

UNIVERSITA' DEGLI STUDI DI PAVIA
CENTRO DI RICERCA SULLE ACQUE

**UN EFFICIENTE CODICE DI VERIFICA E
DIMENSIONAMENTO DI RETI IDRICHE A MAGLIE**

Dott. Ing. Renato IANNELLI

Quaderni del Centro di Ricerca sulle Acque - N. 6 - Luglio 1996

QUADERNI DEL CENTRO DI RICERCA SULLE ACQUE

Serie precedente non numerata

- R. CELLERINO, E. GERELLI, G. IANNELLI, V. VENDEGNA - **Problemi economici derivanti dall'inquinamento: il caso del Mare Adriatico.** - Convegno Internazionale "Odissea Rifiuti: Proposte di riequilibrio", Venezia, 6 dicembre 1985.
- G. IANNELLI - **Alcune note sui trattamenti di depurazione delle acque usate** - Pavia, 1986.
- C. VELTRI, V. VENDEGNA, G. IANNELLI - **L'inquinamento batteriologico del Fiume Ticino in relazione all'utilizzo ricreativo delle sue sponde** - "Studio effettuato per conto del Consorzio Parco Lombardo della Valle del Ticino", Pavia, giugno 1987.

Serie ordinaria

- 1 - R. IANNELLI - **Modello matematico per gestione e controllo di un impianto a fanghi attivi in regime non permanente** - Pavia, aprile 1988.
- 2 - V. VENDEGNA - **Sviluppo e impiego di matrici per l'analisi e la valutazione negli studi di impatto ambientale** - Pavia, giugno 1988.
- 3 - R. IANNELLI - **Taratura e verifica sul depuratore di Fusina di un modello matematico non stazionario di impianto a fanghi attivi** - Pavia, luglio 1991
- 4 - G. BIOLZI - **I costi delle Opere di Acquedotto e Fognatura (Revisione ed attualizzazione dei Criteri del Piano di Risanamento delle Acque della Regione Lombardia)** - Pavia, febbraio 1995.
- 5 - A. BERZERO, M. BALDI - **Guida per la sicurezza del lavoro nei laboratori chimici** - Pavia, gennaio 1996.
- 6 - R. IANNELLI - **Un efficiente codice di verifica e dimensionamento di reti idriche a maglie** - Pavia, luglio 1996.

UN EFFICIENTE CODICE DI VERIFICA E DIMENSIONAMENTO DI RETI IDRICHE A MAGLIE

Renato Iannelli

Dipartimento di Idraulica, Trasporti e Infrastrutture civili - Politecnico di Torino

SOMMARIO

Vengono descritte le caratteristiche generali, i fondamenti teorici e le modalità di implementazione di un programma di calcolo per il dimensionamento e la verifica in moto permanente di reti di distribuzione idrica. La verifica in moto permanente della rete viene condotta con un particolare metodo basato sull'analisi topologica della rete e sull'individuazione della minima matrice ridotta, che viene poi risolta iterativamente con una variante del metodo Newton-Raphson. Il dimensionamento automatico (ovvero la scelta dei diametri, data la conformazione planimetrica della rete, minimizzando il costo totale di costruzione) viene effettuata con un semplice algoritmo di minimizzazione del costo con funzioni di vincolo non lineari, basata sul metodo del gradiente con variabili discrete. Non essendo possibile garantire il raggiungimento del minimo assoluto anche con algoritmi complessi e con lunghi tempi di calcolo, il metodo rinuncia a perseguire il minimo assoluto della funzione costo in favore del contenimento dei tempi di elaborazione, della semplicità e della flessibilità. Il dimensionamento così ottenuto può essere dunque un utile punto di partenza per il progettista, che successivamente potrà perfezionare il dimensionamento in funzione dei molti aspetti peculiari presenti in ogni progetto di rete idrica.

Viene anche descritta l'implementazione del codice di calcolo che, sviluppato in C++ sfruttando le caratteristiche di ereditarietà e polimorfismo della programmazione orientata agli oggetti, consente la modularità del codice, che può essere facilmente arricchito di funzionalità per simulare varie tipologie di apparecchi idraulici in aggiunta a quelli già previsti (serbatoi, erogazioni, condotte con varie leggi di resistenza, pompe e perdite localizzate).

Il codice qui presentato (sorgente o compilato per Windows 95, Macintosh e PowerMacintosh), è disponibile come software di pubblico dominio richiedendolo all'autore all'indirizzo iannelli@polito.it.

INDICE

1. INTRODUZIONE.....	3
2. L'ALGORITMO DI VERIFICA.....	5
2.1 ANALISI TOPOLOGICA E IDENTIFICAZIONE DELLA MATRICE RIDOTTA.....	7
2.1.1 <i>Algoritmo ricorsivo di riconoscimento dell'albero</i>	13
2.2 RISOLUZIONE ITERATIVA COL METODO NR MODIFICATO.....	15
2.3 APPROSSIMAZIONE LINEARE E DEFINIZIONE DELLA CONFIGURAZIONE DI PRIMO TENTATIVO	19
2.4 IMPLEMENTAZIONE DEI VARI TIPI DI ORGANI DI RETE	21
2.4.1 <i>Condotte con legge di resistenza di Darcy</i>	22
2.4.2 <i>Condotte con legge di resistenza di Scimemi</i>	23
2.4.3 <i>Condotte con legge di resistenza di Scimemi-Veronese</i>	23
2.4.4 <i>Condotte con legge di resistenza di De Marchi - Marchetti</i>	24
2.4.5 <i>Condotte con legge di resistenza monomia generica</i>	25
2.4.6 <i>Condotte con legge di resistenza di Colebrook-White + perdite localizzate</i>	26
2.4.7 <i>Pompe con curva caratteristica assegnata</i>	27
3. L'ALGORITMO DI DIMENSIONAMENTO	30
3.1 IL DIMENSIONAMENTO MEDIANTE MINIMIZZAZIONE DEL COSTO DI COSTRUZIONE	31
3.2 DUE SEMPLICI ESEMPI DI APPLICAZIONE DEL METODO DEL GRADIENTE DISCRETO	35
4. L'IMPLEMENTAZIONE DEL CODICE DI CALCOLO.....	43
4.1 IMPLEMENTAZIONE "OBJECT ORIENTED".....	43
4.2 ESPANDIBILITÀ DEL CODICE	48
4.3 GESTIONE DELLA MEMORIA.....	48
5. ISTRUZIONI PER L'UTILIZZO DEL CODICE.....	51
5.1 SIGNIFICATO DEI FILES BINARI GENERATI AL TERMINE DELLE ITERAZIONI	54
6. ESEMPI DI UTILIZZO DEL CODICE: TEST DI VERIFICA E DIMENSIONAMENTO EFFETTUATI PRESSO IL C.R.A.....	55
6.1 VERIFICA DELLA RETE DI DISTRIBUZIONE IDRICA DELLA CITTÀ DI MONZA (MI)	55
6.2 DIMENSIONAMENTO DELLA RETE DISTRIBUTIVA DELL'IMPIANTO DI PLUVIRRIGAZIONE DEL CONSORZIO DI BONIFICA DELLA BARAGGIA VERCELLESE	58
7. CONCLUSIONI E FUTURI SVILUPPI.....	61
8. BIBLIOGRAFIA	62

1. INTRODUZIONE

La risoluzione delle reti idriche a maglie in moto permanente è argomento largamente dibattuto, fin dall'introduzione del metodo risolutivo di Hardy Cross, ed i vantaggi e svantaggi delle diverse classi di metodi di calcolo fino ad oggi proposti sono abbastanza noti e codificati. Non si vuole, pertanto, con questo lavoro, proporre l'ennesima tecnica risolutiva, ma piuttosto focalizzare l'attenzione sull'implementazione del metodo risolutivo in un codice di calcolo sviluppato con l'obiettivo di raggiungere la massima efficienza di elaborazione e di trattare reti anche notevolmente complesse e multiformi (dotate di svariati tipi di apparecchi idraulici e di condotte con leggi di resistenza variabili a piacimento).

Assume, ad esempio, grande importanza l'analisi topologica della rete, al fine di verificarne la congruità individuando il maggior numero possibile di errori nell'introduzione dei dati, e sgravare l'operatore da compiti tediosi quali l'individuazione delle maglie e la definizione delle portate di primo tentativo. Queste operazioni, che venivano tradizionalmente svolte a mano, devono essere necessariamente automatizzate per rendere possibile la verifica di schemi particolarmente complessi quali quelli che si incontrano in fase di analisi di reti già esistenti, operazione che sempre più frequentemente viene effettuata con l'obiettivo di migliorare la condotta gestionale o di progettare l'ampliamento o il completamento di reti esistenti.

Poiché la programmazione ad oggetti si presta particolarmente bene ad adempiere a queste caratteristiche, si ritiene interessante descrivere la tecnica implementativa adottata per sviluppare un codice di calcolo che, pur relativamente complesso, mantenga caratteristiche di razionalità descrittiva e modularità che ne consentano l'agevole potenziamento con l'aggiunta di nuove funzionalità. La programmazione ad oggetti è oggi molto conosciuta ed applicata fra gli sviluppatori di applicazioni commerciali, ma ancora scarsamente utilizzata nella modellistica numerica, dove pure potrebbe essere assai utile per le grandi possibilità di razionalizzazione e semplificazione dei codici di calcolo particolarmente ampi e complessi.

Per quanto riguarda il dimensionamento automatico delle reti idriche mediante minimizzazione della funzione costo, si deve di nuovo segnalare che l'argomento è stato largamente dibattuto per molti anni ma, a differenza di quanto detto a proposito dei metodi di verifica, si è ben lungi dall'aver individuato le tecniche ottimali: ancora oggi, nonostante la disponibilità di numerosissimi algoritmi di ottimizzazione non lineare sempre più elaborati e complessi, il problema del raggiungimento del minimo assoluto della funzione costo in problemi vincolati intrinsecamente non convessi come quello in esame è ancora sostanzialmente irrisolto. I risultati migliori, in termini di raggiungimento di soluzioni particolarmente economiche e di

contenimento dei tempi di calcolo sono stati raggiunti per lo più con tecniche semieuristiche, che però risultano difficilmente generalizzabili ed adattabili a reti complesse.

Viene spesso affermato che la ricerca di tecniche automatiche di dimensionamento ottimale è sconfinata nel puro esercizio speculativo per il fatto che le peculiarità di ciascun progetto rendono i vantaggi economici ottenibili dal dimensionamento ottimale del tutto marginali rispetto ad altri aspetti ben più significativi. Non si condivide del tutto questa opinione: se è indubbio che una tecnica di ottimizzazione valida in termini generali dovrebbe tener conto di una miriade di aspetti talmente complessi ed articolati da rendere praticamente impossibile l'esame del problema in termini generali, e se è pure vero che, anche affrontando il problema in termini molto semplificati non è possibile disporre di un algoritmo che garantisca l'identificazione della soluzione di minimo costo, è altrettanto certo che l'utilizzo delle tecniche di ottimizzazione può risultare assai utile al progettista sia per individuare soluzioni di partenza su cui effettuare ulteriori elaborazioni (consentendo un notevole risparmio di tempo), sia per operare su reti molto complesse analizzando situazioni difficilmente gestibili senza l'ausilio di questo strumento.

In presenza di questa situazione, si è scelto di rinunciare all'obiettivo del raggiungimento della situazione ottimale, ma piuttosto di fornire un metodo semplice e rapido che, pur non garantendo l'individuazione della soluzione di minimo costo assoluto, fornisca al progettista una situazione di primo tentativo su cui operare inserendo gli ulteriori vincoli progettuali di volta in volta presenti, per giungere infine alla scelta progettuale più valida.

Gli argomenti della verifica e del dimensionamento, che rappresentano "blocchi" di programmazione indipendenti, verranno trattati in dettaglio nei prossimi capitoli, dedicando poi attenzione all'implementazione del codice C++ con le tecniche dell'OOP (object oriented programming) ed infine alle modalità di utilizzo, completamento e compilazione del codice, che è disponibile, in versione sorgente e compilata per Windows 95, Macintosh e PowerMacintosh, come software di pubblico dominio.

Viene anche fatto un cenno ai problemi di dimensionamento e verifica di reti di dimensioni e complessità anche notevoli che sono stati svolti presso il Centro di Ricerca sulle Acque e che sono serviti a testare e mettere a punto il programma fino a portarlo alla configurazione attuale, nonché ai possibili sviluppi su cui si intende lavorare in futuro.

2. L'ALGORITMO DI VERIFICA

Una rete idrica può essere schematizzata mediante un numero n di nodi e un numero l di lati, definendo come “lati” i vari dispositivi di rete caratterizzati dall’aver due nodi come estremi: le condotte, le pompe, le valvole, ecc.

Una rete idrica che scambia portata con l’esterno (mediante immissioni o erogazioni di portata ai nodi o distribuite lungo il percorso delle condotte, o anche attraverso serbatoi a quota piezometrica imposta) può essere trasformata in un sistema completamente autonomo aggiungendo ulteriori lati (detti “fittizi”) che congiungeranno i punti di immissione e di erogazione fra loro, in modo da bilanciare le portate circolanti all’interno della rete ottenendo un sistema caratterizzato da scambio di portata nullo con l’esterno. Operando in questo modo, anche i serbatoi, le immissioni e le erogazioni concentrate di portata ai nodi possono essere considerati come particolari tipi di lati: prendendo come riferimento di quota piezometrica uno qualsiasi dei serbatoi presenti in rete (di solito conviene scegliere quello che eroga la portata maggiore), si opererà come di seguito:

- gli ulteriori serbatoi verranno rappresentati come lati fittizi congiungenti i loro nodi di collocazione con il nodo del serbatoio principale, ed aventi portata q incognita e dislivello piezometrico h pari alla differenza di quota piezometrica con il serbatoio principale;
- le immissioni e le erogazioni concentrate di portata imposta verranno rappresentati come lati fittizi congiungenti i loro nodi di collocazione con il nodo del serbatoio principale, ed aventi portata nota e dislivello piezometrico incognito.

I lati possono dunque essere catalogati in tre categorie:

- lati a portata q incognita e dislivello piezometrico imposto (serbatoi oltre al primo);
- lati a dislivello piezometrico incognito e portata imposta (erogazioni o immissioni di portata prestabilita);
- lati a portata q e dislivello piezometrico h entrambi incogniti, caratterizzati da un legame funzionale $q - h$ (condotte, valvole, pompe, ecc.).

In fig. 2.1 viene riportato un esempio di come una rete idraulica con varie condotte e dispositivi di rete possa essere schematizzata secondo i criteri prima definiti.

Indicando con n il numero di nodi, s il numero di lati ad h imposto, e il numero di lati a q imposta, t il numero di lati a q ed h incognite, la verifica della rete consisterà nella soluzione di un sistema costituito da $n-1$ equazioni di continuità ai nodi ed $(s + e + t)$ legami caratteristici $q-h$ relativi ai lati. Questo sistema può essere risolto con tecniche iterative, ma non sono disponibili metodi diretti perché i legami funzionali $q-h$ non sono lineari.

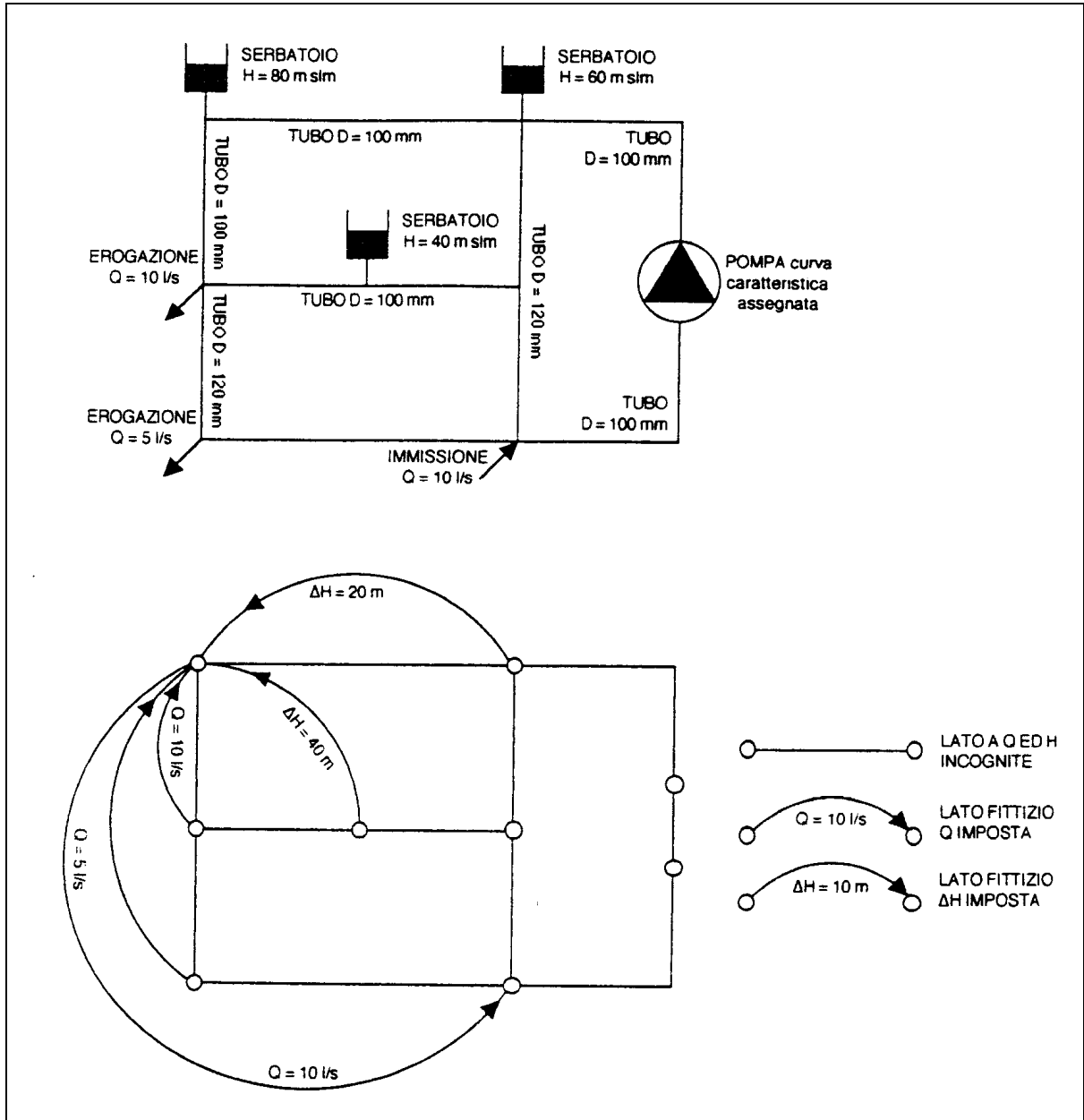


Figura 2.1: Esempio di rete idraulica con relativa schematizzazione topologica

Si nota subito che le equazioni indicate non sono tutte indipendenti, per cui è possibile ricavare un sistema ridotto combinando preliminarmente, dove possibile, le equazioni dipendenti fra loro. Ciò viene effettuato nel metodo di Hardy Cross, in cui, individuando preliminarmente $(s+t-n+1)$ maglie indipendenti ed imponendo la continuità, si calcola la portata di circolazione di ciascuna maglia risolvendo, con un metodo iterativo “sequenziale”, un sistema di $(s+t-n+1)$ equazioni. In pratica il sistema completo viene prima sottoposto (mediante un’analisi topologica da effettuarsi manualmente) ad un metodo diretto di semplificazione che, combinando opportunamente fra loro le equazioni risolutive, ne riduce il numero di $(2n + e - 2)$ equazioni, lasciando alla soluzione iterativa un sistema il più possibile compatto. Come si vedrà, è anche possibile ottenere un diverso sistema ridotto avente come incognite, anziché le

$(s+t-n+1)$ portate di circolazione delle maglie, le $(n-s-1)$ quote piezometriche incognite ai nodi. In questo caso si elimineranno dal sistema completo $(t+e)$ equazioni.

Tuttavia, la tecnica risolutiva sequenziale adottata dal metodo di Cross, se offre gli innegabili vantaggi di consentire anche il calcolo manuale di reti semplici o, in caso di implementazione automatica, di richiedere occupazioni di memoria estremamente contenute, presenta, d'altra parte, problemi di convergenza che si aggravano con l'aumentare delle dimensioni della rete da analizzare.

In alternativa, è possibile risolvere il sistema di equazioni con i classici algoritmi iterativi di soluzione di sistemi non lineari: fra essi i più utilizzati sono il metodo Newton-Raphson (NR) e il metodo dell'analisi lineare (AL). In entrambi i casi l'algoritmo risolutivo può essere applicato al sistema completo, così come ad uno dei due possibili sistemi ridotti di cui si è detto. In quest'ultimo caso si otterrà certamente una considerevole riduzione dell'occupazione di memoria, e, nella maggior parte dei casi, anche una significativa riduzione dei tempi di calcolo ed una maggiore garanzia di convergenza.

Dal confronto fra le due tecniche di riduzione, si osserva che la prima (portate incognite nelle maglie) ottiene il sistema ridotto più piccolo nella maggioranza delle situazioni reali; ma in situazioni di reti fortemente magliate (con un grande numero di maglie rispetto al numero di nodi) e/o in presenza di numerosi serbatoi, la soluzione rispetto alle quote piezometriche incognite ai nodi può risultare più conveniente.

Il codice di calcolo qui presentato prevede entrambe le possibilità, lasciando all'utente la scelta, ma proponendo come "default" quella che, di caso in caso, ottiene il numero di equazioni inferiore.

Dopo essere stato ridotto, il sistema viene risolto con una tecnica derivata dal metodo NR, dopo aver ottenuto una soluzione di primo tentativo mediante risoluzione diretta di un sistema ottenuto linearizzando le equazioni non lineari.

Particolare importanza assume la tecnica di riconoscimento delle maglie, ovvero l'analisi topologica della rete che, per strutture complesse, non può essere lasciata all'utente, ma deve essere gestita in modo automatico. Nel codice qui presentato, l'analisi topologica e la costruzione della matrice ridotta viene effettuata in base ai concetti della teoria dei grafi.

Nei successivi paragrafi viene focalizzata l'attenzione sui tre passi sopra descritti, descrivendo in dettaglio le tecniche implementate nel codice di calcolo: l'analisi topologica; la definizione della configurazione di primo tentativo e la soluzione iterativa finale.

2.1 Analisi topologica e identificazione della matrice ridotta

Secondo la schematizzazione prima presentata, considereremo la rete di distribuzione come un insieme di *nodi* (o vertici) e *lati* (o spigoli), ovvero coppie ordinate di nodi che rappresentino le condotte ed i dispositivi di rete (pompe, valvole, ecc.) immaginabili come "giunzioni"

di due nodi con un verso di percorrenza positivo. I due nodi della coppia ordinata vengono detti *estremi* del lato.

L'insieme dei lati e dei nodi viene detto *grafo orientato*, e in particolare *multigrafo* se contiene anche lati che congiungono gli stessi estremi (lati in parallelo) o che hanno i due estremi coincidenti (lacci, di nessun interesse nell'applicazione specifica). Si userà comunque la dizione "grafo" per indicare indifferentemente entrambe le tipologie. Un sottoinsieme dei lati e dei nodi di un grafo viene detto *sottografo*.

Un nodo e un lato sono *incidenti* se il nodo è uno dei due estremi del lato. L'incidenza consente un'efficace rappresentazione matriciale: un grafo con n nodi ed l lati può essere rappresentato mediante la matrice di incidenza $M = (m_{ij})$ di dimensioni $n \times l$ definita dalla:

$$m_{ij} = \begin{cases} +1 & \text{se il nodo } n_i \text{ è il primo estremo del lato } l_j \\ -1 & \text{se il nodo } n_i \text{ è il secondo estremo del lato } l_j \\ 0 & \text{negli altri casi} \end{cases}$$

Per le elaborazioni che seguono, si indicheranno con lettere maiuscole le grandezze vettoriali o matriciali, e con lettere minuscole le grandezze scalari.

Tutte le colonne della matrice d'incidenza (corrispondenti ai lati) avranno quindi un valore +1 (corrispondente al posto del nodo di entrata), un valore -1 (corrispondente al posto del nodo di uscita) e tutti gli altri nulli.

Due nodi di un grafo sono *connessi* se è possibile raggiungere l'uno partendo dall'altro e percorrendo una successione di lati e nodi. L'intero grafo è connesso se tutti i nodi sono connessi fra loro.

Un grafo si dice *aciclico* se non contiene *maglie* (o cicli, o circuiti), ovvero coppie di percorsi alternativi per raggiungere un nodo partendo da un altro. Un grafo aciclico connesso si dice *albero*, e deve avere necessariamente un numero di lati pari al numero di nodi - 1.

Qualsiasi grafo connesso con n nodi e l lati può essere trasformato in un albero che ne mantenga tutti i nodi, eliminandone opportunamente $(l-n+1)$ lati. L'insieme dei lati eliminati viene detto *coalbero*. I lati facenti parte dell'albero sono detti *rami*; quelli del coalbero *corde*. Pertanto qualsiasi grafo connesso può essere partizionato in due sottografi, detti appunto albero e coalbero.

Tale partizione può essere effettuata con un algoritmo (di solito di tipo ricorsivo) che partendo da un punto del grafo continui ad aggiungere lati fino ad aver raggiunto tutti i nodi. Il codice C++ implementato nel programma è descritto nel seguito come interessante esempio di procedura ricorsiva. L'albero così costruito viene detto *con radice*, in quanto ogni nodo sarà raggiungibile con percorsi unidirezionali dal nodo di partenza, detto appunto *radice*. Nelle reti idrauliche, conviene utilizzare come radice il serbatoio principale, che deve essere collegato (come già detto) alle erogazioni e agli altri eventuali serbatoi mediante lati "fittizi", che danno luogo a tante "pseudo-maglie" quante sono le erogazioni e i serbatoi oltre al primo. Come si

vedrà, è utile costruire l'albero includendo tutti i lati fittizi relativi ai serbatoi ed escludendo tutti quelli relativi alle erogazioni.

Poiché l'albero è il massimo sottografo aciclico, l'aggiunta di ogni corda all'albero darà luogo alla formazione di una nuova maglia, detta *maglia fondamentale*. Pertanto ogni grafo è caratterizzato da un numero di maglie fondamentali coincidente con la dimensione del coalbero, ovvero $l - n + 1$.

Le maglie fondamentali di un grafo possono essere rappresentate mediante una matrice con $l - n + 1$ righe (una per ogni maglia fondamentale) ed l colonne (una per ogni lato) detta *matrice delle maglie* $B_f = (b_{ij})$ e definita dalla:

$$b_{ij} = \begin{cases} +1 & \text{se il lato } l_i \text{ fa parte della maglia } m_j \text{ ed ha verso concorde} \\ -1 & \text{se il lato } l_i \text{ fa parte della maglia } m_j \text{ ed ha verso discorde} \\ 0 & \text{negli altri casi} \end{cases}$$

Poiché ogni maglia fondamentale contiene una ed una sola corda, la matrice può essere costruita con lo stesso ordine di elencazione delle corde ed assegnando ad ogni maglia fondamentale il verso di percorrenza della relativa corda. Riportando ad ogni riga prima i rami e poi le corde, la matrice delle maglie B_f risulta partizionabile come:

$$B_f = \begin{array}{c|c} \text{albero} & \text{coalbero} \\ \hline [B & U] \end{array}$$

in cui B è la matrice delle maglie ridotta e U rappresenta la matrice identica.

L'individuazione della matrice delle maglie è la seconda operazione fondamentale ai fini della descrizione topologica della rete, e può anch'essa essere effettuata con un algoritmo ricorsivo che, partendo dal secondo estremo di ogni corda continui a tentare di connettere rami fino al raggiungimento del primo estremo della stessa corda.

L'ultimo concetto da introdurre è quello di *insieme di taglio* di un grafo connesso, definito come un sottografo che: (1) se eliminato divide il grafo in due parti, (2) non contiene alcun sottografo con la proprietà (1). Poiché un albero è un grafo connesso con il minimo di connessione, l'eliminazione di ogni ramo lo dividerà in due parti, e quindi ogni ramo costituirà da solo un insieme di taglio. Nel caso, invece, di un grafo generico (costituito da un albero ed un coalbero), si riconosce che ogni possibile insieme di taglio comprenderà uno ed un solo ramo associato ad un certo numero di corde, che è necessario eliminare per dividere il grafo in due parti.

Analogamente al caso precedente, si potrà dunque scegliere per ogni grafo un numero $n - 1$ (pari al numero di rami) di *insiemi di taglio fondamentali*, che potranno essere rappresentati mediante una *matrice degli insiemi di taglio* $A_f = (a_{ij})$ avente $n - 1$ righe (una per ogni insieme di taglio fondamentale) ed l colonne (una per ogni lato) e definita dalla:

$$a_{ij} = \begin{cases} +1 & \text{se il lato } l_i \text{ fa parte dell' insieme di taglio } t_j \text{ con verso concorde} \\ -1 & \text{se il lato } l_i \text{ fa parte dell' insieme di taglio } t_j \text{ con verso discorde} \\ 0 & \text{negli altri casi} \end{cases}$$

Ancora in analogia al caso precedente, la matrice degli insiemi di taglio può essere costruita con lo stesso ordine di elencazione dei rami, con versi a loro concordi e riportando ad ogni riga prima i rami e poi le corde. Risulterà pertanto partizionabile come:

$$A_f = \begin{bmatrix} \text{albero} & \vdots & \text{coalbero} \\ [U & & A] \end{bmatrix}$$

in cui U rappresenta la matrice identica ed A la matrice degli insiemi di taglio ridotta.

Un teorema della teoria dei grafi stabilisce che, se ordinate allo stesso modo, le matrici A e B sono fra loro ortogonali, ovvero che $B = -A^T$ ed $A = -B^T$. Grazie a questa importante caratteristica, una volta individuata una delle due matrici, risulta immediatamente determinata anche la seconda.

Le matrici delle maglie e degli insiemi di taglio consentono di esprimere in forma compatta l'insieme delle equazioni di continuità dei carichi e delle portate.

Le equazioni di continuità delle portate in rete in forma matriciale si possono esprimere come:

$$A_f Q = 0$$

in cui Q è il vettore colonna delle portate relative agli l lati. Questa rappresenta l'insieme delle equazioni di continuità negli insiemi di taglio (matematicamente equivalente all'insieme di equazioni di continuità ai nodi), ed indica che la somma delle portate per ciascun insieme di taglio deve essere nulla.

Partizionando la matrice A_f come già visto e il vettore colonna Q nelle componenti Q_T relativa all'albero e Q_{T^c} relativa al coalbero, si ottiene:

$$[U \ \vdots \ A] \begin{bmatrix} Q_T \\ \hline Q_{T^c} \end{bmatrix} = 0 \quad (1)$$

ovvero:

$$Q_T = -A Q_{T^c} \quad (2)$$

Analogamente si può esprimere l'equazione di continuità dei carichi nelle maglie come:

$$B_f H = 0$$

in cui H è il vettore colonna delle differenze di carico piezometrico relative ai lati. Questa equazione indica che la somma delle perdite di carico lungo ciascuna maglia deve essere nulla. Partizionando la matrice B_f e il vettore H nelle parti relative all'albero e al coalbero si ottiene (analogamente al caso precedente) la:

$$[B \ \vdots \ U] \begin{bmatrix} H_T \\ \hline H_{T^c} \end{bmatrix} = 0 \quad (3)$$

ovvero:

$$H_{T'} = -BH_T \quad (4)$$

Le equazioni (2) e (4) consentono di ricavare Q_T in funzione di $Q_{T'}$ e $H_{T'}$ in funzione di H_T . Pertanto le portate nei lati del coalbero e le differenze di carico nei lati dell'albero rappresentano un insieme di variabili primarie, dalle quali si possono immediatamente ricavare le variabili secondarie mediante la (2) e la (4).

Per la risoluzione della rete sarà dunque sufficiente determinare i valori delle sole variabili primarie, dalle quali si potranno poi ricavare immediatamente anche quelle secondarie.

Poichè per i lati fittizi relativi ai serbatoi la differenza piezometrica è imposta e per quelli relativi alle erogazioni o immissioni di portata è quest'ultima ad essere nota, è opportuno includere i primi nell'albero ed i secondi nel coalbero, in modo da ridurre al minimo il numero di variabili principali incognite. Pertanto si ordineranno i lati nel seguente ordine:

1. lati fittizi relativi ai serbatoi (tutti appartenenti all'albero);
2. lati appartenenti all'albero;
3. lati appartenenti al coalbero;
4. lati fittizi relativi ad erogazioni ed immissioni di portata (tutti appartenenti al coalbero).

Operando questa partizione, le equazioni (1) e (3) diventano:

$$\begin{bmatrix} U & 0 & \vdots & A_{11} & A_{12} \\ 0 & U & \vdots & A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} Q_{T_1} \\ Q_{T_2} \\ \hline Q_{T_1'} \\ Q_{T_2'} \end{bmatrix} = 0 \quad (5)$$

$$\begin{bmatrix} B_{11} & B_{12} & \vdots & U & 0 \\ B_{21} & B_{22} & \vdots & 0 & U \end{bmatrix} \begin{bmatrix} H_{T_1} \\ H_{T_2} \\ \hline H_{T_1'} \\ H_{T_2'} \end{bmatrix} = 0 \quad (6)$$

in cui i vettori colonna Q ed H sono partizionati nelle quattro componenti sopra indicate e le matrici A e B sono anch'esse partizionate di conseguenza.

A questo punto è necessario introdurre i legami q - h relativi ai vari lati. Si tratta di funzioni diversificate di caso in caso, potendo includere leggi di resistenza di condotte, ovvero curve caratteristiche di pompe, od altro. Le esprimeremo con un operatore generico "*" come di seguito:

$$h = r * q \text{ (forma resistiva)}$$

ovvero:

$$q = y * h \text{ (forma conduttiva)}$$

La forma resistiva consente di ricavare h da q , la forma conduttiva viceversa.

In forma matriciale possiamo indicare sinteticamente tali relazioni per l'intera rete come:

$$H = R * Q$$

$$Q = Y * H$$

in cui H e Q sono i vettori colonna delle differenze di carico e delle portate nei lati; R ed Y rappresentano le matrici quadrate dei coefficienti, aventi valori non nulli solo sulla diagonale principale, ed “*” avente il significato di operatore generico non lineare, come già detto.

Rispettando la partizione introdotta nella (5) e nella (6), i legami q-h si porranno per i soli lati non fittizi nell’albero e nel coalbero:

$$\begin{bmatrix} H_{T_2} \\ H_{T_1'} \end{bmatrix} = \begin{bmatrix} R_T & 0 \\ 0 & R_{T'} \end{bmatrix} * \begin{bmatrix} Q_{T_2} \\ Q_{T_1'} \end{bmatrix} \quad (7)$$

$$\begin{bmatrix} Q_{T_2} \\ Q_{T_1'} \end{bmatrix} = \begin{bmatrix} Y_T & 0 \\ 0 & Y_{T'} \end{bmatrix} * \begin{bmatrix} H_{T_2} \\ H_{T_1'} \end{bmatrix} \quad (8)$$

A questo punto è possibile ricavare l’equazione risolutiva in forma resistiva (incognite le portate nei lati non fittizi del coalbero) o conduttiva (incogniti i dislivelli piezometrici nei lati non fittizi dell’albero).

Ricaviamo l’equazione in forma resistiva: dalla (6) otteniamo:

$$\begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} H_{T_1} + \begin{bmatrix} B_{12} & U \\ B_{22} & 0 \end{bmatrix} \begin{bmatrix} H_{T_2} \\ H_{T_1'} \end{bmatrix} + \begin{bmatrix} 0 \\ U \end{bmatrix} H_{T_2'} = 0 \quad (9)$$

introducendo la (7):

$$\begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} H_{T_1} + \begin{bmatrix} B_{12} & U \\ B_{22} & 0 \end{bmatrix} \begin{bmatrix} R_T & 0 \\ 0 & R_{T'} \end{bmatrix} * \begin{bmatrix} Q_{T_2} \\ Q_{T_1'} \end{bmatrix} + \begin{bmatrix} 0 \\ U \end{bmatrix} H_{T_2'} = 0 \quad (10)$$

Scrivendo le variabili secondarie Q_{T_2} in funzione delle variabili primarie mediante la (5) otteniamo:

$$\begin{bmatrix} Q_{T_2} \\ Q_{T_1'} \end{bmatrix} = \begin{bmatrix} -A_{21} & -A_{22} \\ U & 0 \end{bmatrix} \begin{bmatrix} Q_{T_1'} \\ Q_{T_2'} \end{bmatrix} \quad (11)$$

che, introdotta nella (10) e ricordando che $A = -B^T$, dà:

$$\begin{bmatrix} B_{11} \\ B_{21} \end{bmatrix} H_{T_1} + \begin{bmatrix} B_{12} & U \\ B_{22} & 0 \end{bmatrix} \begin{bmatrix} R_T & 0 \\ 0 & R_{T'} \end{bmatrix} * \begin{bmatrix} B_{12} & U \\ B_{22} & 0 \end{bmatrix}^T \begin{bmatrix} Q_{T_1'} \\ Q_{T_2'} \end{bmatrix} + \begin{bmatrix} 0 \\ U \end{bmatrix} H_{T_2'} = 0 \quad (12)$$

Eliminiamo infine le variabili secondarie $H_{T_2'}$ prendendo solo la parte superiore della (12):

$$B_{11}H_{T_1} + \begin{bmatrix} B_{12} & U \end{bmatrix} \begin{bmatrix} R_T & 0 \\ 0 & R_{T'} \end{bmatrix} * \begin{bmatrix} B_{12}^T & B_{22}^T \\ U & 0 \end{bmatrix} \begin{bmatrix} Q_{T_1'} \\ Q_{T_2'} \end{bmatrix} = 0 \quad (13)$$

che semplificata diventa:

$$B_{11}H_{T_1} + B_{12}R_T * \begin{bmatrix} B_{12}^T Q_{T_1'} + B_{22}^T Q_{T_2'} \end{bmatrix} + R_T * Q_{T_1'} = 0 \quad (14)$$

La (14) rappresenta l’equazione risolutiva in forma resistiva, avente come incognite il vettore $Q_{T_1'}$ delle portate nei lati non fittizi del coalbero.

L’equazione in forma conduttiva può essere ricavata in modo assolutamente equivalente, e si

presenta come di seguito:

$$Y_T * H_{T_2} + A_{21} Y_T * [A_{11}^T H_T + A_{21}^T H_{T_2}] + A_{22} Q_{T_2} = 0 \quad (15)$$

In questo caso le incognite sono rappresentate dal vettore H_{T_2} dei dislivelli piezometrici nei lati non fittizi dell'albero.

Le due equazioni risolutive espresse in forma matriciale si prestano ad una comoda implementazione dell'algoritmo risolutivo iterativo, come sarà descritto nei successivi paragrafi. Come si vede, si potrà scegliere di caso in caso la forma risolutiva più conveniente. Poiché è nella maggior parte dei casi più rapida la convergenza del sistema risolutivo più compatto, si preferirà la forma resistiva quando i lati non fittizi nel coalbero saranno in numero minore dei lati non fittizi nell'albero, cioè quando $s+t-n+1 < n-1$, come già detto.

2.1.1 Algoritmo ricorsivo di riconoscimento dell'albero

Si riporta il codice C++ dell'algoritmo di ricerca dell'albero per mostrare come la ricorsività (ovvero la possibilità di richiamare una procedura dall'interno della procedura stessa), resa possibile dai linguaggi evoluti quali il C, il C++ o il Pascal, consenta l'elaborazione estremamente chiara e concisa di problemi di ricerca tutt'altro che semplici.

Procedura "cercarami" di costruzione dell'albero (le righe sono numerate solo ai fini della successiva spiegazione):

```

1 void cercarami (const int& n1, int& posto)
2 //procedura ricorsiva che cerca e attacca uno ad uno i rami all'albero
3 {
4   int t, n2; //t=tubo, n1 = 1° nodo del tubo, n2 = 2° nodo del tubo
5   nododafare[n1] = 0;
6   for (t = 0; t < ntubi; t++)
7     if (Inc[t][n1])
8       for (n2=0; n2<nnodi; n2++)
9         if (Inc[t][n2] && (n1 != n2) && nododafare[n2])
10            {posiz[t] = posto++; cercarami(n2,posto);}
11 }
```

Obiettivo della procedura è identificare all'interno del grafo rappresentativo della rete un albero avente come radice il primo serbatoio. Gli altri serbatoi, come già detto, sono schematizzati come lati fittizi che collegano i nodi di tali serbatoi al nodo del primo serbatoio. Tali lati vengono inseriti nell'albero, e vengono collegati direttamente alla radice.

La procedura costruisce, in pratica, la matrice "posiz" che contiene il numero progressivo di ogni tubo nell'ordine di calcolo, ovvero partendo dal primo serbatoio e proseguendo fino ad

aver collegato tutti gli altri nodi alla radice. La variabile “posto” rappresenta il numero d’ordine che, in fase di chiamata della procedura, viene impostata pari alla posizione del primo tubo che, nell’ordine di calcolo, seguirà l’ultimo dei serbatoi. Successivamente, la variabile “posto” viene incrementata man mano che si individuano nuovi tubi da inserire nell’albero.

La riga 1 è l’intestazione C++ della procedura (come noto, nei linguaggi C e C++ le procedure sono particolari funzioni che restituiscono il valore “void” (vuoto), ovrerosia non restituiscono alcun valore. Alla procedura devono essere forniti i valori “n1” (numero del nodo al quale attaccare il successivo ramo) e “posto” (numero progressivo del tubo iniziale nell’ordine di calcolo). I lati vengono preventivamente ordinati disponendo prima i serbatoi, poi i lati generici (denominati “tubi”) ed infine le erogazioni.

Partendo dal primo nodo, il loop alla riga 6 verifica tutti i tubi alla ricerca di uno incidente col nodo stesso. Se lo trova, cerca il secondo nodo di tale tubo (loop di riga 8) verificando che non faccia già parte dell’albero; quindi inserisce tale tubo nella matrice “posiz” (rami ordinati), incrementa il contatore “posto” di 1 e richiama dal suo interno la stessa procedura “cercarami” partendo dal nodo appena trovato (riga 10).

La procedura viene richiamata dal corpo del programma con le seguenti istruzioni:

```

.....
//DICHIARAZIONI ED ALLOCAZIONI MATRICI
20 charvect nododafare;//vettore per marcare nodi da collegare all'albero
21 nododafare.alloc(nnodi);//dimensionamento matrice
22 for (i=0;i<nnodi;i++) nododafare[i]=1;//all'inizio pone tutti=1
23 intvect posiz; //posizione di ogni tubo nell'ordine di calcolo
24 posiz.alloc(ntubi);
25 charmatr Inc; //dichiarazione matrice di incidenza
26 Inc.alloc(ntubi,nnodi); //dimensionamento matrice incidenza
27 // costruzione matrice incidenza Inc (non riportata per brevità)
.....
//COSTRUZIONE ALBERO
28 int posto = nserbatoi;
29 cercarami(nodorif,posto); //chiama la procedura ricorsiva cominciando 30
//dal serbatoio principale
31 for (i=0;i<nserbatoi;i++) cercarami(lato[i].nodo2,posto);//la richiama
32 //partendo dai serbatoi secondari
.....

```

Oltre alle chiamate della procedura “cercarami” sono state riportate per completezza, stralciandole dal corpo del programma, le dichiarazioni, i dimensionamenti e le inizializzazioni delle variabili e delle matrici topologiche inerenti la costruzione dell’albero.

Come si nota, la procedura ricorsiva viene richiamata prima dal serbatoio principale e poi, in sequenza, dai serbatoi secondari. Ciò in quanto i serbatoi secondari sono preventivamente inseriti nell'albero (come già spiegato), e porzioni di rete potrebbero essere unicamente collegati ad uno dei serbatoi secondari.

Poichè i lati vengono sequenzialmente analizzati, partendo dal primo, ed inclusi nell'albero se consentono di raggiungere un nodo non ancora collegato, l'albero verrà costruito inserendo ai primi posti (più vicini alla radice) i lati elencati per primi nell'ordine di immissione. Ciò in quanto l'algoritmo si preoccupa di identificare un albero, senza effettuare alcuna particolare scelta fra i tanti possibili alberi aventi come radice il serbatoio principale. Per migliorare la rapidità di convergenza del successivo processo iterativo, può essere quindi conveniente ordinare i lati (nel file di input) per resistività crescente, in modo che le condotte principali vengano inserite nell'albero più vicine possibile alla radice. Questa possibilità è stata sperimentata su reti molto estese e, in certi casi, ha dato luogo a riduzioni abbastanza significative dei tempi di calcolo necessari alla soluzione della rete.

Si noti che i vettori e le matrici vengono gestiti come oggetti di classe "charvect", "intvect", "charmatr", ecc. Queste classi, definite all'interno del programma, consentono di gestire l'allocazione dinamica delle matrici (ovvero la possibilità di dichiarare le dimensioni delle matrici nel corso dell'esecuzione del programma senza doverle dimensionare il fase di compilazione) e quindi consentono l'adattabilità del codice alle dimensioni della memoria a disposizione dell'utente senza richiedere la ricompilazione. Questa caratteristica è commentata al par.4.3. Come si vede alla riga 26, il dimensionamento è effettuato con un "metodo" della classe "charmatr", mediante l'istruzione `Inc.alloc(ntubi,nnodi)`; che alloca lo spazio in memoria e dimensiona la matrice con le dimensioni "ntubi" e "nnodi".

2.2 Risoluzione iterativa col metodo NR modificato

La tecnica risolutiva utilizzata si basa sul metodo Newton-Raphson (NR), ma vi si differenzia per il fatto di non richiedere il calcolo e l'inversione della matrice jacobiana ad ogni passo iterativo: viene invece utilizzata un'equazione iterativa per aggiornare, in base ai residui delle funzioni, un'approssimazione della matrice Jacobiana inversa che viene poi utilizzata per correggere le variabili al fine di ottenere, ad ogni iterazione, un residuo più piccolo. Questo approccio presenta due vantaggi:

- la soluzione diventa molto più rapida, in quanto l'inversione della matrice è operazione che, per reti estese, risulta assai dispendiosa in termine di tempi di calcolo;
- consente una formulazione agevole di qualsiasi legge non derivabile: ad esempio la legge di resistenza di Colebrook-White, non derivabile in quanto non esplicita.

L'algoritmo utilizza pure una tecnica di variazione del passo di iterazione, che lo riduce nel caso che la convergenza risulti difficoltosa. Ciò migliora significativamente la trattabilità di

reti a convergenza difficile.

Il metodo (denominato “metodo Newton-Raphson modificato”) si basa su una classe di tecniche risolutive, presentate da Broyden (Broyden C.G., 1965), per risolvere sistemi di equazioni senza richiedere il calcolo di derivate parziali. Si presenta nel seguito l’algoritmo di Broyden, così come ulteriormente elaborato da Lam e Wolla (Lam C.F., Wolla M.L., 1972).

Utilizzando ancora la notazione vettoriale usata al cap. 2, la formula di Newton-Raphson (estensione del metodo di Newton a sistemi di equazioni) è la seguente:

$$X_{i+1} = X_i - J_i^{-1} F_i$$

in cui: $J_i = \left. \frac{\partial F}{\partial X} \right|_{X=X_i}$; $F(X) = 0$; $F_i = F(X_i)$.

Il vettore X rappresenta il set di incognite (Q_{T_1} , per la forma resistiva, H_{T_2} per la forma conduttiva), mentre il sistema di equazioni di risolvere (la (14) in forma resistiva oppure la (15) in forma conduttiva) è rappresentato dall’equazione vettoriale $F(X) = 0$.

Per derivare il metodo di Broyden, si ponga $P_i = -B_i^{-1} F_i$ con $B_i \approx J_i$. Si potrà pertanto scrivere un’approssimazione della formula NR come

$$X_{i+1} = X_i + t_i P_i \quad (16)$$

in cui è stato pure introdotto il parametro t_i di controllo del passo di iterazione (come si vedrà meglio in seguito).

Poichè il vettore F può essere considerato funzione di t_i , si può sviluppare in serie F, trascurando i termini di ordine superiore, come:

$$F(X_i + t_i P_i) = F_{i+1} \approx F[X_i + (t_i - s_i) P_i] + s_i \frac{dF}{dt_i} \quad (17)$$

La derivata di F rispetto a t_i può essere espressa come:

$$\frac{dF}{dt_i} = \frac{\partial F}{\partial X_i} \frac{dX_i}{dt_i} = J_i P_i.$$

Sostituendo nella (17) si ottiene:

$$F_{i+1} - F[X_i + (t_i - s_i) P_i] \approx s_i J_i P_i.$$

Siccome B_i è un’approssimazione della matrice jacobiana J_i , per migliorare l’approssimazione si prende B_{i+1} in modo che sia:

$$F_{i+1} - F[X_i + (t_i - s_i) P_i] = s_i B_{i+1} P_i.$$

Eguagliando s_i e t_i si ottiene:

$$\left. \begin{aligned} F_{i+1} - F_i &= t_i B_{i+1} P_i \\ \text{ovvero } \frac{F_{i+1} - F_i}{t_i} &= B_{i+1} P_i = \frac{dF_j}{dt} \end{aligned} \right\} \quad (18)$$

in cui X_j si trova fra X_i e X_{i+1} . Pertanto, il vettore $B_{i+1} P_i$ è pari a dF/dt valutato in un punto compreso fra X_i e X_{i+1} . Anche se questa stima è peggiore di quella al punto X_{i+1} , assicurerà

comunque che B_{i+1} si riferisca a punti più vicini alla soluzione di X_i .

Se B approssima J , per approssimare J^{-1} si dovrà calcolare B^{-1} ad ogni iterazione. Questa inversione può essere evitata: definendo $H_i = -B_i^{-1}$ e ponendo $Y_i = F_{i+1} - F_i$ si otterrà:

$$P_i = H_i F_i$$

che sostituito nell'eq. (18) darà:

$$\left. \begin{array}{l} Y_i = t_i B_{i+1} P_i \\ \text{ovvero } H_{i+1} Y_i = -t_i P_i \end{array} \right\} \quad (19)$$

Questa equazione lega la variazione del vettore funzione F alla variazione di X nella direzione di P_i . Non fornisce, invece, alcuna informazione sulla variazione di F quando X varia diversamente da P_i . Ma l'uso di t_i e H_i può far variare X in una direzione diversa da quella di P_i , per cui bisogna prendere qualche precauzione. Si prenda B_{i+1} in modo che la variazione di F prevista da B_{i+1} in una direzione Q_i ortogonale a P_i sia pari a quella prevista da B_i . In termini matematici ciò può essere scritto come:

$$B_{i+1} Q_i = B_i Q_i \quad \text{con } Q_i' P_i = 0 \quad (20)$$

La soluzione delle equazioni (19) e (20) è:

$$B_{i+1} = B_i + \frac{(Y_i - t_i B_i P_i) P_i'}{t_i P_i' P_i}.$$

Usando la formula iterativa di inversione, questa equazione può essere scritta come:

$$H_{i+1} = H_i - \frac{(t_i P_i + H_i Y_i) P_i' H_i}{P_i' H_i Y_i}. \quad (21)$$

Il metodo NR modificato può ora essere scritto come $X_{i+1} = X_i + t_i P_i$ in cui $P_i = H_i F_i$; $Y_i = F_{i+1} - F_i$ ed H_i viene aggiornato mediante la (21).

Il valore di t_i può essere scelto in modo da minimizzare la norma di F . Tuttavia, per ridurre il tempo di calcolo, anziché minimizzare la norma di F rispetto a t_i , si può ricercare un valore di t_i in modo che esso riduce la norma di F_i . Esistono diverse tecniche di stima per minimizzare o ridurre funzioni di una variabile scalare. Quella proposta da Broyden è di tipo predittivo.

Se ciascuna componente di F è una funzione lineare di t e se $H_i = -J_i^{-1}$, allora la norma di F , $|F|$, è una funzione quadratica di t . Se $t = 1$ (passo completo), allora $F(X+tP) = F(X+HF) = F(X-J^{-1}F) = F(X) = 0$. Quindi, $F = 0$ quando $t = 1$. Questa ideale approssimazione quadratica ha la forma di $(1 - 2t + t^2) |F(t=0)| + t^3 |F(t=1)|$. Il minimo di questa funzione interpolante si trova a

$$t_{\min} = \frac{\sqrt{1 + 6RF} - 1}{3RF} \quad (22)$$

in cui $RF = \frac{|F(t=1)|}{|F(t=0)|}$.

Questo valore di t può essere utilizzato per stimare il minimo della norma di F ad ogni iterazione.

Ricapitolando, l'algoritmo NR modificato si riassume nei seguenti punti:

1. Si imposta una stima iniziale X_0 della soluzione (valori di primo tentativo).
2. Si calcola un valore iniziale di H_0 , mediante metodi numerici o semplicemente utilizzando la matrice identica.
3. Si calcola $F_i = F(X_i)$.
4. Si calcola $P_i = H_i F_i$.
5. Si pone $t=1$ e si aggiorna X mediante la (16). Se $F(t=1)$ è minore di $F(t=0)$ si va al passo successivo, altrimenti si usa la (22) per stimare t_{\min} e aggiornare X mediante la (16).
6. Si verifica la convergenza confrontando $|F_{i+1}|$ con la tolleranza ϵ , se raggiunta, si esce.
7. Si calcola $Y_i = F_{i+1} - F_i$ e si usa la (21) per calcolare H_{i+1} .
8. Si torna al passo 4. e si ripete il procedimento.

Al passo 5, se la norma di F valutata per t_{\min} risulta maggiore di $F(t=0)$, significa che il valore di t_{\min} stimato mediante la (22) non è corretto. Si è allora utilizzato un ulteriore passo di affinamento della stima di t_{\min} (Lam C.F., Wolla M.L., 1972): in prossimità del minimo, la norma di F viene approssimata con un polinomio del secondo ordine passante per $F(t=0)$, $F(t=t_{\min})$ ed $F(t=1)$, con t_{\min} calcolato dalla (22). Il nuovo valore di t_{\min} può ora essere calcolato cercando il minimo di tale polinomio quadratico. Il passo 5 prevede dunque tre successive stime del valore di t e di $F(t)$ ciascuna delle quali viene eseguita in sequenza solo se la precedente non ha fornito un valore di $|F(t)| < |F(t=0)|$.

Un approfondito confronto di questo metodo col Newton Raphson standard è stato effettuato da Lam e Wolla nel lavoro più volte citato. Il metodo modificato è risultato di convergenza più rapida soprattutto nella soluzione di reti di grandi dimensioni. La rapidità si manifesta, più che in un ridotto numero di iterazioni, in un minor tempo necessario ad eseguire ciascuna iterazione per il fatto di non dover invertire la matrice jacobiana: Lam e Wolla riportano un confronto fra i due metodi nella risoluzione di una rete con 30 nodi e 50 lati, in cui il metodo NR raggiunge la convergenza in 574 secondi con 14 iterazioni, mentre il metodo modificato, pur richiedendo 68 iterazioni, impiega 354 secondi, con un tempo di calcolo per iterazione pari a circa 1/8 di quello richiesto dal metodo standard (i tempi si riferiscono ad un calcolatore dell'epoca, con velocità di elaborazione notevolmente inferiore agli odierni personal computer). Tale rapporto diventa ancora più favorevole per reti più estese.

Un altro vantaggio segnalato è la minore sensibilità alla qualità dei valori di primo tentativo: il metodo modificato pare infatti in grado di convergere anche per condizioni di primo tentativo sensibilmente diverse dalle condizioni effettive (perlomeno per reti interamente monotone resistive), a differenza del metodo standard. In particolare, quando si tema di disporre di valori di primo tentativo lontani dalla soluzione finale, Lam e Wolla suggeriscono, per la stima iniziale di H_0 al passo 2, di utilizzare la matrice identica piuttosto che ricorrere al calcolo con metodi numerici, in quanto valori errati delle derivate locali potrebbero portare verso configurazioni ancora più lontane dall'obiettivo finale.

Lam e Wolla sottolineano come il metodo sia stato concepito per trattare elementi monotoni di tipo resistivo, quali sono le condotte o gli altri organi dissipativi. Poichè la preventiva riduzione della matrice elimina dalla soluzione iterativa i lati fittizi relativi a serbatoi ed immissioni di portata in rete, l'unico elemento non dissipativo la cui soluzione viene affidata al metodo iterativo è la pompa con curva caratteristica assegnata. Alcune prove effettuate presso il C.R.A. su reti di estensione non molto elevata hanno comunque ottenuto un'agevole convergenza che però, secondo Lam e Wolla, può diventare difficile per curve caratteristiche con spiccata tendenza non lineare. Si è portati a concludere che una buona stima delle condizioni iniziali (qual è quella ottenibile con la linearizzazione delle funzioni $h-q$) consenta il raggiungimento della convergenza anche per sistemi non del tutto monotoni crescenti. La questione deve comunque essere presa in considerazione per verifiche di reti estese con numerose pompe, eventualmente realizzando una preventiva approssimazione lineare o quasi-lineare delle curve caratteristiche adottate.

2.3 Approssimazione lineare e definizione della configurazione di primo tentativo

Per la formulazione del primo tentativo, necessario per iniziare il processo iterativo di risoluzione del sistema, si è operato linearizzando il sistema e risolvendolo con il metodo di riduzione di Gauss. Le equazioni del moto sono state linearizzate mediante una retta passante per il punto $q=0 - h=0$ e per un altro punto per il quale viene predeterminata la velocità media del fluido: prefissando il valore di velocità media si determina immediatamente la portata q_0 (moltiplicandola per la sezione della relativa condotta) e poi la corrispondente perdita complessiva h_0 calcolandola con l'equazione completa. Per il valore iniziale di velocità è stato scelto 1 m/s, in quanto idoneo, come ordine di grandezza, a trattare applicazioni di tipo acquedottistico. Si è osservato che la scelta di tale valore influenza comunque abbastanza poco i tempi di calcolo del successivo processo iterativo.

Per quanto riguarda le curve caratteristiche delle pompe, che vengono successivamente gestite con un'approssimazione polinomiale basata su tre punti forniti dall'utente, l'approssimazione lineare è stata effettuata interpolando i tre punti con una retta mediante il metodo dei minimi quadrati. Da diverse prove effettuate, si è visto che tale approssimazione fornisce un buon punto di partenza, per il fatto che le curve caratteristiche di solito, nel campo di effettivo utilizzo, non si discostano molto da una curva caratteristica effettivamente lineare.

La risoluzione del sistema linearizzato viene effettuata con il metodo di riduzione di Gauss. Si tratta di un metodo semplice e rapido che, non essendo iterativo, garantisce sempre il raggiungimento della soluzione. Nel codice è stato anche implementato il metodo di Gauss-Jordan, ovvero una variante del metodo di Gauss che evita l'errore numerico nel caso che l'elemento pivotale risulti pari a 0 o molto piccolo. Tuttavia questo secondo metodo, richiedendo un maggior numero di elaborazioni rallenta i tempi di calcolo e quindi, poiché la for-

mulazione del sistema ridotto rende assai improbabile la generazione di coefficienti nulli, si è preferito lasciare come opzione principale il metodo di Gauss nella sua forma più semplice. In effetti, nelle numerose prove effettuate col programma non è mai capitato di imbattersi in errori numerici dovuti a valori troppo piccoli del coefficiente pivotale.

Il metodo scelto (così come tutti i metodi diretti) tende ad aumentare i tempi di calcolo e l'occupazione di memoria all'aumentare delle dimensioni del sistema ridotto da analizzare, per cui, per reti estremamente estese, la scelta di un metodo iterativo potrebbe risultare più opportuna, anche se sarebbe da verificare l'effettiva capacità di convergere sempre e rapidamente verso la soluzione. Il metodo scelto è stato comunque testato in reti dell'ordine di 2000 condotte e 2000 nodi, denotando tempi di calcolo ed occupazioni di memoria compatibili con i personal computer.

Per evitare il ricalcolo della soluzione di primo tentativo nel caso di successive rielaborazioni di reti con la stessa topologia (operazione che viene ripetuta molte volte nei casi di taratura di modelli di rete) è stato introdotto un meccanismo di conservazione e successivo ripristino della soluzione di primo tentativo.

Una volta linearizzate, le equazioni vettoriali $H = R * Q$ e $Q = Y * H$ si semplificano rispettivamente in $H = R' * Q + S$ e $Q = Y' * H + Z$, in cui i vettori S e Z contengono valori non nulli in corrispondenza delle sole pompe, in quanto in tutti gli altri casi le leggi resistive passano per il punto $q = 0$; $h = 0$. Sostituendo tali espressioni nelle equazioni risolutive (14) e (15) si ottengono le espressioni vettoriali lineari da trattare con il metodo di Gauss.

L'espressione (14) relativa alla forma resistiva diventa:

$$B_{11} H_{T_1} + B_{12} R_T * [B_{12}^T Q_{T_1} + B_{22}^T Q_{T_2}] + R_T * Q_{T_1} = 0$$

che, grazie alle proprietà associativa e distributiva che divengono valide sostituendo l'operatore non lineare "*" con l'usuale prodotto linee per colonne, con alcuni passaggi può essere scritta:

$$(R_T + B_{12} R_T B_{12}^T) Q_{T_1} = -B_{11} H_{T_1} - B_{12} (R_T B_{22}^T Q_{T_2} + S_T) - S_T$$

ovvero:

$$C Q_{T_1} = N$$

in cui è Q_{T_1} il vettore delle incognite, C la matrice dei coefficienti e N il vettore dei termini noti.

Col metodo di Gauss, mediante una sequenza di trasformazioni elementari, la matrice C incrementata del vettore N viene trasformata in triangolare alta, e viene poi risolta in cascata. Con la variante di Gauss-Jordan, prioritariamente alla trasformazione elementare di ogni riga, si verifica che il valore del primo coefficiente (detto "pivotale") non sia troppo piccolo (tale da dar luogo ad errori numerici se posto al denominatore di operazioni di divisione), ed eventualmente si permuta preventivamente l'intera riga con un'altra. Si omette, comunque, la presentazione dei due metodi che sono descritti in qualsiasi testo di calcolo matriciale.

Per la forma conduttiva, con procedimento analogo, la semplificazione dell'equazione risolutiva (15) dà luogo alla seguente:

$$(Y_T + B_{12}^T Y_T B_{12}) H_{T_2} = B_{22}^T Q_{T_2} - B_{12}^T (Y_T B_{11} H_{T_1} - Z_T) - Z_T$$

ovvero:

$$C' H_{T_2} = N'$$

in cui è H_{T_2} il vettore delle incognite, C' la matrice dei coefficienti e N' il vettore dei termini noti. La soluzione è analoga al caso della forma resistiva.

Le modalità di linearizzazione delle equazioni descrittive dei vari organi di rete vengono presentate, assieme alle modalità di implementazione delle formule risolutive, nei prossimi paragrafi.

2.4 Implementazione dei vari tipi di organi di rete

Si è già detto al par. 2.1 dell'implementazione dei serbatoi e delle erogazioni o immissioni di portata assegnata mediante lati fittizi, con la conseguente analisi topologica. Restano da definire le modalità di implementazione degli altri possibili organi di rete, quali le condotte, le pompe o gli organi dissipativi. Si riportano qui le leggi adottate, rimandando al cap. 4 la descrizione della strutturazione del codice secondo le metodologie della programmazione orientata agli oggetti.

L'implementazione delle leggi che legano il dislivello piezometrico h alla portata q che percorre le varie tipologie di lati, deve essere espressa, come già detto, in due diverse forme, a secondo del metodo risolutivo adottato. Tali leggi sono state indicate con le espressioni simboliche:

$$h = r * q \text{ (forma resistiva)}$$

$$q = y * h \text{ (forma conduttiva).}$$

Per le condotte, sono state implementate nel codice le seguenti leggi di resistenza:

- formula di Darcy (proposta per tubazioni in ghisa senza rivestimento interno, ovvero con rivestimento bituminoso leggero);
- formula di Scimemi (per condotte in fibrocemento ovvero in ghisa sferoidale con rivestimento interno cementizio);
- formula di Scimemi-Veronese (per condotte in acciaio semplicemente bitumato);
- formula di De Marchi - Marchetti (per condotte in acciaio con rivestimento interno in bitume centrifugato);
- espressione monomia generica della cadente piezometrica $j = \beta \frac{Q^\alpha}{D^\gamma}$;
- formula di Colebrook-White per le perdite distribuite + somma delle perdite localizzate.

L'ultima espressione è anche in grado di simulare organi di intercettazione e regolazione quali

valvole o saracinesche.

In aggiunta è stata implementata la gestione di pompe di curva caratteristica assegnata sia per immissione in rete (con prelievo dalla falda o da serbatoi esterni) sia di rilancio all'interno della rete.

2.4.1 Condotte con legge di resistenza di Darcy

La legge di resistenza di Darcy stima la cadente piezometrica j con la ben nota formula:

$$j = \beta \frac{q^2}{d^5}$$

con j = cadente piezometrica [adim.]; q = portata [m^3/s]; d = diametro interno della condotta [m] e β = coefficiente di resistenza di Darcy pari a:

$$\beta = 0,00164 + \frac{0,000042}{d}$$

Poichè lo stesso Darcy suggerì di raddoppiare il valore di β per tubi usati, la formula è stata implementata con un coefficiente moltiplicativo k inseribile da parte dell'utente.

La legge di resistenza si presenta pertanto in forma resistiva:

$$h = r * q = \frac{k \cdot \beta \cdot l}{d^5} \cdot q^2$$

in cui l = lunghezza della condotta [m], $r = \frac{k \cdot \beta \cdot l}{d^5}$ e l'operatore "*" \equiv " $\cdot (\dots)^2$ ".

In forma conduttiva sarà invece:

$$q = y * h = \left(\frac{d^5}{k \cdot \beta \cdot l} \right)^{1/2} \cdot q^{1/2}$$

con $y = \left(\frac{d^5}{k \cdot \beta \cdot l} \right)^{1/2}$ e l'operatore "*" \equiv " $\cdot (\dots)^{1/2}$ ".

L'utente può inserire il coefficiente k pari ad 1 o 2, come suggerito da Darcy, ovvero con diverso valore attribuibile mediante taratura.

Per l'approssimazione lineare di primo tentativo si è utilizzata una retta passante per i punti $q=0$; $h=0$ e $q=q_0$; $h=h(q_0)$ in cui q_0 è stato posto pari a $v_0 \cdot \frac{\pi d^2}{4}$, con v_0 = valore di primo tentativo della velocità media in condotta, assunto pari ad 1 m/s. L'espressione risultante diventa:

in forma resistiva $h = r' \cdot q$ con

$$r' = \frac{k \cdot \beta \cdot l \cdot q_0}{d^5} = r \cdot q_0$$

in forma conduttiva $q = y' \cdot h$ con

$$y' = \frac{1}{r'} = \frac{d^5}{k \cdot \beta \cdot l \cdot q_0} = \frac{y^2}{q_0}$$

2.4.2 Condotte con legge di resistenza di Scimemi

La legge di resistenza di Scimemi stima la cadente piezometrica j per condotte in fibrocemento con la formula:

$$j = 0,000984 \frac{q^{1,78}}{d^{4,78}}$$

con gli stessi simboli e le stesse unità di misura prima indicate.

Le implementazioni saranno: in forma resistiva:

$$h = r * q = \frac{0,000984 \cdot l}{d^{4,78}} \cdot q^{1,78}$$

con $r = \frac{0,000984 \cdot l}{d^{4,78}}$ e l'operatore “*” \equiv “. (...) ^{1,78}”.

In forma conduttiva:

$$q = y * h = \left(\frac{d^{4,78}}{0,000984 \cdot l} \right)^{1/1,78} \cdot q^{1/1,78}$$

con $y = \left(\frac{d^{4,78}}{0,000984 \cdot l} \right)^{1/1,78}$ e l'operatore “*” \equiv “. (...) ^{1/1,78}”.

Per l'approssimazione lineare di primo tentativo si è utilizzata anche in questo caso una retta passante per i punti $q=0; h=0$ e $q=q_0; h=h(q_0)$ in cui q_0 è stato posto pari a $v_0 \cdot \frac{\pi d^2}{4}$, con $v_0 =$ valore di primo tentativo della velocità media in condotta, assunto pari ad 1 m/s. L'espressione risultante diventa:

in forma resistiva $h = r' \cdot q$ con

$$r' = \frac{0,000984 \cdot l \cdot q_0^{0,78}}{d^{4,78}} = r \cdot q_0^{0,78}$$

in forma conduttiva $q = y' \cdot h$ con

$$y' = \frac{1}{r'} = \frac{d^{4,78}}{0,000984 \cdot l \cdot q_0^{0,78}} = \frac{y^{1,78}}{q_0^{0,78}}$$

2.4.3 Condotte con legge di resistenza di Scimemi-Veronese

La legge di resistenza di Scimemi-Veronese stima la cadente piezometrica j per condotte in acciaio semplicemente bitumato con la formula:

$$j = 0,00141 \frac{q^{1,82}}{d^{4,72}}$$

con gli stessi simboli e le stesse unità di misura prima indicate.

Le implementazioni saranno: in forma resistiva:

$$h = r * q = \frac{0,00141 \cdot l}{d^{4,72}} \cdot q^{1,82}$$

con $r = \frac{0,00141 \cdot l}{d^{4,72}}$ e l'operatore “*” \equiv “ $\cdot (\dots)^{1,82}$ ”.

In forma conduttiva:

$$q = y * h = \left(\frac{d^{4,72}}{0,00141 \cdot l} \right)^{1/1,82} \cdot q^{1/1,82}$$

con $y = \left(\frac{d^{4,72}}{0,00141 \cdot l} \right)^{1/1,82}$ e l'operatore “*” \equiv “ $\cdot (\dots)^{1/1,82}$ ”.

Per l'approssimazione lineare di primo tentativo si è utilizzata anche in questo caso una retta passante per i punti $q=0; h=0$ e $q=q_0; h=h(q_0)$ in cui q_0 è stato posto pari a $v_0 \cdot \frac{\pi d^2}{4}$, con $v_0 =$ valore di primo tentativo della velocità media in condotta, assunto pari ad 1 m/s. L'espressione risultante diventa:

in forma resistiva $h = r' \cdot q$ con

$$r' = \frac{0,00141 \cdot l \cdot q_0^{0,82}}{d^{4,72}} = r \cdot q_0^{0,82}$$

in forma conduttiva $q = y' \cdot h$ con

$$y' = \frac{1}{r'} = \frac{d^{4,72}}{0,00141 \cdot l \cdot q_0^{0,82}} = \frac{y^{1,82}}{q_0^{0,82}}$$

2.4.4 Condotte con legge di resistenza di De Marchi - Marchetti

La legge di resistenza di De Marchi - Marchetti stima la cadente piezometrica j per condotte in acciaio con rivestimento interno in bitume centrifugato con la formula:

$$j = 0,000917 \frac{q^{1,81}}{d^{4,81}}$$

con gli stessi simboli e le stesse unità di misura prima indicate.

Le implementazioni saranno: in forma resistiva:

$$h = r * q = \frac{0,000917 \cdot l}{d^{4,81}} \cdot q^{1,81}$$

con $r = \frac{0,000917 \cdot l}{d^{4,81}}$ e l'operatore “*” \equiv “ $\cdot (\dots)^{1,81}$ ”.

In forma conduttiva:

$$q = y * h = \left(\frac{d^{4,81}}{0,000917 \cdot l} \right)^{1/1,81} \cdot q^{1/1,81}$$

con $y = \left(\frac{d^{4,81}}{0,000917 \cdot l} \right)^{1/1,81}$ e l'operatore “*” \equiv “ $\cdot (\dots)^{1/1,81}$ ”.

Per l'approssimazione lineare di primo tentativo si è utilizzata anche in questo caso una retta

passante per i punti $q=0$; $h=0$ e $q=q_0$; $h=h(q_0)$ in cui q_0 è stato posto pari a $v_0 \cdot \frac{\pi d^2}{4}$, con $v_0 =$ valore di primo tentativo della velocità media in condotta, assunto pari ad 1 m/s. L'espressione risultante diventa:

in forma resistiva $h = r' \cdot q$ con

$$r' = \frac{0,000917 \cdot l \cdot q_0^{0,81}}{d^{4,81}} = r \cdot q_0^{0,81}$$

in forma conduttiva $q = y' \cdot h$ con

$$y' = \frac{1}{r'} = \frac{d^{4,81}}{0,000917 \cdot l \cdot q_0^{0,81}} = \frac{y^{1,81}}{q_0^{0,81}}$$

2.4.5 Condotte con legge di resistenza monomia generica

E' stata anche implementata una legge di resistenza di tipo monomio, con tutti i coefficienti definibili dall'utente:

$$j = \beta \frac{q^\alpha}{d^\gamma}$$

con j in [m/m], q in [m³/s], d in [m] ed i tre coefficienti α , β e γ inseribili da parte dell'utente. E' così possibile implementare formule di resistenza monomie non esplicitamente previste, come quella di Hazen Williams.

Le implementazioni saranno: in forma resistiva:

$$h = r * q = \frac{\beta \cdot l}{d^\gamma} \cdot q^\alpha$$

con $r = \frac{\beta \cdot l}{d^\gamma}$ e l'operatore “*” \equiv “. (...) $^\alpha$ ”.

In forma conduttiva:

$$q = y * h = \left(\frac{d^\gamma}{\beta \cdot l} \right)^{1/\alpha} \cdot q^{1/\alpha}$$

con $y = \left(\frac{d^\gamma}{\beta \cdot l} \right)^{1/\alpha}$ e l'operatore “*” \equiv “. (...) $^{1/\alpha}$ ”.

Per l'approssimazione lineare di primo tentativo si è anche in questo caso utilizzata una retta passante per i punti $q=0$; $h=0$ e $q=q_0$; $h=h(q_0)$ in cui q_0 è stato posto pari a $v_0 \cdot \frac{\pi d^2}{4}$, con $v_0 =$ valore di primo tentativo della velocità media in condotta, assunto pari ad 1 m/s. L'espressione risultante diventa:

in forma resistiva $h = r' \cdot q$ con

$$r' = \frac{\beta \cdot l \cdot q_0^{\alpha-1}}{d^\gamma} = r \cdot q_0^{\alpha-1}$$

in forma conduttiva $q = y' \cdot h$ con

$$y' = \frac{1}{r'} = \frac{d^\gamma}{\beta \cdot l \cdot q_0^{\alpha-1}} = \frac{y^\alpha}{q_0^{\alpha-1}}$$

2.4.6 Condotte con legge di resistenza di Colebrook-White + perdite localizzate

E' stata infine implementata la formula di Darcy-Weisbach:

$$j = \lambda \frac{v^2}{2 \cdot g \cdot d}$$

con l'indice di resistenza λ calcolato mediante l'espressione di Colebrook-White:

$$\frac{1}{\sqrt{\lambda}} = -2 \log_{10} \left(\frac{1}{3,715} \frac{\varepsilon}{d} + \frac{2,51}{\text{Re} \sqrt{\lambda}} \right)$$

in cui: j = cadente piezometrica [m/m]; v = velocità media in condotta [m/s]; g = accelerazione di gravità [m/s²]; d = diametro interno della condotta [m]; ε = asperità media della parete interna della condotta [m]; Re = numero di Reynolds = $v \cdot d / \nu$ con ν = viscosità cinematica del fluido [m²/s].

Per la valutazione delle perdite complessive fra i due nodi di estremità della condotta, alla perdita distribuita $j \cdot l$ è stato aggiunto un termine di stima della somma delle perdite concentrate, espresso come:

$$\sum h_{loc} = \xi \cdot \frac{v^2}{2g}$$

in cui, oltre ai termini già visti, si ha $\sum h_{loc}$ = somma delle perdite localizzate presenti lungo la condotta [m]; ξ = somma dei coefficienti di perdita localizzata presenti lungo la condotta [adim.]. Si è prevista tale implementazione in quanto, a differenza delle formule pratiche precedentemente esaminate che comprendono nelle perdite distribuite un valore mediato delle perdite localizzate nei giunti e nelle altre singolarità solitamente presenti lungo la condotta, l'espressione di Colebrook-White tiene conto delle sole perdite distribuite propriamente dette, mentre le perdite localizzate devono essere opportunamente stimate a parte.

Questa implementazione consente anche di tener conto di organi di intercettazione o altri tipi di dispositivi di dissipazione, assegnando lunghezza nulla e diametro e coefficiente ξ corrispondenti all'organo dissipativo da simulare.

L'espressione complessiva diventa dunque:

$$h = \left(\frac{\lambda}{d} \cdot l + \xi \right) \cdot \frac{q^2}{2g \frac{\pi d^2}{4}}$$

Ovviamente, nell'implementazione, bisogna tener conto del fatto che il coefficiente λ dipende da Re , e quindi dalla portata. L'implementazione è possibile in forma esplicita per la forma conduttiva, la cui espressione, omettendo i semplici passaggi, risulta pari a:

$$q = \sqrt{\frac{h}{\left[\pi \sqrt{\frac{gd^3}{2l}} \log_{10} \left(\frac{\varepsilon}{3,715d} + 2,51v \sqrt{\frac{l}{2gd^3h}} \right) \right]^2 + \frac{8\xi}{g\pi^2 d^4}}}$$

mentre per la forma resistiva è necessario un procedimento di calcolo iterativo: si è adottata la seguente espressione:

$$h = \left\{ \left[\pi \sqrt{\frac{gd^3}{2l}} \log_{10} \left(\frac{\varepsilon}{3,715d} + 2,51v \sqrt{\frac{l}{2gd^3h_0}} \right) \right]^2 + \frac{8\xi}{g\pi^2 d^4} \right\} \cdot q^2$$

in cui con h_0 si indica il valore di h all'iterazione precedente. Visto lo scarso peso di h_0 rispetto agli altri membri, assegnando un valore iniziale di h_0 di 1 m la convergenza viene raggiunta in tre o quattro iterazioni al massimo.

Per l'approssimazione lineare di primo tentativo si è anche in questo caso utilizzata una retta passante per i punti $q=0; h=0$ e $q=q_0; h=h(q_0)$ in cui q_0 è stato posto pari a $v_0 \cdot \frac{\pi d^2}{4}$, con $v_0 =$ valore di primo tentativo della velocità media in condotta, assunto pari ad 1 m/s. Si è pure trascurato (ponendolo = 1) il termine h che compare sotto radice all'interno dell'espressione logaritmica, che impedisce di esplicitare la formula. L'espressione risultante diventa:

in forma resistiva $h = r' \cdot q$ con

$$r' = \left\{ \left[\pi \sqrt{\frac{gd^3}{2l}} \log_{10} \left(\frac{\varepsilon}{3,715d} + 2,51v \sqrt{\frac{l}{2gd^3h_0}} \right) \right]^2 + \frac{8\xi}{g\pi^2 d^4} \right\} \cdot q_0$$

in forma conduttiva $q = y' \cdot h$ con

$$y' = \frac{1}{r'}$$

2.4.7 Pompe con curva caratteristica assegnata

E' stato infine implementato un elemento di rete costituito da una pompa con curva caratteristica assegnata. La curva caratteristica viene inserita dall'utente mediante tre coppie di punti portata - prevalenza (i due limiti estremi della curva più un punto centrale) calcolata fra i due nodi di collegamento alla rete.

Mediante questo elemento sarà possibile simulare sia pompe di rilancio, sia pompe di immissione in rete:

- per le pompe di rilancio, l'utente assegnerà semplicemente il nodo di presa, il nodo di mandata e la curva caratteristica della pompa come rilevabile dai dati tecnici del fabbricante;
- per le pompe di immissione in rete si dovrà inserire un serbatoio (con quota pari al livello di presa della pompa) ed una pompa con presa nel serbatoio e mandata nel nodo di immis-

sione in rete;

- in caso di pompe prelevanti direttamente dalla falda, si dovrà inserire un serbatoio fittizio con quota piezometrica pari al livello dinamico della falda;
- nel caso si disponga di un certo numero di letture simultanee di portata e pressione alla mandata della pompa, si implementerà un serbatoio fittizio con quota pari al riferimento altimetrico delle pressioni, e curva caratteristica interpolante le coppie portata-pressione rilevate.

L'implementazione è effettuata approssimando la curva caratteristica con una parabola $h = a \cdot q^2 + b \cdot q + c$ passante per i tre punti q_1-h_1 , q_2-h_2 , q_3-h_3 , inseriti dall'utente. I valori di a , b , c vengono calcolati con le seguenti espressioni:

$$a = \frac{1}{q_1 - q_3} \left(\frac{h_2 - h_1}{q_2 - q_1} - \frac{h_2 - h_3}{q_2 - q_3} \right)$$

$$b = \frac{h_2 - h_3}{q_2 - q_3} - a(q_2 + q_3)$$

$$c = -aq_1^2 - bq_1 + h_1$$

Il calcolo viene considerato valido se, a convergenza, il punto di funzionamento risulta compreso fra le due coppie di valori estremi inseriti dall'utente. Se il punto ricade all'esterno, il calcolo viene comunque portato a termine e nella presentazione dei risultati viene segnalato che il punto di funzionamento ricade all'esterno della curva caratteristica.

Per migliorare la linearità dell'implementazione e quindi la convergenza, per portate inferiori al valore q_1 indicato dall'utente (e quindi fuori dal campo di funzionamento della pompa) la curva è stata implementata mediante una retta tangente alla parabola in corrispondenza del punto $q_1 - h_1$. Questa implementazione ha consentito, nelle diverse prove effettuate, di raggiungere un'agevole convergenza per una vasta gamma di curve caratteristiche ordinarie. Le espressioni adottate sono pertanto: in forma resistiva:

$$\text{per } q < q_1: \quad h = (2aq_1 + b)q + (c - aq_1^2)$$

$$\text{per } q \geq q_1: \quad h = aq^2 + bq + c$$

ed in forma conduttiva:

$$\text{per } h > h_1: \quad q = \frac{h_0 - (c - aq_1^2)}{(2aq_1 + b)}$$

$$\text{per } h \leq h_1: \quad q = \frac{-b + \sqrt{b^2 - 4a(c - h)}}{2a}$$

Per la corretta applicazione della forma conduttiva è necessario che l'approssimazione quadratica $h(q)$ sia monotona decrescente nel campo di funzionamento specificato. Questa condizione è opportuno sia verificata anche in forma resistiva ai fini del raggiungimento della con-

vergenza.

Per l'approssimazione lineare di primo tentativo è stata utilizzata una retta interpolante le tre coppie di valori $q_i - h_i$ col metodo dei minimi quadrati.

Nella forma resistiva la retta assume l'espressione $h = rq + s$ con le costanti r ed s aventi i seguenti valori:

$$r = \frac{(q_1 - \bar{q})(h_1 - \bar{h}) + (q_2 - \bar{q})(h_2 - \bar{h}) + (q_3 - \bar{q})(h_3 - \bar{h})}{(q_1 - \bar{q})^2 + (q_2 - \bar{q})^2 + (q_3 - \bar{q})^2}$$

$$s = \bar{h} - r\bar{q}$$

in cui $\bar{q} = \frac{q_1 + q_2 + q_3}{3}$ e $\bar{h} = \frac{h_1 + h_2 + h_3}{3}$.

Nella forma conduttiva si assume l'espressione $q = yh + z$ in cui:

$$y = \frac{1}{r}$$

$$z = -\frac{s}{r}$$

3. L'ALGORITMO DI DIMENSIONAMENTO

Il problema del dimensionamento, a differenza di quello di verifica precedentemente illustrato, è non banale anche per reti a maglie aperte, e consiste nell'individuare la configurazione di rete (diametri, materiali, quote dei serbatoi, caratteristiche delle pompe, ecc.) atta a soddisfare i requisiti di funzionalità (pressioni, portate, grado di sicurezza e ridondanza, semplicità gestionale, ecc.) con il minimo costo globale.

Posto in questi termini il problema si presenta talmente vasto e complesso da risultare praticamente non affrontabile con una certa generalità. Viene allora generalmente semplificato lasciando come unica variabile la scelta dei diametri da assegnare alle diverse condotte da progettare, ed attribuendo al progettista la gestione di tutti gli ulteriori parametri dimensionali.

Fatta questa prima semplificazione, la soluzione più tradizionale del problema viene fornita dal cosiddetto criterio di economia degli acquedotti consortili (valido per reti prive di maglie chiuse), che, mediante alcune ipotesi semplificative, giunge a dedurre che in una condotta con portate derivate lungo il percorso in punti discreti, conviene che le perdite di carico unitarie nei vari tronchi stiano tra loro come le radici cubiche delle portate.

Viene pure applicata un'estensione del metodo alle reti a maglie, mediante una preventiva eliminazione delle maglie stesse tramite il "taglio" di alcuni dei tronchi in punti preliminarmente scelti e denominati "punti neutri".

Poichè tale metodo risulta certamente di applicazione lunga e tediosa, con molti limiti di applicabilità (ad esempio alle reti da dimensionare solo parzialmente, come negli ampliamenti di sistemi esistenti) e non privo di aspetti non del tutto condivisibili e semplificazioni piuttosto drastiche negli stessi fondamenti teorici, sono stati prodotti molti sforzi, specie nei primi anni '70, alla ricerca di metodi più precisi e soprattutto più generali, in grado cioè di analizzare in modo più rigoroso le reti magliate e i problemi di ampliamento di strutture esistenti.

Tutti i metodi alternativi, sempre basati sul criterio della minimizzazione del costo globale, rientrano nel campo della cosiddetta "ricerca operativa". Si tratta di metodi di ottimizzazione di una certa funzione scalare di una variabile vettoriale (la funzione "costo globale") sottoposta al vincolo della funzionalità richiesta alla rete. In ciò si affianca a molti problemi di ottimizzazione tipici del moderno approccio ingegneristico ma, come si vedrà, rientra nel campo dei problemi più difficilmente gestibili, ovvero quelli di tipo non lineare con funzioni di vincolo di tipo non convesso.

Vista la non disponibilità di metodi generali in grado di garantire l'individuazione della configurazione ottimale in problemi di questo tipo, e visto anche il nutrito numero di ipotesi

semplificative poste alla base del problema, che delegano al progettista l'esame, di volta in volta, di molte variabili che complicano ulteriormente il procedimento, nel presente lavoro si è deciso di rinunciare all'obiettivo del raggiungimento dell'ottimo assoluto, ma piuttosto di realizzare uno strumento di tipo euristico, semplice e rapido che, pur non garantendo il raggiungimento dell'ottimo, fornisca risultati significativi in tempi di calcolo molto rapidi, al fine di fornire al progettista un ausilio alle tradizionali tecniche di dimensionamento certamente molto utile per ridurre drasticamente i tempi di comprensione, analisi e trattamento dei problemi di dimensionamento di reti a maglie chiuse sia ex-novo sia di ampliamento o completamento.

Nei successivi paragrafi sarà presentato il metodo utilizzato, anche mediante alcuni semplici esempi, non senza confrontarlo con gli altri possibili algoritmi di ottimizzazione non lineare oggi più accreditati ed utilizzati.

3.1 Il dimensionamento mediante minimizzazione del costo di costruzione

Il problema del dimensionamento viene di solito posto nei seguenti termini: noto l'andamento planimetrico della rete da dimensionare (o ampliare) e l'esigenza funzionale da soddisfare, definire l'entità delle condotte e dei manufatti.

L'andamento planimetrico viene solitamente considerato esterno alla problematica analitica, in quanto collegato a parametri più squisitamente progettuali quali le caratteristiche orografiche del territorio, la distribuzione territoriale delle utenze, l'andamento delle strade, ecc.

Le principali esigenze funzionali da soddisfare si riconoscono nei valori delle portate che la rete dovrà essere in grado di fornire alle utenze e nelle quote piezometriche che dovranno essere rispettate nelle varie condizioni di erogazione; ma ve ne sono anche altre, come quella di garantire possibilità alternative di convogliamento in caso di interruzione di alcuni tratti (che rende la struttura magliata particolarmente adatta alle reti di distribuzione), o quella di consentire il monitoraggio per facilitare la gestione (che ha dato origine al concetto, di origine anglosassone, della "distrettizzazione" delle reti).

Per rendere il problema univocamente definito, è inoltre necessario introdurre il concetto di minimizzazione del costo dell'opera, che consiste nell'individuare, fra tutte le possibili configurazioni che soddisfano le diverse esigenze funzionali, quella di minimo costo globale.

In questi termini, si ottiene una tipica applicazione del problema generale di individuazione di estremi vincolati: determinare gli estremi di un campo scalare $f(X)$ (funzione obiettivo) con X variabile in un assegnato sottoinsieme del dominio di f . Nel nostro caso la variabile vettoriale X ha come componenti i diametri delle condotte, le quote dei serbatoi, le potenze delle pompe e le caratteristiche delle altre parti da dimensionare; il campo scalare $f(X)$ è il costo globale dell'opera (esprimibile come somma del costo di costruzione e dell'attualizzazione del costo di esercizio delle varie componenti della rete); il sottoinsieme del dominio è rappresentato da

tutte le possibili configurazioni di rete che soddisfano le esigenze funzionali sopra definite.

I problemi di estremo vincolato sono spesso molto difficili, e non è dato alcun metodo matematico per affrontarli in generale. Nel caso specifico poi, il problema risulta ulteriormente complicato da diversi aspetti, come ad esempio:

- alcune delle componenti del vettore X delle variabili sono discrete anziché continue: infatti i diametri tra cui scegliere sono quelli commerciali disponibili, ed anche la scelta dei materiali delle condotte potrebbe a rigore essere considerata nel processo di ottimizzazione;
- le varie componenti di X sono eterogenee: infatti, nel caso più generale, oltre ai diametri delle condotte, possono variare le quote dei serbatoi, le caratteristiche delle pompe, il numero e le caratteristiche delle eventuali valvole di dissipazione o di altri dispositivi di linea, ecc.;
- nel caso delle reti a maglie, le caratteristiche analitiche delle condizioni di vincolo sono difficili, in quanto richiedono soluzioni iterative;
- nel caso si vogliano calcolare le perdite di carico delle tubazioni con la formula di Colebrook-White (l'unica in grado di trattare egualmente bene le situazioni di moto puramente turbolento e di transizione), vi è l'ulteriore difficoltà della non esplicitabilità della relazione rispetto alla perdita di carico.

Sono pertanto state proposte numerose tecniche risolutive, tutte richiedenti un grado più o meno accentuato di semplificazione del problema generale sopra enunciato. Nessuna, tuttavia, viene ancora oggi riconosciuta come universalmente applicabile in quanto, come vedremo, ciascuna presenta punti a favore ma anche problemi di vario genere non ancora completamente risolti. Tali tecniche possono essere di tipo analitico o numerico.

La tecnica analitica di ottimizzazione vincolata più conosciuta è quella cosiddetta dei *moltiplicatori di Lagrange*, che può essere enunciata come di seguito: se un campo scalare $f(x_1, \dots, x_n)$ soggetto a m vincoli $g_1(x_1, \dots, x_n) = 0, \dots, g_m(x_1, \dots, x_n) = 0$ con $m < n$ ha un estremo relativo, allora esistono m scalari $\lambda_1, \dots, \lambda_m$ tali che $\nabla f = \lambda_1 \nabla g_1, \dots, \lambda_m \nabla g_m$ in ciascun punto di estremo.

Vista la complessità del problema e la potenziale vastità di possibili formulazioni delle equazioni di vincolo, la soluzione analitica del sistema di $n+m$ equazioni scalari prevista dal metodo dei moltiplicatori di Lagrange risulta praticamente non ottenibile, se non in casi particolari.

In alternativa ai metodi analitici è disponibile una nutrita gamma di metodi (prevalentemente di tipo numerico), che possono essere classificati in (Colorni A., 1992):

- *evolutivi*, in cui un procedimento per passi successivi raggiunge la soluzione ottima a partire da una configurazione di primo tentativo;
- *esaustivi*, in cui tutte le soluzioni vengono elencate e si sceglie la migliore;
- *enumerativi* (variante del precedente), in cui si fa uso di qualche criterio per limitare le soluzioni possibili, fino alla scelta finale;

- *decomposizione* in sottoproblemi semplici, collegati o non collegati fra loro (è importante chiarire i collegamenti e le relative variabili di “coordinamento”);
- *euristici*, in cui, quando non si riesce ad avere la soluzione ottima, ci si accontenta di una soddisfacente. Si tratta di metodi “ad hoc” utilizzabili per certi problemi particolari.

E' abbastanza chiaro che i metodi esaustivi ed enumerativi non sono proponibili nel caso di reti di distribuzione di media o grande estensione, in quanto il numero di possibili combinazioni risolutive (anche in presenza di variabili discrete e limitate) raggiunge dimensioni tali da rendere il problema praticamente non trattabile con gli odierni strumenti di elaborazione. La decomposizione è invece possibile in molti casi, e certamente molto conveniente, ma comunque, nel caso delle reti di distribuzione a maglie, non affrontabile con generalità.

Nell'ambito dei metodi evolutivi si collocano praticamente tutte le tecniche ad oggi proposte, che però, vista la complessità del problema, vengono spesso sottoposte a semplificazioni od approssimazioni che le fanno sconfinare nell'ambito dei metodi euristici o (come talvolta li si classifica) semi-euristici. In ogni caso, quest'ultima classificazione spetta, a rigore, a tutti i metodi oggi conosciuti per la trattazione del problema in oggetto, in quanto la non convessità del problema non consente in nessun caso di garantire il raggiungimento del minimo assoluto piuttosto che di uno dei tanti possibili minimi relativi.

I metodi evolutivi possono a loro volta essere classificati in:

- metodi primali (es. metodo delle direzioni ammissibili), che si basano sull'esame del problema tenendo conto dei vincoli esplicitamente;
- penalizzazione (es. metodo della funzione di penalità) che incorporano le funzioni di vincolo nella funzione obiettivo mediante opportuni “pesi” che la allontanano dall'ottimo quando i vincoli risultano non rispettati;
- linearizzazione (es. programmazione separabile), in cui si “deforma” il problema dato fino a renderlo lineare, in modo da consentire l'applicazione dei metodi risolutivi dei problemi lineari (metodo del semplice).

Qualunque sia la tecnica scelta, il maggiore problema sta nel fatto che le funzioni di vincolo, in particolare nelle reti a maglie, risultano di tipo non convesso, e quindi presentano numerosi minimi relativi in grado di “intrappolare” il procedimento iterativo impedendogli di raggiungere il minimo assoluto.

Vista la vastità della questione, è impossibile, in questa sede, esaminare, seppur sinteticamente, le numerose tecniche risolutive classificabili nell'ambito delle categorie prima illustrate. E' altrettanto problematico individuare, fra tutte, la tecnica che garantisca, per il problema specifico del dimensionamento delle reti idrauliche, il conseguimento dei migliori risultati in termini di riduzione della funzione costo globale. Ci si limiterà, quindi, a citare alcuni fra i metodi proposti in letteratura, sintetizzandone le conclusioni riportate dagli stessi autori.

Metodi evolutivi primali, per lo più derivati dal metodo del gradiente, sono stati utilizzati, fra

gli altri, da Jacoby (1968), Watanada (1973), Cembrowicz e Harrington (1973), Lam (1973): si tratta di metodi che consentono di affrontare il problema con grande generalità: anche se in tutti i casi il problema è stato impostato considerando come uniche variabili da ottimizzare i diametri di tutte o parte (ampliamento) delle condotte, l'impostazione permette di includere fra le variabili e fra le funzioni di vincolo le condizioni più disparate, come ad esempio la scelta delle quote di imposta dei serbatoi, il dimensionamento ottimale di pompe di immissione o di rilancio, ecc. In particolare, Lam imposta il criterio di minimizzazione sul fatto che le variabili (i diametri delle condotte) risultano di tipo discreto, consentendo, per questo, una certa semplificazione del problema ed un contenimento dei tempi di calcolo necessari.

I risultati ottenuti in termini di effettiva qualità dei vari processi di minimizzazione sono difficilmente confrontabili, ma pressochè tutti gli autori annotano il fatto che non si può avere la garanzia del raggiungimento del minimo assoluto del costo, e che i vari procedimenti proposti possono ottenere risultati più o meno validi mediante ripetute prove ottenute partendo da diverse condizioni iniziali o variando con tecniche più o meno euristiche diversi parametri di controllo dei processi iterativi. La maggior parte dei lavori citati riportano esempi, che però si riferiscono sempre a problemi di estensione molto limitata, con tempi di elaborazione che aumentano esponenzialmente quanto più risulta raffinata la metodologia o gli artifici di volta in volta adottati per avvicinarsi al minimo assoluto.

Metodi basati sulla linearizzazione del problema sono stati proposti da Mele (1971), Sassoli e Milano (1973), Ciaponi e Papiri (1985), Morgan e Goulter (1985), Cappelli d'Orazio e al. (1992). I metodi di linearizzazione sono diversi: in certi casi, partendo dal concetto che fra tutte le possibili configurazioni quelle che prevedono l'apertura delle maglie sono sempre le più economiche, il problema viene linearizzato assumendo come incognite le lunghezze dei tronchi anziché i diametri. In pratica ciascun tronco viene suddiviso in tanti segmenti quanti sono i diametri commerciali fra cui si vuole scegliere e ricercando poi il minimo della funzione costo con tecniche di programmazione lineare che, variando le lunghezze dei tratti, giungono ad individuare la soluzione di minimo costo in cui ogni tronco risulta suddiviso in due soli tratti di diametri commerciali di solito contigui. Cappelli d'Orazio e al. si servono di una tecnica ibrida che, per certi versi, utilizza pure le funzioni di penalizzazione.

Termini e al., (1994) propongono un metodo di tipo probabilistico adatto per variabili di tipo discreto, la cui applicazione a reti di dimensioni reali appare però penalizzata dai lunghissimi tempi di elaborazione citati dagli autori.

Il metodo utilizzato nel presente codice deriva da quello di Lam prima citato (metodo del gradiente discreto): si tratta di una tecnica che presenta i pregi della semplicità, generalità e rapidità, che lo rendono facilmente applicabile a reti anche di grande estensione. Trattandosi di un metodo "primale", le condizioni di vincolo non entrano a far parte della funzione obiettivo da minimizzare, ma vengono applicate separatamente, iterazione dopo iterazione, per verificare che siano ancora soddisfatte. Inoltre, siccome le variabili sono discrete (per le condotte hanno

come dominio la serie dei diametri commerciali), la minimizzazione della funzione obiettivo viene effettuata seguendo la direzione del gradiente negativo, e variando di una unità il diametro di quelle componenti che formano con essa un angolo inferiore ad un certo valore di confronto. In questo modo si evita di calcolare ed invertire la matrice jacobiana ad ogni iterazione. L'applicazione di questo metodo può dar luogo ad interessanti considerazioni sulla validità del processo di minimizzazione del costo, che vengono presentate nel prossimo paragrafo mediante semplici esempi.

3.2 Due semplici esempi di applicazione del metodo del gradiente discreto

Il metodo del gradiente discreto mantiene separato il campo scalare da minimizzare (rappresentato dal costo globale della rete) dalle condizioni di vincolo (rappresentate da disequazioni rispetto ai valori minimi e massimi delle portate e pressioni in rete che devono essere verificate). Per illustrare il procedimento, prendiamo in considerazione due esempi il più possibile semplificati.

E' innanzitutto necessario definire la funzione costo globale del sistema. Prendendo in considerazione una rete planimetricamente definita, senza pompe o altri dispositivi di linea, con il materiale prefissato e le quote dei serbatoi già definite, le uniche variabili risultano essere i diametri delle condotte da dimensionare.

Il costo di esercizio non varia significativamente col diametro (in quanto il dispendio energetico risulta fissato dalle quote dei serbatoi e dall'assenza di impianti di sollevamento in rete) per cui è lecito considerare il solo costo di costruzione. Questo viene generalmente espresso in funzione del diametro in forma polinomiale: $C = aD + bD^2 + cD^3...$, ma per materiali come l'acciaio o la ghisa, e per diametri nell'ambito di quelli ordinari (fino a 60 cm) la legge lineare è già perfettamente significativa: infatti, considerando il costo proporzionale al peso della tubazione, a parità di pressione nominale lo spessore aumenta proporzionalmente al diametro, ed il peso, e quindi il costo, proporzionalmente al suo quadrato. In realtà per i materiali citati e le pressioni ordinarie, gli spessori minimi necessari alle lavorazioni portano a pressioni nominali molto superiori a quelle richieste nelle usuali applicazioni acquedottistiche (40-50 bar contro i 10 usualmente richiesti nelle reti di distribuzione), per cui all'aumentare del diametro si accetta una diminuzione della pressione nominale (le tubazioni in ghisa sferoidale a catalogo del maggiore produttore italiano hanno pressione nominale che si riduce dai 50 bar del DN 60 mm ai 40 bar del DN 600 mm) a fronte di un aumento dello spessore meno che lineare. Inoltre i costi fissi di lavorazione e di posa in opera fanno avvicinare ancor più la funzione di costo alla proporzionalità diretta. A titolo di esempio si riporta in fig. 3.1 l'andamento del costo in opera di tubazioni di acciaio con rivestimento interno di bitume centrifugato come rilevato da una media di alcuni prezzi di capitolato per lavori recentemente eseguiti. Come si vede, l'interpolazione lineare indicata in figura rappresenta molto bene l'anda-

mento. Andamenti simili si ottengono per tutti i materiali più usati nelle applicazioni acquedottistiche.

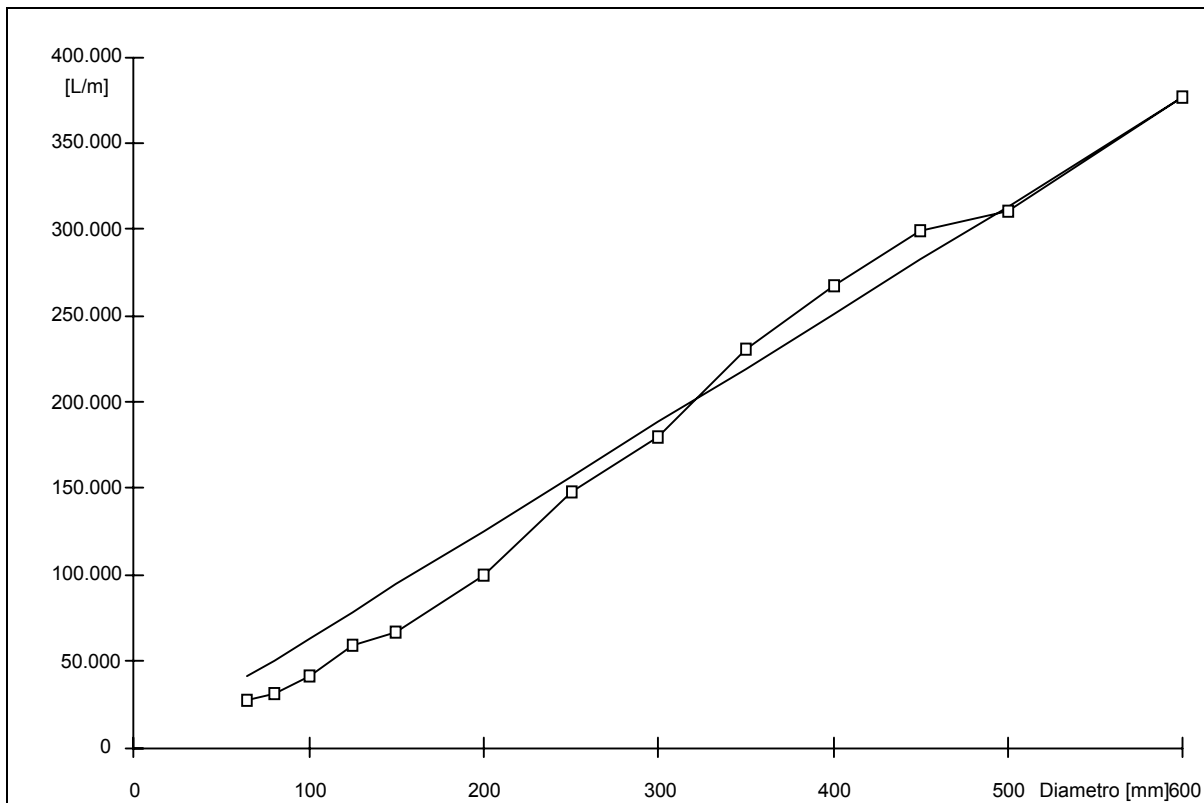


Fig. 3.1 - Costo in opera di tubazioni d'acciaio rivestite in bitume centrifugato

Prendiamo quindi in considerazione la configurazione di fig. 3.2: la rete è semplicemente costituita da due condotte lunghe rispettivamente 100 e 200 m disposte in serie e percorse da una portata di $0,1 \text{ m}^3/\text{s}$ e con una massima perdita di carico ammessa di 50 m.

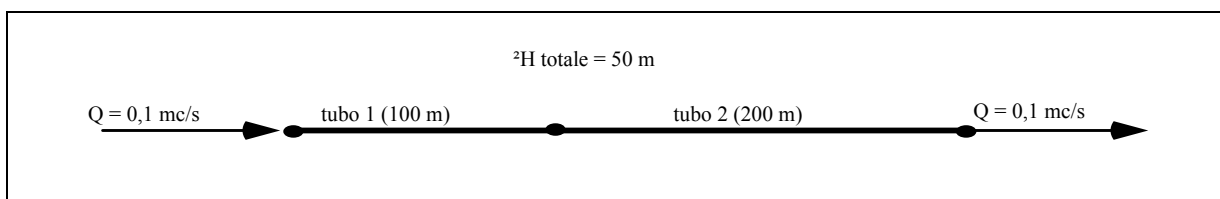


Fig. 3.2 - Esempio n. 1: configurazione di rete

Si consideri una formula di resistenza monomia, come quella di Scimemi $j = 0,000984 Q^{1,78}/D^{4,78}$ precedentemente citata. Siccome i diametri da dimensionare sono solo due, il problema può essere visualizzato sul piano cartesiano formato dal campo di variabilità del diametro delle due condotte (fig. 3.3).

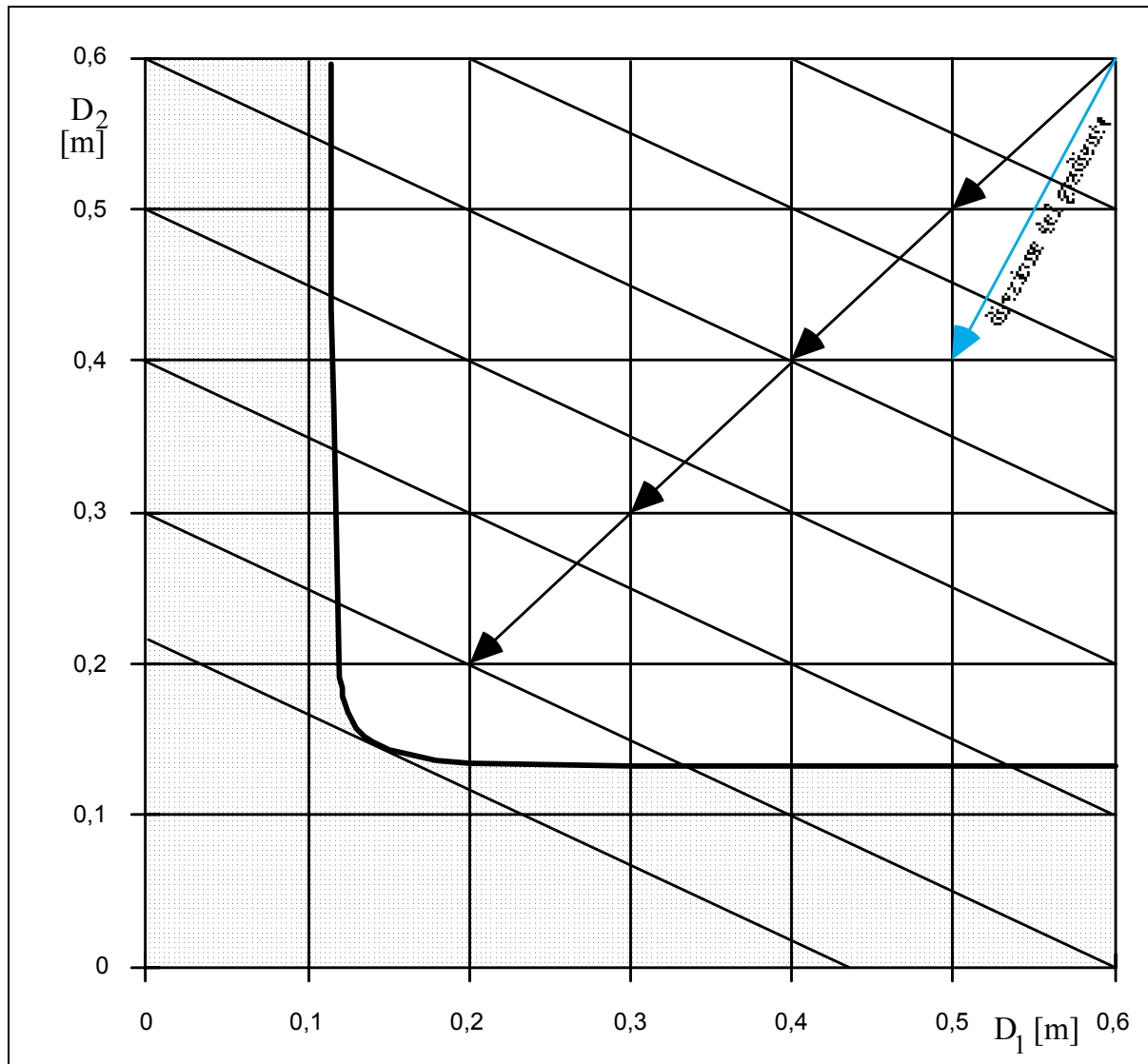


Fig. 3.3 - Esempio n. 1: rappresentazione cartesiana del problema di ricerca del minimo

Applicando la formula di Scimemi ad entrambe le condotte otteniamo:

$$\Delta h_1 = L_1 \cdot 0,000984 \cdot \frac{Q^{1,78}}{D_1^{4,78}}$$

$$\Delta h_2 = L_2 \cdot 0,000984 \cdot \frac{Q^{1,78}}{D_2^{4,78}}$$

che possono essere unite tenendo conto che $\Delta h = \Delta h_1 + \Delta h_2 = 50 \text{ m}$. Esplicitando D_2 in funzione di D_1 ricaviamo:

$$D_2 = \left(\frac{L_2 \cdot 0,000984 \cdot Q^{1,78}}{\Delta h - L_1 \cdot 0,000984 \cdot \frac{Q^{1,78}}{D_1^{4,78}}} \right)^{\frac{1}{4,78}}$$

che rappresenta il luogo delle coppie di diametri che danno luogo esattamente alla perdita ri-

chiesta di 50 m. Questa curva viene riportata in fig. 3.3 e delimita il luogo delle possibili configurazioni che soddisfano le condizioni di vincolo (la zona puntinata dà luogo a una perdita di carico superiore a 50 m e quindi non soddisfa le condizioni di vincolo).

Sul piano cartesiano possiamo anche rappresentare le curve di livello del campo scalare obiettivo (il costo totale del sistema). Nell'ipotesi di proporzionalità diretta e se le due condotte sono dello stesso materiale, il costo sarà dato da $C = k L_1 D_1 + k L_2 D_2$ e le curve di livello saranno semplicemente delle rette, il cui andamento è visualizzato in fig. 3.3. La soluzione ricercata è dunque quella rappresentata dal punto di tangenza fra la retta del costo minimo e la curva che delimita il campo di validità delle condizioni di vincolo, che corrisponde ad un diametro di circa 0,15 m, uguale per entrambe le condotte come era intuitivo aspettarsi. Supponendo che i diametri discretizzati siano quelli rappresentati dalle linee orizzontali e verticali in figura, la soluzione "commerciale" più economica risulta essere quella con entrambi i diametri pari a 0,2 m.

Per ricercare tale soluzione, il metodo del gradiente discreto opera come di seguito:

- si pongono entrambi i diametri pari al massimo valore (nel nostro caso 0,6 m);
- si verifica che sia rispettata la perdita di carico massima ammessa;
- si riducono di un'unità i diametri dei tronchi per i quali la derivata del costo forma col gradiente un angolo inferiore ad un valore di raffronto, ad esempio 75° (l'angolo di raffronto è una sorta di parametro di taratura il cui criterio di scelta sarà commentato nel prosieguo). In pratica si opera valutando il coseno dell'angolo fra il gradiente e ciascuna delle sue componenti i (le derivate parziali della funzione costo C rispetto a ciascun diametro) con la formula:

$$\cos \theta_i = \frac{\left| \frac{\partial C}{\partial D_i} \right|}{|\nabla C|}$$

e poi confrontandolo col coseno dell'angolo di raffronto. Nel nostro caso entrambe le componenti formano col gradiente un angolo inferiore a quello di raffronto, per cui entrambi i diametri possono essere diminuiti di un'unità;

- si verifica nuovamente la rete, e si procede nella riduzione fino a quando la rete non risulterà più verificata, cosa che nel nostro caso avverrà per entrambi i diametri pari a 0,1 m;
- riprendendo l'ultima configurazione verificata, si prova a ridurre i diametri uno per volta, partendo da quello che forma l'angolo minore con la direzione del gradiente (nel nostro caso entrambe le riduzioni porteranno ad uscire dal campo di validità delle condizioni di vincolo);
- l'ultima configurazione verificata (nel nostro caso entrambi i diametri pari a 0,2 m) rappresenta quella di minimo costo.

Come si vede, il procedimento, estremamente semplice e logico, giunge alla soluzione in sei

tentativi: quattro riduzioni simultanee di entrambi i diametri e due riduzioni finali di un diametro per volta per verifica. Si noti che le riduzioni simultanee di più diametri consentono di avvicinarsi rapidamente all'obiettivo anche nel caso in cui i tronchi da dimensionare siano molto numerosi. Di contro, il metodo richiede di verificare ripetutamente la rete, cosa che per reti estese ed a maglie chiuse (in cui è necessario il procedimento iterativo) potrebbe allungare eccessivamente i tempi di calcolo. Ciò in realtà, come si vedrà meglio nel prosieguo, non rappresenta un grosso problema, in quanto il risultato finale di una verifica rappresenta un'ottima configurazione di primo tentativo per la verifica successiva, che richiederà quindi un numero di iterazioni molto limitato per giungere a convergenza.

Il secondo esempio viene presentato in fig. 3.4: la rete è ancora costituita da due condotte lunghe 100 e 200 m, ma questa volta disposte in parallelo in modo da costituire una maglia chiusa. Si ha una portata entrante di 1 mc/s che si ripartisce nelle due condotte ed esce dall'altro nodo provocando una perdita di carico che non deve essere superiore a 2 m.

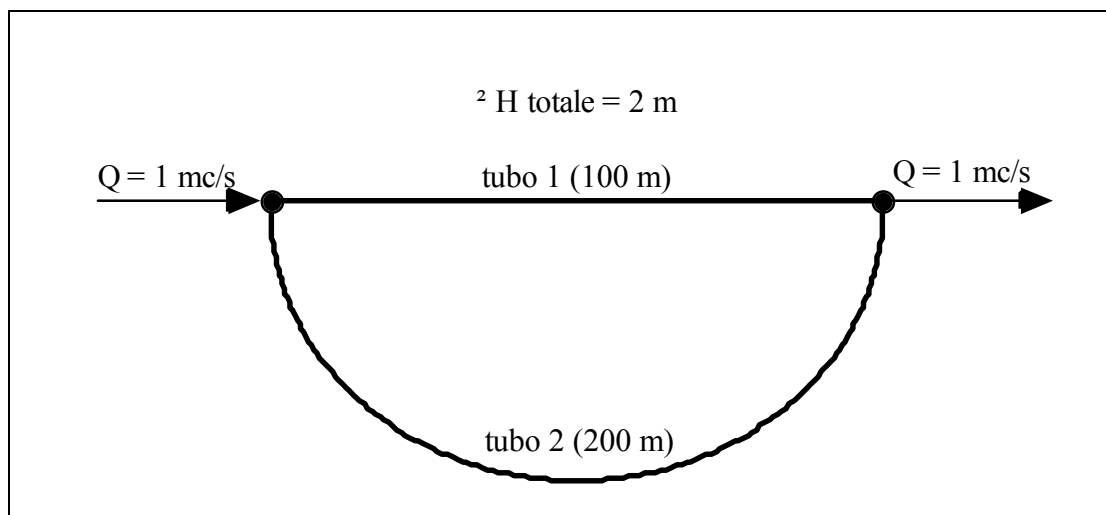


Figura 3.4 - Esempio n. 2: configurazione di rete

In questo caso è possibile esplicitare attraverso le formule di resistenza la portata Q_1 in funzione di D_1 e quindi, siccome $Q_1 + Q_2 = Q = 1 \text{ mc/s}$, ricavare D_2 in funzione di D_1 analogamente a quanto visto nell'esempio precedente. Riportando la curva sul grafico cartesiano costituito dai diametri variabili D_1 e D_2 si ottiene la situazione illustrata in fig. 3.5, in cui, di nuovo, vengono pure riportate le curve di livello della funzione costo, uguali a quelle del caso precedente.

Si nota subito che la configurazione di minimo costo prevede l'annullarsi di uno dei due diametri (quello della condotta di maggiore lunghezza). La significatività di questa caratteristica (che a prima vista potrebbe sembrare banale) sta nel fatto che è generalizzabile a tutte le reti a maglie chiuse, come è stato mostrato analiticamente (Cenedese e Mele, 1976). È proprio in virtù di questa considerazione che alcuni dei metodi di ricerca della soluzione ottimale sem-

plificano il problema trasformando le reti magliate in reti aperte mediante eliminazione dei tratti a portata minima di ciascuna maglia ed ottenendo una configurazione in cui le portate dei tronchi possono essere calcolate a priori anziché costituire anch'esse delle incognite.

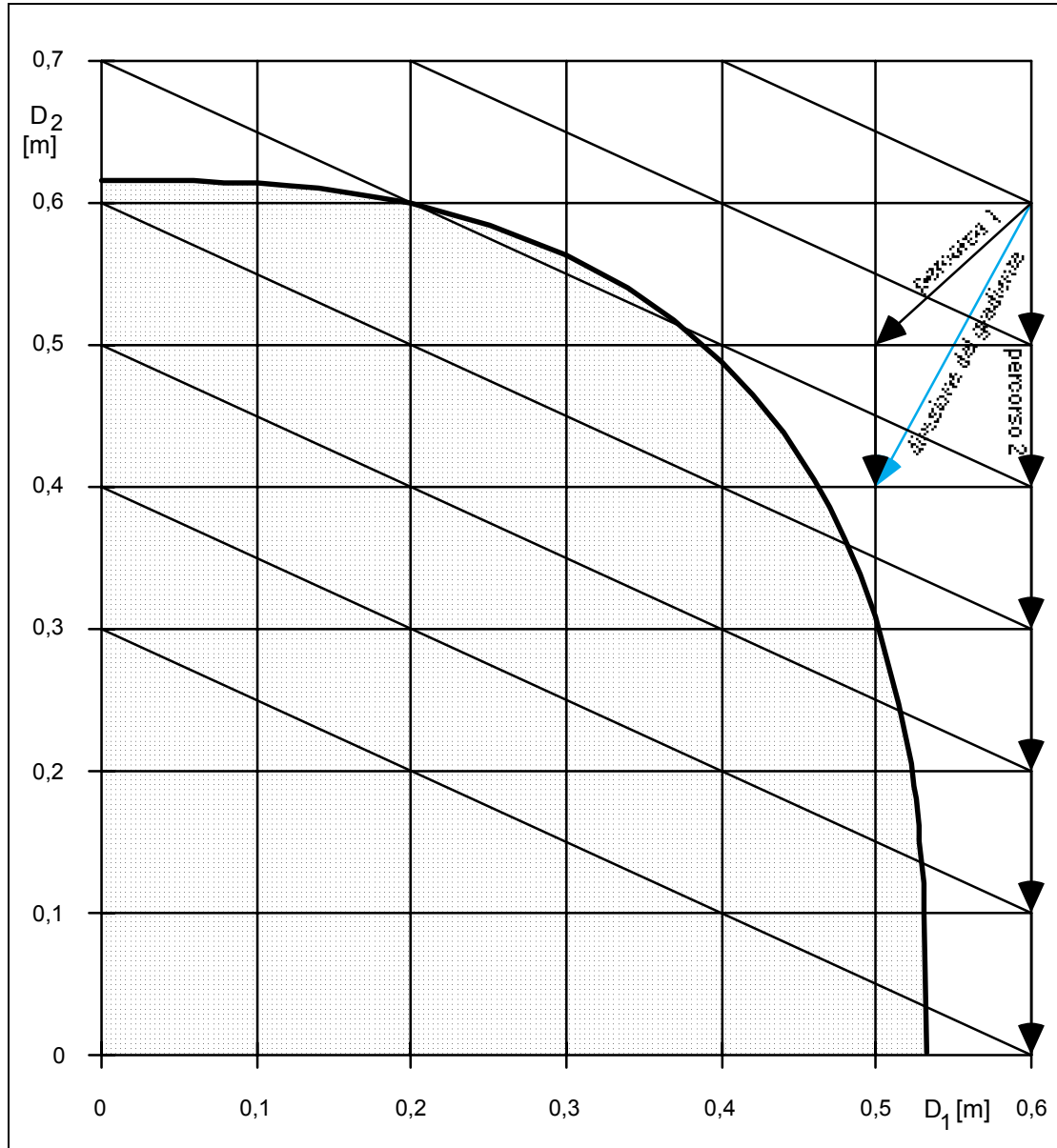


Fig. 3.5 - Esempio n. 2: rappresentazione cartesiana del problema di ricerca del minimo

Il procedimento di ricerca del minimo col metodo del gradiente discreto è il seguente: anche in questo caso si parte dalla configurazione con entrambi i diametri pari al massimo valore e poi, seguendo il percorso 1 indicato in figura, si raggiunge la configurazione 0,5 - 0,5. Verificato poi che la 0,4 - 0,4 non soddisfa le condizioni di vincolo, verrebbero diminuiti prima D_1 e poi D_2 . In entrambi i casi la rete non sarebbe verificata, per cui il processo risulterebbe intrappolato da una configurazione che non rappresenta il minimo.

Il minimo sarebbe invece stato raggiunto se l'angolo di confronto non fosse stato di 75° , ma

inferiore (ad esempio 60°): in questo caso sarebbe stato seguito il percorso 2 che raggiunge rapidamente il minimo. Ciò evidenzia il problema della difficoltà di raggiungere effettivamente il minimo: in questo semplice caso è sufficiente utilizzare un angolo di raffronto abbastanza piccolo, ma nei casi reali possono anche risultare necessari artifici numerici che richiedono un compromesso fra l'affidabilità della soluzione raggiunta ed i tempi di calcolo necessari. In ogni caso, tuttavia, l'incertezza riguarda solo il fatto che la soluzione individuata sia effettivamente la più economica possibile, essendo invece le condizioni di vincolo sicuramente verificate.

Per quanto riguarda il valore più adatto da attribuire all'angolo di confronto, si vede che valori maggiori riducono il numero di iterazioni necessarie, mentre valori minori sono più affidabili nel raggiungere il minimo della funzione costo. Da alcune prove effettuate, il valore di 60° suggerito da Lam pare rappresentare il miglior compromesso per problemi a due dimensioni (due diametri da dimensionare). Per un problema ad n dimensioni sembra invece più opportuno generalizzare tale valore mediante la:

$$\alpha = \frac{2}{3} \cdot \arctan(\sqrt{n-1}) + 30^\circ$$

che può essere interpretata come:

$$\alpha = \text{angolo medio} + 1/3 (90^\circ - \text{angolo medio})$$

che per $n = 2$ si riconduce appunto al valore di 60° .

Tornando alla considerazione che la rete più economica è sempre del tipo ad albero, si può osservare che le caratteristiche di funzionalità delle reti magliate che le rendono preferibili rispetto a quelle ad albero non vengono prese in considerazione nel processo di ottimizzazione. La caratteristica principale è il grado di "ridondanza", ovvero la capacità della rete di mantenere un certo grado di funzionalità anche in caso di interruzione di uno o più tronchi.

Generalmente si risolve il problema chiudendo le maglie della soluzione ad albero individuata con tronchi del diametro minimo. Nel secondo esempio, si adotterebbero i valori $D_1 = 0,6 \text{ m}$ e $D_2 = 0,1 \text{ m}$. Si può misurare il grado di ridondanza di questa soluzione valutando la portata convogliabile dal solo tronco di diametro inferiore a parità di perdita di carico (2m): tale portata sarebbe di 7,6 l/s, cioè appena lo 0,76% della portata di dimensionamento. Considerando invece, ad esempio, la configurazione con $D_1 = 0,5 \text{ m}$ e $D_2 = 0,4 \text{ m}$, che soddisfa egualmente le condizioni di vincolo ma è più costosa della precedente, si ha una portata convogliabile dal solo tronco 2 di 310 l/s, con una ridondanza del 31% sulla portata di dimensionamento.

Ciò porta a concludere che il criterio di minimizzare il costo con le sole condizioni di vincolo sulle piezometriche, pone a confronto soluzioni diverse dal punto di vista della funzionalità, conducendo alla scelta della soluzione che al costo minimo associa anche il più basso grado di funzionalità (la ridondanza nulla, appunto, che corrisponde alla configurazione ad albero di minimo costo). In realtà il grado di ridondanza rappresenta una condizione di vincolo altret-

tanto desiderabile, ed il fatto che si ricerchino configurazioni a maglie chiuse ne è una chiara conferma.

Nella normale pratica progettuale la soluzione individuata viene pertanto sottoposta a varie verifiche - fra cui quella di funzionalità in caso di interruzione di uno o più tronchi principali [2] - ed eventualmente corretta. Tuttavia questi correttivi apportati a conclusione del processo di minimizzazione del costo, quanto più sono rilevanti tanto più possono inficiare il processo di minimizzazione stesso, nel senso che la soluzione individuata potrebbe non essere la più economica fra quelle che soddisfano anche queste condizioni di vincolo supplementari.

E' chiaro che la moltitudine di fattori che entra in gioco nei problemi reali richiede inevitabilmente alcune approssimazioni delle tecniche progettuali che vengono suggerite dallo studio teorico, ma alcuni indirizzi possono comunque essere tratti dalle considerazioni suindicate:

- il problema, comune a tutte le tecniche di minimizzazione proposte, della non certezza di individuare il minimo assoluto fra tutti i possibili minimi relativi, è da ridimensionare nella sua gravità, in quanto altri fattori che entrano in gioco nel processo progettuale possono diventare preponderanti rispetto ad esso;
- potrebbe essere invece interessante arricchire l'insieme di condizioni di vincolo utilizzate durante il processo di minimizzazione almeno di quelle che vengono sistematicamente verificate al termine di ogni progetto di rete idrica.

Da questo punto di vista il metodo del gradiente discreto sembra essere particolarmente interessante in quanto, richiedendo semplicemente la verifica delle condizioni di vincolo, piuttosto che la loro introduzione nella funzione obiettivo, lascia la massima libertà nella loro definizione.

4. L'IMPLEMENTAZIONE DEL CODICE DI CALCOLO

L'implementazione del codice di calcolo sin qui presentato è riportata in fig. 4.1: come viene meglio descritto ai successivi paragrafi, la struttura è di tipo tradizionale per quanto riguarda l'algoritmo risolutivo vero e proprio, mentre viene utilizzata la tecnica orientata agli oggetti per quanto riguarda il modulo di verifica della rete, ed in particolare la descrizione delle leggi q-h dei vari dispositivi di rete.

Le principali caratteristiche del programma, in termini di strutturazione del codice di calcolo, vengono illustrate in questo capitolo, riportando anche alcuni stralci di codice ad esemplificare le caratteristiche peculiari.

Le modalità di utilizzo del programma, nonché un cenno alle prove di validazione e messa a punto realizzate presso il Centro di Ricerca sulle Acque dell'Università di Pavia verranno invece presentate nei capitoli conclusivi.

4.1 Implementazione “object oriented”

Caratteristica del programma, che è sviluppato in linguaggio C++, è di sfruttare l'implementazione per classi ed oggetti (OOP, programmazione orientata agli oggetti) tipica di questo linguaggio di programmazione. In particolare, questa struttura è stata utilizzata nell'implementazione delle leggi q-h dei vari dispositivi implementati.

La strutturazione per classi consente di “incapsulare” le caratteristiche di implementazione delle varie parti, escludendole completamente dal resto del codice, e quindi rendendole semplici da comprendere e da potenziare con successivi interventi.

I vantaggi della programmazione ad oggetti sono essenzialmente legati alle caratteristiche dette “ereditarietà” e “polimorfismo”: ogni classe incorpora le strutture dei dati ad essa necessarie, ed anche le procedure (dette “metodi”) richieste per operare su quella classe. Da una classe è possibile costruirne altre derivate (*ereditarietà*), che ereditano dalla prima (detta “classe base”) tutte le sue strutture dati e i suoi metodi, ai quali potranno essere aggiunti o sostituiti quelli peculiari della classe derivata.

Per utilizzare le classi devono essere allocati degli “oggetti” appartenenti a quella classe. Ad esempio, vi sarà una classe “condotte”, dalla quale derivare le classi “condotte legge Darcy”, “condotte legge Colebrook” ecc. Per ogni condotta facente parte della struttura sarà creato un oggetto di classe “condotte”, che però potrà essere diversificato di caso in caso a seconda della classe derivata di appartenenza..

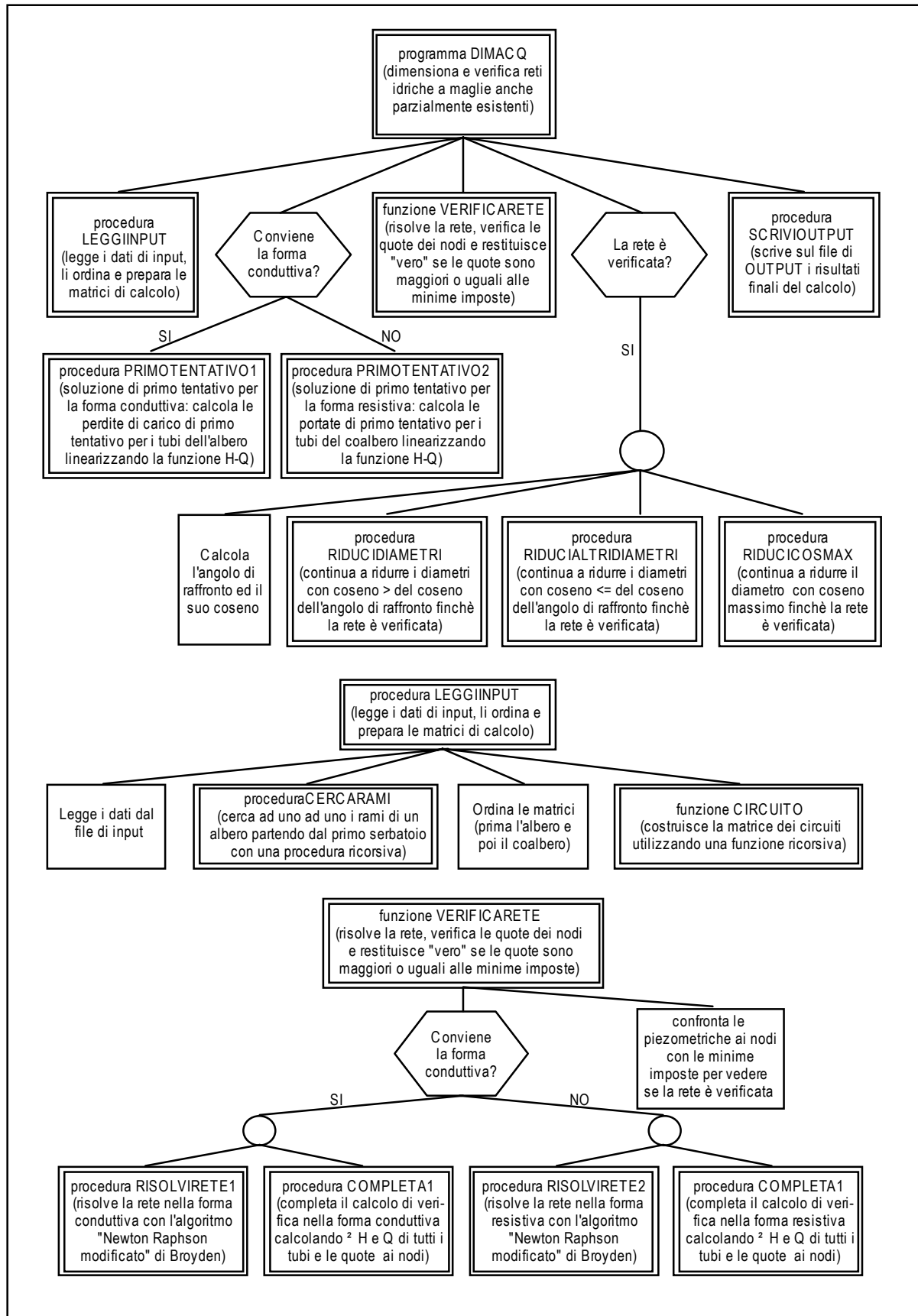


Figura 4.1 - Diagramma di struttura del programma di calcolo

In questo modo, dovendo operare su una sequenza di oggetti appartenenti a classi differenziate, ma derivate da una classe comune, richiamando un metodo della classe di base si otterrà, per ciascun oggetto, un comportamento diversificato a seconda della classe di ognuno, ma finalizzato alla realizzazione dello stesso obiettivo (*polimorfismo*).

Nel caso specifico, è stata realizzata una classe “astratta” denominata “CLink” che comprende tutte le possibili giunzioni fra due nodi (C sta per “classe”, Link sta per “giunzione”).

Col termine “astratta” si intende una classe che ha il solo scopo di essere la base per derivare altre classi: da essa non possono essere direttamente allocati oggetti: questi verranno allocati solo dalle classi derivate. Una classe astratta ha quindi lo scopo di raccogliere caratteristiche comuni ad altre classi, che verranno successivamente derivate da essa.

La classe “CLink” incorpora come dati gli identificativi dei due nodi di estremità, oltre al nome della condotta. Come metodi incorpora tutti quelli che verranno richiamati nel corso del programma, in una forma di default che dovrà essere sostituita nelle varie implementazioni specifiche delle classi derivate. Vi sono quindi procedure di input e output, di calcolo di h da q, di calcolo da q ad h, di approssimazione lineare delle due funzioni per il primo tentativo, ecc. Questi metodi vengono dichiarati come “virtuali” (ovvero che devono essere modificati dalle classi derivate). Una classe è “astratta” quando tutti i suoi metodi sono “virtuali”.

Dalla classe CLink vengono derivate la classe astratta “CTubo” che rappresenta le condotte, e la classe “CPompa” che rappresenta le pompe con curva caratteristica. Quest’ultima conterrà, fra i suoi dati, i punti necessari alla costruzione della curva caratteristica, mentre la classe sovrascriverà le procedure di input, output e calcolo della sua classe base “CLink”. La classe “CTubi” sarà invece ancora astratta, in quanto da essa verranno derivate altre classi per implementare le diverse possibili leggi di resistenza. Questa classe aggiungerà ai dati di “CLink” il diametro e la lunghezza della condotta.

Infine da “CTubo” vengono derivate le classi “CDarcy”, “CScimemi”, “CColebrook” ecc. che implementano le diverse leggi di resistenza. Si noti che queste classi hanno un numero di parametri di input diversificati l’una dall’altra, cui corrisponde un tracciato record diverso nel file di input. Questo tipo di gestione risulterebbe assai più complesso in un linguaggio di programmazione di tipo tradizionale.

Dovendo intervenire successivamente sul codice, ad esempio per aggiungere nuovi tipi di condotte o di apparecchi idraulici di rete, sarà sufficiente derivare una nuova classe per gestire le nuove funzionalità, scrivendone la nuova struttura dati e i nuovi metodi, ma ereditando dalla classe base i dati ed i metodi che non richiedono modifiche. Questa operazione risulterà semplificata dal fatto che non si dovrà esaminare il codice di tutto il programma, ma solo quello della classe base.

In C++ ogni classe, ogni dato (semplice o strutturato) ed ogni procedura (o metodo) deve essere “dichiarato” prima di essere implementato od utilizzato. Ciò è anche utile per la successiva analisi del codice, in quanto esaminando le dichiarazioni si potrà capire la funzione dei

dati e dei metodi senza entrare nel dettaglio del codice implementativo dei metodi stessi. Per sfruttare ancor più questa caratteristica, nei codici estesi è opportuno raccogliere tutte le dichiarazioni in files separati, che potranno successivamente essere esaminati per comprendere la struttura e la funzione dei vari metodi. Si riporta di seguito la porzione di codice con la dichiarazione della gerarchia di classi derivate da CLink, di cui si è detto (per sinteticità si omettono le dichiarazioni di alcuni metodi meno significativi).

```
class CLink //abstract class che rappresenta le giunzioni fra due nodi
//caratterizzate da una legge H-Q (es.: tubi, pompe, valvole, ecc.).
// Tutti i metodi sono virtuali e quindi devono essere sovrascritti.
{public:
virtual void input() = 0; //gestisce l'input da file dei dati necessari al link
virtual void output() = 0; //gestisce l'output su file dei dati finali
virtual void outnodi(signed char); //gestisce l'output su file di nome link, nodo1 e nodo2
virtual void setcoeff(signed char formaconduttiva) = 0;
//calcola i coeff. caratteristici della legge H-Q
virtual double Ylin() = 0; //fornisce Ylin approssimando linearmente la legge-Q-H
virtual double Rlin() = 0; //fornisce Rlin approssimando linearmente la legge H-Q
virtual double Z() = 0; //fornisce Z approssimando linearmente la legge-Q-H
virtual double S() = 0; //fornisce S approssimando linearmente la legge H-Q
virtual double H(double Qin, double Hold) = 0; //calcola H=f(Q) con la legge di
//resistenza completa. Hold è richiesto come valore di 1° tentativo per formule
//iterative tipo tipo Colebrook)
virtual double Q(double Hin) = 0; //calcola Q=f(H) con la legge di resistenza completa
//metodi di accesso ai dati protetti
virtual void setD(double){cerr<<"tentativo di setD in un link non tubo";exit(-1);}
virtual double D(){cerr<<"richiesta di D in un link non tubo";exit(-1);return(-1);}
//per brevità si omettono gli altri metodi di accesso ai dati protetti
protected:
int nodo1val,nodo2val;
char nomeval[11];
};

class CTubo: public CLink //abstract class derivata da CLink che rappresenta tubi
//circolari con D ed L, da cui derivare classi per varie leggi di resistenza
{public:
//metodi di accesso ai dati protetti
void setD(double newD); //fissa il diametro e calcola i coeff.
inline double D() {return Dval;} //restituisce il valore attuale del diametro;
//per brevità si omettono gli altri metodi di accesso ai dati protetti
protected:
double Dval; //diametro del tubo [m]
double Lval; //lunghezza del tubo [m]
};

class CDarcy: public CTubo //classe di tubi con legge di resistenza di Darcy
{public:
void input(); //gestisce l'input da file dei dati necessari al link
void output(); //gestisce l'output su file dei dati finali contenuti nel link
void setcoeff(signed char formaconduttiva); //calcola i coefficienti della legge H-Q
double Ylin(); //fornisce Ylin approssimando linearmente Y nei dintorni di V0
double Rlin(); //fornisce Rlin approssimando linearmente R nei dintorni di V0
double H(double Qin, double Hold); //calcola H=f(Q) con la legge di resistenza completa
double Q(double Hin); //calcola Q=f(H) con la legge di resistenza completa
private:
double n; //moltiplicatore di beta (1 = tubi nuovi, 0 = tubi usati)
double Y,R; // Q = R * sqrt(H) (forma conduttiva); H = R * sqr(Q) (forma resistiva)
};

class CScimemi: public CTubo //classe di tubi con legge di resistenza di Scimemi
{public:
void input(); //gestisce l'input da file dei dati necessari al link
```



```

void output();          //gestisce l'output su file dei dati finali contenuti nel link
void setcoeff(signed char formaconduttiva);//calcola i coefficienti della legge H-Q
double Ylin();//fornisce Ylin approssimando linearmente Y nei dintorni di V0
double Rlin();//fornisce Rlin approssimando linearmente R nei dintorni di V0
double H(double Qin, double Hold);//calcola H=f(Q) con la legge di resistenza completa
double Q(double Hin);//calcola Q=f(H) con la legge di resistenza completa
private:
double Y,R; //  $Q = R * \sqrt{H}$  (forma conduttiva);  $H = R * \text{sqr}(Q)$  (forma resistiva)
};

class CScimVer: public CTubo//classe di tubi con legge di resistenza di Scimemi-Veronese
{public:
//si omettono i metodi, che sono analoghi a quelli della classe CScimemi
private:
double Y,R; //  $Q = R * \sqrt{H}$  (forma conduttiva);  $H = R * \text{sqr}(Q)$  (forma resistiva)
};

class CDeMarMar: public CTubo//classe di tubi con legge di resistenza di De Marchi-Marchetti
{public:
//si omettono i metodi, che sono analoghi a quelli della classe CScimemi
private:
double Y,R; //  $Q = R * \sqrt{H}$  (forma conduttiva);  $H = R * \text{sqr}(Q)$  (forma resistiva)
};

class CGeneric: public CTubo//classe di tubi con legge di resistenza monomia generica
{public:
void input();//gestisce l'input da file dei dati necessari al link
void output();          //gestisce l'output su file dei dati finali contenuti nel link
void setcoeff(signed char formaconduttiva);//calcola i coefficienti della legge H-Q
double Ylin();//fornisce Ylin approssimando linearmente Y nei dintorni di V0
double Rlin();//fornisce Rlin approssimando linearmente R nei dintorni di V0
double H(double Qin, double Hold);//calcola H=f(Q) con la legge di resistenza completa
double Q(double Hin);//calcola Q=f(H) con la legge di resistenza completa
private:
double alfa,beta,gamma;//coefficienti di  $H = \text{beta} * Q^{\text{alfa}} / D^{\text{gamma}}$ 
double Y,R; //  $Q = R * H^{(1/\text{alfa})}$  (forma conduttiva);  $H = R * Q^{\text{alfa}}$  (forma resistiva)
};

class CColebrook: public CTubo//classe di tubi con legge di resistenza monomia generica
{public:
void input();//gestisce l'input da file dei dati necessari al link
void output();          //gestisce l'output su file dei dati finali contenuti nel link
void setcoeff(signed char formaconduttiva);//calcola i coefficienti della legge H-Q
double Ylin();//fornisce Ylin approssimando linearmente Y nei dintorni di V0
double Rlin();//fornisce Rlin approssimando linearmente R nei dintorni di V0
double H(double Qin, double Hold);//calcola H=f(Q) con la legge di resistenza completa
double Q(double Hin);//calcola Q=f(H) con la legge di resistenza completa
private:
double eps,csi,nu; //parametri della formula
double A,B,C,E; // costanti di calcolo (indipendenti da Q e H)
};

class CPompa: public CLink//classe derivata da CLink che rappresenta le pompe
//con curva caratteristica descritta da tre punti di funzionamento
{public:
void input();//gestisce l'input da file dei dati necessari al link
void output();          //gestisce output su file dei dati finali
char range(double Q,double H);//se fuori campo funzionam. fornisce stringa di avviso
void setcoeff(signed char formaconduttiva);//calcola i coeff. della legge H-Q
double Ylin();//fornisce Ylin approssimando linearmente la curva caratteristica

```

```

double Rlin();//fornisce Rlin approssimando linearmente la curva caratteristica
double Z();//fornisce Z approssimando linearmente la curva caratteristica
double S();//fornisce S approssimando linearmente la curva caratteristica
double H(double Qin, double Hold);//calcola H=f(Q) con legge di resistenza completa
double Q(double Hin);//calcola Q=f(H) con la legge di resistenza completa
private:
double Q1, Q2, Q3; //portate nei tre punti di funzionamento
double H1, H2, H3; //prevalenze nei tre punti di funzionamento
double a, b, c, a1, b1; //coefficienti del polinomio approssimante
};

```

Si può notare che in ogni classe vi è una parte “pubblica” ed una “privata”: in quest’ultima sono elencati i dati a cui non si può accedere se non utilizzando metodi presenti nella parte “pubblica”. In questo programma tutti i dati sono privati, e quindi richiedono l’accesso attraverso gli appositi metodi. Ciò per preservare l’integrità dei dati e quindi della struttura stessa del programma.

4.2 Espandibilità del codice

Con la strutturazione del codice sopra definita, risulta molto agevole arricchire il programma con nuove classi derivate: ad esempio, volendo implementare una diversa legge di resistenza, si potrà derivare una nuova classe da “CtTubo” o da una delle classi da essa derivate, aggiungendo le procedure e i dati necessari alla classe specifica ed eventualmente sovrascrivendo le procedure da modificare. Le procedure e i dati già adeguati (ad esempio il diametro, la lunghezza, il nome, gli identificativi dei nodi di estremità) potranno direttamente essere ereditati dalla classe base.

4.3 Gestione della memoria

Un’altra caratteristica del programma è l’allocazione dinamica della memoria: tutte le grosse matrici utilizzate dal programma sono in realtà degli oggetti che vengono allocati (mediante un puntatore alla locazione di memoria iniziale) durante l’esecuzione del programma, una volta che ne siano state identificate le dimensioni. Pertanto non sarà necessario dimensionare le matrici in fase di compilazione.

In questo modo il programma potrà funzionare con la sola memoria effettivamente necessaria: per risolvere una rete piccola potrà essere eseguito su un computer con poca memoria RAM, ma lo stesso programma (senza richiedere ricompilazione) potrà risolvere reti molto grandi su un elaboratore con adeguata dotazione di memoria RAM.

Si riporta di seguito il codice di dichiarazione della gerarchia di classi creata per gestire l’allocazione dinamica di matrici monodimensionali e bidimensionali.

```

//Header file "Array2D.h":
//Definisce le classi "Array1D" e "Array2D" che consentono di allocare
//dinamicamente matrici di dimensioni definibili durante l'esecuzione del programma.
//Viene inoltre definita la funzione "MemoriaEsaurita" che interrompe il programma
se
//non c'è spazio sufficiente sulla heap per allocare le matrici dinamiche.

```

```

void MemoriaEsaurita();

template<class T> class Array1D
{public:
    Array1D ();
    Array1D (int new_max);
    ~Array1D ();
    void alloc (int new_max);
    void scrivi (ofstream& file);
    void leggi (ifstream& file);
    inline T& operator [] (int ind);
private:
    int max;
    T *dati;};

template<class T> class Array2D
{public:
    Array2D ();
    Array2D (int new_max_x, int new_max_y);
    ~Array2D ();
    void alloc (int new_max_x, int new_max_y);
    void scrivi (ofstream& file);
    void leggi (ifstream& file);
    Array1D<T>& operator [] (int ind);
private:
    int max;
    Array1D<T> **righe;};

template<class T> inline T& Array1D<T>::operator[] (int ind) {return dati[ind];}

template<class T> inline Array1D<T>& Array2D<T>::operator[] (int ind) {return
*righe[ind];}

```

L'implementazione è riportata di seguito:

```

// Implementation file "Array2D.cp"
#include <new.h>
#include <iostream.h>
#include <string.h>
#include <stdlib.h>
#include "Array2D.h"

void MemoriaEsaurita()
{cerr<<"memoria insufficiente! Aumentare la memoria dell'applicazione.\n"; exit(-
1);}

//implementazione metodi di Array1D
template<class T> Array1D<T>::Array1D () {}

template<class T> Array1D<T>::Array1D (int new_max)
{alloc(new_max);}

template<class T> void Array1D<T>::alloc (int new_max)
{max = new_max;dati = new T[new_max];}

template<class T> void Array1D<T>::scrivi (ofstream& file)
{file.write((unsigned char*) dati, max*sizeof(T));}

template<class T> void Array1D<T>::leggi (ifstream& file)
{file.read((unsigned char*) dati, max*sizeof(T));}

template<class T> Array1D<T>::~Array1D ()
{delete [] dati;}

//Implementazione metodi di Array2D
template<class T> Array2D<T>::Array2D () {}

template<class T> Array2D<T>::Array2D (int new_max_x, int new_max_y)
{    alloc(new_max_x,new_max_y);}

template<class T> void Array2D<T>::alloc (int new_max_x, int new_max_y)

```

```
{
    max = new_max_x;
    righe = new Array1D<T>*[max];
    for (register int i = 0; i<max; i++) righe[i] = new Array1D<T>(new_max_y);}

template<class T> void Array2D<T>::scrivi (ofstream& file)
{
    for (register int i = 0; i<max; i++) righe[i]->scrivi(file);}

template<class T> void Array2D<T>::leggi (ifstream& file)
{
    for (register int i = 0; i<max; i++) righe[i]->leggi(file);}

template<class T> Array2D<T>::~~Array2D ()
{
    for (register int i=0;i<max;i++) delete righe[i];
    delete righe;}
```

5. ISTRUZIONI PER L'UTILIZZO DEL CODICE

Il programma descritto nel presente lavoro è denominato “Dimacq”, ed è stato scritto e compilato con il compilatore “CodeWarrior C++” su piattaforme macintosh, PowerMacintosh e Windows 95. Il programma, è di pubblico dominio, e può essere richiesto all'autore nella sua versione aggiornata, sia come sorgente che compilato per le tre piattaforme citate, all'indirizzo *iannelli@polito.it*.

Si riportano di seguito le istruzioni operative per l'uso del programma.

Per utilizzare Dimacq bisogna preparare, utilizzando un text editor o un foglio di calcolo, il file di input (un file di testo con un nome qualsiasi) rispettando il formato spiegato nel seguito; inserirlo nella stessa directory del programma e quindi lanciare Dimacq.

Si aprirà una finestra che chiederà il nome del file di input, e quindi sarà eseguito il calcolo fornendo a video alcune informazioni sulle varie fasi dell'elaborazione.

Al termine verranno generati due files di testo e tre files binari, aventi il nome del file di input più un breve suffisso: .out, .ctr, .top, .pri, .fin.

I due file di testo (con suffissi .out e .ctr) possono essere aperti con qualsiasi text editor o foglio di calcolo per visualizzare i risultati.

Porre attenzione al fatto che i numeri vengono scritti col punto come separatore decimale. Pertanto utilizzando un foglio di calcolo che riconosca la virgola decimale si avrà una interpretazione scorretta.

Il file .out riporta il risultato dell'elaborazione, mentre il file .ctr riporta le informazioni sull'elaborazione mostrate a video durante l'esecuzione.

Sul contenuto dei tre file binari si tornerà nel seguito.

Il file deve essere scritto rispettando il formato descritto nel seguito. I campi devono essere separati da spazi o tabulatori; alla fine di ogni riga bisogna inserire un caporiga non preceduto da spazi o tabulatori.

E' obbligatorio rispettare la sequenza di inserimento (nodi - serbatoi - punti di erogazione - condotte - pompe - diametri commerciali - parametri del processo iterativo) inserendo una riga vuota dopo ogni gruppo e riportando tutte le righe di intestazione.

Prima dell'esecuzione il programma effettua alcune verifiche di congruità del file di input, e se individua errori nella topologia arresta l'esecuzione indicando i motivi dell'incongruenza che renderebbe impossibile il calcolo.

Si descrivono di seguito i tracciati da rispettare per ciascun gruppo di dati.

NODI: dopo le due righe di intestazione scrivere una riga per ogni nodo terminando con una

riga vuota. In ogni riga inserire:

- nome del nodo (max 16 caratteri alfanumerici senza spazi o tabulatori);
- piezometrica minima richiesta per ognuna delle configurazioni di verifica [m slm]

SERBATOI: dopo le due righe di intestazione scrivere una riga per ogni serbatoio terminando con una riga vuota. In ogni riga inserire:

- nome del nodo di posizionamento del serbatoio (uno dei nodi già inseriti);
- quota del pelo libero nel serbatoio [m slm].

PUNTI DI EROGAZIONE: dopo le due righe di intestazione scrivere una riga per ogni punto di erogazione terminando con una riga vuota. In ogni riga inserire:

- nome del nodo di posizionamento dell'erogazione (uno dei nodi già inseriti);
- portata erogata in ognuna delle configurazioni di verifica [l/s].

CONDOTTE: dopo le due righe di intestazione scrivere una riga per ogni condotta terminando con una riga vuota. In ogni riga inserire:

- tipo della condotta: un carattere alfanumerico scelto fra
 - D = Darcy (Ghisa senza rivestimento)
 - S = Scimemi (fibrocemento, ghisa con rivestimento cementizio)
 - V = Scimemi-Veronese (acciaio semplicemente bitumato)
 - M = De Marchi - Marchetti (acciaio con rivestimento di bitume centrifugato)
 - G = espressione monomia generica $j = \beta * Q^\alpha / D^\gamma$ (consente di adottare qualsiasi altra formula monomia, come ad esempio quella di Hazen - Williams)
 - C = Colebrook-White + perdite localizzate $csi * V^2 / (2g)$ (è la formula più precisa e completa, specie per tubazioni molto lisce come quelle plastiche, ed inoltre consente di trattare anche fluidi diversi dall'acqua; le perdite localizzate consentono anche di inserire dispositivi di dissipazione quali valvole parzialmente chiuse, od altro. Il calcolo risulta però più lento in quanto la formula richiede una valutazione iterativa)
- nome della condotta (max 10 caratteri senza spazi o tabulatori);
- nome del nodo iniziale;
- nome del nodo finale;
- diametro [mm]
- lunghezza [m]
- coefficienti: a seconda del tipo di condotta riportare:
 - D: un coeff. moltiplicativo del beta di Darcy (1 = tubi nuovi, 2 = tubi usati)
 - S: non sono necessari coefficienti
 - V: non sono necessari coefficienti
 - M: non sono necessari coefficienti
 - G: alfa, beta, gamma dell'espressione monomia di cui sopra
 - C: asperità media epsilon [mm], csi dell'espressione $csi * V^2 / 2g$, viscosità cinematica ν [m²/s] (= 1e-6 m²/s per l'acqua a 20°C)

POMPE: le pompe devono essere descritte indicando tre punti della curva caratteristica portata-prevalenza Q1-H1, Q2-H2, Q3-H3 (solitamente i due estremi della curva caratteristica e il punto di funzionamento al massimo rendimento).

Dopo le due righe di intestazione scrivere una riga per ogni pompa terminando con una riga vuota. In ogni riga inserire:

- nome della pompa (max 10 caratteri senza spazi o tabulatori);
- nome del nodo di prelievo;
- nome del nodo di restituzione;
- Q1 [l/s];
- H1 [m];
- Q2 [l/s];
- H2 [m];
- Q3 [l/s];
- H3 [m].

Se non si inseriscono pompe, bisogna comunque riportare le due righe di intestazione e la riga vuota finale.

DIAMETRI COMMERCIALI: sono i diametri fra cui l'algoritmo di ottimizzazione sceglie quelli di dimensionamento. Ovviamente, se si inseriscono meno diametri il calcolo sarà più rapido. Se viene richiesta solo la verifica non saranno usati (ma bisogna comunque inserirne almeno uno).

Dopo la riga di intestazione scrivere una riga contenente i diametri [mm] in ordine crescente separati da spazi o tabulatori. Dopo l'ultimo inserire un caporiga senza spazi o tabulatori e poi una riga vuota.

PARAMETRI DEL PROCESSO ITERATIVO: dopo le due righe di intestazione scrivere una riga contenente i seguenti tre numeri:

- tolQ [l/s] per la forma conduttiva: dopo ogni iterazione viene calcolata la norma delle portate dei tubi dell'albero ($\sqrt{\sum(Q_i^2)}$): se $<$ tolQ termina il calcolo;
- tolH [m] per la forma resistiva: dopo ogni iterazione viene calcolata la norma dei deltaH dei tubi del coalbero ($\sqrt{\sum(H_i^2)}$): se $<$ tolH termina il calcolo;
- n.max iter.: se dopo questo numero di iterazioni non è stata raggiunta la convergenza il calcolo viene interrotto stampando comunque i risultati.

Se il programma non riesce a raggiungere la convergenza dopo qualche decina (o centinaio per le reti più estese) di iterazioni, probabilmente il file di input contiene qualche errore od incongruenza che bisognerà individuare e correggere. E' stato osservato che ordinando le condotte per resistività crescente la convergenza risulta leggermente più rapida. (ipotizzando una legge di resistenza monomia $H = L * \beta * Q^\alpha / D^\gamma$, la resistività di una condotta è pari prodotto $L * \beta / D^\gamma$)

5.1 Significato dei files binari generati al termine delle iterazioni

Come già detto, al termine dell'esecuzione vengono generati tre files binari con suffisso .top, .pri, .fin. Questi files vengono utilizzati da Dimacq nel caso si ripeta l'elaborazione dello stesso file di dati. Il file .top contiene la descrizione della topologia della rete; il file .pri contiene il risultato del calcolo al termine del primo tentativo (approssimazione lineare) e il file .fin contiene il risultato del calcolo al termine del processo iterativo. Prima di utilizzare tali files, il programma effettua una verifica di congruenza e, se non soddisfatta, effettua nuovamente il calcolo. Tali files servono a velocizzare il ricalcolo di reti molto estese per elaborazioni successive in cui sia stato variato solo qualche diametro o lunghezza, o portata erogata, lasciando invariata la topologia. Ciò può ridurre i tempi di elaborazione fino a circa la metà. Se dopo un'elaborazione viene modificato il file .fin in .pri, alla successiva esecuzione di Dimacq il calcolo riprenderà dall'ultima iterazione dell'esecuzione precedente. Se il riutilizzo della topologia e del primo tentativo precedentemente generati comporta problemi, è sufficiente eliminare questi files dalla directory prima di eseguire Dimacq. In questo modo sarà effettuata l'elaborazione fin dall'inizio.

6. ESEMPI DI UTILIZZO DEL CODICE: TEST DI VERIFICA E DIMENSIONAMENTO EFFETTUATI PRESSO IL C.R.A.

Il programma “Dimacq” è stato sviluppato per la prima volta nel 1993 in linguaggio “Pascal”, e in seguito totalmente riscritto in C++, ampliato ed ottimizzato.

In questo periodo è stato ripetutamente utilizzato, presso il Centro di Ricerca sulle Acque dell’Università di Pavia, per trattare problemi di verifica e dimensionamento di reti idrauliche via via più estese e complesse. Molte delle caratteristiche attuali sono state implementate in seguito a vari tipi di problemi ed esigenze manifestatesi durante tale utilizzo.

Si fa cenno, nel presente capitolo, alle due più significative applicazioni, l’una di sola verifica (rete di distribuzione idrica della città di Monza) e l’altra di dimensionamento (rete distributiva dell’impianto di pluvirrigazione del Consorzio di bonifica della Baraggia Vercellese).

6.1 Verifica della rete di distribuzione idrica della città di Monza (MI)

La rete di distribuzione idrica della città di Monza (MI) copre una superficie di circa 18 km² servendo una popolazione di 122.000 abitanti. L’approvvigionamento idrico è garantito da 36 pozzi, distribuiti nel territorio servito, che alimentano direttamente la rete attingendo dalla falda sottostante, come avviene in diverse reti di distribuzione di grandi centri urbani nel territorio padano. La rete è dotata di un unico piccolo serbatoio a torre, di influenza solo locale.

La strutturazione attuale della rete è il risultato di numerosi potenziamenti ed ampliamenti effettuati nel tempo con lo scopo di adattare il servizio di distribuzione alle sempre crescenti richieste idriche, sia per l’espansione del territorio servito, sia per l’incremento di densità insediativa (civile ed industriale) e di consumo unitario. Il risultato è un sistema idraulico fortemente interconnesso ed articolato, il cui funzionamento risulta difficile da comprendere e prevedere, nonostante la presenza di un sistema di telecontrollo delle stazioni di pompaggio dai pozzi e di un sistema di gestione cartografica della rete e degli allacciamenti.

Lo sviluppo e la taratura di un accurato modello di verifica è stato affrontato con lo scopo di comprendere il funzionamento idraulico del sistema al fine di razionalizzare la gestione delle pompe, definire corrette procedure operative per far fronte ai momenti di crisi e programmare opportuni interventi di potenziamento della rete per eliminarne le attuali carenze.

La schematizzazione adottata per il modello è stata ricavata direttamente dalle informazioni dell’archivio cartografico. La rete non presenta una chiara suddivisione fra condotte di trasporto principali e di distribuzione all’utenza, ma è caratterizzata da una grande varietà di

diametri, con gruppi di piccole condotte che spesso assolvono (a volte in modo insufficiente) chiare funzioni di redistribuzione generale delle portate e delle piezometriche.

E' stata pertanto introdotta nel modello l'intera struttura di rete, ottenendo uno schema di calcolo di 1.613 tronchi e 1.173 nodi, di cui 36 di immissione di portata (pozzi con pompe sommerse) e i rimanenti di erogazione. Lo schema è riprodotto in fig. 6.1.



Figura 6.1: Schema della rete di distribuzione idrica di Monza

Le verifiche di funzionamento in regime permanente sono state effettuate con il programma "Dimacq", utilizzando per tutte le condotte la formula di Darcy, con i coefficienti moltiplicativi k (vedi par. 2.4.1) messi a punto mediante una taratura ottenuta dal confronto fra i risultati di diverse simulazioni e numerosi valori di portata e pressione rilevati in rete dal sistema di telecontrollo.

I pozzi sono stati simulati mediante immissioni in rete di portata imposta, pari ai valori rilevati dal sistema di telecontrollo. I valori di pressione di mandata rilevati dal telecontrollo sono stati confrontati con i risultati del calcolo ai fini della taratura del modello.

In alternativa si è imposta la quota piezometrica ai nodi di immissione (mediante "serbatoi virtuali"), confrontando i relativi valori di portata calcolata e misurata.

Le verifiche sono state effettuate con la forma conduttiva, risultata più conveniente in termini di occupazione di memoria e tempi di elaborazione.

La convergenza (alla tolleranza di 0,1 l/s sulla norma delle portate incognite) è stata ottenuta

mediamente in circa 500 iterazioni, con 45 minuti di tempo di elaborazione su un Power-Computing PowerCenter 150 (PowerMac compatibile) dotato di PowerPC 604e a 150 MHz. L'occupazione complessiva di memoria RAM è stata di circa 8 Mb.

Si è visto che del tempo di elaborazione complessivamente richiesto, circa 1/6 viene impiegato per la costruzione della topologia (ricerca dell'albero e costruzione della matrice delle maglie), ed ancora 1/6 del tempo complessivo per il calcolo del primo tentativo. In seguito a questa osservazione è stato implementato il meccanismo di riutilizzo della topologia e del primo tentativo descritto al precedente capitolo, che ha consentito di ridurre i tempi di calcolo per elaborazioni successive alla prima di circa 15 minuti.

Le prime simulazioni hanno pure evidenziato la necessità di un corredo di verifiche di pre-processo necessarie ad individuare eventuali errori nella costruzione del file di input, altrimenti non gestibili in una rete così complessa. Il programma è stato quindi dotato di una diagnostica sulla topologia di input, che segnala all'utente diversi tipi di anomalie (nodi mancanti, parti di rete non connesse, ecc.).

Per facilitare la comprensione e l'analisi del sistema, è stata pure studiata la possibilità di definire criteri di semplificazione topologica della rete (Iannelli e Brega, 1996), giungendo da un lato a definire algoritmi di semplificazione topologica automatizzata, dall'altro ad evidenziare l'effetto di diverse possibili tecniche semplificative sui tempi di elaborazione necessari, e sul numero complessivo di lati e nodi costituenti la rete.

I motivi che hanno spinto a tale studio sono i seguenti:

- le operazioni di calibratura preliminare sono risultate difficoltose a causa delle grandi dimensioni della rete: poiché l'attribuzione dei coefficienti di perdita viene effettuata variandone i valori nelle varie condotte fino ad ottenere la migliore corrispondenza con la piezometria misurata, la possibilità di far riferimento ad uno schema con un minor numero di condotte consente di semplificare significativamente la procedura;
- la prevedibilità del comportamento della rete è risultata scarsa: poiché gli interventi da proporre non possono che essere ipotizzati dal progettista e quindi verificati attraverso il modello, la disponibilità di una struttura topologica semplificata risulta di grande aiuto anche al progettista in fase di definizione delle varie alternative progettuali da sperimentare mediante verifica sul modello;
- visti i tempi di elaborazione relativamente lunghi necessari per ogni verifica, un modello in grado di essere eseguito più rapidamente risulta particolarmente utile, specialmente nelle operazioni di calibratura, che richiedono numerose elaborazioni di verifica e confronto con i valori misurati sulla rete reale.

Le varie possibili tecniche semplificative sono state applicate in cinque passi:

1. le parti di rete connesse alla struttura generale mediante un solo punto, sono state sostituite con singole erogazioni concentrate;
2. gruppi di condotte in parallelo sono stati sostituiti con un'unica condotta equivalente;

3. gruppi di condotte in serie sono stati sostituiti con un'unica condotta equivalente previa ripartizione sui nodi restanti delle erogazioni ai nodi eliminati;
4. sono state eliminate le condotte a portata minima;
5. sono state eliminate le condotte a resistività massima (in alternativa a 4.)

Tutti i passi semplificativi sono stati realizzati curando la possibilità di realizzazione automatizzata, mediante alcuni algoritmi descritti nel lavoro citato.

I risultati ottenuti nelle varie fasi della semplificazione sono stati confrontati con quelli del modello completo sia dal punto di vista della corrispondenza col modello completo che dei tempi di elaborazione e del numero di iterazioni richieste per giungere a convergenza.

I confronti sono sintetizzati in Tabella 6.1, in cui la qualità delle semplificazioni viene quantificata catalogando percentualmente i nodi in funzione dell'errore (ovvero della differenza di quota piezometrica calcolata rispetto al modello completo) commesso per ciascuno di essi. (I tempi di elaborazione si riferiscono ad un PowerMac 6100/66 con PowerPC601 a 66 Mhz, significativamente meno veloce della macchina precedentemente citata).

Tabella 6.1 - Sintesi dei risultati del processo di semplificazione

<i>passi di semplificazione</i>	<i>num. tronchi residui</i>	<i>num. nodi residui</i>	<i>tempo di calcolo per la ve- rifica</i>	<i>numero iterazio- ni per la verifica</i>	<i>% nodi con errore sulla piezometri- ca < 0,5 m</i>	<i>% nodi con errore sulla piezometri- ca < 1 m</i>	<i>% nodi con errore sulla piezometrica < 1,5 m</i>
Rete completa	1613	1173	2h 04'	490	100	100	100
Eliminazione strutture ad una connessione	1442	1008	1h 23'	441	100	100	100
Sostituzione tronchi in serie e in parallelo	1198	788	52'	453	45,3	79,2	95,7
Taratura con ridistribuzione dell'erogazione	1198	788	52'	453	94,8	97,7	99,1
Eliminazione tronchi con portata < 0,2 l/s	1148	788	41'	405	94,6	97,7	99,1
Eliminazione tronchi con $r > 3 \cdot 10^5 \text{ s}^2 \cdot \text{m}^{-5}$	1137	788	34'	362	93,7	97,7	99,4

La tabella evidenzia che:

- al termine del procedimento di semplificazione i tempi di calcolo si sono ridotti a circa 1/4 di quelli richiesti dal modello completo, avendo eliminato il 29,5% delle condotte e il 32,8% dei nodi;
- gli errori generati sono risultati molto piccoli, largamente inferiori a quelli ottenibili da una buona calibrazione del modello completo rispetto alla rete reale.

6.2 Dimensionamento della rete distributiva dell'impianto di pluvirrigazione del Consorzio di bonifica della Baraggia Vercellese

Un'altra interessante applicazione del programma, questa volta rivolta alla fase di dimensio-

namento automatico, si riferisce alla rete di distribuzione di un grande impianto di irrigazione a pioggia previsto nel comprensorio della Baraggia Vercellese in seguito alla realizzazione della diga di Mongrando lungo il corso del torrente Ingagna (Sammartano e al., 1996).

Si tratta di una rete primaria strutturata a maglie, dalla quale sono derivate le reti secondarie di alimentazione degli irrigatori, che vengono attivati in sequenza mediante un complesso sistema di elettrovalvole comandate da un sistema di telecontrollo centralizzato (fig. 6.2)

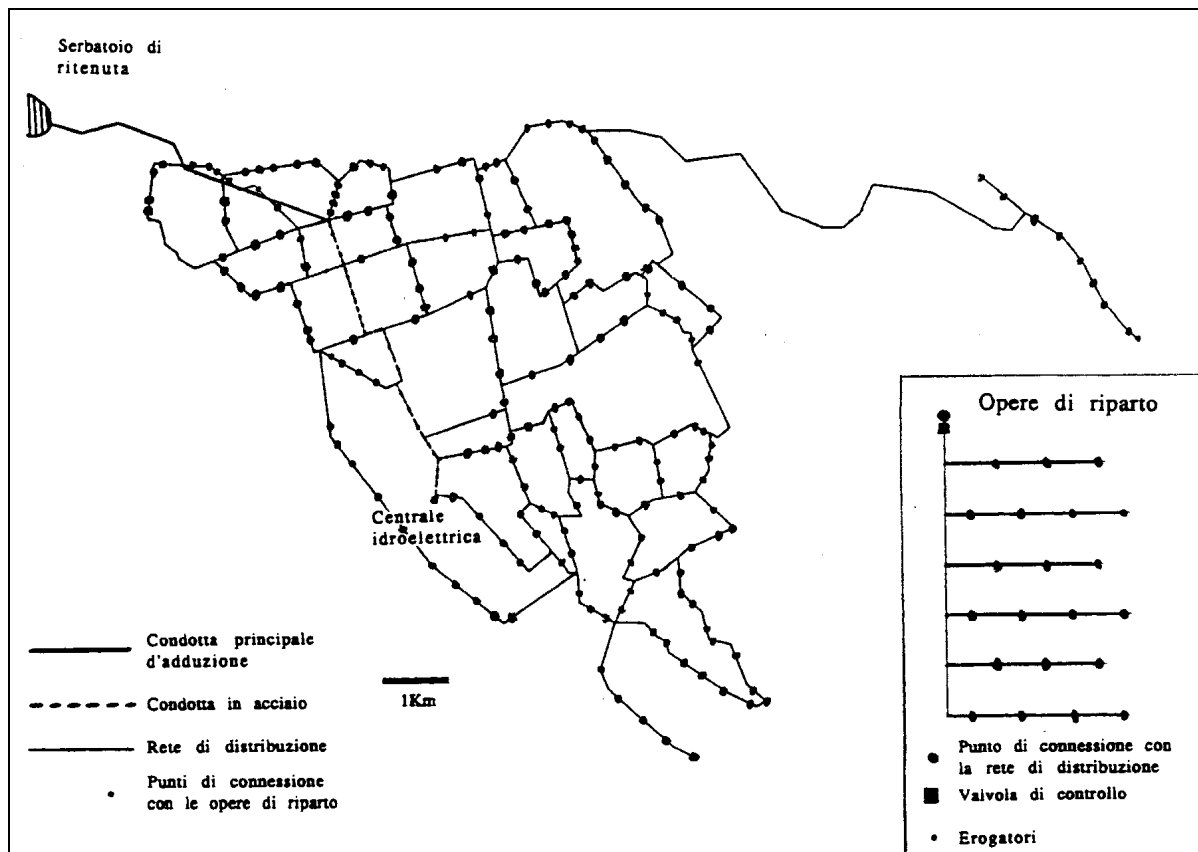


Figura 6.2: Schema della rete distributiva del previsto impianto di pluvirrigazione della Baraggia Vercellese

L'irrigazione del comprensorio è prevista mediante suddivisione in 32 comizi, attivati sequenzialmente mediante un meccanismo di turnazione accuratamente studiato.

All'interno della rete è pure prevista una condotta in acciaio, interconnessa alle altre ma disconnettibile mediante azionamento di opportune elettrovalvole destinata, nei periodi di inutilizzo agricolo, ad alimentare una centrale di sfruttamento idroelettrico.

Il problema di dimensionamento si presenta quantomai interessante, in quanto il processo di ottimizzazione (necessario al dimensionamento ottimale dei diametri delle condotte) deve tener conto contemporaneamente dei 32 diversi regimi di funzionamento relativi ai vari turni irrigui.

Per trattare questo problema il programma è stato modificato in modo da consentire, nell'ambito del processo iterativo di definizione dei diametri, l'analisi simultanea di diverse

configurazioni di verifica.

La rete è stata schematizzata mediante 57 nodi e 91 condotte, di cui 8 in acciaio di diametro prefissato e 83 in vetroresina da dimensionare. Il ciclo di ottimizzazione ha previsto 32 verifiche complete ad ogni passo di iterazione, per un totale di circa 200 iterazioni in un tempo complessivo mediamente pari ad 11 ore su PowerMac 6100/66 con powerPC601 a 66 Mhz, corrispondente a circa 6 secondi per ogni verifica completa.

Il procedimento di ottimizzazione è stato ripetuto più volte variando le condizioni iniziali, il set di diametri commerciali fra cui scegliere ed anche alcuni dettagli implementativi dell'algoritmo di ricerca, scegliendo alla fine la soluzione più funzionale ed economica.

Il processo di ottimizzazione è stato complicato dalla necessità di tener conto anche di diverse possibili pressioni operative degli irrigatori, con diversi costi specifici dovuti alla maggior gittata degli irrigatori a pressione nominale maggiore. Questo aspetto è stato preso in considerazione manualmente, mediante confronto fra diverse possibili soluzioni individuate dal programma di ottimizzazione. ulteriori dettagli su questo lavoro sono rilevabili dalla Tesi di Laurea citata in bibliografia.

7. CONCLUSIONI E FUTURI SVILUPPI

Il codice di calcolo accuratamente descritto nel presente lavoro ha raggiunto, dopo oltre tre anni di perfezionamenti e collaudi, una certa maturità, che ne garantisce affidabilità, efficienza e comodità di utilizzo.

Non volendo certo competere con i numerosi prodotti commerciali disponibili, si caratterizza però per il fatto di consentire anche un dimensionamento automatico delle reti, ottenibile prefissando le quote minime richieste a tutti i nodi, caratteristica questa non disponibile, a conoscenza dello scrivente, nei software commerciali più diffusi.

L'algoritmo di verifica ha denotato stabilità ed efficienza nel trattamento di reti anche molto complesse ed estese.

il processo di dimensionamento è invece da considerarsi di primo orientamento, in quanto utile soprattutto per definire possibili soluzioni di base su cui il progettista può successivamente intervenire prendendo in considerazione i molti ulteriori aspetti peculiari tipici di ciascuna rete.

Al procedimento di ottimizzazione si intende dedicarsi ancora in futuro, sia implementando possibili algoritmi alternativi più efficaci nell'avvicinarsi al minimo assoluto della funzione costo, sia, soprattutto, prevedendo l'implementazione, nelle funzioni di vincolo, di altre caratteristiche essenziali delle reti idrauliche, prime fra tutte una definizione ben codificata delle caratteristiche di ridondanza ed affidabilità.

8. BIBLIOGRAFIA

- ANDERSON E.J., AL JAMAL K.H. (1995), "Hydraulic-network simplification", *Journal of Water Resources Planning and Management.*, ASCE, vol.121, n.3, p.235.
- APOSTOL T.M. (1978), "Calcolo - volume terzo - analisi 2", *Boringhieri*, pp. 264-269.
- AWWA RESEARCH COMMITTEE ON DISTRIBUTION SYSTEMS (1974), "Water-distribution research and applied development needs", *Journal*, AWWA, vol.68, n.4, p.189.
- BRANDON T.W. (1984), "Water distribution systems", *the Inst. of Water Engineers and Scientists*, London, England, p.153.
- BROYDEN C.G. (1965), "A class of methods for solving nonlinear simultaneous equations", *Math of computation*, pp. 577, 593.
- CAPPELLI D'ORAZIO M., FONTANA D.M., SANTARPIA L. (1992), "Optimization of the capital cost of closed water supply systems: direct finding of solutions close to the absolute minimum", *Ingegneria sanitaria-ambientale*, sett.-ott., pp. 45-55.
- CEMBROWICZ R.G., HARRINGTON J.J. (1973), "Capital-cost minimization of hydraulic network", *Journal of the hydraulic division*, proceedings ASCE, Vol. 99, No HY3, March, pp. 431-440.
- CENEDESE A., MELE P. (1976), "Un metodo per il dimensionamento diretto delle reti idriche a maglie", *Atti del XV convegno di idraulica e costruzioni idrauliche*, Roma.
- CIAPONI C., PAPIRI (S. 1985), "Contributo al dimensionamento ottimale delle reti di condotte in pressione a maglie chiuse", *Ingegneria sanitaria*, n. 4, luglio-agosto, pp. 209-217.
- COLORNI A. (1992), "Ricerca operativa", *CLUP - Città Studi*, Milano.
- CURTO G., TUMBIOLO A. (1995), "La calibrazione dei modelli delle reti di distribuzione idrica", 16° corso di aggiorn. *Tecniche per la difesa dall'inquinamento*, Guardia Piemontese Terme.
- EGGENER C.L., POLKOWSKI L.B. (1976), "Network models and the impact of modeling assumptions", *Journal*, AWWA, vol.68, n.4, p.189.
- FLETCHER R. (1996), "Practical methods of optimization" 2nd ed., *J. Wiley & sons*.
- HAMBERG D., SHAMIR U. (1988), "Schematic models for distribution systems design. I: Combination concept", *Journal of Water Resources Planning and Management*, ASCE, vol.114, n.2, p.129.
- IANNELLI G. (1963), "La verifica delle reti di distribuzione di fluidi incomprimibili" *VIII Convegno Nazionale di Idraulica e Costruzioni Idrauliche*, Pisa.

- IANNELLI G. (1967), "Elementi e dati per il proporzionamento delle reti di distribuzione" *Ingegneria sanitaria*, n. 4.
- IANNELLI R. (1993), "Metodi di progettazione per l'ampliamento delle reti di distribuzione idrica" - Atti del XIV corso di aggiornamento *Tecniche per la difesa dall'inquinamento*, Università degli studi della Calabria, Giugno.
- IANNELLI R., BREGA A. (1996), "Metodi di semplificazione topologica per lo studio delle reti di distribuzione idrica", Atti del *XXV Convegno Nazionale di Idraulica e Costruzioni Idrauliche*, Torino.
- JACOBY S.L.S. (1968), "Design of optimal hydraulic networks", *Journal of the hydraulic division*, proceedings ASCE, Vol. 94, No HY3, May, pp. 641-661.
- JEPPSON R.W. (1982), "Equivalent hydraulic pipe for parallel pipes", *Journal of the Hydraulics Division*, ASCE, vol.108, n.HY1, p.35.
- KESAVAN H.K., CHANDRASHEKAR M. (1972), "Graph-theoretic models for pipe network analysis", *Journal of the hydraulic division*, proceedings ASCE, Vol. 98, No HY2, Feb., pp. 345-364.
- LAM C. F. (1973), "Discrete gradient optimization of water systems", *Journal of the hydraulic division*, proceedings ASCE, Vol. 99, No HY6, June, pp. 863-872.
- LAM C.F., WOLLA M.L. (1972), "Computer analysis of water distribution systems: part II-numerical solution", *Journal of the hydraulic division*, proceedings ASCE, Vol. 98, No HY3, March, pp. 447-460.
- LIPSCHUTZ S. (1976), "Discrete mathematics", *Mc Graw-Hill Book Company*, New York.
- MELE P. (1971), "Il dimensionamento idraulico di sistemi di condotte con i metodi della programmazione lineare", *L'acqua*, fasc. 6, pp. 89-94.
- MORGAN D.R., GOULTER I.C. (1985), "Optimal urban water distribution design", *Water resources research*, vol. 21, n. 5, pp. 642-652.
- SAMMARTANO M., IANNELLI G., IANNELLI R. (1996), "Studio di ottimizzazione economica per il dimensionamento di un impianto di irrigazione a pioggia", *Tesi di laurea di M. Sammartano, relatore G. Iannelli, correlatore R. Iannelli.*
- SASSOLI F., MILANO V. (1973), "Sul calcolo di economia di un particolare tipo di rete di distribuzione urbana ad anello", *Giornale del Genio civile*, n. 1-2, pp. 21-30.
- TERMINI D., TUCCIARELLI T., MAZZOLA M.R. (1994), "La progettazione delle reti idriche con il metodo "Simulated annealing"", Atti del *XXIV Convegno Nazionale di Idraulica e Costruzioni Idrauliche*, Napoli.
- WALSKI, T.M. (1983), "Using water distribution system models", *Journal*, AWWA, vol.75, n.2, p.58.
- WATANADA T. (1973), "Least-cost design of water distribution systems", *Journal of the hydraulic division*, proceedings ASCE, Vol. 99, No HY9, Sept., pp. 1497-1513.