



UNIVERSITÀ DI FIRENZE

## Precision Agriculture

Student:

**Gennai Gian Maria**

---

ANNO ACCADEMICO 2023/2024

# 1 Introduzione

La *Precision Agriculture* è un approccio più tecnologico rispetto alla classica agricoltura, con un sistema di monitoraggio e automatizzazione all'interno di un terreno agricolo e con l'aiuto di dispositivi **IoT** all'interno di una rete a supporto per lo scambio ed elaborazione dei dati.

La rete che si crea, va ad utilizzare tecnologie differenti a seconda dell'utilizzo finale. Vedremo prima una spiegazione più approfondita di tutte le parti che mi andranno a comprendere la rete e dopo andremo a modellarla su un particolare caso specifico. Tutti i riferimenti che troviamo all'interno del testo, se non è presente il link alla fonte in locale, è presente nella sezione finale dedicata.

## 1.1 Physical Layer

Al livello più basso della struttura dello stack protocollare di comunicazione, responsabile della trasmissione e della ricezione del segnale grezzo fra i dispositivi, possiamo trovare i sensori ed attuatori.

### 1.1.1 Sensori

I sensori all'interno della rete sono principalmente i seguenti:

- Temperatura: Per misurare la temperatura dell'ambiente circostante alla pianta in modo da poterla mantenerla sempre all'interno del suo range vitale.
- Umidità: Misura dell'umidità dell'aria che oltre a stare dentro i range per il comfort della pianta permette, insieme ad altri sensori, di prendere decisioni sulle varie sessioni d'irrigazione
- Pressione: Misura la pressione dell'aria
- Umidità del suolo: Misurazione dell'umidità e permeabilità del suolo
- Camera: Permette di catturare immagini della pianta in modo da trarne informazioni visive

I sensori hanno differenti tipi di connessioni, alcuni hanno bisogno di essere elaborati prima di essere mandati al *Cloud* e confrontati con i livelli di riferimento. Un esempio di quello appena detto possono essere i sensori per l'umidità del terreno, i quali solo inseriti a differenti profondità nel terreno e di conseguenza sarebbe uno spreco di risorse mandare le informazioni di tutti i sensori singolarmente senza prima aggregarne alcuni sul posto.

Per avere informazioni ulteriori ho trovato un sensore al quale mi posso affidare per fare misurazioni dell'umidità del suolo e trarne le relative conclusioni dal sito ufficiale del dispositivo. Da questo dispositivo io posso andare a ricavare informazioni riguardanti umidità, temperatura e salinità del terreno a profondità che vanno fino

a 2/6 metri. Ogni informazioni che viene ricavata è inviata, senza l'ausilio di alcuna scheda sim, ad una *Base Station*(BS) oppure, nel caso i sensori siano molto lontani dalla base station, possono essere inseriti dei ripetitori che permettono l'invio delle informazioni alla BS.

Seguendo la logica appena riportata sopra, i sensori devono però essere inseriti abbastanza in profondità da non essere portati via da eventuali macchinari agricoli. Per quanto riguarda la durata della batteria è di circa 20 anni e di conseguenza non devono essere rimossi dal terreno tutte le volte.

Possiamo implementare tecniche di *beamforming*(tecnica che usa antenne multiple per mandare il segnale verso una specifica direzione), in modo da poter consumare meno energia, direzionando i ripetitori direttamente verso la BS.

### 1.1.2 Attuatori

Gli attuatori che possiamo trovare sono di tanti tipi e tutto a secondo di quello che vogliamo automatizzare, ma nel contesto agricolo e in particolare di coltivazioni fisse di grandi dimensioni come possono essere Vigneti o Uliveti, l'utilizzo di strutture fisse è molto scomodo, al contrario possiamo andare ad utilizzare droni per alleggerire il peso delle strutture fisiche.

Tramite i droni si possono gestire anche l'immissione di alcuni prodotti sulle piante(ad esempio il ramato per le viti), tenendosi aggiornati grazie ai feedback che arrivano direttamente dalle webcam posizionate lungo i filari.

### 1.1.3 Droni

L'utilizzo dei droni può essere molto utile in questo ambiente e merita un approfondimento.

L'utilizzo del drone può essere molteplici e quindi non solo per quanto riguarda magari la parte di irrigazione o immissione di trattamenti sulle piante, ma anche per quanto riguarda il prelievo delle informazioni direttamente tramite una camera attaccata al drone. Tramite un eventuale camera io posso riuscire a rilevare lo stato di salute della pianta e anche riuscire a capire se ha bisogno o meno di determinati prodotti.

La possibilità di sviluppo di una rete con l'aiuto di un drone si affaccia a due possibili prospettive, andare ad inglobare una rete neurale all'interno del drone o meno. Vanno considerati molti aspetti tra cui:

- Dimensioni
- Peso del drone
- Peso Massimo Al Decollo
- Videocamere

- Licenza di guida necessaria
- Autonomia di volo
- Costo

Per avere queste stime mi sono basato su una tesi di laurea del Politecnico di Torino che è inserita nella sezione Link.

L'autonomia di volo di droni che arrivano fino a circa *1500€* è tra i 25 e 50 minuti in caso di volo senza vento, cosa da considerare in casi di spazi aperti e che aumentando l'attrito porta ad un consumo più rapido della batteria. Esistono già comunque dei droni che forniscono la possibilità di programmare il percorso di volo e loro riescono a catturare tutta una serie di informazioni tramite la fotocamera che poi sono elaborate e fornite all'operatore. Ovviamente le operazioni che sono fatte su queste strutture non devono essere troppo onerose in modo da non dover consumare ulteriore energia causata dal dover montare un hardware troppo complesso.

C'è anche la possibilità di avere alcuni droni che hanno come unico scopo agire sulle varie piantagioni in base alle informazioni che ricevono ed altri fare dei voli di ricognizione per andare a raccogliere i dati.

## 2 Network Layer

Gli indirizzi sono davvero necessari rispetto alla rete che andiamo a costruire? Possiamo pensare a differenti utilizzi per cui andare ad utilizzare indirizzi all'interno di questa struttura.

Andando ad inserire sensori sparsi per il terreno, avere un indirizzo che mi indica ogni singolo cluster di sensori sarebbe un inutile spreco di indirizzi. Potremmo però anche avere più terreni con differenti coltivazioni e grazie ad un indirizzo io potrei distinguere da dove arriva l'informazione e gestirla in modo differente per ogni piantagione; sapendo quali e dove sono i sensori potrei decidere di non prelevare informazioni o di mandarli in sleep nei periodi in cui non è necessario il loro utilizzo. Andremo ad utilizzare indirizzi IPv6.

Sopra ho parlato di cluster di sensori perché non ci saranno strutture con un singolo sensore che mi fa una unica misurazione ma dispositivi che riescono a controllare più parametri o comunque più sensori messi insieme in un unico punto per andare poi ad elaborare tutte le informazioni di quell'area.

## 3 Transport Layer

UDP, o User Datagram Protocol, è un protocollo di comunicazione leggero e senza connessione. A differenza del TCP (Transmission Control Protocol), UDP non

stabilisce una connessione prima di trasmettere dati e non garantisce la consegna ordinata e affidabile dei pacchetti. Tuttavia, questa caratteristica lo rende veloce, il che lo rende una scelta adatta per determinati scenari di comunicazione.

Nel contesto dell'IoT, l'uso di UDP è comune per diverse ragioni. La natura senza connessione di UDP si adatta bene a scenari in cui la velocità e la tempestività della trasmissione dei dati sono cruciali. Ad esempio, sensori IoT possono inviare rapidamente dati di lettura a una stazione di monitoraggio senza l'onere delle procedure di connessione tipiche di altri protocolli. L'utilizzo di UDP nell'IoT offre vantaggi in termini di efficienza e riduzione della latenza. Tuttavia, poiché non fornisce garanzie sulla consegna dei pacchetti, potrebbero verificarsi perdite di dati in caso di congestione di rete o altre condizioni avverse.

## 3.1 Application Layer

L'architettura che mi permette il funzionamento della rete all'interno della rete agricola utilizza sia il protocollo **CoAP** (Constrained Application Protocol) che **MQTT-NS** (Message Queuing Telemetry Transport-Network Sensor).

### 3.1.1 CoAP

Il CoAP, acronimo di Constrained Application Protocol, è un protocollo di comunicazione progettato per l'Internet delle Cose (IoT) e per dispositivi con risorse limitate, come sensori, attuatori e nodi di rete. Si basa sul modello di comunicazione RESTful (Representational State Transfer), simile al protocollo HTTP utilizzato nel web, ma è ottimizzato per l'ambiente delle reti a basso consumo energetico e a bassa larghezza di banda.

Le caratteristiche principali del CoAP includono:

- **Leggerezza:** Il CoAP è progettato per essere leggero e efficiente, con un overhead di trasmissione dei dati ridotto rispetto ad altri protocolli più complessi. Questo lo rende adatto per dispositivi con risorse limitate, come sensori alimentati a batteria.
- **Affidabilità:** Pur essendo un protocollo UDP (User Datagram Protocol)-basato, il CoAP fornisce opzioni per la trasmissione affidabile dei dati, consentendo la ritrasmissione dei pacchetti persi o danneggiati. Questo è particolarmente importante in scenari in cui è necessaria una consegna garantita dei dati, come nel controllo remoto di dispositivi critici.
- **Supporto per la Multicast:** Il CoAP supporta la comunicazione multicast, consentendo a un singolo messaggio di essere inviato a più destinazioni contemporaneamente. Questo è utile per aggiornamenti di gruppo o notifiche broadcast all'interno di una rete IoT.

- Interoperabilità: Il CoAP è progettato per interoperare facilmente con altri protocolli web e servizi basati su HTTP. Ciò consente ai dispositivi CoAP di comunicare con server web e servizi cloud utilizzando gli stessi meccanismi utilizzati nel web tradizionale.

### 3.1.2 MQTT/MQTT-NS

MQTT è un protocollo di messaggistica leggero e efficiente progettato per la comunicazione tra dispositivi IoT (Internet of Things). È basato sul modello *publish-subscribe*, dove i dispositivi pubblicano (inviano) messaggi su specifici argomenti e altri dispositivi si iscrivono (sottoscrivono) a questi argomenti per ricevere i messaggi. MQTT è noto per il suo basso overhead di trasmissione e il consumo di risorse ridotto, rendendolo adatto per la comunicazione in reti a banda limitata e con dispositivi con risorse limitate.

MQTT-NS è una variante del protocollo MQTT ottimizzata per reti con connettività via satellite. È progettato per gestire le sfide uniche associate alle comunicazioni satellitari, come la latenza elevata e la connettività intermittente. MQTT-NS incorpora specifiche modifiche e ottimizzazioni per garantire una trasmissione affidabile dei messaggi anche in condizioni di rete difficili. Questo protocollo consente la comunicazione efficiente tra dispositivi IoT e reti satellitari, consentendo la trasmissione di dati in aree remote o inaccessibili tramite reti terrestri convenzionali.

## 4 Connessione con rete 5G

Introduciamo l'utilizzo della tecnologia **5G** in modo da poter sfruttare tutti i vantaggi che ne derivano, come:

- Più dispositivi connessi rispetto al 4G
- M2M a basso costo
- Risparmio di batteria per i dispositivi con limitazioni
- Traffico maggiore rispetto al 4G
- Molto affidabile
- Bassa latenza

Per l'accesso alla rete 5G, visto che viene fornito da un privato, andrà messo in conto un costo fisso in termini di denaro.

La rete 5G è fondata da molti componenti tra i quali:

- User Equipment: dispositivi degli utenti finali.

- gNB: (Next-Generation NodeB) che è l'equivalente di una stazione radio del 5G.
- RAN: (Radio Access Network) è la rete di accesso radio vera e propria e comprende anche i gNB. Qui abbiamo le antenne vere e proprie.
- NG-RAN: è una componente importante del 5G ed è dove si trovano la Centralized Unit e la Distributed Unit.
- Core Network: è la parte centrale della rete e serve per l'elaborazione, il trasferimento e il controllo dei dati. Include UPF (User Plane Function) per la trasmissione dati, il SMF (Session Management Function) per la gestione delle sessioni, e altri componenti come l'AMF (Access and Mobility Management Function) e l'UDM (Unified Data Management).

NFV (Network Functions Virtualization) e SDN (Software-Defined Networking) sono due concetti chiave nel campo delle reti di comunicazione. NFV è una tecnologia che punta a virtualizzare le tradizionali funzioni di rete, eseguite precedentemente su hardware dedicato. Con NFV, queste funzioni possono essere implementate come software eseguito su infrastrutture di server standard, portando vantaggi come flessibilità nell'allocazione delle risorse e velocità di creazione dei servizi di rete.

SDN, d'altra parte, è un paradigma che separa il piano di controllo dal piano di dati nelle reti di comunicazione. In un'architettura SDN, il controllo della rete è centralizzato e programmabile attraverso software, consentendo una gestione più dinamica e flessibile della rete in tempo reale.

Entrambe le tecnologie rivoluzionano il modo in cui progettiamo e gestiamo le reti di comunicazione. NFV trova applicazioni in vari contesti, come i provider di servizi di telecomunicazioni, semplificando la gestione delle reti e consentendo una distribuzione più dinamica delle risorse. SDN, con le sue applicazioni diversificate, migliora la gestione del traffico, ottimizza le risorse di rete e semplifica l'implementazione di nuovi servizi, ad esempio nei data center per ottimizzare le prestazioni della rete e la distribuzione delle risorse.

L'adozione di queste tecnologie apre la strada a reti più flessibili, efficienti e scalabili, portando benefici significativi sia per i fornitori di servizi che per gli utenti finali.

Con l'utilizzo di queste tecnologie ho anche l'introduzione del *Network Slicing* il quale mi permette di creare **reti virtuali ottime** per ogni applicazione. Passiamo da un approccio dove una rete si adatta a tutti i servizi a una struttura che si modifica dinamicamente dove reti logiche multiple(slice) create sopra una struttura fisica fissa per incontrare differenti tipi di servizi da fornire all'utente. Abbiamo più reti *end-to-end* create grazie al software in una struttura distribuita e dinamicamente

controllate da un orchestratore.

La struttura a slice per il Network Slicing è composta da 3 livelli differenti:

- Infrastructure Layer: che ha lo scopo di provvedere alle risorse fisiche o virtuali.
- Instance Layer: ha lo scopo di creare ogni slice di rete in base alle richieste d'istanza di servizio fornite dal livello superiore.
- Service Layer: gira al di sopra di tutti gli altri livelli ed è quello che fornisce l'interfaccia per l'utente finale.

La rete 5G utilizza un'architettura basata su Cloud.

## 5 Flusso dei dati

Il flusso dei dati è un aspetto importante da visualizzare all'interno di questa struttura, abbiamo molte informazioni e alcuni devono avere la priorità rispetto ad altri. **Data-flow** (flusso dei dati) è definito come ogni trasmissione singola o multipla che avviene tra ognuno dei due relativi dataset considerati e invece indichiamo come **Data-stream** una sequenza di data-flow che hanno una o più risorse distinte di dati di destinazione che saranno direttamente raccolti o utilizzati in PA(Physical Activity). Cioè, un flusso di dati collega gli input riconosciuti e gli output desiderati per una particolare analisi o insieme di analisi in PA mentre un flusso di dati può rappresentare nient'altro che una fase di elaborazione intermedia.

Il flusso di dati può variare a seconda di quello che deve essere trasportato. Prendiamo per esempio il prelievo di informazioni su una determinata piantagione, dove in questo caso è più corretto prendere campioni dei dati ad ogni periodo  $T$ , il che vuol dire in modo *periodico* così da avere punti di riferimento ben stabiliti. Se al contrario andiamo invece a trasportare un flusso di informazioni relativo ai vari allarmi che ci possono essere, il flusso di dati non può essere periodico dato che, se arrivasse un segnale di allarme all'istante  $T+1$  io dovrei aspettare fino al nuovo istante  $T+T$  per ottenere la trasmissione del messaggio, ma avendo informazioni relative ad un allarme è più corretto che l'invio sia immediato.

In sintesi io avrò due distinti flussi di informazioni, un flusso periodico e uno aperiodico.

## 6 Architettura Lambda

L'architettura Lambda è un modello di progettazione per sistemi di elaborazione dati che combina l'elaborazione batch e l'elaborazione in tempo reale per gestire



grandi volumi di dati in modo efficiente. Prende il nome dalla forma della lettera greca lambda, che rappresenta una combinazione di un flusso batch e di un flusso in tempo reale.

Questo modello è particolarmente adatto per i sistemi che devono elaborare dati in tempo reale, ma che richiedono anche un'analisi approfondita e storica dei dati. L'architettura Lambda permette di ottenere risultati immediati e aggiornati in tempo reale, mentre contemporaneamente consente anche l'analisi di grandi quantità di dati storici.

Per far questo, l'architettura Lambda è composta da diversi strati o layer, ognuno dei quali svolge un ruolo specifico nell'elaborazione e nell'analisi dei dati. Questi layer includono il Batch Layer, lo Speed Layer e il Serving Layer, ognuno dei quali ha responsabilità distinte nell'elaborazione e nella presentazione dei dati (figura 1).

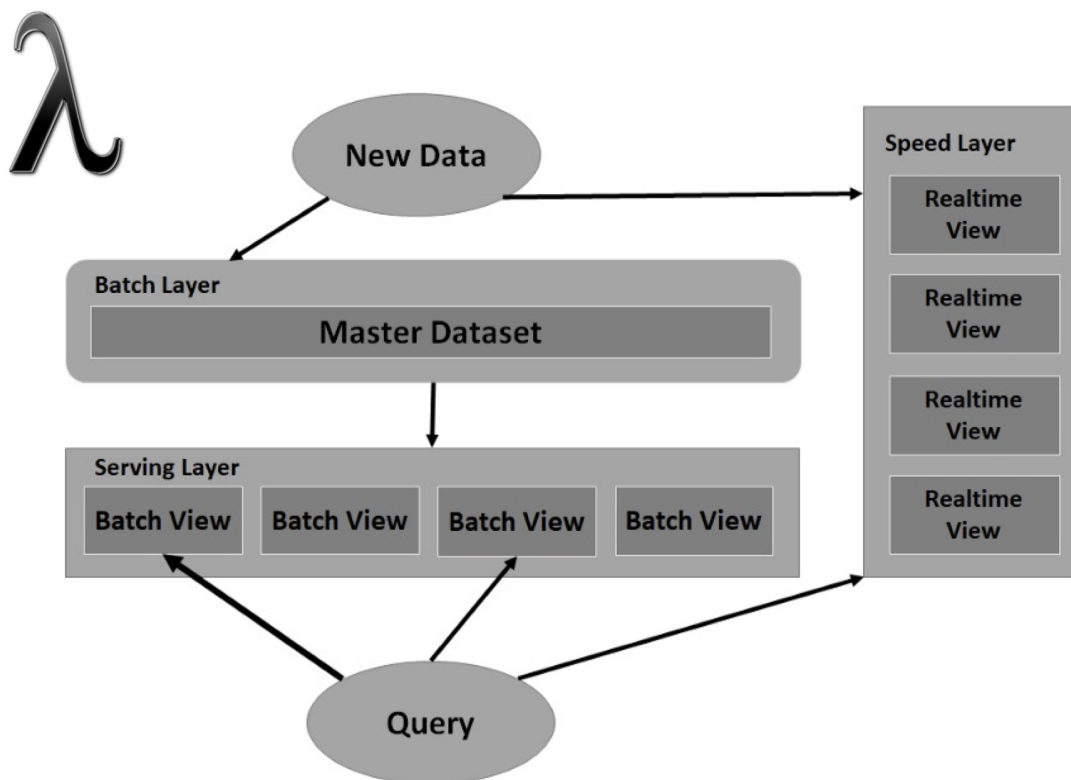


Figura 1: Architettura Lambda

- **Batch Layer (Strato Batch):** Il Batch Layer è responsabile dell'elaborazione dei dati in batch su dataset completi e storici. Utilizza sistemi di elaborazione distribuita come Apache Hadoop o Apache Spark per analizzare grandi quantità di dati in modo scalabile e affidabile. L'elaborazione batch avviene su dati storici e completi, il che consente di ottenere risultati accurati e completi nel tempo.

- **Speed Layer (Strato di Velocità):** Lo Speed Layer si occupa dell'elaborazione in tempo reale dei dati provenienti dalle fonti di streaming. Utilizza sistemi di elaborazione in tempo reale come Apache Storm o Apache Flink per analizzare i flussi di dati in ingresso e generare risultati immediati. Questo strato è in grado di elaborare dati in tempo reale e fornire risultati immediati per rispondere rapidamente a eventi e flussi di dati in continuo arrivo.
- **Serving Layer (Strato di Servizio):** Il Serving Layer combina i risultati generati dal Batch Layer e dallo Speed Layer per fornire risposte a query interattive e in tempo reale. Utilizza sistemi di archiviazione ad accesso rapido come Apache HBase o Apache Cassandra per memorizzare i risultati pre-elaborati e renderli disponibili per le query degli utenti. Questo strato offre un'interfaccia di interrogazione che permette agli utenti di ottenere informazioni in modo rapido ed efficiente, combinando sia i dati batch che quelli in tempo reale.
- **Batch Views (Viste Batch):** Le Batch Views sono le visualizzazioni pre-elaborate dei dati generate dal Batch Layer. Consentono di eseguire query complesse su grandi quantità di dati storici in modo efficiente e scalabile. Le Batch Views vengono aggiornate periodicamente in base ai nuovi dati che arrivano nel sistema.
- **Real-time Views (Viste in Tempo Reale):** Le Real-time Views sono le visualizzazioni dei dati generate dallo Speed Layer. Queste viste forniscono risultati immediati basati sui flussi di dati in ingresso e sono aggiornate in tempo reale. Le Real-time Views consentono di rispondere rapidamente a eventi in tempo reale e di fornire risultati aggiornati all'istante.

L'architettura Lambda è una struttura molto robusta, resistente ai guasti, ci permette di scalare la nostra struttura orizzontalmente, mi permette di utilizzare dati di diverse tipologie e non si limita all'utilizzo esclusivo di alcuni, posso aggiungere nuovo codice per implementare nuove strutture di *batch views*, la manutenzione da fare è minima e semplice, ho la possibilità di fare debugging delle queries e riesco ad avere bassa latenza in lettura e negli aggiornamenti.

## 7 Precision Agriculture in Viticoltura

Adesso andiamo ad analizzare e strutturare un caso vero e proprio, verranno rielaborate le informazioni fornite sopra.

Consideriamo un'area di **100 ettari** con una piantagione di viti da vino che giustifichi l'utilizzo di struttura di *Precision Agriculture* con rete 5G, in Italia.

Adesso passiamo a descrivere la rete.

## 7.1 Layer della rete IoT

### 7.1.1 Livello fisico

Nel livello fisico troviamo quindi sensori per il monitoraggio di piante e per il rilevamento di eventuali problematiche. Come abbiamo già descritto sopra ci saranno diversi tipi di sensori che andranno messi in posizioni e a distanze uno dall'altro differenti.

Consideriamo come prima cosa i sensori per l'umidità del terreno, questi vanno inseriti abbastanza in profondità da non essere portati via durante lavorazioni agricole e devono essere messi anche ad altezze differenti in modo da poter monitorare il tasso di assorbimento del terreno, anche se per le viti, le radici non vanno troppo in profondità, di fatti nei primi tre anni quelle principali arrivano a circa 30-35cm. La distanza che dobbiamo tenere da uno all'altro è ovviamente relativa alla precisione con cui noi vogliamo andare a monitorare il terreno ma metterli troppo vicini sarebbe comunque uno spreco di risorse e denaro.

Per i sensori di temperatura invece possiamo prendere distanze molto più ampie rispetto ai sensori di umidità. Ovviamente qui la differenza di temperatura varia pochissimo da una zona ad un'altra, dato che spesso nei campi di viti abbiamo terreni molto uniformi e tutti esposti alla stessa luce solare, almeno in centro Italia. L'unico motivo per cui uno potrebbe pensare di aumentare un po' il numero di sensori di temperatura è per rilevare incendi e intervenire il prima possibile dato che negli ultimi anni, a causa della siccità stanno aumentando, così che se vedo un sensore con temperatura molto più alta rispetto alla media degli altri vuol dire che ho un problema.

Abbiamo poi sensori per la pressione dell'aria e umidità dell'aria, che anche questi vanno messi a descrizione del proprietario. Nel caso della viticoltura però, questo è un tipo di sensore che non va a influenzare molto le informazioni della pianta ma magari potrebbe esserci una stazione per controllare il tempo in quella zona in modo da poter tenere i droni al riparo nel caso in cui ci fossero problemi.

Le telecamere possono invece essere utilizzate per rilevare malattie o parassiti sulla pianta o comunque rilevarne la salute generale. Qui, a secondo della scelta, possono essere fatti dei controlli a campione oppure dei controlli più precisi che ovviamente mi vanno ad alzare i costi. C'è anche la possibilità di utilizzare le telecamere integrate sui droni ma non tratteremo questo aspetto. Le telecamere fisse sul luogo però hanno il problema della pulizia che deve essere fatta ogni tot per garantire una buona qualità dell'immagine.

Passando invece agli attuatori che troviamo in questa rete, sono principalmente droni. Questi ci permettono di non dover creare strutture fisiche apposite e fisse così che una volta effettuato il lavoro, viene sgombrata la zona. La scelta del drone varia molto a secondo del peso che deve portare, ad esempio questo drone qui, può portare fino a 5kg di materiale, che possono essere semi o anche spruzzare pesticidi sulla pianta, ma nel caso dell'irrigazione è ben diverso perché per innaffiare un terreno di 100 ettari i litri di acqua da utilizzare sono molti e il drone dovrebbe sempre fare

avanti e indietro. Per far fronte a questo problema ci sarà una struttura di supporto mobile assieme al drone, che si muove tra i filari, con una cisterna d'acqua.

Per la comunicazione tra i sensori strutturiamo la cosa come segue. In un punto comodo da raggiungere fisicamente (per far sì di agevolare possibili azioni di manutenzione), andiamo a inserire quella che chiameremo *Base Station (BS)* o meglio una struttura che mi permette di prendere le informazioni provenienti dai sensori e poi rispedirle a chi di dovere. I sensori posso farli comunicare con un protocollo **Zigbee**, che grazie anche a dei ripetitori, posso far arrivare il segnale da tutti i sensori fino alla BS. Una volta raccolte le informazioni necessarie, queste verranno elaborate e/o inviate ad una struttura *Cloud*, con tecnologia cellulare 5G.

Utilizzeremo questa tecnologia perché, nell'esempio da noi preso e quindi in Italia, la copertura è abbastanza sviluppata e in più ci offre tutti i vantaggi già elencati nella sezione apposita. Le informazioni sono poi inviate da un router 5G (tipo [questo](#)) che a sua volta le riceve tramite un antenna per la rete Zigbee.

## 7.2 Livello di rete

Per questa struttura andremo ad utilizzare indirizzi **IPv6**.

## 7.3 Livello di trasporto

Avendo già analizzato le differenze tra il protocollo **UDP** e **TCP**, nella sezione dedicata, qui ci limiteremo ad osservare che per questo caso particolare andremo ad utilizzare UDP, in modo che si elimini la parte di set up, diminuendo la latenza. In più come vedremo nella seguente parte, utilizzeremo CoAP e MQTT-NS che entrambi utilizzano UDP.

## 7.4 Livello di applicazione

Come già annunciato prima avremo che, a questo livello utilizzeremo sia **CoAP** che **MQTT-NS** e questo perché entrambe offrono caratteristiche differenti rispetto al lavoro che vogliamo fare.

Quando vogliamo inviare i dati dei messaggi dai sensori verso la BS, soltanto per raccogliere informazioni relative alla piantagione, andiamo ad utilizzare CoAP che, in verità, funziona in modalità *Client-Server* e di conseguenza la BS magari ogni T secondi, invierà una richiesta di informazioni e queste ritorneranno dai sensori.

Per la segnalazione degli allarmi invece funziona in modo differente perché in questo caso ho un problema che deve essere risolto nell'immediato non posso aspettare troppo e di conseguenza preferisco utilizzare un modello *Publisher-Subscriber* così che ogni qual volta devo fare una segnalazione perché c'è stata una modifica nei dati, questa viene inviata e tutti coloro che sono interessati a riceverla.

## 7.5 Cloud

La soluzione Cloud è sicuramente la migliore ipotesi in caso si volesse affiancare un'applicazione, ad esempio per smartphone, che invia tutte le informazioni della piantagione e i messaggi di allerta.

L'utilizzo del Cloud mi permetterà di avere accesso ai dati in qualsiasi parte del mondo avente una connessione ad Internet ma, al contrario, il tempo di elaborazione dei messaggi è maggiore. Per questo la struttura generale dovrebbe essere di tipo lambda.

La struttura quindi sarà formata da un sistema hardware in locale, capace di elaborare in modo rapido e continuo le informazioni che arrivano dai sensori in modo da poter implementare la parte *Real Time* dell'architettura. Per fare questo possiamo molto semplicemente andare ad aggiungere la struttura direttamente nella BS dove arrivano i dati e questi una volta elaborati andranno ad essere inviati in modo rapido ai droni.

Per la parte di *Batch* invece i dati una volta arrivati alla BS, andranno inviati al Cloud ed elaborati.

Questa soluzione riesce ad escludermi il *Fog Computing* dato che riesce a fare operazioni rapide in locale e tutte le altre operazioni più onerose invece vengono fatte sul Cloud, con un ritardo maggiore ma che non crea nessun problema.

## 8 Conclusioni

Abbiamo visto la struttura di **Precision Agriculture** di un terreno con una piantagione di Viti da Vino di circa 100 ettari situato in Italia.

Come abbiamo potuto osservare sopra, dopo la spiegazione generale delle singole parti, la struttura finale è molto meno complessa, questo perché non è possibile far fronte a come sarà effettivamente un caso reale, dato che, una volta che il progetto viene portato avanti verranno tirate fuori sempre più problematiche relative spesso anche a fattori naturali.

Abbiamo quindi creato una prima idea per la gestione automatica di un terreno di viti da vino con tecnologia cellulare 5G e una serie di sensori per il monitoraggio delle piante e della zona con un sostegno meccanico fatto da droni. Tramite questa rete può essere completamente automatizzata l'irrigazione, la manutenzione delle piante tramite pesticidi e altri medicinali.

Sicuramente questa non è una struttura che copre tutto il lavoro che c'è da svolgere sul terreno coltivato ma, associata ad un'applicazione per smartphone può andare a segnalare informazioni su come, quando e dove, far fare determinate operazioni all'operatore che andrà direttamente a svolgere i suoi compiti in un tempo minore, in modo efficiente e tempestivo.

## 9 Link alla documentazioni

All'interno di questa sezione troveremo tutti i riferimenti su cui mi sono basato per svolgere questo progetto.

- Lightweight messaging protocol for precision agriculture
- 5G Network: Architecture, Protocols, Challenges and Opportunities
- Revolutionizing Agriculture with IoT –A Review
- CoAP vs. MQTT-SN: Comparison and Performance Evaluation in Publish-Subscribe Environments
- Wireless Sensor Network for Precision Agriculture
- Distributed pub/sub model in CoAP-based Internet-of-Things networks
- Hybrid MQTT-COAP Protocol for Data Communication in Internet of Things
- Wireless Communication Technologies for Internet of Things and Precision Agriculture: A Review
- Wireless Soil Moisture Sensor For Agriculture
- Development of a model of data-flows for precision agriculture based on a collaborative research project
- Classificazione automatica di piante sane o malate con uso di immagini
- Droni con fotocamera
- NSEDS: One Implementation of Lambda Architecture in Large-scale Seismic Exploration Scenes
- Guida per il viticoltore