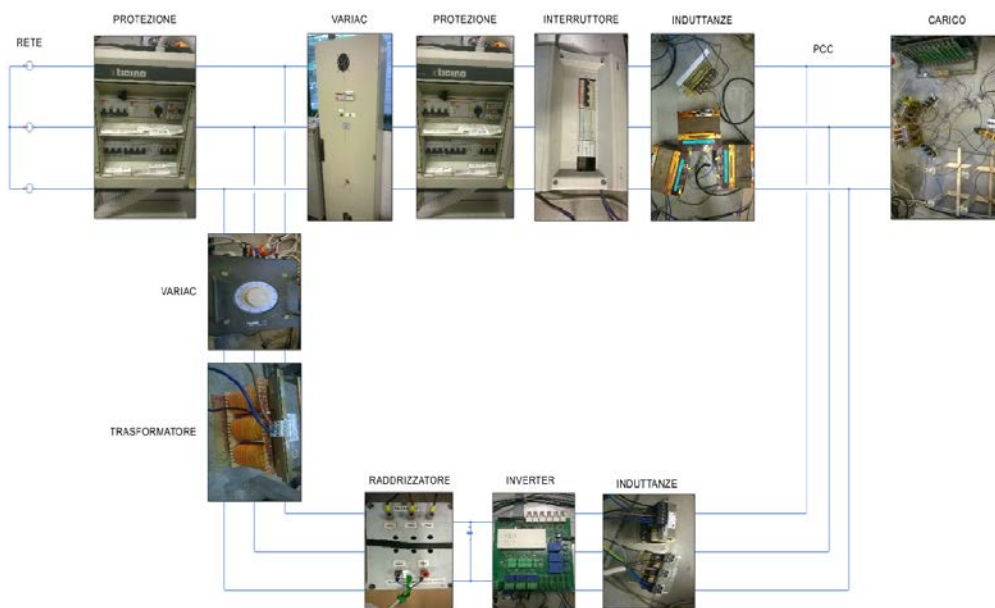
Inoltre, sono considerati due metodi attivi per l'*islanding detection*, ritenuti idonei in applicazioni *smart grid* e generazione distribuita dell'energia elettrica. Il metodo *voltage drift* genera un disturbo nella potenza attiva immessa in rete dal convertitore d'interfaccia, mentre il metodo *frequency drift* genera

un disturbo di potenza reattiva.

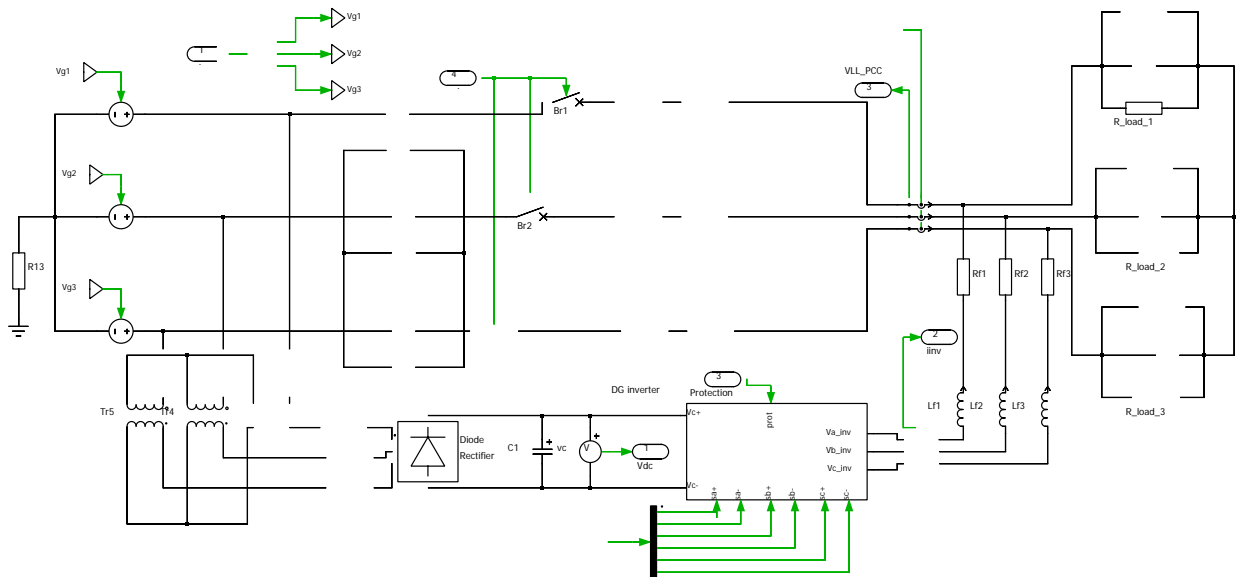
b) Adattamento di un prototipo di convertitore in scala ridotta e verifica sperimentale degli algoritmi di controllo

Presso il laboratorio LEMAD del DIE di UNIBO è disponibile un banco di prova idoneo ad applicazioni grid-connected, allestito in parte durante le attività del PAR 2013. Il prototipo è stato modificato in modo da consentire la verifica sperimentale delle tecniche di controllo sviluppate sia per il miglioramento della qualità della corrente di rete, sia delle tecniche di *islanding-detection*.

Lo schema elettrico impiegato per tali prove è definito anche da alcuni standards quali UL 1741, la IEEE 929 e IEEE 1547, è quello della Fig. 39. Il Distributed Power Generation System (DPGS) è connesso al Point of Common Coupling (PCC) mediante un inverter. Al PCC risulta connesso anche il carico, schematizzato come un'impedenza RLC di tipo parallelo, progettato in modo da entrare in risonanza con la frequenza di rete (50 Hz).



a)



b)

Fig. 39. Sistema sperimentale per *anti-islanding*. Appartati sperimentali (a). Schema (b).

Si suppone che la rete sia di potenza prevalente, ossia in grado di imporre il valore della tensione e della frequenza, che divengono quindi parametri costanti. Il caso peggiore è quello in cui le potenze attiva e reattiva assorbite dal carico risonante eguagliano rispettivamente i valori di potenze attiva e reattiva fornite dal DPGS. Infatti, il funzionamento del sistema potrebbe permanere indisturbato anche dopo il distacco della rete, senza variazioni di frequenza e di tensione al carico apprezzabili.

Il sistema descritto è stato impiegato con successo nelle prove di *islanding detection* in conformità con gli standard attuali.

b) *Definizione del sistema di controllo di un convertitore di interfaccia nelle configurazioni “raddrizzatore PWM” e “filtro attivo” con controllo di corrente sino alla 19a armonica, basato su regolatori ripetitivi.*

L’obiettivo primario di questa attività è la definizione di un sistema di controllo in grado di abbattere il contenuto armonico della corrente d’ingresso e di migliorare il fattore di potenza, nell’ipotesi che il convertitore d’interfaccia operi da filtro attivo. Il sistema di controllo del convertitore ha il compito di

- stabilire il valore dei riferimenti di tensione che consentono di generare una corrente in ingresso tale da rendere sinusoidale la corrente di linea;
- mantenere la tensione del condensatore del bus DC costante, così da poter generare la corrente in ingresso richiesta, ed alimentare l’eventuale carico DC in uscita
- migliorare il fattore di potenza nel punto di connessione alla rete.

In un sistema di riferimento sincrono con la tensione di rete, opportunamente orientato, è possibile controllare la tensione del bus DC tramite la sola componente *d* della corrente in ingresso, mentre la componente *q* rappresenta un grado di libertà per regolare la potenza reattiva scambiata con la rete. Lo schema di controllo adottato è mostrato nella Fig. 40.

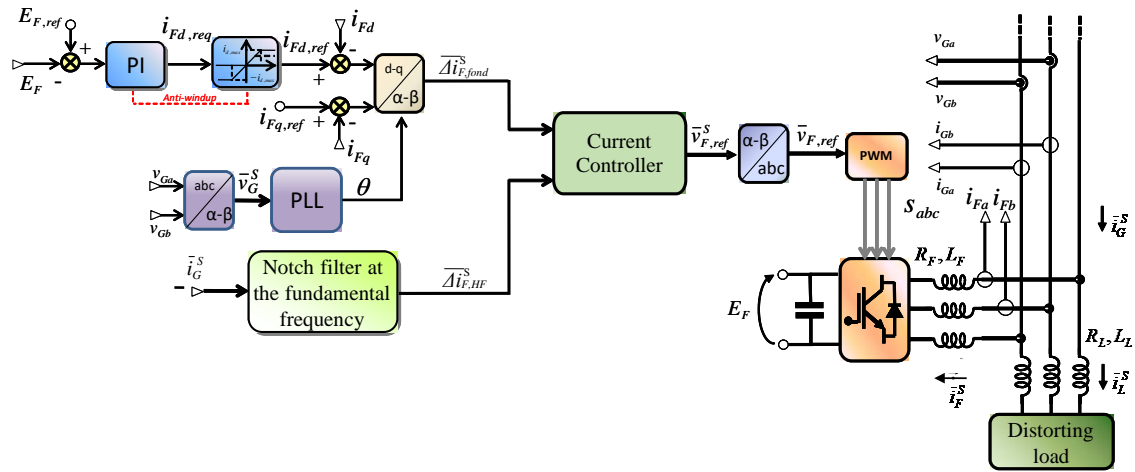


Fig. 40. Schema di controllo (architettura filtro attivo)

Per il miglioramento della qualità della corrente in ingresso, si sono impiegati dei regolatori di tipo ripetitivo. I regolatori ripetitivi sono dei sistemi di regolazione in catena chiusa che, nella loro formulazione più classica, sommano al loro ingresso lo stesso segnale, ritardato di un certo periodo di tempo. La funzione di trasferimento presenta frequenze di risonanza multiple della frequenza fondamentale, alle quali il guadagno del regolatore risulta idealmente infinito. Questa caratteristica, favorevole all'attenuazione dei disturbi, può tuttavia causare fenomeni di instabilità, in quanto il regolatore mantiene il guadagno elevato in una banda assai ampia.

I regolatori ripetitivi si basano sul Principio del Modello Interno (IMP, Internal Model Principle) che afferma che una condizione sufficiente per l'inseguimento asintotico del riferimento è che la funzione di trasferimento ottenuta combinando quella del regolatore e dell'impianto controllato contenga il polinomio generatore del segnale di riferimento al denominatore. Se il segnale di riferimento è periodico, con periodo T_p , in teoria, la funzione di trasferimento da includere nel regolatore è del tipo:

$$G_r(s) = \frac{1}{1 - e^{-T_p s}}$$

Alcune soluzioni per migliorare la stabilità sono state proposte alla comunità scientifica in anni recenti. Si tratta di schemi di controllo che tentano di compensare i ritardi di fase dovuti ai tempi di calcolo del sistema di controllo o ai sensori e che impiegano filtri non causali a sfasamento nullo per ridurre il guadagno ad alta frequenza. Nel progetto sono state analizzate 6 diverse varianti, denominate Architettura A, B, C, D, E ed F.

Architettura A

Questa architettura di controllo rappresenta l'evoluzione della struttura proposta da Bin Zhang, Danwei Wang, Keliang Zhou e Yigang Wang nel 2008. Lo schema di funzionamento dell'architettura è mostrato in Fig. 41. Il parametro K è un guadagno di retroazione, mentre Q' è un filtro a fase nulla, modificato per soddisfare la condizione di causalità. Il diagramma di Bode del regolatore è mostrato in Fig. 42.

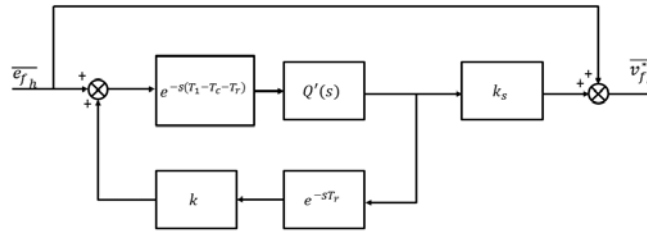


Fig. 41 - Schema dell'architettura A.

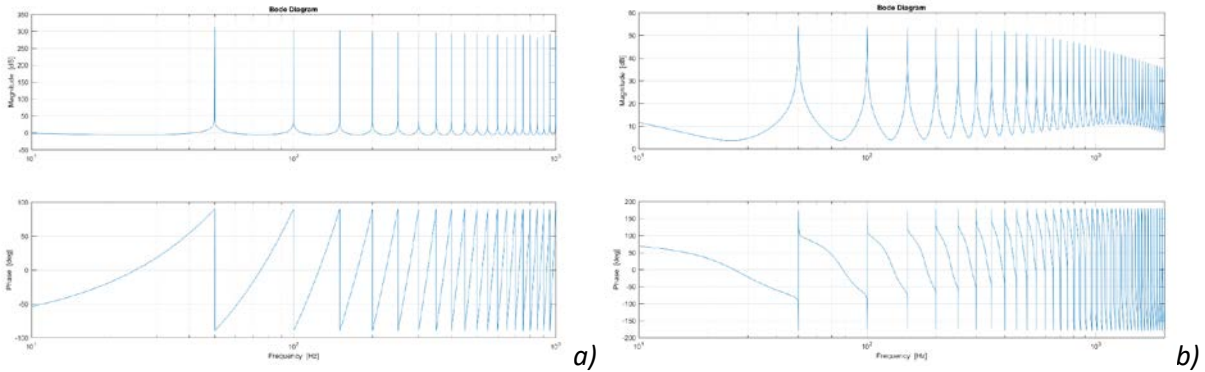


Fig. 42 - Diagramma di Bode dell'architettura G_{rip}^A senza filtro stabilizzatore (a) e con compensazione dei ritardi e filtro stabilizzatore (b).

L'espressione generale di Q' è del tipo:

$$Q'(s) = C_1 e^{-s2T_c} + C_0 e^{-sT_c} + C_1$$

Il filtro Q' può essere usato per generare un anticipo di fase per compensare sia il tempo di calcolo, sia il ritardo causato dalla conversione digitale-analogica dei segnali.

Architettura B

Questa architettura rappresenta un'evoluzione della architettura A; cambiando i segni della retroazione è possibile modificare le frequenze di risonanza del sistema, che diventano multipli interi dispari della frequenza fondamentale. I picchi di risonanza sono a 150 Hz, 250 Hz, ..., ossia ogni 100 Hz.

Poiché lo spettro delle correnti di carico presenta in genere solo armoniche dispari, questa soluzione rappresenta il miglior compromesso tra qualità di abbattimento armonico e memoria necessaria nel sistema di controllo.

Architettura C

Questa architettura rappresenta una variante dell'architettura di controllo B in cui il buffer di memoria è ¼ del precedente. La funzione di trasferimento presenta risonanze ogni 200 Hz. Nel caso ideale, lo spettro delle correnti di rete dovrebbe presentare solo armoniche di ordine dispari non multiple di 3, alternativamente di sequenza diretta e inversa, ossia +1, -5, +7, -11, +13, ... Il comportamento della rete però differisce da quello ideale; a causa degli inevitabili squilibri di carico tra le fasi della rete, dovuti principalmente ai carichi monofase, nel punto di connessione può essere presente una prima armonica di tensione di sequenza inversa. Questa armonica, sebbene limitata, richiama nello spettro delle correnti anche le armoniche -1, +5, -7, +11, -13, ...

L'architettura di controllo non è in grado di eliminare queste armoniche e quindi si ha un peggioramento nelle prestazioni del sistema.

Architettura D

Questa architettura di controllo trae spunto dalla soluzione proposta da G. Escobar, A. Valdez, J. Leyva-Ramos e P. Mattavelli nel 2007, che proponeva di aggiungere, a monte del controllo, una azione integrale. Poiché tale soluzione non fornisce risultati soddisfacenti, e, soprattutto, non permette di ottenere l'anticipo di fase necessario alla stabilità dell'anello di corrente, lo schema di controllo è stato modificato come mostrato nella Fig. 43.

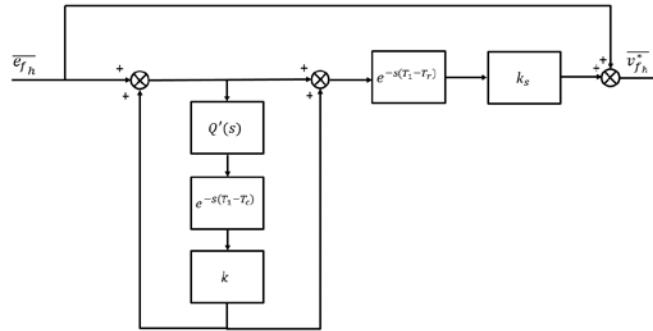


Fig. 43 - Schema dell'architettura D.

In essa, l'anticipo di fase è ottenuto aggiungendo un ulteriore uno sfasamento temporale pari a T_1 . Dal punto di vista delle fasi, l'introduzione del termine e^{-sT_1} non cambia la fase della funzione di trasferimento alle frequenze di risonanza. Un'ulteriore particolarità di questa architettura è l'azione in avanti; questa azione introduce una serie infinita di zeri, la cui risonanza produce delle gole nella funzione di trasferimento. L'effetto di queste zone a guadagno nullo è di non amplificare i disturbi a frequenze intermedie, rendendo il sistema più robusto.

La Fig. 44 mostra il diagramma del regolatore senza l'introduzione di nessun accorgimento atto alla stabilità.

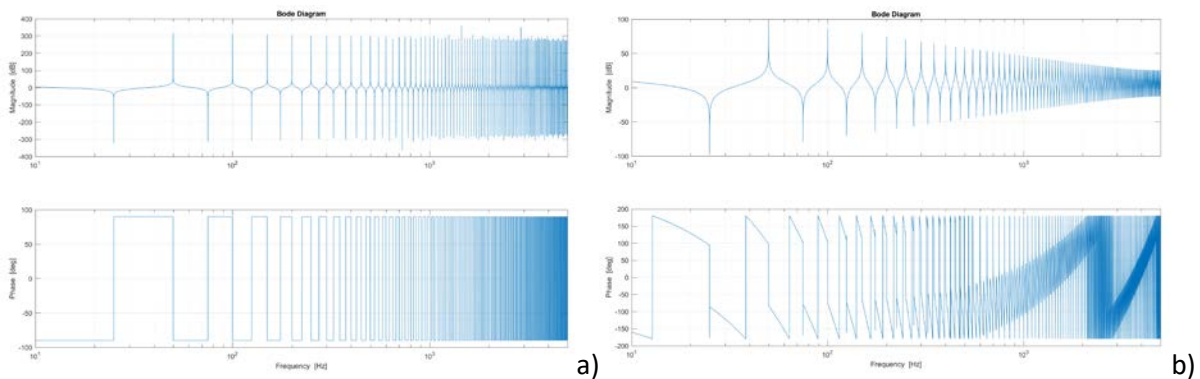


Fig. 44 - Diagramma di Bode dell'architettura G_{rip}^D senza filtro stabilizzatore (a) e con filtro stabilizzatore e compensazione dei ritardi (b).

Architettura E

Questa architettura rappresenta la combinazione delle architetture B e D. Le risonanze dovute agli zeri della funzione di trasferimento sono descritte dalla relazione:

$$f_{risonanza}^{zeri'} = \left(\frac{1}{2} + 2k\right) f_1$$

Architettura F

Questa architettura rappresenta l'evoluzione dell'architettura E, implementata però nel sistema di riferimento sincrono all'armonica fondamentale.

c) *Attività sperimentale*

Una serie di prove sperimentali è stata volta a determinare le prestazioni di ogni architettura, per diversi valori dei guadagni e dei filtri stabilizzanti. La forma d'onda delle correnti di rete generate dal carico distortente è mostrato in Fig. 45. Il THD della corrente è pari al 21.56%.

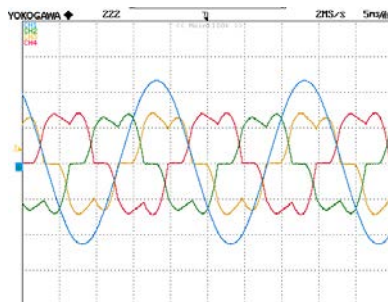


Fig. 45 - Correnti di rete (2A/div) e prima armonica della tensione al PCC (in blu, 40V/div), senza l'intervento di nessun controllo (2ms/div).

L'attivazione del sistema di controllo consente un ragguardevole miglioramento della qualità della corrente di linea. Le architetture di controllo che hanno ottenuto le prestazioni migliori sono state A, B ed E, che hanno ridotto la distorsione armonica della corrente di rete sino al 2.21%.

La Fig. 46 mostra i transitori di avvio delle architetture di controllo B ed E.

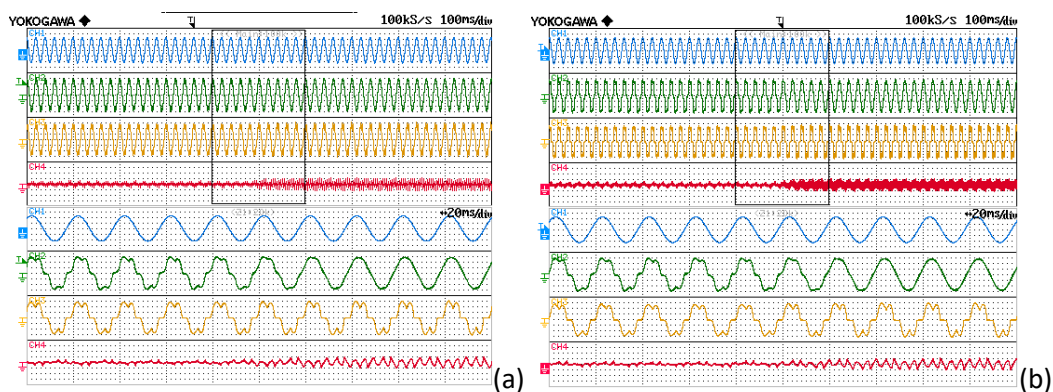


Fig. 46: Transitorio di avvio dell'architettura di controllo B (a) ed E (b). Dall'alto verso il basso: prima armonica della tensione di rete, corrente di rete, corrente di carico, corrente di inverter. Scale: 100 ms/div primo grafico, 20 ms/div secondo grafico, 40 V/div, 1 A/div.

La Tab. I confronta i risultati conseguiti da ciascuna architettura in termini di THD delle correnti di rete.

Tab. I: Confronto tra i risultati ottenuti con le varie architetture.

Architettura di controllo	THD
A	2.73%
B	2.29%
C	3.57%
E	2.21%
F	3.50%

d) *Definizione di tecniche per l'islanding detection, e implementazione negli schemi di controllo.*

L'*islanding* è la situazione in cui una porzione di rete si trova ad operare in isola (condizione di disconnessione dal resto della rete), alimentata da uno o più Distributed Power Generation Systems (DPGS). I principali problemi derivanti dalla condizione operativa di *islanding* incontrollato sono i seguenti:

- Mancato controllo, da parte del gestore, di tensione e frequenza all'interno della parte di rete isolata (isola), con possibilità di arrecare danni ad apparecchiature ed utenti.
- Rischio per manutentori, che potrebbero non essere al corrente della condizione di alimentazione dell'isola.
- Interferenza con i tentativi di ripristino del servizio da parte del gestore, in quanto il comportamento dell'isola risulta imprevedibile ed incontrollato e non è possibile procedere con la richiusura degli interruttori di linea.
- Eventuale mancanza di congruenza di fase (il cui andamento è, nell'isola, non controllato) all'atto della riconnessione dell'isola, con conseguenti gravi danni ai dispositivi connessi alla rete.

Il funzionamento isolato di una porzione della rete pubblica non è permesso. In caso d'interruzione della rete (intenzionale o meno), il sistema generatore deve rilevare la condizione di funzionamento in "isola" e immediatamente interrompere il funzionamento. Questo meccanismo di sorveglianza è noto come protezione *anti-islanding*. Esistono numerose tecniche di *islanding detection*. Tipicamente la rilevazione dell'*islanding* diviene una necessità nel momento in cui i DPGS sono di piccola taglia (specialmente piccoli impianti fotovoltaici, ma anche ad esempio piccole turbine eoliche), dal momento che i DPGS di elevata potenza (con riferimento ad esempio ad impianti eolici di potenza rilevante) possono essere invece impiegati per stabilizzare la rete proprio nel momento in cui questa smetta di fornire energia.

Nonostante i tentativi di armonizzazione da parte degli enti europei (CENELEC), ciascun paese ha sviluppato i propri standard e normative per i dispositivi produttori interfacciati alla rete elettrica, specialmente nei paesi dove sono presenti sistemi di incentivazione (ad esempio in Italia, con il Conto Energia). Lo standard che definisce le caratteristiche dei sistemi interfacciati alla rete è la normativa ENEL DK5940, secondo cui il sistema deve essere trifase sopra ai 6 kW, è necessario un trasformatore di isolamento oltre i 20 kW, e può essere in BT fino a 50 kW.

Le principali caratteristiche che un metodo diagnostico *anti-islanding* dovrebbe possedere sono le seguenti:

- affidabilità: la condizione di funzionamento in Isola dovrebbe avvenire in ogni condizione di funzionamento

- selettività: la capacità di discriminare tra funzionamento in isola e perturbazione di rete
- minima perturbazione: l'impatto sulla qualità dell'energia dovrebbe essere il minore possibile.

Nel corso del progetto, sono stati implementati nel sistema di controllo di un convertitore grid-connected due metodi attivi, ossia basati sulla generazione di piccoli disturbi in uscita dal convertitore per cambiare qualcuno dei parametri di rete. Essi sono:

Voltage drift method – il metodo si basa su principio di generare un disturbo nella potenza attiva immessa in rete dal convertitore d'interfaccia. Nel caso di funzionamento in isola, la perturbazione tende a destabilizzare rapidamente l'impianto in isola, causando una variazione della tensione e della frequenza facilmente rilevabile. Il metodo è anche detto *potenza attiva-tensione*, e il suo schema di implementazione è mostrato in Fig. 47(a).

Frequency drift method – il metodo si basa sul principio di generare un disturbo di potenza reattiva immessa in rete dal convertitore di interfaccia. Nel caso di funzionamento in isola, la perturbazione tende a destabilizzare rapidamente l'impianto in isola, causando una repentina variazione della frequenza. Il metodo è anche detto *potenza reattiva-frequenza* e il suo schema di implementazione è mostrato in Fig. 47(b).

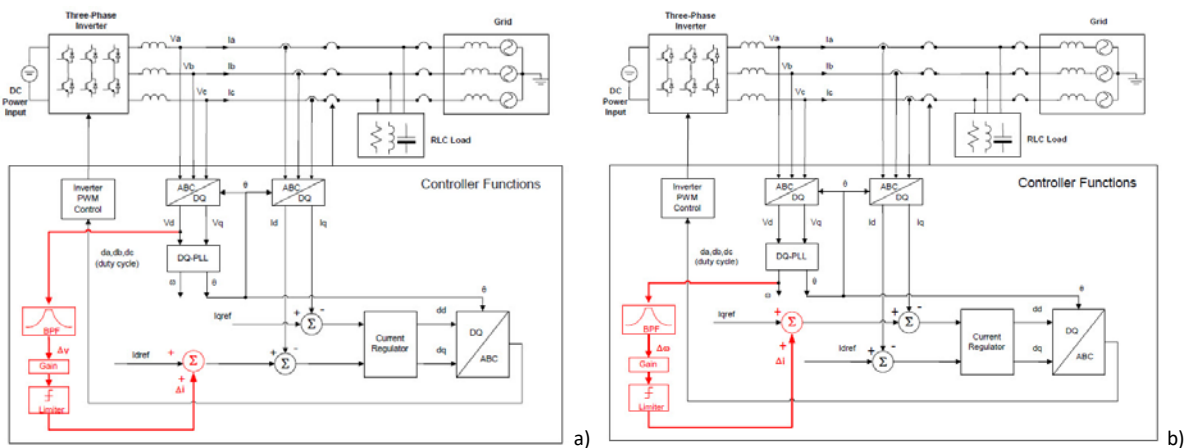


Fig. 47. Schemi di controllo per islanding detection. (a) Metodo voltage-drift, (b) Metodo frequency drift.

Indipendentemente dall'implementazione e dalla strategia di individuazione del problema adottata, la norma IEC 62116 stabilisce una procedura di test con riferimento ai metodi di rilevamento sull'inverter (inverter resident detection). Con riferimento alla Fig. 48, il test prevede come carico un circuito RLC parallelo.

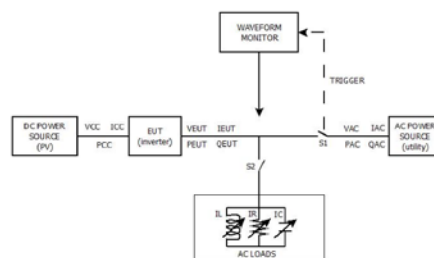


Fig. 48 - Struttura dell'apparato sperimentale per l'esecuzione di test sugli algoritmi anti-islanding.

Il test consiste nella misura del tempo per cui il convertitore grid-connected continua ad alimentare il carico in isola. Lo standard IEC prevede un limite di 2 s, come gli standard IEEE 1547 e IEEE 929. Invece, la norma CEI 0-2 individua un tempo di 0.16 s.

Entrambi i metodi implementati si sono dimostrati idonei al riconoscimento di condizioni di funzionamento in isola con tempi inferiori a 0.12s, rispettando le specifiche di tutte le norme anzidette.

La Fig. 49 mostra alcuni il funzionamento di entrambi i metodi, con ottimi risultati.

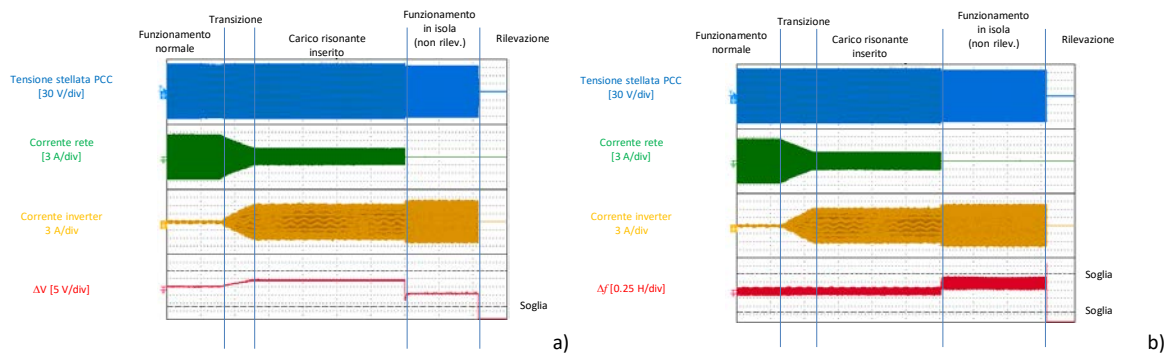


Fig. 49 – Prove sperimentali di *islanding detection* con il metodo voltage-drift (a), frequency drift (b).

Le attività di ricerca hanno richiesto un intenso scambio di informazioni tra UNIBO-DEI e CNR-IM, utili da un lato per la definizione della miglior configurazione progettuale e della progettazione dei sistemi prototipali, dall'altro alla comprensione dei problemi di interfacciamento delle microreti e delle smart-grid.

In particolare, UNIBO-DEI ha valutato il comportamento del convertitore di interfaccia della microrete messa a punto da CNR-IM per valutare le sue prestazioni in diverse configurazioni circuitali e in diverse condizioni di carico. Si è inoltre valutato l'effetto delle tecniche di controllo di corrente di tipo ripetitivo applicate al convertitore di interfaccia per applicazioni smartgrid o microgrid.

Le simulazioni effettuate si sono focalizzate su numerosi aspetti: sollecitazioni dei componenti, qualità delle correnti generate, stabilità di controllo, dimensionamento dei filtri.

A titolo d'esempio si riportano alcune immagini, relative allo studio delle prestazioni del convertitore d'interfaccia della microrete predisposta dal CNR-IM. Nella Fig. 50 si mostrano la corrente DC richiesta dal carico, la tensione ai capi del condensatore lato DC dell'inverter, le tensioni e le correnti di rete. Al tempo $t=0.1$ s, l'inverter si avvia a piena corrente. Dopo un secondo, la corrente di carico si riduce al 50% e si riporta al valore nominale $t=1.5$. L'impiego di regolatori ripetitivi e risonanti consente ottime prestazioni dinamiche, benché stabili. La tensione del bus DC mostra una variazione inferiore al 2% in queste gravose condizioni di funzionamento.

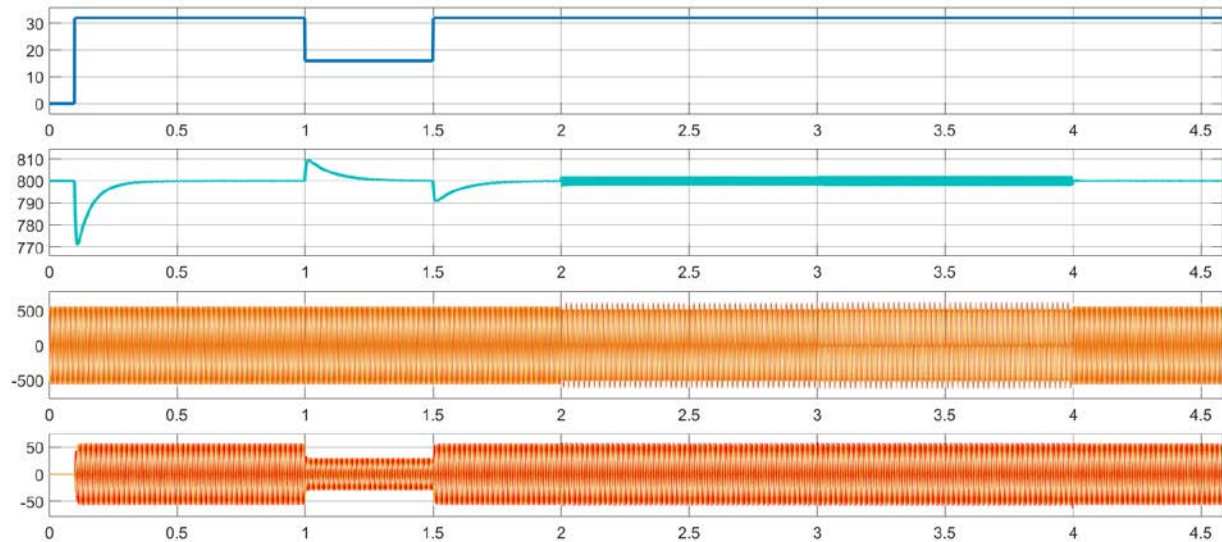


Fig. 50. Transitorio con variazioni di carico per il convertitore della microrete e comportamento con distorsione d'ingresso.

Al tempo $t=2s$, viene introdotta una distorsione della tensione di rete. In particolare, la tensione di rete acquista le seguenti componenti di disturbo:

- una armonica fondamentale di sequenza inversa con ampiezza pari al 10% della tensione fondamentale
- una tensione di quinta armonica di sequenza inversa con ampiezza pari al 5% della tensione fondamentale
- una tensione di settima armonica con ampiezza pari al 3% della tensione fondamentale

Al tempo $t=3s$ si aumenta la distorsione della tensione di rete. Tutte le armoniche di tensione precedentemente introdotte assumono una tensione pari al 10% della tensione fondamentale.

Nonostante la notevole distorsione introdotta, la qualità della corrente iniettata in rete rimane elevata grazie al sistema di controllo ripetitivo. Il THD delle correnti di rete ha un valore inferiore all'1% in condizioni di rete non distorta, mentre rimane ad un valore inferiore al 3% in condizioni di tensione di rete distorta. Ad esempio, la Fig. 51 mostra il transitorio successivo all'accessione del convertitore nel caso di massima distorsione di tensioni di rete e pieno carico d'uscita.

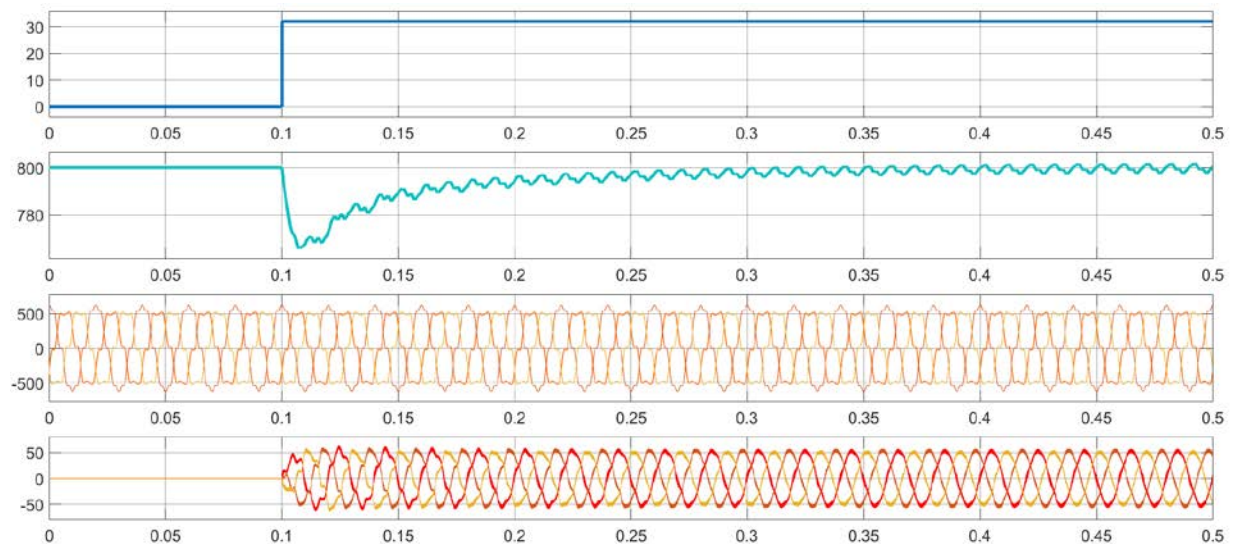


Fig. 51. Transitorio con variazioni di carico per il convertitore della microrete e comportamento con distorsione d'ingresso.

I parametri delle simulazioni delle Fig. 50 e 51 si basano sul convertitore della microrete, supposta collegata a una rete trifase a 400 V RMS, 50 Hz, con induttanza di disaccoppiamento di 0.75 mH e 0.02 Ohm. La tensione nominale in uscita è posta a 800 V, la capacità del condensatore d'uscita è di 3 mF, la frequenza di commutazione è di 10 kHz. I componenti sono supposti con forward voltage degli IGBT pari a 0.8 V, dei diodi pari a 0.8 V, con tempi morti pari a 2 μ s.

Gli schemi di controllo, l'analisi delle prestazioni e i risultati delle simulazioni sono stati condivisi con CNR-IM per la prosecuzione del progetto e la realizzazione dei sistemi prototipali.

Confronto con i risultati attesi ed eventuali scostamenti rilevati

I sistemi di controllo sviluppati basati su regolatori ripetitivi si sono dimostrati in grado di soddisfare la normativa di riferimento.

I target di progetto erano i seguenti:

- corretto funzionamento in modalità raddrizzatore attivo e filtro attivo in condizioni tipiche delle applicazioni industriali;
- abbattimento della distorsione delle correnti in ingresso indicativamente di un fattore 10 (es: riduzione del THD delle correnti in ingresso da 20 a 2), e comunque tali da rientrare nelle norme IEEE-STD 519 e EN 61000-3-12;
- controllo indipendente delle singole armoniche di corrente sino alla 17^a armonica.

Nello specifico caso in esame, il coefficiente TDD (total distortion demand) dovrebbe essere inferiore al 5% (Tab. II).

Tab. II – Standard 519-2014

Maximum harmonic current distortion in percent of I_L						
Individual harmonic order (odd harmonics) ^{a, b}						
I_{sc}/I_L	$3 \leq h < 11$	$11 \leq h < 17$	$17 \leq h < 23$	$23 \leq h < 35$	$35 \leq h \leq 50$	TDD
< 20 ^c	4.0	2.0	1.5	0.6	0.3	5.0
20 < 50	7.0	3.5	2.5	1.0	0.5	8.0
50 < 100	10.0	4.5	4.0	1.5	0.7	12.0
100 < 1000	12.0	5.5	5.0	2.0	1.0	15.0
> 1000	15.0	7.0	6.0	2.5	1.4	20.0

Si è sperimentalmente verificato che il convertitore d'interfaccia con architetture di controllo di tipo A, B ed E è in grado di abbattere la distorsione da circa il 21.5% a circa il 2%, compensando la distorsione della corrente di carico ben oltre alla ventesima armonica (teoricamente sino alla 50a armonica con un buffer di 100 campioni), con tempi di calcolo inferiori di circa un ordine di grandezza rispetto alle prestazioni di un controllo di corrente basato su un banco di regolatori PIR sino alla 20^a armonica.

Nel sistema sperimentale utilizzato, ciò consentirebbe il dimezzamento del volume dell'induttanza di ingresso al convertitore, pur rimanendo nei rispetti della normativa con carichi aventi un THD sino al 21%.

Le regole tecniche di riferimento più stringenti per la gestione del fenomeno *dell'islanding* sono contenute all'interno della norma CEI 0-21, dal titolo: "Regola tecnica di riferimento per la connessione di utenti attivi e passivi alle reti di BT delle imprese distributrici di energia elettrica", entrata in vigore il 01.10.2014. Nella norma si individuano i tempi di intervento delle protezioni applicabili anche a sistemi *anti-islanding*, al variare della tensione e della frequenza, come riportato nella Fig. 52.

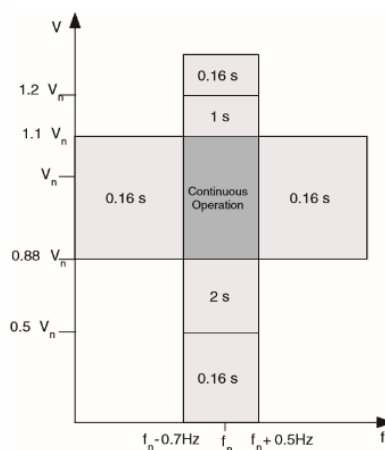


Fig. 52 – Tempi di intervento

Per i sistemi di *anti-detection* sviluppati, i tempi di intervento ottenuti durante le prove pratiche sono stati inferiori a 120ms, toccando minimi di 20 ms, ben al di sotto dei 0.16s previsti dalle norme.

L'obiettivo del PAR2015 era di portare le linee di ricerca UNIBO-DEI da TRL3 a TRL4, e in alcuni casi a TRL5. Si osserva che, in genere, una volta che una tecnologia al livello TRL3 (proof-of-concept) è pronta, la tecnologia avanza fino a TRL 4. Durante TRL 4, i sottosistemi vengono testati l'uno con l'altro. Il livello TRL 5 è una continuazione del TRL 4. Tuttavia, una tecnologia TRL5 è identificata come una

tecnologia *breadboard* e deve essere sottoposta a test più rigorosi della tecnologia al livello TRL 4. Le simulazioni devono essere eseguite in ambienti che sono il più vicino possibile a quello industriale.

Pertanto, si può ritenere che gli obiettivi di avanzamento siano stati raggiunti, in quanto ciascuna tecnologia ha raggiunto lo stadio della verifica sperimentale in accordo con le norme tecniche, seppur in scala ridotta, al di là della dimostrazione del mero principio di base. Lo stadio successivo delle attività di ricerca precederebbe la realizzazione in scala reale di prototipi per test più vicini agli ambienti industriali.

Deliverables

Unità di ricerca	DESCRIZIONE DELIVERABLE	RISULTATI QUANTITATIVI
UNIBO-DEI	Il report è diviso in due parti. La prima è relativa al confronto tra le diverse tecniche di controllo della corrente con regolatori ripetitivi, mentre la seconda illustra le tecniche di islanding detection e le modalità di prova.	Il report include la modellistica dei convertitori e dei sistemi di controllo e le relative simulazioni numeriche in ambiente Matlab/Simulink o PLECS con verifiche sperimentali e l'analisi del problema dell'islanding in sistemi di generazione distribuita.

Benefici per gli utenti finali

Le tecniche di controllo sviluppate per il miglioramento della qualità della corrente assorbita dalla rete comportano diversi vantaggi, sia per l'utente finale, sia per il gestore della rete.

In primo luogo, è possibile una riduzione dei requisiti dei filtri EMC in ingresso e di rifasamento, e conseguentemente del loro ingombro, che potrebbero essere ridotti in virtù della migliore qualità della corrente.

In secondo luogo, la riduzione delle armoniche in rete e delle perdite del convertitore comporta un aumento potenziale dell'efficienza del gestore, che potrebbe auspicabilmente riflettersi in una riduzione del costo dell'energia per l'utente finale, o comunque un risparmio delle risorse energetiche.

Diffusione dei risultati

Una prima diffusione dei risultati conseguiti durante l'attività di ricerca è avvenuta attraverso le pubblicazioni seguenti:

- Albino Amerise, Michele Mengoni, Luca Zarri, Angelo Tani, Giovanni Serra, Domenico Casadei, "Control system for shunt active power filters with adaptive voltage saturation," IEEE Energy Conversion Congress and Expo (ECCE 2017), Cincinnati, US, Sept. 30- Oct. 5, 2017, pp.682-687, ISBN: 978-1-5090-2997-6, IEEE Catalog Number: CFP17ECD-USB, 10.1109/ECCE.2017.8095850.

- A. Amerise, M. Mengoni, G. Rizzoli, L. Zarri, A. Tani, D. Casadei, "Adaptive Voltage Saturation Algorithms for Selective Harmonic Control in Shunt Active Power Filters," IEEE Energy Conversion Congress and Expo (ECCE 2018), Portland, US, September 23-27, 2018, pp. 5692-5698, IEEE Catalog Number: CFP18ECD-USB, ISBN USB: 978-1-4799-7311-8.

Entrambi i lavori scientifici analizzano il sistema di controllo di un convertitore active-front end nella configurazione da filtro attivo per il miglioramento della qualità della corrente. In particolare, viene considerato il caso della saturazione dei regolatori di corrente qualora la tensione disponibile per gli obiettivi di controllo sia insufficiente.

IEEE Energy Conversion Congress and Expo (ECCE) è una tra le più note conferenze nel settore dell'elettronica di potenza e degli azionamenti elettrici, vede la partecipazione di circa 1500 persone, una fiera industriale e circa 1000 presentazioni scientifiche. I lavori sono selezionati sulla base del giudizio di 5 esperti del settore e la percentuale di accettazione è spesso inferiore al 50%. I lavori presentati sono indicizzati sul database IEEE Xplore.

Apparecchiature disponibili e acquisite

Lo schema del sistema di controllo dei convertitori elettronici è stato implementato nell'ambiente Matlab/Simulink, prodotto da Mathworks, di uso comune in ambito universitario.

Il modello circuitale del convertitore, la rete di alimentazione, i carichi distorcenti sono stati invece realizzati in due ambienti software diversi:

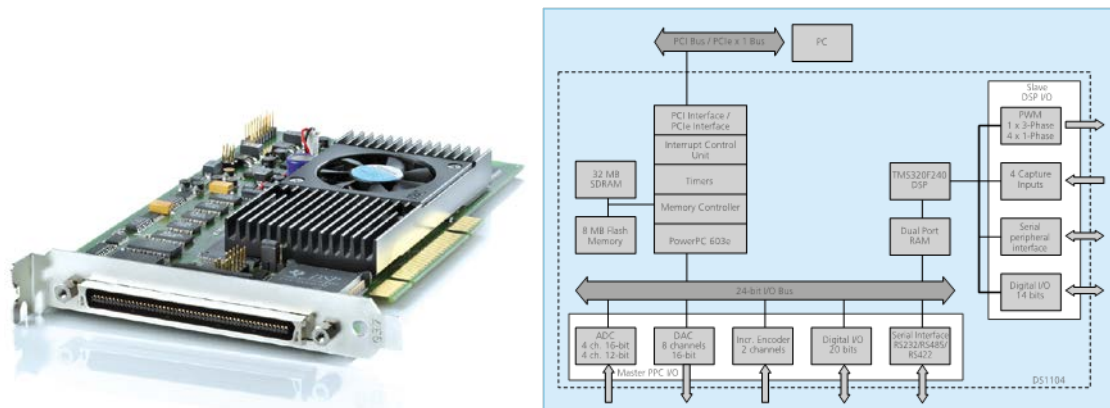
- Powersystem toolbox, di Mathworks
- PLECS toolbox, di PLECSIM

Il primo è un toolbox di Matlab ampiamente noto per la realizzazione di simulazione di sistemi elettrici. Ha il vantaggio di essere perfettamente integrato in Simulink, ma non nasce specificatamente per la simulazione di sistemi elettronici di potenza in commutazione.

Diversamente, PLECS Blockset (Piece-wise Linear Electrical Circuit Simulation) è sviluppato in particolare per la simulazione dei circuiti elettrici in commutazione. Il vantaggio principale consiste nella buona velocità di esecuzione, perché l'algoritmo d'integrazione si limita a calcolare lo stato del convertitore statico solo negli istanti di commutazione, approssimando per quanto possibile le forme d'onda inter-ciclo in modo lineare.

L'implementazione sperimentale degli schemi di controllo è avvenuta mediante la piattaforma dSpace DS1104 installata in un PC in configurazione miditower (Fig. 50).

La scheda DS1104 comunica all'esterno mediante il pannello connettore CP1104 (Fig. 51), che permette l'acquisizione e l'invio di segnali gestiti dalla scheda. Il pannello connettore è provvisto di sedici connettori BNC (otto relativi ai canali ADC e otto ai canali DAC), di un connettore sub-D a 37 pin (Digital I/O Connector), di un secondo connettore sub-D a 37 pin (Slave I/O PWM Connector) per i segnali di modulazione PWM, di due connettori sub-D a 15 pin (Incremental Encoder Interface Connectors) per l'utilizzo di due encoder incrementali, di due connettori sub-D a 9 pin (UART RS232 e UART RS485/RS422 Connectors) che permettono la comunicazione con dispositivi esterni (UART significa Universal Asynchronous Receiver and Transmitter) e di 54 led indicanti lo stato dei segnali digitali. I dispositivi utilizzati durante le prove possono quindi essere facilmente collegati e scollegati.



Parameter	Specification
Processor	<ul style="list-style-type: none"> MPC8240 processor with PPC 603e core and on-chip peripherals 64-bit floating-point processor CPU clock: 250 MHz 2 x 16 KB cache, on-chip
Memory	<ul style="list-style-type: none"> Global memory: 32 MB SDRAM Flash memory: 8 MB
Timer	<ul style="list-style-type: none"> 4 general-purpose timers: <ul style="list-style-type: none"> 32-bit down counter Reload by hardware 80-ns resolution 1 sampling rate timer (decrementer): <ul style="list-style-type: none"> 32-bit down counter Reload by software 40-ns resolution 1 time base counter: <ul style="list-style-type: none"> 64-bit up counter 40-ns resolution
Interrupt controller	<ul style="list-style-type: none"> 5 timer interrupts 2 incremental encoder index line interrupts 1 UART interrupt 1 slave DSP interrupt 1 slave DSP PWM interrupt 5 A/D converter (end of conversion) interrupts 1 host interrupt 4 external interrupts (user interrupts)

Fig. 50 – Sistema dSpace DS1104. Schema di massima e specifiche.

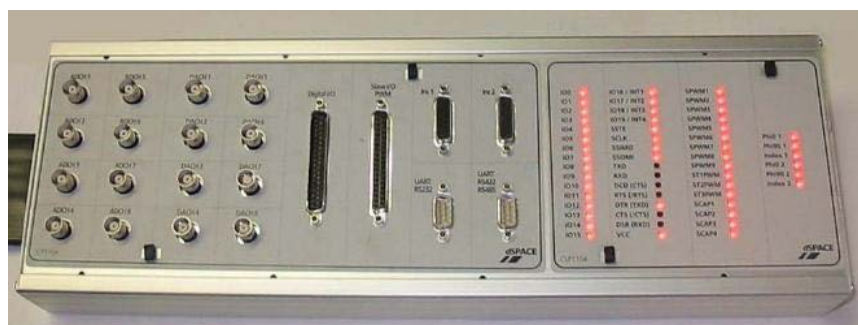


Fig. 51- Pannello connettore CP1104.

Attività 2.2 Sviluppo di un dimostratore di smart micro-grid, basato su un'architettura in corrente continua che integra fonti rinnovabili - sistemi di accumulo stazionari e carichi energivori (CNR-IM)

Obiettivi

Progettazione ed allestimento di una architettura in corrente continua a supporto di utenze energivore in contesti di tipo smart district, al fine di ridurre l'impatto sulla rete elettrica e favorire l'impiego di energia da fonti rinnovabili.

Le attività di questa linea proseguiranno nelle successive annualità del Piano Triennale (POA 2016 2017).

Risultati conseguiti

L'attività di ricerca svolta presso il CNR Istituto Motori di Napoli ha riguardato lo sviluppo di un dimostratore da laboratorio di architettura in corrente continua, a supporto dell'integrazione di fonti rinnovabili ed utenze energivore in contesti di tipo smart district. I principali risultati conseguiti possono essere sintetizzati nei seguenti due punti:

- Scelta della architettura di potenza e dei singoli componenti che costituiscono il dimostratore di smart micro-grid, mediante l'analisi delle diverse soluzioni presentate in letteratura e valutazioni preliminari in ambiente di simulazione.
- Allestimento del dimostratore di smart micro-grid e dei suoi principali componenti con particolare riferimenti ai sistemi di conversione, accumulo e generazione dell'energia elettrica da fonte fotovoltaica.

Si riporta di seguito una descrizione dettagliata dei risultati conseguiti.

Caso Studio

L'attività di ricerca 2.2 proposta da CNR - IM è stata riferita ad un caso studio di micro-grid in corrente continua rappresentativa di uno smart district, che integra utenze elettriche in corrente continua, sistemi di generazione di energia elettrica da fonti rinnovabili e sistemi di accumulo stazionario. Il caso studio prevede inoltre la presenza di utenze energivore di tipo industriale, che operano in corrente alternata e possono scambiare energia con la micro-grid in base a logiche *demand side management*. A riguardo in Fig. 2.2.1 è riportato uno schema a blocchi della applicazione considerata.

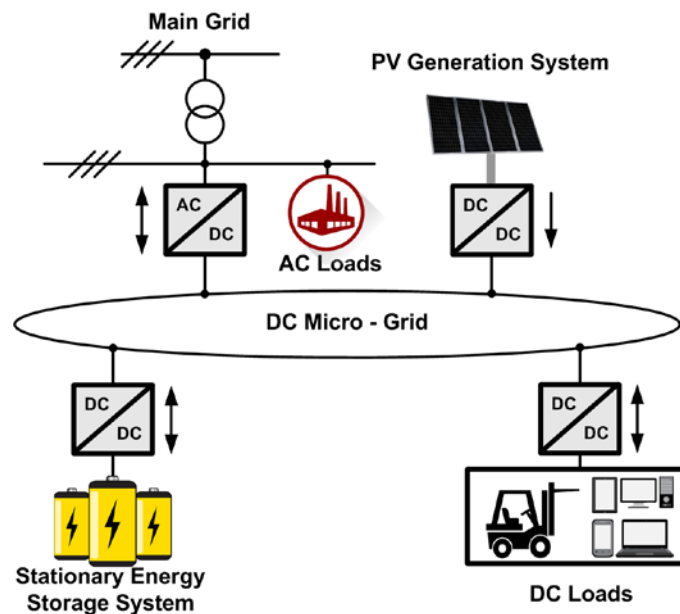


Fig. 2.2.1. Schema a blocchi della micro-grid considerata come caso studio per l'attività 2.2.

In particolare, l'architettura proposta è basata su un bus in corrente continua, che consente lo scambio energetico tra i diversi attori dello smart district, ed è ottenuto attraverso l'impiego di un convertitore di potenza AC/DC di tipo bidirezionale (grid-tied converter). Il convertitore AC/DC permette di prelevare o fornire energia da e verso la rete, in funzione delle condizioni operative richieste per la micro-grid. L'integrazione di sistemi di accumulo stazionari, carichi in corrente continua e fonti rinnovabili è gestita mediante l'impiego di convertitori DC/DC bidirezionali o unidirezionali, basati su configurazioni a singolo o multiplo stadio di conversione (es. multilivello). L'architettura micro-grid in corrente continua presenta notevoli vantaggi in termini di efficienza, complessità, modularità e costi. Infatti, l'ottimizzazione dei flussi energetici all'interno della micro-grid consente un impiego ottimale delle fonti rinnovabili, dando la possibilità anche a carichi industriali, che operano in corrente alternata, di poter usufruire di energia pulita e a basso costo.

Per la suddetta architettura, il grid-tied converter riveste un ruolo di particolare rilievo in quanto ha il compito di gestire lo scambio energetico tra micro-grid e rete, controllando allo stesso tempo che vengano rispettati tutti i parametri di power quality richiesti dagli standard di rete. Per tali ragioni, l'ottimizzazione delle prestazioni del grid tied converter ha richiesto uno studio dettagliato di specifiche architetture e controlli ottimali di tale tipologia di convertitore AC/DC, oggetto delle attività di consulenza di UNIBO-DEI (2.1.). Ciò garantisce quindi anche un'ottimizzazione delle prestazioni globali della micro-grid studiata dal CNR-IM.

Attività di Simulazione

A partire dal caso studio descritto in precedenza, una prima fase di attività è stata rivolta all'implementazione di un modello Matlab Simulink rappresentativo del dimostratore da allestire presso i laboratori del CNR-IM. Le attività di simulazione sono state principalmente finalizzate a supportare la fase di dimensionamento dei convertitori, che compongono il dimostratore e dei singoli componenti connessi alla micro-grid. In particolare, il grid tied converter considerato è basato su una classica configurazione ad IGBT ed un'architettura di conversione trifase AC/DC a tensione impressa,

che si interfaccia con un bus in corrente continua ad 800 V DC. Lo schema elettrico di tale architettura è riportato in Fig. 2.2.2.

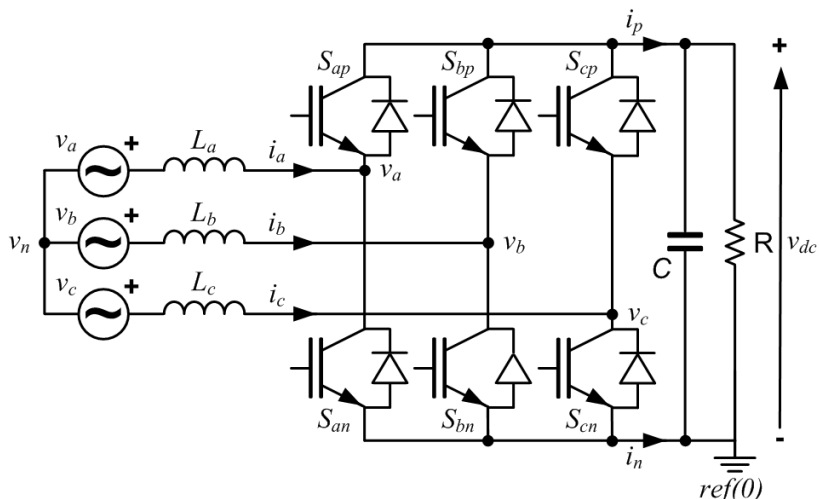


Fig. 2.2.2 Schema elettrico dell'architettura del grid-tied converter.

Per tale convertitore è stata inoltre prevista una logica di controllo della potenza reattiva attraverso uno schema basato su componenti in quadratura di tensione e corrente [1]-[3]. Tale schema è riportato in Fig. 2.2.3, in cui il riferimento di potenza reattiva è impostato uguale a 0.

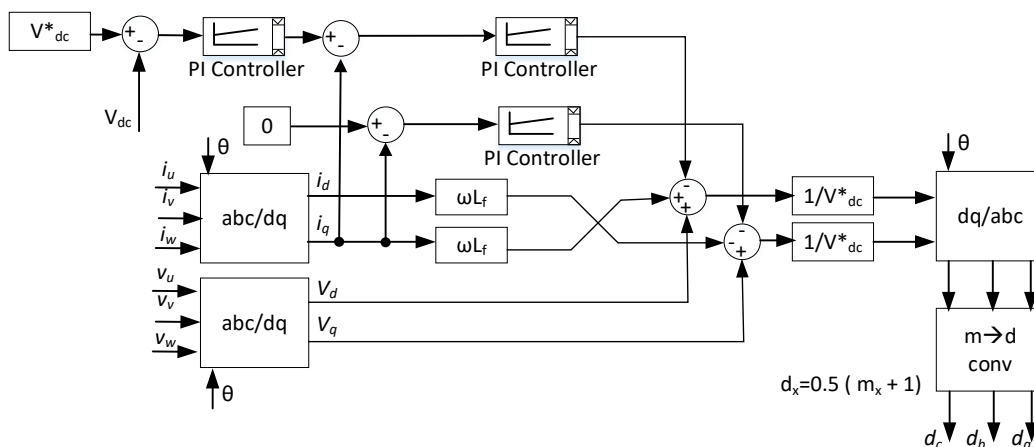


Fig. 2.2.3 Schema di controllo della potenza reattiva

Come accennato precedentemente, il convertitore AC/DC è stato il principale oggetto dell'attività di consulenza di UNIBO (2.1). Tali attività sono state focalizzate su schemi di controllo avanzati, di cui quello riportato in Fig. 2.2.3 rappresenta un caso particolare, e su specifiche architetture in grado di ottimizzare la power quality e la gestione della potenza reattiva nelle diverse condizioni di funzionamento della micro-grid considerata dal CNR-IM.

Per i convertitori DC/DC, che interfacciano sistemi di accumulo stazionario al bus in corrente continua, si è fatto invece riferimento ad un'architettura ad IGBT di tipo *interleaved*, il cui schema elettrico è riportato in Fig. 2.2.4.

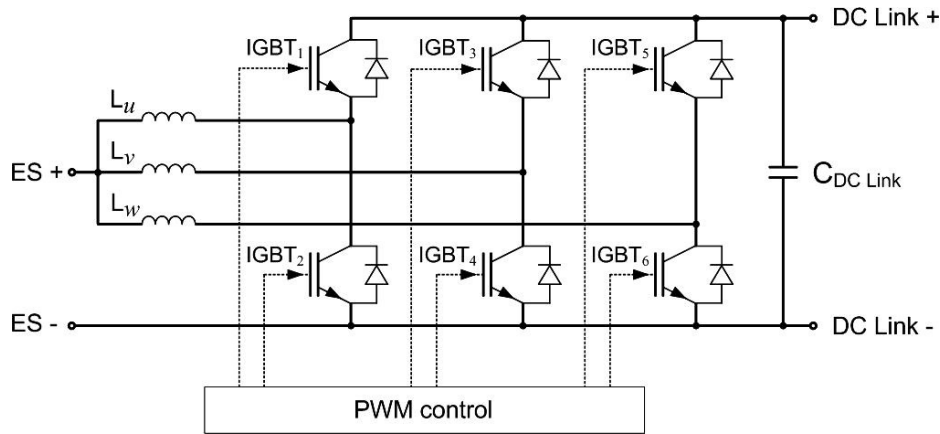


Fig. 2.2.4 Schema elettrico architettura convertitori DC/DC.

Tale architettura può essere ricavata anche attraverso la modifica di inverter trifase da fotovoltaico, cortocircuitando le tre fasi in ingresso e controllando opportunamente la corrente sui rami L_u , L_v ed L_w . Questa configurazione, come mostrato in letteratura [4] [5], consente di ottenere notevoli vantaggi in termini di costi, semplicità di controllo e *ripple* di corrente. Nel caso specifico, il convertitore DC/DC connesso a sistemi di accumulo può essere controllato utilizzando come riferimento la corrente di batteria, I_{ES} , sia in fase di carica che di scarica. Ulteriori anelli di controllo esterni sono inoltre previsti per poter utilizzare riferimenti basati sulla corrente in uscita dal convertitore lato DC-Link, $I_{DC-Link}$, e/o la tensione sul bus DC-Link, $V_{DC-Link}$.

Lo schema complessivo del modello Matlab-Simulink sviluppato nell'ambito delle attività oggetto di tale relazione è riportato in Fig.2.2.5.

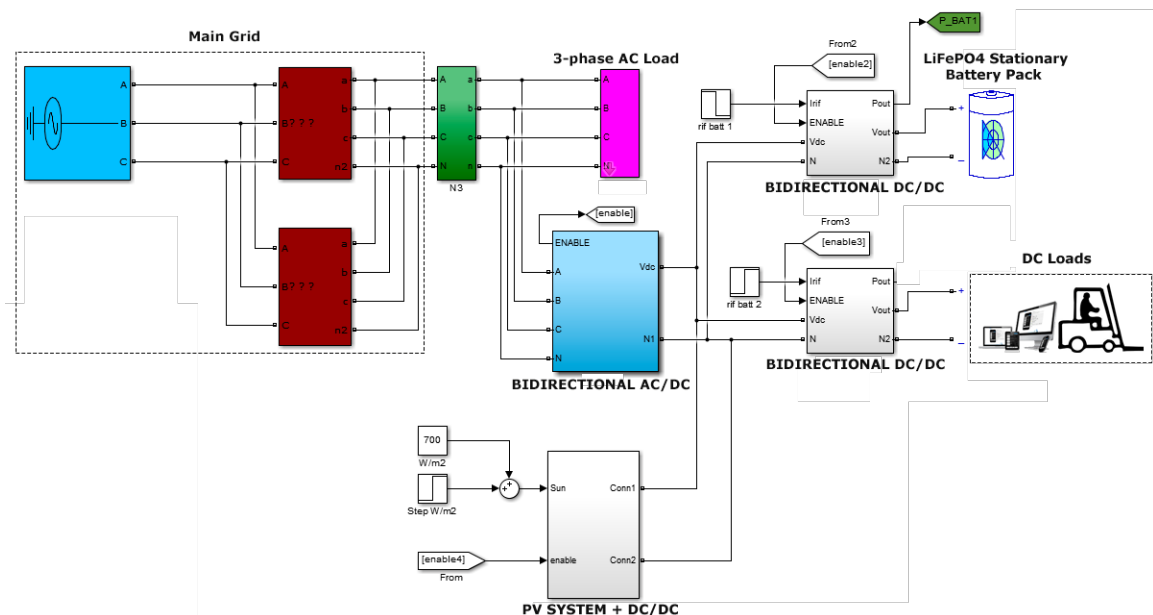


Fig. 2.2.5. Schema del modello Matlab Simulink (gerarchia di livello più alto).

A riguardo va precisato che, per la modellazione dei sistemi di conversione in ambiente Matlab-Simulink, si fa generalmente riferimento all'impiego di modelli estremamente semplificati, in cui i singoli convertitori sono simulati con generatori equivalenti di tensione e/o corrente, o di modelli ad elevata complessità computazionale, in cui i convertitori sono simulati a livello di singolo componente di switching. La prima tipologia di modelli di simulazione consiste nel considerare i singoli convertitori come black box e consente una rapida valutazione di strategie di gestione dei flussi energetici

all'interno di micro-grid, con possibilità di simulare condizioni di funzionamento su larga scala temporale. Tuttavia, con questo approccio, l'efficienza dei sistemi di conversione, i parametri di power quality e le dinamiche associate alla modulazione PWM sono completamente trascurate. D'altro canto la modellazione dei sistemi di conversione, a partire dall'azione della modulazione PWM sui singoli componenti di switching, risulta spesso troppo onerosa, in termini di tempi e complessità computazionale, per la simulazione di fenomeni di lunga durata (es. carica/scarica pacchi batterie, variazioni di carico giornaliere, ecc..).

Per i suddetti motivi, si è scelto di utilizzare un approccio di simulazione con un livello di complessità intermedio, basato sul modello generalizzato di modulazione PWM, indicato in letteratura come *Equivalent Average Circuit Model* [6]. Tale metodologia si basa sulla schematizzazione del singolo componente di switching riportata in, in Fig. 2.2.6:

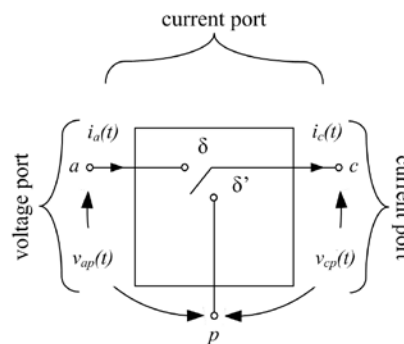


Fig. 2.2.6. Schematizzazione singolo componente di switching.

dove δ rappresenta il duty cycle per il componente considerato e $\delta' = 1 - \delta$ il suo complementare. In base a tale schematizzazione, i valori istantanei di corrente e tensione possono essere espressi in base alle equazioni 1.a e 1.b.

$$i_a(t) = \begin{cases} i_c(t), & 0 \leq t \leq \delta T_s \\ 0, & \delta T_s \leq t \leq T_s \end{cases} \tag{1.a}$$

$$v_{cp}(t) = \begin{cases} v_{ap}(t), & 0 \leq t \leq \delta T_s \\ 0, & \delta T_s \leq t \leq T_s \end{cases} \tag{1.b}$$

Partendo dalle suddette equazioni si può dimostrare che, i valori medi di corrente e tensione possono essere approssimati come $\overline{I_a} = \delta I_c$ e $\overline{V_{cp}} = \delta V_{ap}$ [6]. Pertanto, i singoli componenti di switching sono stati modellati con generatori equivalenti di tensione e corrente basati sui valori medi $\overline{I_a}$ e $\overline{V_{cp}}$. L'*Equivalent Average Circuit Model* del singolo componente può quindi essere rappresentato mediante lo schema riportato in Fig. 2.2.7.

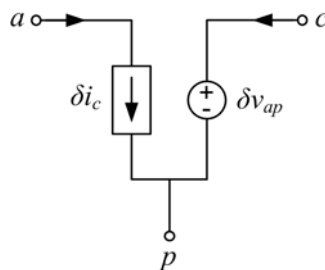


Fig. 2.2.7. *Equivalent Average Circuit Model* del singolo componente di switching.

In accordo al modello equivalente del singolo componente riportato in Fig. 2.2.7, Il modello di simulazione del grid tied converter può essere semplificato come riportato in Fig. 2.2.8, dove i termini d_a, d_b, d_c rappresentano gli indici di modulazione compresi nel range [0-1].

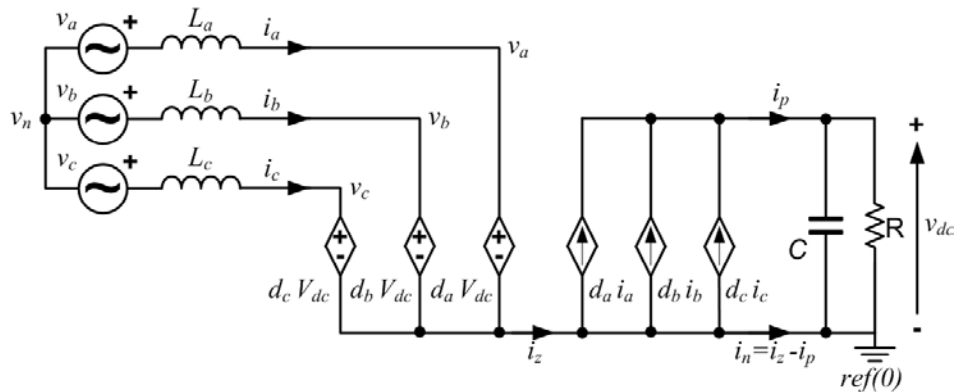


Fig. 2.2.8. Modello semplificato del convertitore AC/DC.

La stessa logica di modellazione è stata impiegata per la simulazione dei convertitori DC/DC connessi alla micro-grid.

Per quanto riguarda l'integrazione di fonti rinnovabili con la micro-grid si è fatto riferimento ad un sistema di generazione di tipo fotovoltaico da circa 7.5 kWp. L'implementazione del campo fotovoltaico nel modello Matlab-Simulink è stata realizzata facendo riferimento a condizioni di funzionamento in regime quasi-stazionario. Pertanto, le caratteristiche tensione-corrente dei singoli pannelli al variare delle condizioni di irraggiamento sono state implementate attraverso l'impiego di *look-up Table*. A riguardo, le caratteristiche utilizzate sono riportate in Fig. 2.2.9.

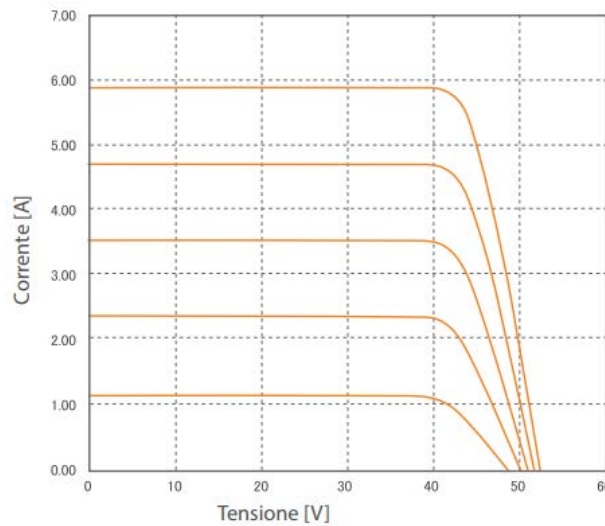


Fig. 2.2.9. Caratteristiche tensione corrente dei Pannelli Fotovoltaici connessi alla micro-grid [7].

A partire dalle suddette curve è stato quindi possibile implementare il comportamento di un campo fotovoltaico costituito da 4 stringhe, composte da 8 moduli, connesse tra loro elettricamente in parallelo.

Le attività ed i risultati ottenuti in ambiente di simulazione hanno quindi consentito di ricavare utili informazioni a supporto del dimensionamento del dimostratore di micro-grid da laboratorio oggetto di queste attività.

Allestimento del dimostratore di micro-grid

Allestimento dei sistemi di conversione e della micro-grid

L'allestimento del dimostratore di micro-grid, rappresentativa di uno smart district, è avvenuto mediante l'impiego e la configurazione di opportuni sistemi di conversione statica dell'energia elettrica, sistemi di accumulo stazionari, sistemi di generazione di tipo fotovoltaico e sistemi di acquisizione e controllo.

In particolare, il grid tied converter è stato realizzato a partire da un inverter trifase di tipo fotovoltaico ed è stato configurato, con il supporto dall'azienda produttrice, per gestire flussi di potenza bidirezionali e per lavorare come convertitore a tensione impressa sul lato in corrente continua. Le schede elettroniche ed i componenti di potenza sono state inoltre protette con un opportuno case in metallo. A riguardo, in Fig. 2.2.10 sono riportate immagini esterne ed interne del convertitore AC/DC installato presso il CNR-Istituto Motori.



Fig. 2.2.10. Immagine esterna ed interna del convertitore AC/DC installato presso il CNR Istituto Motori.

Il grid tied converter, a partire dalla rete trifase in bassa tensione AC, realizza un bus in corrente continua ad 800 V, che rappresenta il DC-Link della microgrid considerata. La gestione dei flussi di potenza bidirezionali avviene mediante il controllo della tensione di bus, $V_{DC-Link}$. Infatti, quando le diverse utenze connesse alla microgrid assorbono potenza elettrica, la tensione sul bus tende a diminuire ed il controllo PWM del convertitore interviene richiedendo un'aliquota di potenza maggiore alla rete elettrica AC. Viceversa, quando i sistemi di accumulo o di generazione dell'energia elettrica forniscono potenza alla micro-grid, la tensione sul bus tende ad aumentare ed il controllo PWM del convertitore interviene riversando in rete l'energia accumulata sul condensatore di bus, secondo quanto rappresentato nello schema elettrico del convertitore in Fig. 2.2.2. In Tabella 1, sono riportate le principali caratteristiche tecniche del convertitore AC/DC.

Tabella 1. Caratteristiche tecniche del convertitore AC/DC.

DC Link - Voltage	800 V DC
Rated current	32 A
Main grid voltage	380 – 480 V AC, 50/60 Hz
Rated current	32 A
Rated power	20 kVA
Control mode	DC bus voltage reference
Maximum efficiency	96%
Size (wxdxh)	160 × 561 × 291 mm
Weight	20 kg
Cooling	Air
Operative temperature	-10°C – +40°C
Isolation degree	IP 20
Input signal	Digital: ON/OFF, Reset, external relè (+24V DC)
Output signal	Digital output for fault operative conditions

Come evidente dalle caratteristiche riportate in tabella, il convertitore AC/DC è caratterizzato da una potenza nominale di 20 kVA. Pertanto, in tutte le condizioni di funzionamento la potenza richiesta e/o fornita alla rete elettrica non supera i 20 kW, mentre lo scambio di potenza sul bus in corrente continua può raggiungere valori di gran lunga maggiori, grazie al contributo dei sistemi di accumulo e di generazione dell'energia elettrica da fonte fotovoltaica. Lo studio di specifiche architetture e strategie di gestione e controllo ottimali per tale convertitore è stato approfondito attraverso le attività di consulenza di UNIBO (2.1). Infatti, per l'applicazione in esame, il convertitore AC/DC, oltre a operare da raddrizzatore controllato, ha anche il compito di migliorare il fattore di potenza nel punto di connessione alla rete, compensando la potenza reattiva e le correnti armoniche assorbite da eventuali carichi non lineari presenti in rete. A riguardo, per lo scambio di potenza tra micro-grid e rete, l'impiego del sistema di controllo con prestazioni ottimizzate, proposto da UNIBO-DEI, consente di ridurre il THDi della corrente di linea, a parità di filtro di ingresso, oppure di ridurre l'ingombro del filtro d'ingresso e conseguentemente anche le perdite complessive del dispositivo.

I convertitori DC/DC connessi al dimostratore sono stati configurati per integrare fonti rinnovabili e sistemi di accumulo con la micro-grid in corrente continua. In questo caso, il controllo dei convertitori avviene utilizzando come riferimento un segnale analogico, fornito da un sistema di controllo esterno, in corrente (4- 20 mA) o in tensione (0-10 V). Tale segnale è utilizzato per pilotare il sistema di controllo embedded dei convertitori e, in funzione della modalità di controllo scelta, è direttamente proporzionale al riferimento di corrente lato utenza/batteria/sistema di generazione, al riferimento di corrente lato DC-Link, o al riferimento di tensione lato DC-Link. La gestione dei flussi di potenza bidirezionali avviene invece attraverso l'impiego di un segnale digitale, che è fornito da un sistema di controllo esterno in funzione delle specifiche condizioni di funzionamento richieste dalle strategie di gestione delle richieste energetiche della micro-grid. L'attività di allestimento ha quindi riguardato anche la disposizione di morsettiere per la gestione dei segnali di acquisizione e controllo dei singoli convertitori. A riguardo, si riporta in Fig. 2.2.11 uno schema del convertitore DC/DC bidirezionale,

connesso al sistema di accumulo stazionario, con i principali morsetti di potenza, acquisizione e controllo.



Fig. 2.2.11 Schema del convertitore DC/DC bidirezionale con morsetti di potenza, acquisizione e controllo.

Il convertitore DC/DC ha subito alcune modifiche rispetto alla sua configurazione originale, che hanno peraltro comportato l'acquisto di schede elettroniche di potenza per la conversione statica dell'energia elettrica. Tali modifiche sono state implementate in collaborazione con il produttore ed hanno consentito la gestione dei flussi bidirezionali ed il controllo del convertitore mediante i riferimenti esterni sia di tipo analogico che digitale. Le schede elettroniche ed i componenti di potenza sono state inoltre coperte con un opportuno case di protezione. A riguardo, in Fig. 2.2.12 si riportano immagini esterne ed interne del convertitore DC/DC bidirezionale.



Fig. 2.2.12 Immagine esterna ed interna del convertitore DC/DC bidirezionale installato presso il CNR Istituto Motori.

Per il suddetto convertitore è stata prevista una corrente massima fino a 200 A in entrambe le direzioni di funzionamento, con un range di tensione, lato sistema di accumulo, che può andare da 50 V a 300 V. Tale convertitore può quindi essere utilizzato per supportare la micro-grid con potenze fino a circa 60 kW. A riguardo, in Tabella 2 si riportano le principali caratteristiche del convertitore DC/DC utilizzato per la gestione del sistema di accumulo stazionario.

Tabella 2. Principali Caratteristiche del convertitore DC/DC che alimenta il sistema di accumulo stazionario.

Input	
Voltage range [V]	35-750, DC
Rated Current [A]	20 A
Maximum Current [A]	100, 5 s/10 min
Output	
Voltage [V]	100-750, DC
Rated Current	20 A
Maximum Current	100 A, 5 s/10 min
Control Mode	- ES Current Reference - DC Link Voltage Reference
Rated Efficiency [p.u.]	0.96
Input Signals	Digital Signal: Charge/Discharge, Control mode selector (V/I) Analog Signals: Current Reference on the ES side (0-10 V or 0-20 mA)
Output Signals	Analog Signals: Voltage reference, Energy Storage and/or DC/Link Voltage (0-10 V)

I convertitori che realizzano il dimostratore di micro-grid, la sensoristica per l'acquisizione dei principali parametri elettrici di funzionamento ed i relativi dispositivi ausiliari e di protezione, sono stati integrati all'interno di un rack da laboratorio. Un'immagine completa del dimostratore realizzato al CNR-Istituto Motori è riportata di seguito in Fig. 2.2.13.



Fig. 2.2.13. Immagine esterna ed interna del quadro che integra il dimostratore di micro-grid.

Come visibile dalle foto, il bus in corrente continua è stato realizzato mediante due barre di rame a cui sono connessi il grid tied converter ed i convertitori DC/DC, che integrano sistemi di generazione di energia elettrica da fonti rinnovabili, sistemi di accumulo stazionari ed utenze elettriche in corrente

continua. Sono stati inoltre installati selettori e pulsanti fronte-quadro per poter controllare manualmente l'attivazione dei singoli convertitori e l'interconnessione della micro-grid con i singoli componenti elettrici esterni. Sono stati infine previsti un display digitale, che segnala la presenza di tensione sul bus in corrente continua, ed uno smart meter *Diris A40*, per l'analisi delle principali grandezze di rete. Il *Diris A40* è stato inoltre equipaggiato con un modulo di comunicazione aggiuntivo basato su protocollo RS-485.

Il dimostratore di micro-grid è stato dotato di un sistema di acquisizione e controllo basato su Hardware modulare National Instruments. In Particolare è stata impiegata una Ni cDaQ-9188, equipaggiata con:

- 2 moduli NI 9221 per l'acquisizione di segnali analogici in tensione con valori compresi nel range $\pm 60\text{ V}$
- 1 modulo NI 9207, per l'acquisizione degli ingressi analogici in corrente con valori compresi nel range $\pm 20\text{ mA}$
- 1 modulo di controllo NI 926, per l'impiego di uscite analogiche in tensione con valori compresi nel range $0\text{-}10\text{ V}$.

Si riporta di seguito in Fig. 2.2.14 un'immagine completa del sistema National Instruments configurato per i test sperimentali sul dimostratore.

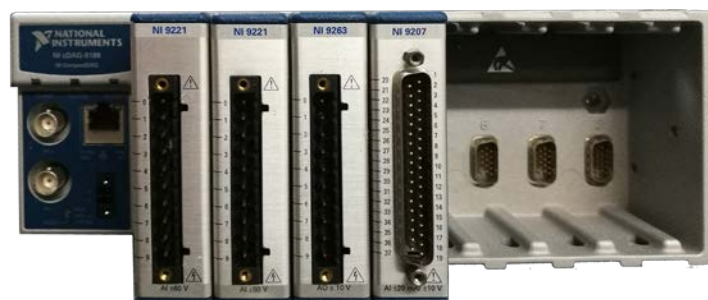


Fig. 2.2.14. Sistema di acquisizione National Instruments basato su NI cDaq -9188.

L'Hardware National Instruments descritto in precedenza è gestito mediante l'impiego di interfacce software implementate in ambiente Labview. Per operazioni di acquisizione e controllo real time è stato inoltre previsto l'impiego di sistemi Nidec con processore embedded, che sono stati configurati attraverso il software *Sypt Pro*. La licenza per l'impiego di tale software è stata fornita da Nidec Industrial Automation nell'ambito del POA 2015.

Il sistema smart meter *Diris A40*+modulo RS485 consente di acquisire, attraverso un'interfaccia software remota, utili informazioni sugli scambi energetici tra micro-grid e rete elettrica in termini di efficienza e power quality. L'analizzatore è munito di sensori di corrente, che sono stati posizionati sulle singole fasi in corrispondenza del punto di scambio tra micro-grid e rete AC. A riguardo, si riporta in Fig. 2.2.15 lo schema di collegamento del *Diris A40*, comprensivo dei relativi sensori.

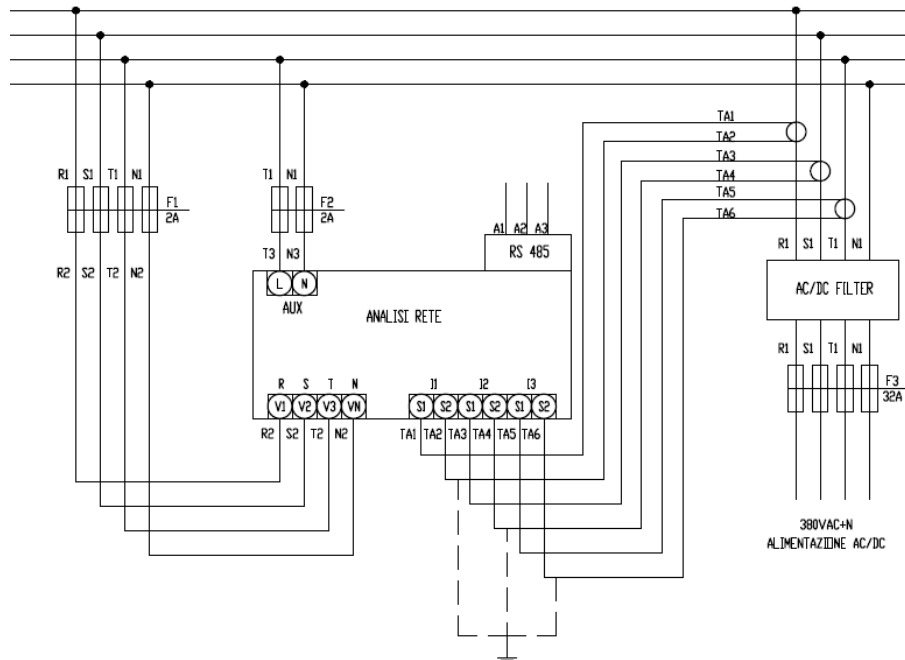


Fig. 2.2.15. schema di collegamento del Diris A40.

Come mostrato nello schema di collegamento sono stati inoltre previsti dei fusibili di protezione per il collegamento ai morsetti di misura della tensione (F1), per l'alimentazione dell'analizzatore di rete (F2) e per l'alimentazione del convertitore AC/DC (F3). Il sistema di smart metering realizzato consente quindi di focalizzare particolarmente l'attenzione sui parametri di power quality durante lo svolgimento dei test di caratterizzazione del dimostratore. E' quindi possibile confrontare soluzioni tradizionali di controllo del convertitore AC/DC con soluzioni ottimali di gestione della power quality, basate sulle tecniche di controllo avanzate, oggetto delle attività di consulenza di UNIBO (2.1). L'elenco delle misure ed analisi che è possibile ottenere attraverso l'interazione da remoto con il Diris A40 è riportato in Tabella 3.

Tabella 3. Principali Misure ed Analisi registrate a bordo del Diris A40

Principali Misure	Correnti, Tensioni, Frequenza, Potenze attive, reattive e apparenti, Fattore di potenza
	Correnti, tensioni e frequenza medie
	Potenze medie
Analisi della qualità	THD 3U, 3V, 3 I/In Fino al grado 63
	Singole componenti armoniche fino al grado 63

Le misure di tensione e corrente all'interno della micro-grid sono state eseguite attraverso l'impiego di sensori LEM che prevedono un'alimentazione bipolare a $\pm 15V$. A titolo di esempio, in Fig. 2.2.16, si riporta lo schema di misura di tensione e corrente sul bus in corrente continua.

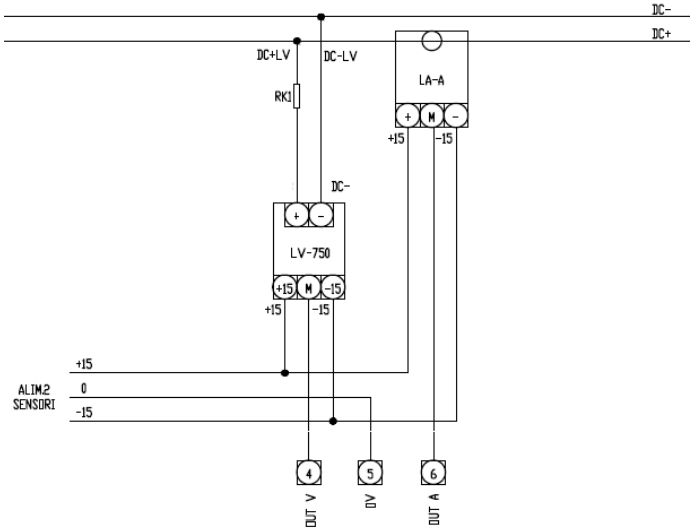


Fig. 2.2.16. Schema di misura tensione e corrente su bus in corrente continua.

In particolare, per la misura di tensione è stato utilizzato un trasduttore un LEM LV-100, che accetta valori di tensione in ingresso fino a 2500 V. Le principali caratteristiche di tale trasduttore sono riportate in Tabella 4.

Tabella 4. Principali Caratteristiche LEM LV-100 [8].

Electrical data			
I_{PN}	Primary nominal current rms	10	mA
I_{PM}	Primary current, measuring range	0 ± 20	mA
R_M	Measuring resistance with ± 15 V	@ ± 10 mA _{max}	$R_{M \min}$ $R_{M \max}$ 0 150 Ω
		@ ± 20 mA _{max}	0 50 Ω
I_{SN}	Secondary nominal current rms	50	mA
K_N	Conversion ratio	10000 : 2000	
V_C	Supply voltage (± 5 %)	± 15	V
I_C	Current consumption	$31 + I_S$	mA
Accuracy - Dynamic performance data			
X_S	Overall accuracy @ I_{PN} , $T_A = 25^\circ\text{C}$	± 0.7	%
ϵ_L	Linearity error	< 0.1	%
I_O	Offset current @ $I_P = 0$, $T_A = 25^\circ\text{C}$	Typ	± 0.2 mA
		Max	± 0.2 mA
I_{OT}	Temperature variation of I_O 0°C .. + 70°C	± 0.2	± 0.3 mA
t_r	Response time ¹⁾ to 90 % of I_{PN} step	20 .. 100	μs
General data			
T_A	Ambient operating temperature	0 .. 70	$^\circ\text{C}$
T_S	Ambient storage temperature	- 25 .. + 85	$^\circ\text{C}$
R_P	Primary coil resistance @ $T_A = 70^\circ\text{C}$	1900	Ω
R_S	Secondary coil resistance @ $T_A = 70^\circ\text{C}$	60	Ω
m	Mass	460	g
	Standards	EN 50178: 1997	

Come suggerito dal suo datasheet, il LEM LV-100 presenta la massima precisione (± 0.7 %) quando al primario circola una corrente corrispondente al valore nominale $I_{PN} = 10$ mA. Il dimensionamento della resistenza in ingresso al sensore, $R_{K1} = 100$ k Ω , è stato eseguito per ottenere il valore di corrente nominale al primario in corrispondenza del valore di tensione, $V_{DC-Link} = 1000$ V, che è prossimo alla tensione del bus, $V_{DC-Link}$. In questo caso, è stato necessario utilizzare resistenze corazzate in quanto la potenza elettrica dissipata sulla resistenza R_{K1} è pari a circa 10 W. Analogamente, al valore di corrente primaria, $I_{PN} = 10$ mA, corrisponde una corrente nominale al secondario, $I_{SN} = K_N \cdot I_{PN} = 50$ mA. Tale valore

non rientra nel range ($\pm 20 \text{ mA}$) delle correnti ammissibili per la scheda di acquisizione NI 9207. Pertanto, l'uscita del LEM LV-100 deve essere ulteriormente convertita per essere acquisita su un ingresso analogico in tensione del modulo National nel range $\pm 10 \text{ V}$. Per tale conversione è stata impiegata una resistenza di misura R_M , dimensionata su un valore di tensione acquisita pari a circa 5 V , secondo la formula $R_M = 5 \text{ V} / 50 \text{ mA} = 100 \Omega$.

La misura di corrente sul bus DC è invece realizzata mediante un trasduttore ad effetto Hall LEM LA 125-P, che consente di misurare valori di corrente compresi nel range $\pm 200 \text{ A}$. Le principali caratteristiche di tale trasduttore sono riportate in Tabella 5.

Tabella 5. Principali Caratteristiche LEM LA 125-P [9].

Electrical data								
I_{PN}	Primary nominal current rms		125	A				
I_{PM}	Primary current, measuring range		0 .. ± 200	A				
R_M	Measuring resistance		$T_A = 70 \text{ }^\circ\text{C}$	$T_A = 85 \text{ }^\circ\text{C}$				
			$R_{M \text{ min}}$	$R_{M \text{ max}}$				
		with $\pm 12 \text{ V}$	@ $\pm 125 \text{ A}_{\text{max}}$	5	52	14	50	Ω
			@ $\pm 200 \text{ A}_{\text{max}}$	5	20	14	18	Ω
		with $\pm 15 \text{ V}$	@ $\pm 125 \text{ A}_{\text{max}}$	25	74	40	72	Ω
	@ $\pm 200 \text{ A}_{\text{max}}$	25	34	40 ¹⁾	40 ¹⁾	Ω		
I_{SN}	Secondary nominal current rms		125	mA				
K_N	Conversion ratio		1 : 1000					
U_C	Supply voltage ($\pm 5 \%$)		$\pm 12 \dots 15$	V				
I_C	Current consumption		16 (@ $\pm 15 \text{ V}$) + I_S	mA				
Accuracy - Dynamic performance data								
X	Accuracy @ I_{PM} , $T_A = 25 \text{ }^\circ\text{C}$	@ $\pm 15 \text{ V} (\pm 5 \%)$	± 0.60	%				
		@ $\pm 12 \dots 15 \text{ V} (\pm 5 \%)$	± 0.80	%				
ϵ_L	Linearity error		< 0.15	%				
I_O	Offset current @ $I_p = 0$, $T_A = 25 \text{ }^\circ\text{C}$		Typ	Max				
I_{OM}	Magnetic offset current ²⁾ @ $I_p = 0$ and specified R_M			± 0.40				
		after an overload of $3 \times I_{PN}$		± 0.50				
I_{OT}	Temperature variation of I_O	0 $^\circ\text{C}$.. + 70 $^\circ\text{C}$	± 0.15	± 0.50				
		- 40 $^\circ\text{C}$.. + 85 $^\circ\text{C}$	± 0.30	± 0.95				
t_{ra}	Reaction time		< 500	ns				
t_r	Step response time ³⁾⁴⁾ to 90 % of I_{PN}		< 1	μs				
dI/dt	dI/dt accurately followed ⁴⁾		> 200	A/ μs				
BW	Frequency bandwidth ⁴⁾ (- 1 dB)		DC .. 100	kHz				
General data								
T_A	Ambient operating temperature		- 40 .. + 85	$^\circ\text{C}$				
T_S	Ambient storage temperature		- 40 .. + 90	$^\circ\text{C}$				
R_S	Secondary coil resistance	@ $T_A = 70 \text{ }^\circ\text{C}$	32	Ω				
		@ $T_A = 85 \text{ }^\circ\text{C}$	33.5	Ω				
m	Mass		40	g				
		Standard		EN 50178: 1997				

Tale trasduttore presenta un'uscita in corrente al secondario che, in base al rapporto di conversione $K_N = 1:1000$, è compresa nel range $\pm 200 \text{ mA}$. Pertanto, anche in questo caso, è stato necessario convertire l'uscita del sensore di corrente, attraverso l'impiego di una resistenza di misura R_M , ed acquisirla su un ingresso analogico in tensione nel range $\pm 10 \text{ V}$.

L'allestimento del dimostratore, dei sistemi di conversione, del sistema di acquisizione/controllo e della relativa sensoristica rende possibile lo svolgimento di test sperimentali volti a caratterizzare il dimostratore nelle diverse condizioni di funzionamento.

La fase di allestimento è stata quindi ulteriormente completata con l'allestimento di un sistema di accumulo stazionario e di un sistema di generazione dell'energia elettrica da fonte fotovoltaica.

Allestimento del sistema di accumulo stazionario

Come detto precedentemente, i sistemi di accumulo stazionari rappresentano uno degli elementi di maggiore importanza per un'efficiente gestione dei flussi energetici all'interno di micro-grid, in contesti di tipo smart district. Nell'ambito del POA 2015, le attività sono state quindi rivolte anche all'allestimento di un sistema di accumulo stazionario, in scala reale, interconnesso con il dimostratore di micro-grid.

Il sistema di accumulo che è stato allestito in Istituto Motori è basato su celle LiFePO_4 . Tale tecnologia sta avendo una crescente diffusione nel settore delle applicazioni di accumulo stazionario di energia in quanto consente, rispetto a tecnologie basate su composti del litio a più elevata densità energetica, di ridurre notevolmente i costi e di incrementare sicurezza e vita utile dei sistemi di accumulo. Le celle al litio basate su catodo LiFePO_4 , sono state proposte per la prima volta nel 1996 dall'Università del Texas [10]. I principali vantaggi di questa tecnologia, in termini di costi, sono legati all'utilizzo del ferro, che è un elemento di basso costo e facilmente reperibile in natura. I vantaggi in termini di sicurezza sono invece dovuti alla forza del legame covalente Fe-P-O, che è caratterizzato dalla ridotta possibilità di rilascio di ossigeno, con conseguente riduzione del rischio di esplosione. Le celle di accumulo LiFePO_4 presentano una tensione nominale di cella di 3.2 V a 25 °C e da un'energia specifica che raggiunge valori fino a 140 Wh/kg.

Le celle di accumulo impiegate per la realizzazione del pacco batteria sono LiFePO_4 da 90 Ah prodotte dalla *Winston Batteries*. Le principali caratteristiche e limiti operativi di tali celle sono riportate in Tabella 6.

Tabella 6. Principali caratteristiche e limiti operativi delle celle Winston LiFePO_4 - 90 Ah

Model name	LFP090AHA	
Nominal voltage	3.2 V	Operating voltage under load is 3.0 V
Capacity	90 AH	+/- 5%
Operating voltage	max 4.0V - min 2.8V	At 80% DOD
Deep discharge voltage	2.5 V	The cells is damaged if voltage drops below this level
Maximal charge voltage	4 V	The cells is damaged if voltage exceeds this level
Optimal discharge current	< 45 A	0.5 C
Maximal discharge current	< 270 A	3 C, continuous for max 15 minutes from full charge
Max peak discharge current	< 900 A	10 C, maximal 5 seconds in 1 minute
Optimal charge current	< 45 A	0.5 C
Maximal charge current	< 270 A	< 3 C with battery temperature monitoring
Maximal continuous operating temperature	80 °C	The battery temperature should not increase this level during charge and discharge
Dimensions	143 x 218 x 61	Millimeters (tolerance +/- 2 mm)
Weight	3 kg	Kilograms (tolerance +/- 150g)

Il pacco batterie allestito presso il CNR-Istituto Motori è basato su 90 celle collegate elettricamente in serie, per una tensione nominale complessiva pari a circa 288 V. Lo schema elettrico di collegamento delle 90 celle che realizzano il sistema di accumulo stazionario è riportato in Fig. 2.2.17.

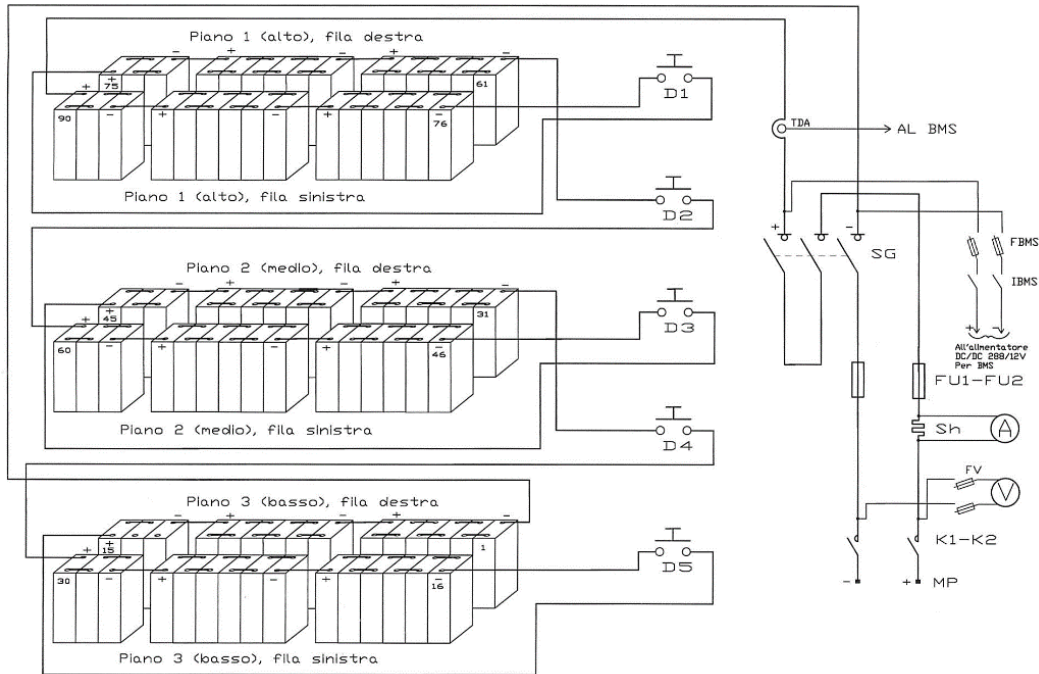


Fig. 2.2.17. Schema elettrico di collegamento del sistema di accumulo stazionario.

Come mostrato, sono stati previsti 5 sezionatori intermedi (D1÷D5) al fine di isolare singoli gruppi da 18 celle. Il sistema di sezionatori così realizzato consente quindi di ottenere valori di tensione intermedi di circa 60 V, che favoriscono lo svolgimento delle operazioni di manutenzione in condizioni di sicurezza. Sono stati inoltre previsti fusibili ad azione rapida (FU1, FU2) ed un sezionatore generale (SG) per garantire la sicurezza durante le operazioni la connessione/disconnessione del sistema di accumulo stazionario con la micro-grid.

La gestione del sistema di accumulo stazionario avviene attraverso un Battery Management System (BMS) SCB 90 prodotto da Alchemy Italia S.r.l.. Il BMS SCB 90 è basato su un'architettura distribuita, caratterizzata da 6 unità slave e da un'unità master. Le unità slave acquisiscono i parametri significativi di ogni singola cella per poi trasmetterli via CAN Bus all'unità master. In questo modo il sistema SCB 90 monitora lo stato di ciascuna delle 90 celle, assicurandone il corretto funzionamento entro i limiti operativi previsti dal costruttore. In Fig. 2.2.18. si riporta un'immagine di un'unità slave connessa al BMS.



Fig. 2.2.18. Immagine di una delle unità slave connessa al BMS.

Il controllo del BMS avviene attraverso la misura delle 90 tensioni di singola cella e di 36 valori di temperatura in punti significativi del pacco batteria. La scheda master dell'SCB90 è alimentata

direttamente dal pacco batteria, mediante l'impiego di un convertitore DC/DC con uscita a 12 VDC.

Il BMS consente di far operare l'intero pacco batteria in condizioni di funzionamento ottimali, evitando anomalie che potrebbero avere effetti negativi sul funzionamento dell'intero sistema e sulla vita utile delle singole celle. In particolare sono monitorate le seguenti condizioni di anomalia:

- Superamento della tensione massima di carica di singola cella
- Stato di carica < 20%
- Superamento della tensione minima di scarica di singola cella
- Temperatura elevata (comando sistema di ventilazione)
- Sovratemperatura dissipatori
- Sovra-corrente di carica/scarica.
- Guasto sonda corrente
- Guasto SCB90

Tutte le misure fatte dal BMS possono essere acquisite su un computer remoto attraverso protocollo CAN. A titolo di esempio, in Fig. 2.2.19, si riporta un'acquisizione della tensione a vuoto di singola cella, realizzata con Stato di Carica (SoC) pari a 40%.

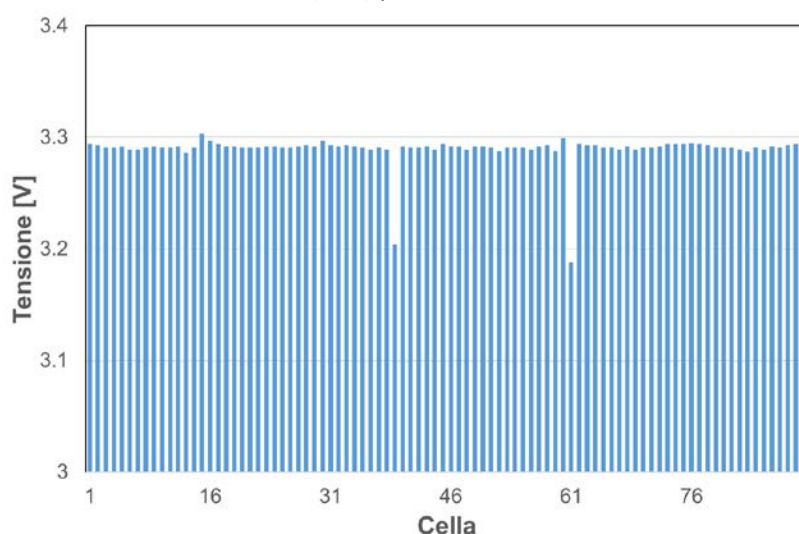


Fig. 2.2.19. Misura delle tensioni di singola cella per uno stato di carica pari al 40 %.

Dalle suddette acquisizioni è possibile osservare che il valore minimo di tensione, $V_{min} = 3.18 V$, è misurato per la cella 61, mentre il valore massimo di tensione, $V_{max} = 3.30 V$, è misurato per la cella 15. In queste condizioni di funzionamento, il BMS fornisce informazioni anche sul valore medio delle tensioni di cella, $V_{avg} = 3.29 V$, e la tensione totale del pacco batteria $V_{tot} = 296 V$.

Le 90 celle che compongono il pacco batteria, insieme con i sistemi di protezione e controllo e con la relativa sensoristica, sono state disposte in un armadio mobile. Tale armadio è composto da una parte otticamente accessibile, in cui sono disposte le singole celle di accumulo con le relative unità slave, e da un quadro elettrico per la gestione e l'interconnessione del sistema di accumulo, in cui sono disposti i sezionatori, il BMS con il relativo convertitore per l'alimentazione a 12 V ed i trasduttori di tensione e corrente. Sono stati inoltre previsti indicatori analogici fronte quadro per poter monitorare i valori di tensione, corrente e stato di carica del pacco batteria. A riguardo, in Fig. 2.2.20 sono mostrate immagini interne ed esterne del quadro elettrico per la gestione e l'interconnessione del sistema di accumulo stazionario.

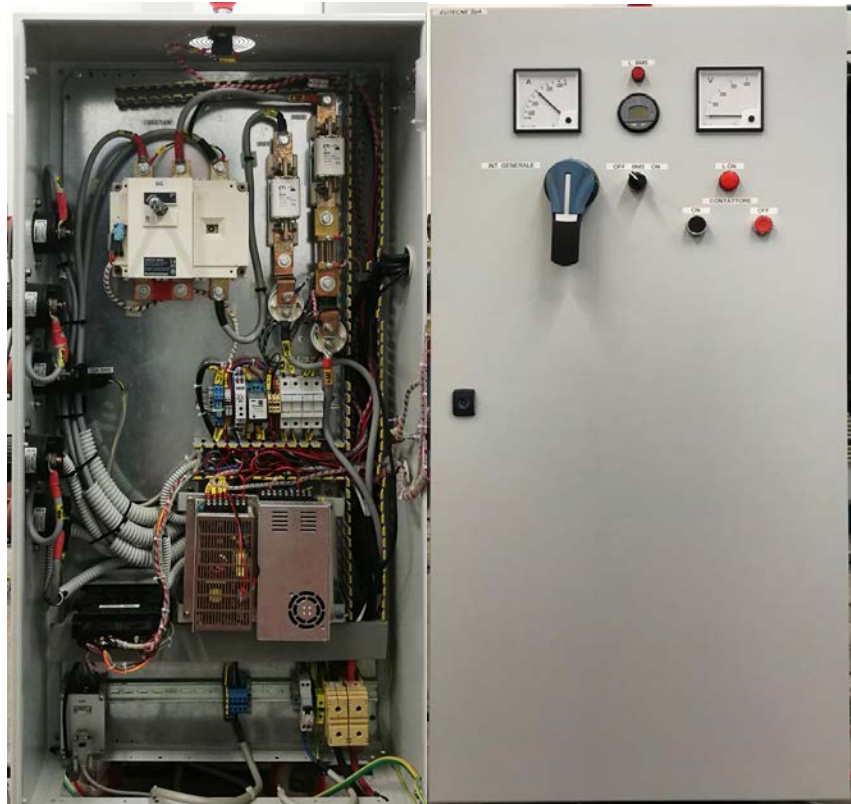


Fig. 2.2.20. Immagini interne ed esterne del quadro elettrico per la gestione ed interconnessione del sistema di accumulo stazionario.

L'armadio è stato dotato di ruote con elevata capacità di carico, per poterne favorire lo spostamento durante le prove di laboratorio. Un'immagine completa del sistema di accumulo stazionario allestito presso il CNR-Istituto Motori è mostrata in Fig. 2.2.21.



Fig. 2.2.21. Sistema di Accumulo Stazionario allestito presso il CNR – Istituto Motori

E' stata infine prevista la possibilità di poter ricaricare il pacco batteria direttamente dalla rete AC in bassa tensione mediante l'impiego di un caricabatteria ZIVAN. Tale caricabatteria è dotato di un modulo di comunicazione CAN, ed è stato opportunamente configurato per l'interfacciamento con il BMS durante le operazioni di ricarica.

Allestimento del sistema fotovoltaico

Nell'ambito dell'attività 2.2 proposta per il POA 2015 e per il proseguo delle attività nel POA 2016-2017, è stato dato particolare rilievo all'integrazione del dimostratore di micro-grid con sistemi di generazione dell'energia elettrica da fonti rinnovabili. Al fine di supportare con evidenze sperimentali tali attività, è stato allestito presso il CNR - Istituto Motori un sistema fotovoltaico da circa 7.5 kWp, composto da 4 stringhe di 8 moduli fotovoltaici.

Per l'allestimento di tale impianto sono stati impiegati moduli fotovoltaici basati su tecnologia ibrida ad etero-giunzione[11], che allo stato attuale della tecnologia consente di ottenere un'efficienza di conversione più elevata rispetto ad altre soluzioni reperibili sul mercato. La tecnologia ad etero-giunzione è considerata di tipo ibrido, in quanto si basa sull'impiego combinato di tecnologie di prima (silicio mono o policristallino) e seconda generazione (silicio amorfo). In particolare, uno strato di film ultra-sottile di silicio amorfo è depositato tra le intercapedini delle celle di un modulo di silicio mono o policristallino, al fine di massimizzare la superficie attiva esposta alla radiazione solare [12]. Ad oggi tale tecnologia fotovoltaica è commercializzata da Panasonic (ex Sanyo) ed è identificata con l'acronimo HIT (Heterojunction with Intrinsic Thin layer). A riguardo, in Fig. 2.2.22 si riporta schematicamente la composizione di una cella HIT.

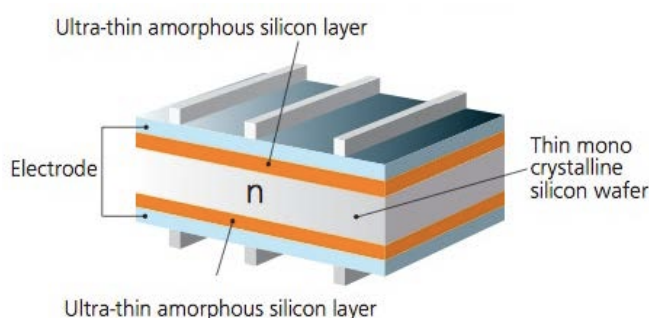


Fig. 2.2.22. Schema di una cella fotovoltaica HIT.

Per tale tecnologia i valori di efficienza di conversione di singola cella e dell'intero pannello fotovoltaico si attestano rispettivamente al 22 % ed al 19.4 % [7][13]. I moduli HIT risultano inoltre caratterizzati da una densità di potenza di circa 180 W/m² che rappresenta il valore più elevato rispetto alle altre soluzioni presenti sul mercato.

Le principali caratteristiche elettriche dei moduli scelti sono riportate in Tabella 4 relativamente alle condizioni standard di test (STC).

Tabella 7. Principali caratteristiche, valutate in STC, dei pannelli Panasonic HIT N240.

Pannello Fotovoltaico	Panasonic HIT N240
Potenza Nominale (P_{MPP}) [W]	240
Tensione alla potenza nominale (V_{MPP}) [V]	43.7
Corrente alla potenza nominale (I_{MPP}) [A]	5.51
Tensione di circuito aperto (V_{OC}) [V]	52.4

Corrente di corto circuito (I_{sc}) [A]	5.86
Massima tensione di funzionamento [V_{MAX}]	1000

Per le valutazioni preliminari alla fase di allestimento, è stata considerata una potenza elettrica di riferimento da circa 7.5 kWp, che corrisponde, per i moduli fotovoltaici considerati, ad una superficie totale occupata di circa 50 m². In base a tali requisiti è stato identificato il sito di installazione riportato in Fig. 2.2.23.

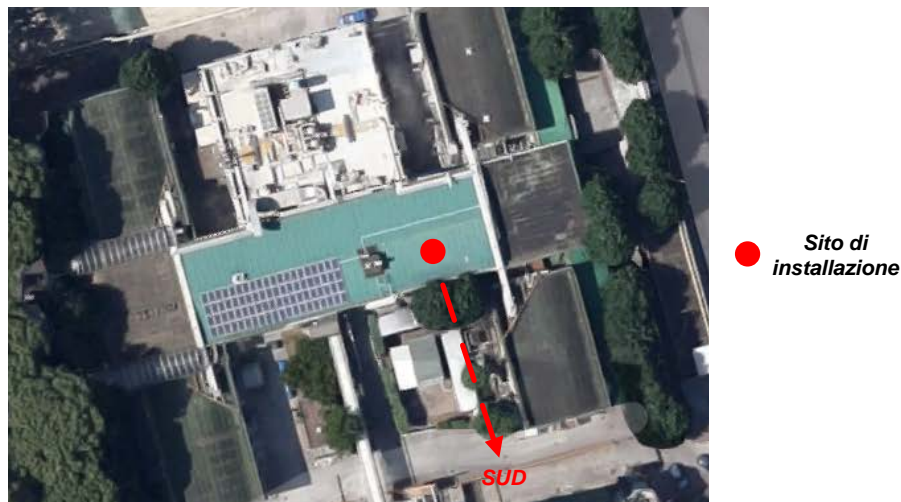


Fig. 2.2.23. Sito di installazione dell'impianto fotovoltaico.

L'area di installazione identificata per l'impianto è adiacente ad un impianto preesistente realizzato da 20 kWp. Le valutazioni energetiche preliminari sono state realizzate mediante i dati storici di radiazione solare per la città di Napoli con riferimento ad i parametri riportati in Tabella 8.

Tabella 8. Parametri utilizzati per le analisi energetiche dell'impianto fotovoltaico.

Località:	Napoli
Latitudine:	040°51'00"
Longitudine:	014°15'00"
Altitudine:	17 m
Albedo:	60 % Superfici chiare di edifici

Sulla base delle precedenti considerazioni sono state realizzate, mediante l'utilizzo del tool Socomec PV-CAD 3.0, diverse simulazioni finalizzate alla valutazione dell'energia annua prodotta dall'impianto in funzione dell'angolo di tilt dei pannelli. I risultati delle simulazioni sono riportati in Fig. 2.2.24.

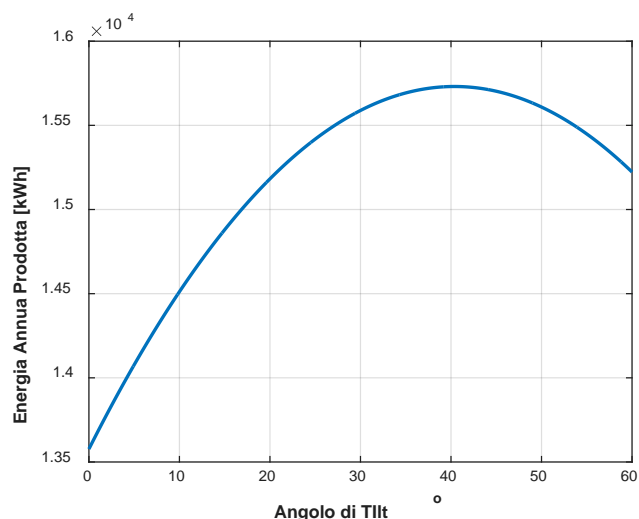


Fig. 2.2.24. Energia annua prodotta dall'impianto fotovoltaico in funzione dell'angolo di tilt.

In particolare, da tali valutazioni risulta che l'energia elettrica annua prodotta dall'impianto raggiunge il suo valore massimo, pari a circa 15740 kWh, per un angolo di tilt pari a circa 40°. Tale valore è superiore di soli 560 kWh annui rispetto a quanto si riesce ad ottenere con un angolo di tilt pari a 20°, con un miglioramento complessivo del 3%. Al contrario si evidenzia un miglioramento del 15% rispetto ad una installazione in piano con angolo di tilt pari a 0°.

In seguito alle suddette valutazioni energetiche, l'allestimento dell'impianto fotovoltaico in Istituto Motori è stato realizzato in conformità a quanto previsto dalla circolare "Guida per l'installazione degli Impianti Fotovoltaici" Prot. n. 1324 - 07/02/2012 del Dipartimento dei Vigili del Fuoco del Soccorso Pubblico e della Difesa Civile. In particolare, nel rispetto delle normative anti-incendio previste da tale circolare, il sito di installazione dell'impianto fotovoltaico è stato ricoperto con lastre Unipan in cemento alleggerito fibro-rinforzato. Le lastre Unipan sono classificate, ai fini della reazione al fuoco, in classe A1 –incombustibili, secondo D.M. 10/03/2005 e EN 13501-1. Le lastre Unipan spessore 16 mm posate in doppio strato forniscono uno strato di materiale di resistenza al fuoco certificato come EI 30.

Il campo fotovoltaico allestito presso i laboratori del CNR Istituto Motori è stato quindi realizzato mediante la connessione in parallelo di quattro stringhe, composte da 8 moduli Panasonic HIT. Le caratteristiche tecniche dell'impianto fotovoltaico realizzato in Istituto Motori sono riportate in Tabella 9.

Tabella 9. Caratteristiche tecniche dell'impianto fotovoltaico allestito al CNR – Istituto Motori

<i>PV Module</i>	<i>Panasonic HIT N240</i>
Number of PV Panels per String	8
Number of Strings	4
Open Circuit Voltage [V]	420
MPP Voltage [V]	350
MPP Power [kW]	7.68

Il piano di installazione dei moduli fotovoltaici è inclinato rispetto all'orizzontale di 10° ed ha un orientamento azimutale di 0° rispetto al sud. I moduli sono stati disposti su supporti in acciaio zincato regolabili, che offrono la possibilità di modificare manualmente l'angolo di tilt da 10° fino a 40°. Tale funzionalità consente di analizzare le prestazioni delle singole stringhe, per diverse inclinazioni, al

variare delle condizioni climatiche. A riguardo, in Fig. 2.2.25 si riporta una vista laterale di tali supporti regolabili.

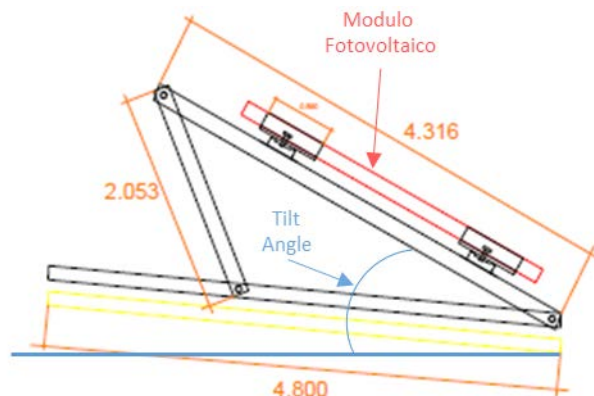


Fig. 2.2.25. Vista laterale dei supporti regolabili per la variazione dell'angolo di tilt.

Gli ancoraggi della struttura sono stati praticati avendo cura di ripristinare la tenuta stagna dell'attuale copertura e con proprietà di resistenza al vento conformi al DM 14/01/08. In Fig. 2.2.26 si riporta un'immagine dell'impianto fotovoltaico allestito presso il CNR – Istituto Motori.



Fig. 2.2.26. Impianto Fotovoltaico allestito presso il CNR – Istituto Motori

Nell'ambito del POA 2015 è stata inoltre completato il cablaggio e l'allestimento dei sistemi di protezione per il collegamento dell'impianto fotovoltaico al dimostratore di micro-grid, disposto all'interno dei laboratori dell'Istituto Motori.

Conclusioni

In tale documento sono stati descritti i principali risultati ottenuti per le attività svolte nella Linea Progettuale 2 - 2.2. "Sviluppo di un dimostratore di smart micro-grid, basato su un'architettura in corrente continua che integra fonti rinnovabili - sistemi di accumulo stazionari e carichi energivori" nell'ambito del POA 2015.

I risultati presentati fanno riferimento sia ad attività di dimensionamento preliminari, svolte in ambiente di simulazione, sia ad attività sperimentali, svolte nei laboratori del CNR – Istituto Motori di Napoli.

In particolare, le attività di simulazione sono state implementate mediante lo sviluppo di modelli basati sull'approccio *Equivalent Average Circuit Model*. Tale approccio è caratterizzato da un buon rapporto accuratezza/complessità computazionale ed ha consentito di ottenere utili informazioni sul dimensionamento, controllo e gestione dei sistemi di conversione che compongono il dimostratore, nelle diverse condizioni operative. Sono stati inoltre implementati modelli di simulazione per i principali attori energetici che interagiscono con la micro-grid, tra cui: carichi in corrente alternata/continua, sistemi di accumulo stazionari, sistemi di generazione dell'energia elettrica da fonti rinnovabili.

Sulla base dei risultati ottenuti in ambiente di simulazione, le attività sperimentali descritte in questo documento sono state rivolte all'allestimento di un dimostratore di micro-grid in corrente continua, rappresentativo di uno smart district urbano. In particolare, tali attività hanno riguardato la configurazione dei singoli convertitori, e dei relativi sistemi di acquisizione e controllo che consentono di gestire i flussi energetici all'interno della micro-grid realizzata. L'attività di consulenza da parte di UNIBO-DEI è stata utile ad evidenziare l'importanza dell'ottimizzazione delle tecniche di controllo del grid-tied converter, in termini di power quality e di impatto sul dimensionamento dei filtri in ingresso a tale convertitore. E' stato poi dimensionato ed allestito un sistema di accumulo stazionario basato su celle LiFePO₄, con gli opportuni sistemi di monitoraggio, controllo e protezione, da essere integrato con la micro-grid a supporto dell'impiego di fonti di energia rinnovabile. E' infine stata svolta una attività di allestimento e configurazione di un impianto fotovoltaico da 7.5 kWp composto da moduli fotovoltaici basati su tecnologia HIT.

I risultati ottenuti nell'ambito del POA 2015, forniscono le basi per attività di approfondimento, che saranno svolte nel PAR 2016-2017, e che riguarderanno la caratterizzazione sperimentale del dimostratore e l'analisi di strategie di gestione ottimali dei flussi energetici in contesti smart district. A riguardo, per l'interazione microgrid-rete AC saranno analizzati sperimentalmente i principali parametri di power quality e, grazie al contributo della consulenza di UNIBO-DEI, saranno valutate le potenzialità di soluzioni innovative basate su convertitori con componenti Silicon-Carbide o di tipo wide bandgap.

Riferimenti Bibliografici

- [1] Singh, Bhim, et al. "A review of three-phase improved power quality AC-DC converters." *IEEE Transactions on industrial electronics* 51.3 (2004): 641-660.
- [2] Espinoza, Jose R., and Geza Joos. "State variable decoupling and power flow control in PWM current-source rectifiers." *IEEE transactions on industrial electronics* 45.1 (1998): 78-87.
- [3] Milosevic, Mirjana. "Decoupling control of d and q current components in three-phase voltage source inverter." *Power Systems Conference and Exposition (PSCE), Atlanta, USA, October 2006*. 2003.
- [4] Wang, Yi-Feng, et al. "Interleaved high-conversion-ratio bidirectional dc-dc converter for distributed energy-storage systems—circuit generation, analysis, and design." *IEEE Transactions on Power Electronics* 31.8 (2016): 5547-5561.
- [5] Dwari, Suman, and Leila Parsa. "An efficient high-step-up interleaved DC-DC converter with a common active clamp." *IEEE Transactions on Power Electronics* 26.1 (2011): 66-78.

- [6] V. Vorperian, "Simplified analysis of PWM converters using model of PWM switch. Continuous conduction mode," in IEEE Transactions on Aerospace and Electronic Systems, vol. 26, no. 3, pp. 490-496, May 1990. doi: 10.1109/7.106126.
- [7] <ftp://ftp.panasonic.com/solar/specsheet/n240-spec-sheet.pdf>.
- [8] https://www.lem.com/sites/default/files/products_datasheets/lv_100_-_1000_sp9.pdf.
- [9] <http://www.europowercomponents.com/media/uploads/la125-p.pdf>.
- [10] K. Padhi, K. Nanjundaswamy e J. Goodenough, «Phospho-olivines as positive-electrode materials for rechargeable lithium batteries,» Journal of the electrochemical society, vol. 144, n. 4, pp. 1188-1194, 1997.
- [11] Olson, Jerry M. "Heterojunction solar cell." U.S. Patent No. 5,342,453. 30 Aug. 1994.
- [12] V. Tyagi, N. A. Rahim, N. Rahim e J. A. Selvaraj, «Progress in solar PV technology: Research and achievement,» Renewable and Sustainable Energy Reviews, vol. 20, n. 0, pp. 443-461, 2013.
- [13] S. De Wolf, A. Descoedres, Z. C. Holman e C. Ballif, «High-efficiency silicon heterojunction solar cells: A review,» Green, vol. 2, n. 1, pp. 7-24, 2012.

Confronto con i risultati attesi ed eventuali scostamenti rilevati

Per tale attività non si riscontrano particolari scostamenti con i risultati attesi.

Infatti, l'obiettivo principale dell'attività 2.2. è stato pienamente raggiunto mediante il dimensionamento, l'allestimento e la configurazione del dimostratore di laboratorio di micro grid in grado di rappresentare in scala i principali flussi energetici all'interno di uno smart district. Il dimensionamento è avvenuto mediante opportune attività di simulazione in ambiente Matlab-Simulink, che hanno fornito informazioni preliminari sulle principali condizioni operative del dimostratore oggetto di tali attività. La micro-grid considerata interconnette un sistema fotovoltaico da 7.5 kWp, installato presso il CNR – Istituto Motori, ed un pacco batterie da 90 Ah – 288 V, che opera a supporto del dimostratore come sistema di accumulo stazionario. Il convertitore (*active front-end*) interposto tra micro-grid e rete è in grado di gestire flussi di potenza fino a circa 20 kW, ottimizzando l'impiego di sistemi di accumulo e fonti rinnovabili a supporto di utenze energivore che operano in corrente alternata o continua.

Deliverables

Unità di ricerca	DESCRIZIONE DELIVERABLE	RISULTATI QUANTITATIVI
CNR-IM	Report D_2.2: Sviluppo di un dimostratore di smart micro-grid a supporto di carichi energivori, interconnessa con fonti rinnovabili e sistemi di accumulo stazionari. (CNR-IM)	<ul style="list-style-type: none"> - Criteri di progettazione, attività di simulazione ed allestimento di un dimostratore di smart micro-grid in corrente continua - Allestimento e dimensionamento sistemi di accumulo stazionari e sistemi di generazione da fonte fotovoltaica

Diffusione dei risultati

Al fine di dare ampia diffusione ai risultati dell'attività di Ricerca di Sistema ottenuti dal CNR-IM nell'ambito del POA 2015 sono state effettuate iniziative di disseminazione, che vengono di seguito riportate.

- Partecipazione alla International Conference on Applied Energy (ICAE 2017) che si è tenuta a Cardiff dal 21 al 24 Agosto 2017.
- Partecipazione al 72° Congresso Nazionale ATI che si è tenuto a Lecce dal 6 all'8 settembre 2017.

In particolare, l'ICAE 2017 è la nona edizione dell'International Conference on Applied Energy. Il tema principale di tale conferenza per il 2017 è stato "Energizing the Future". Per la conferenza sono state organizzate sessioni plenarie, workshop e special session sui seguenti topic: Clean Energy Conversion Technologies, Energy Management, Policy, Economics and Sustainability, Energy Sciences, Energy Storage, Intelligent Energy Systems, Mitigation Technologies, Renewable Energy.

Il 72° Congresso Nazionale dell'Associazione Termotecnica Italiana (ATI) punta a promuovere la produzione ed il trasferimento delle conoscenze nel campo della conversione dell'energia tra le Università, i centri di ricerca ed il mondo produttivo e della pubblica amministrazione.

Ulteriori attività di diffusione dei risultati sono state svolte mediante la presentazione delle attività di progetto in occasione della partecipazione alla manifestazione "Futuro Remoto 2017", che si è tenuta a Napoli a maggio 2017.

Azioni di Networking intese a creare, promuovere e supportare network di ricercatori ed imprenditori del settore con lo scopo principale di riunire risorse e competenze e creare connessioni tra le parti coinvolte, con obiettivi a medio e lungo termine, per promuovere ricerche di comune interesse ma anche per sviluppare progetti/prodotti/processi favorendo la sinergia tra gli attori coinvolti su attività di ricerca comuni che soddisfano le più ampie esigenze di mercato.

Apparecchiature disponibili e acquisite

Per lo svolgimento dell'attività 2.2 sono state utilizzate le seguenti attrezzature disponibili presso il CNR – Istituto Motori: schede di acquisizione e controllo National Instruments/dSpace; oscilloscopi ed alimentatori Tecktronix; software di simulazione Matlab/Simulink. E' stata inoltre acquisita una licenza del software Sypt Pro, fornito da Nidec Industrial Automation Italy S.p.a., per il controllo di sistemi di conversione dell'energia elettrica. Non sono state invece acquisite nuove attrezzature HW da laboratorio per lo svolgimento delle attività del progetto.

Linea 2: Benefici per gli utenti finali

Le attività di ricerca proposte nella Linea 2, nel loro complesso, contribuiscono a favorire gli utenti ed il gestore del sistema elettrico in termini di stabilità, power quality, risparmio energetico e sicurezza dell'infrastruttura di distretti energetici, garantendo un impiego ottimale dei sistemi distribuiti di produzione ed accumulo dell'energia elettrica, migliorando l'efficienza di conversione energetica e la qualità/affidabilità del servizio fornito. In generale, quindi, le attività proposte contribuiscono a mettere a disposizione dei beneficiari know-how tecnico scientifico di supporto per la realizzazione di politiche energetiche e la valutazione preliminare delle reali prestazioni energetiche di apparati e sistemi che compongono uno smart district. I beneficiari di tali attività sono pertanto tutti i principali attori del distretto energetico, dalla mobilità elettrica fino ad utenze aziendali ed alle singole utenze domestiche in edifici intelligenti ad alta efficienza energetica.