

POLITECNICO DI MILANO

Facoltà di Ingegneria dei Processi Industriali

Corso di Studi in INGEGNERIA CHIMICA

Anno Accademico 2010-2011



Tesi di Laurea Specialistica

**Programmazione Lineare: confronto tra metodi
esistenti e un metodo innovativo**

Relatore: Prof. Guido Buzzi Ferraris

Correlatore: Ing. Flavio Manenti

Tesi di laurea specialistica di:

Marco Gabba

matr. 750228

Indice

| | |
|---|----|
| Capitolo 1 – Introduzione..... | 1 |
| Capitolo 2 - Metodi numerici: Stato dell'arte..... | 7 |
| 2.1 Programmazione lineare..... | 7 |
| 2.2 Il metodo del Simpleso..... | 11 |
| 2.2.1 Metodo del Simpleso – Algoritmo..... | 13 |
| 2.2.2 Metodo del Simpleso – Algoritmo modificato..... | 16 |
| 2.2.3 Metodo del Simpleso – Aspetti avanzati..... | 18 |
| 2.3 Il metodo dell'Interior Point..... | 23 |
| 2.3.1 Il metodo dell'Ellissoide..... | 24 |
| 2.3.2 Interior Point..... | 25 |
| Capitolo 3 – Il metodo dell'Attico..... | 29 |
| 3.1 Introduzione..... | 29 |
| 3.2 Concetti base..... | 30 |
| 3.3 Il metodo dell'Attico..... | 30 |
| 3.4 Differenze rispetto agli approcci tradizionali..... | 35 |
| Capitolo 4 - Caso studio..... | 39 |
| 4.1 Il modello matematico..... | 47 |
| Capitolo 5 - Futuri sviluppi..... | 59 |
| 5.1 Mixed-Integer Linear Programming (MILP)..... | 59 |
| 5.1.1 MILP - formulazione..... | 59 |
| 5.2 Il metodo Branch and Bound..... | 61 |
| 5.2.1 Separazione..... | 61 |
| 5.2.2 Rilassamento..... | 62 |
| 5.2.3 Fathoming..... | 63 |
| 5.2.4 Struttura del Branch&Bound..... | 64 |
| Capitolo 6 – conclusioni..... | 67 |
| Bibliografia..... | 71 |

Indice delle figure

| | |
|--|----|
| Figura 1: spazio delle soluzioni accettabili | 10 |
| Figura 2: soluzione di ottimo | 10 |
| Figura 3: Flowsheet della raffineria | 42 |

Indice delle tabelle

| | |
|--|----|
| Tabella 1: tableau del Simplex, iterazione 0 | 14 |
| Tabella 2: tableau del Simplex, iterazione 1 | 15 |
| Tabella 3: tableau del Simplex, iterazione 2 | 15 |
| Tabella 4: tableau del Simplex, iterazione 3 | 15 |
| Tabella 5: Tagli del greggio Mid-Continent a 38.0° API | 40 |
| Tabella 6: Prodotti, prima del blending | 41 |
| Tabella 7: Rese del cracking a singolo passaggio | 43 |
| Tabella 8: Prodotti finali, specifiche | 44 |
| Tabella 9: Coefficienti del LP | 51 |

Abstract

La *Programmazione Lineare* (LP) è una classe di problemi di ottimizzazione vincolata nella quale sia la funzione obiettivo che i vincoli cui il problema è soggetto sono lineari rispetto alle variabili. La programmazione lineare ed i primi problemi ad essa connessi risalgono al 1945, quando il premio Nobel George Stigler formula “il problema della dieta”, ovvero la formulazione di un menu per soddisfare una dozzina circa di richieste nutrizionali con la minima spesa. Nel 1947, Dantzig formalizza il concetto di programmazione lineare come viene oggi inteso e crea anche il metodo risolutivo conosciuto come “metodo del Simplex”. Tale metodo è rimasto l’unico esistente fino a quando Karmakar, nel 1984, propone un nuovo metodo risolutivo, il metodo cosiddetto dell’Interior Point, al fine di superare alcuni problemi che limitavano le possibili applicazioni del metodo del Simplex. Tale metodo, infatti, nasce con l’intento di fornire un algoritmo risolutivo che assicuri un numero massimo di iterazioni *polinomiale* rispetto al numero delle variabili. Nel 2011 il professor Buzzi-Ferraris ha formulato e proposto il metodo dell’Attico. Obiettivo di tale nuovo metodo è fornire uno strumento che possa sfruttare al meglio gli avanzamenti matematici effettuati negli ultimi trent’anni nel campo dei linguaggi di programmazione (viene utilizzata, infatti, la programmazione orientata agli oggetti tipica del linguaggio di programmazione C++) e della fattorizzazione di matrici, in modo da risultare di più facile utilizzo e, di solito, maggiormente performante.. Le caratteristiche matematiche peculiari e i primi testi e confronti effettuati in questo lavoro di tesi fanno ipotizzare che, una volta sviluppato adeguatamente l’algoritmo, attraverso il metodo dell’Attico sarà possibile avere uno strumento efficace e performante per la risoluzione di problemi LP e MILP (Mixed Integer Linear Programming) di dimensioni maggiori rispetto a quanto oggi possibile e permetterà dunque di aumentare ulteriormente l’importanza e l’impatto di questa già fondamentale classe di problemi.

Abstract

The *Linear Programming* (LP) is a class of constrained optimization problems in which both the objective function and the constraints which the problem is subject to are linear with respect to the variables. LP and the first problems related to it date back to 1945, when the Nobel George Stigler formulates "the diet problem", or the formulation of a menu to meet a dozen or nutritional requirements at minimum expense. In 1947, Dantzig formalized the concept of LP as it is now understood and also created the resolution method known as "simplex method". This method is the only one existing until Karmakar, in 1984, proposes a new resolution method, the so-called Interior Point method, in order to overcome some problems that limited the possible applications of the simplex method. This method, in fact, was created with the intent to provide a resolution algorithm that ensures a maximum number of iterations which is of polynomial order with respect to the number of variables. In 2011 Professor Buzzi-Ferraris formulated and proposed the Attic method. The aim of this new method is to provide a tool that can make the most of the mathematical advances made over the last thirty years in the field of programming languages (it is used, in fact, the object-oriented programming typical of the C++ language) and the factorization of matrices so that this method is easier to use and, usually, more powerful. The peculiar mathematical features and the first tests and comparisons made in this thesis led to assume that, once properly developed the algorithm, through the Attic method will be possible to have a powerful and effective tool for solving in a faster way LP and MILP (Mixed Integer Linear Programming) problems larger than what is currently possible and will thus further increase the relevance and impact of this already important class of problems.

Capitolo 1 – Introduzione **La Programmazione Lineare**

La *Programmazione Lineare* (Linear Programming, LP) è una classe di metodi di ottimizzazione vincolata nella quale sia la funzione obiettivo che i vincoli, cui il problema è soggetto, sono lineari rispetto alle variabili.

La programmazione lineare ed i primi problemi ad essa connessi risalgono al 1945, quando il premio Nobel George Stigler formula “il problema della dieta”, ovvero la formulazione di un menu per soddisfare una dozzina circa di richieste nutrizionali con la minima spesa. Durante lo studio di tale tipologia di problema, Stigler realizza che può essere considerato un caso particolare di una classe più generali di problemi a funzione obiettivo e vincoli lineari e che non esiste ancora un metodo diretto per risolvere tale tipo di problemi.

Nello stesso periodo Dantzig sta lavorando per trovare una soluzione per risolvere velocemente i problemi di schieramento militare, addestramento e rifornimento logistico delle derrate e realizza che molti problemi organizzativi possono essere formulati sotto forma di sistemi di disequazioni lineari. Nel 1947, quindi, formalizza il concetto di programmazione lineare come viene oggi inteso e sviluppa anche il primo metodo di calcolo risolutivo, oggi giorno conosciuto come “metodo del Simplex”.

Tale metodo è rimasto l'unico esistente fino a quando Karmakar, nel 1984, ha proposto un nuovo metodo risolutivo, il metodo cosiddetto dell'Interior Point, al fine di superare alcuni problemi che limitavano le possibili applicazioni del metodo del Simplex. Dal momento che, infatti, il numero di iterazioni massime possibili richieste da tale metodo risultavano esponenziali con il numero di variabili, non era talvolta possibile risolvere problemi di grandi dimensioni in quanto i tempi di calcolo erano proibitivi. Il metodo dell'Interior Point nasce quindi con l'intento di fornire un algoritmo risolutivo che assicuri un numero massimo di iterazioni cosiddetto *polinomiale* rispetto al numero delle variabili. In altre parole, l'Interior Point assicura che il numero massimo di iterazioni necessarie per risolvere il problema di programmazione lineare sia vincolato superiormente da una espressione polinomiale rispetto al numero di vincoli del problema.

La Programmazione Lineare conosce quindi una sempre maggior diffusione in termini di numero di applicazioni e riceve sempre più attenzioni da parte di studiosi da tutto il mondo. I due metodi basilari vengono sempre più ottimizzati e i problemi risolvibili diventano sempre più numerosi. Non appena tale pratica diventa però ben assimilata ci si rende conto che i valori frazionari di alcune variabili, nella fattispecie delle variabili decisionali, non sono accettabili. Per esempio, se le derrate alimentari di Dantzig debbono essere spostate da A a B, non ha senso spostarle fino a metà strada o spostarne solo la metà. Ci si rende conto, quindi, che arrotondare tale tipo di variabili all'intero più vicino porta ad errori e imprecisioni rilevanti. Viene quindi ideato un nuovo tipo di problema che affianca alle variabili continue delle variabili di natura discreta: il *Mixed Integer Linear Programming* (MILP). Dal momento che la tecnica più utilizzata per risolvere tale tipo di problemi (il metodo detto *Branch&Bound*) consiste nella risoluzione di un elevato numero di problemi LP fino ad identificare la soluzione ottimale del problema MILP (Capitolo 5, paragrafo 2), diventa sempre più importante rendere gli algoritmi ancora più efficienti e performanti in maniera tale da poter risolvere anche questa classe di problemi in tempi ragionevoli.

Nel seguito vengono indicate le principali tipologie di applicazione dei problemi di tipo LP e MILP.

- **Miscelazione.**

Il problema della dieta è un esempio specifico di tale categoria. Come suggerisce il nome, questa tipologia di problemi implica la miscelazione di prodotti grezzi in maniera tale da soddisfare dei requisiti specifici sui prodotti in uscita. Problemi di miscelazione si trovano frequentemente nell'industria petrolifera, dove diverse tipologie di greggio devono essere miscelati in benzine che rispettino le specifiche commerciali.

- **Movimentazione greggio**

Concettualmente opposte alle operazioni di miscelazione sono quelle di segregazione, spesso chiamate movimentazione di greggio; tale tipo di operazione è essenziale quando vi è la necessità di mantenere separate le diverse miscele, soprattutto per quanto riguarda i prodotti finali (non ha senso ad esempio mescolare nuovamente un diesel e una benzina dopo aver speso molti soldi ed energia per separarli).

Tale classe di problemi può riguardare in ogni caso l'intera raffineria (per esempio, può essere preferibile tenere separati grezzi paraffinici dai grezzi naftenici).

- **Gestione di capitali.**

Uno dei principali problemi in cui si rimanda a problematiche di tipo MILP è quello di gestione dei capitali. Tale problema consiste nel decidere quali progetti finanziare dato un vincolo sul capitale disponibile. La modellazione avviene considerando i *net present value* (NPV) dei diversi progetti e massimizzando la funzione obiettivo data dalla loro somma.

- **Vehicle routing.**

I modelli matematici sono stati particolarmente importanti nel ridurre i costi della logistica agevolando i processi di *vehicle routing*. Il problema più caratteristico di questa classe è il "problema di assegnazione generalizzato": assegnare ciascuno degli N compiti a M server senza superare le capacità dei singoli server. La maggior parte dei sistemi di logistica è caratterizzata da un elevato numero di variabili (distribuzione carburanti, Trenitalia, compagnie aeree...).

- **Programmazione della produzione**

Il problema più rappresentativo di questa categoria per quanto concerne l'ingegneria chimica è il problema del dimensionamento di lotti multi-prodotto: determinare la produzione di N prodotti su T periodi di tempo. L'obiettivo da raggiungere è il soddisfacimento della domanda di mercato per ciascun prodotto in ciascun periodo e non sono permessi esaurimenti di scorte. Vi sono costi marginali di produzione e stoccaggio, costi fissi di produzione e i prodotti competono per l'utilizzo delle macchine di produzione in ogni periodo di tempo. Tale problema va spesso sotto il nome di scheduling di produzione, ovvero l'assegnazione di compiti specifici ai vari treni produttivi di un unico sito, considerando le capacità massime di ogni treno produttivo.

Tuttavia, vi sono svariate applicazioni nell'ambito dell'Ingegneria Chimica e i problemi LP e MILP più ricorrenti sono:

- **Progettazione di processo**
 - Reti di scambiatori
 - Sequenze di distillazione
 - Sistemi di reattori

- **Design, Scheduling, Planning di reattori Batch.**
 - Design di impianti multi prodotto.
 - Design di impianti multiscopo.

Nel 2011 il professor Buzzi-Ferraris ha proposto una nuova innovazione nel campo della Programmazione Lineare: il metodo dell'Attico. Obiettivo di tale nuovo metodo è fornire uno strumento che possa sfruttare al meglio tutti gli avanzamenti matematici effettuati negli ultimi trent'anni, soprattutto nel campo dei linguaggi di programmazione (viene utilizzata, infatti, la programmazione orientata agli oggetti in C++) e della fattorizzazione di matrici, in modo da risultare di più facile utilizzo e, di solito, maggiormente performante; una delle caratteristiche più importanti del metodo è infatti quella per cui l'Attico può risolvere un problema LP così com'è, senza dover necessariamente modificare il modello per poterlo rendere adatto alla risoluzione attraverso i metodi del Simplexso o dell'Interior Point.

Nello svolgimento di questa attività di tesi verranno quindi inizialmente presentati nel dettaglio i metodi di Simplexso e Interior Point. Partendo dai lavori di Dantzig (Dantzig, 1963) e Karmakar (Karmakar, 1984), nel corso degli anni sempre nuove caratteristiche e tecniche sono state aggiunte nel corso degli anni per sviluppare algoritmi capaci di risolvere una quantità sempre maggiore di problemi ed un crescente numero di gradi di libertà, variabili e vincoli. L'evoluzione di tali algoritmi verrà quindi percorsa grazie anche ai lavori di Luenberger (Luenberger & Ye, 2010), Martin (Martin, 1998) e Vanderbei (Vanderbei, 2001).

Verrà quindi effettuata un'analisi approfondita del metodo dell'Attico, così come presentata nel lavoro del Professor Buzzi-Ferraris (Buzzi-Ferraris, 2011). Verranno dunque esaminate le caratteristiche peculiari di tale metodo, le sue basi matematiche e la procedura di risoluzione.

Le prestazioni di questi metodi verranno quindi confrontate su di un caso studio (Manne, 1963). La scelta del caso studio è stata effettuata ponderando complessità di risoluzione, facilità di implementazione e disponibilità di tutti i dati necessari alla risoluzione numerica. Per tale confronto saranno utilizzati gli algoritmi più recenti e diffusi nelle comunità scientifiche e basati su Simplex e Interior Point: CPLEX (IBM), SIMPO (Vanderbei, 2001), MINOS 5.5 (AMPL) e la funzione "linprog" di Matlab (MathWorks).

Verrà infine presentata in dettaglio la teoria del MILP, basandosi soprattutto sul lavoro di Floudas (Floudas, 1995), in quanto tale tipo di problemi saranno la più probabile e importante applicazione del metodo dell'Attico non appena l'algoritmo raggiungerà la sua piena maturità.

Tra le soluzioni accettabili del problema si troveranno quindi anche le *soluzioni di base accettabili*, secondo quanto affermato dal teorema fondamentale della programmazione lineare.

Teorema fondamentale della programmazione lineare: “dato un problema lineare in forma standard (3), ove **A** è una matrice di dimensioni (m x n) di rango m,

1. Se esiste una soluzione accettabile esiste anche una soluzione accettabile di base.
2. Se esiste una soluzione accettabile di ottimo esiste anche una soluzione accettabile di base di ottimo.

A questo punto può essere particolarmente utile osservare un esempio in due dimensioni, al fine di visualizzare meglio i concetti sopra esposti.

Esempio

minimizzare $-2x_1 - x_2$

soggetta a: $x_1 + \frac{8}{3}x_2 \leq 4$

$$x_1 + x_2 \leq 2$$

$$2x_1 \leq 3$$

$$x_1 \geq 0, x_2 \geq 0$$

Come visto in precedenza è possibile trasformare questo problema in forma standard mediante l'aggiunta di tre variabili ausiliarie, ottenendo quindi:

minimizzare $-2x_1 - x_2$

soggetta a: $x_1 + \frac{8}{3}x_2 + x_3 = 4$

$$x_1 + x_2 + x_4 = 2$$

$$2x_1 + x_5 = 3$$

$$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0$$

Le soluzioni si base del problema si possono ottenere ponendo pari a zero due variabili e risolvendo il sistema rispetto alle altre. Dal punto di vista grafico ponendo pari a zero una variabile si ottiene una retta, e quindi le soluzioni di base del problema sono i vertici del politopo ottenuto intersecando le varie rette (Figura 1).

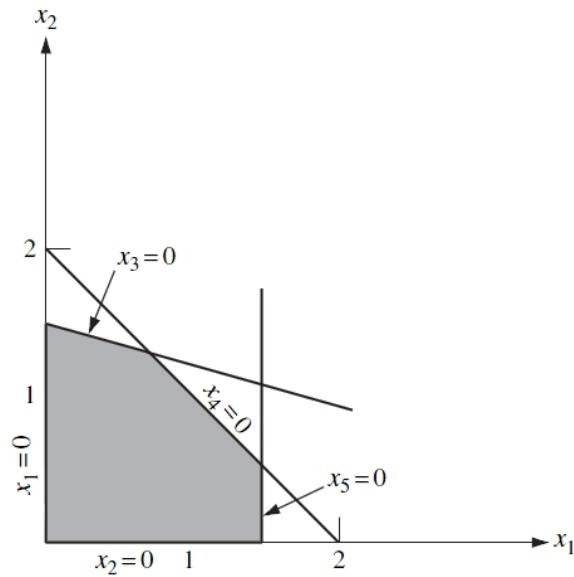


Figura 1: spazio delle soluzioni accettabili

Anche le curve di livello della funzione obiettivo sono rappresentabili con una retta; il problema di ottimizzazione può quindi essere risolto graficamente cercando il minor valore di $z = -2x_1 - x_2$ per cui la retta interseca lo spazio delle soluzioni accettabili. Questo fatto avviene sempre in corrispondenza di un punto estremo (quindi soluzione di base) e nell'esempio mostrato ciò avviene per $z = -3.5$ (Figura 2).

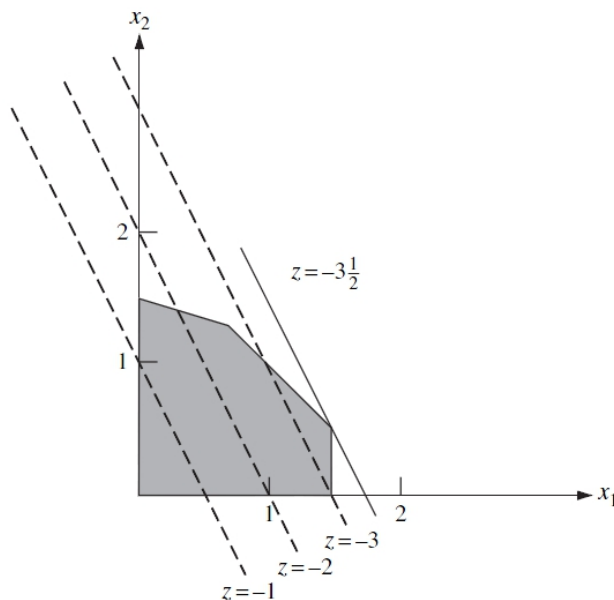


Figura 2: soluzione di ottimo

2.2 Il metodo del Simpleso

Proprio partendo da queste considerazioni di tipo geometrico Dantzig sviluppò nel 1947 il metodo del simpleso, che si basa su un semplice principio, ovvero quello di spostarsi tra punti estremi adiacenti fino a raggiungere il minimo della funzione obiettivo. Dal punto di vista matematico, questo consiste nel passare da una soluzione accettabile di base ad un'altra adiacente in modo tale da diminuire progressivamente il valore della funzione obiettivo. Per fare questo è necessario eseguire un'operazione di *pivoting*, in modo tale che una variabile di base diventi non di base (esce dalla base) mentre una variabile non di base diventi di base (*entra* nella base).

Il *pivoting* consiste nel trasformare la matrice **A**, in maniera simile a quanto avviene nei metodi di eliminazione e trasformazione di Gauss, attraverso la scelta di una variabile (detta appunto *pivot*) ed il calcolo di moltiplicatori al fine di fornire alla matrice la struttura desiderata (nel caso del metodo del Simpleso, rendere una determinata colonna della matrice nulla eccetto un elemento pari a 1)

Volendo quindi sostituire la *l*-sima variabile (appartenente alla base) con la *k*-sima variabile (non appartenente alla base) è possibile farlo, a patto che a_{lk} sia diverso da zero, effettuando le seguenti sostituzioni (ove a'_{ij} sono i coefficienti della matrice **A** modificata e b'_i sono i coefficienti del vettore **b** modificato):

$$\left\{ \begin{array}{l} a'_{ij} = a_{ij} - \frac{a_{lj}}{a_{lk}} \cdot a_{ik} \quad \text{se } i \neq l \\ a'_{lj} = \frac{a_{lj}}{a_{lk}} \end{array} \right. \quad \left\{ \begin{array}{l} b'_i = b_i - b_l \cdot \frac{a_{ik}}{a_{lk}} \quad \text{se } i \neq l \\ b'_l = \frac{b_l}{a_{lk}} \end{array} \right.$$

Bisogna tuttavia capire quale variabile debba uscire dalla base e quale invece vi debba entrare per far sì che la funzione obiettivo diminuisca rispetto al valore attuale. Il problema può essere risolto nel seguente modo: per prima cosa va individuato tra i coefficienti della funzione obiettivo quello negativo e maggiore in valore assoluto; questo significa infatti che se la variabile associata all'indice di questo coefficiente entra nella base si ottiene la massima diminuzione della funzione obiettivo.

Per decidere quale variabile debba lasciare la base bisogna invece considerare il seguente fatto: la variabile che entra nella base passa dal valore zero ad un valore positivo e così facendo modifica il valore delle altre variabili di base:

$$x_i = b_i - a_{ik} \cdot x_k \quad i \in B$$

Bisogna perciò assicurarsi che tutte le variabili rimangano non negative, per cui tra quelle associate a coefficienti $a_{ik} > 0$ devo trovare quella per cui sia minimo il rapporto b_i / a_{ik} . Questa variabile sarà quella che lascerà la base.

Il metodo del simplesso consiste quindi in una serie di operazioni di pivoting successive (ovvero nello spostamento lungo diversi vertici adiacenti) fino ad arrivare al punto in cui tutti i coefficienti della funzione obiettivo sono positivi: qualsiasi variazione del valore delle variabili farebbe aumentare il valore di tale funzione. La soluzione di base trovata è di conseguenza la soluzione di base di ottimo.

Da quanto detto finora dovrebbe essere chiaro che per far partire il metodo del simplesso è necessario conoscere una soluzione di base di partenza. In molti casi, questa è fornita dalle variabili ausiliarie ma, nel caso in cui ciò non fosse possibile, potrebbe essere molto difficile individuare la base di partenza; per ovviare al problema si introducono quindi delle *variabili artificiali*, tali per cui:

- Sia facile trovare una soluzione di base
- La soluzione di base trovata sia accettabile per il problema di partenza.

Per soddisfare queste caratteristiche una volta introdotte le variabili ausiliarie bisogna introdurre anche una *funzione obiettivo artificiale*, che andrà ottimizzata anch'essa sfruttando il metodo del simplesso.

2.2.1 Metodo del Simplexso – Algoritmo

Dal punto di vista dell'implementazione vengono modificate le matrici di partenza, nota la matrice \mathbf{A} di dimensioni $(m \times n)$ e il vettore dei termini noti \mathbf{b} di dimensioni m , nel seguente modo:

- i. Viene aggiunta la riga $(m + 1)$, i cui coefficienti sono quelli della funzione obiettivo per la matrice \mathbf{A} e il valore della funzione obiettivo per il vettore \mathbf{b} .
- ii. Viene aggiunta la riga $(m+2)$, i cui coefficienti sono quelli della funzione obiettivo artificiale per la matrice \mathbf{A} e il valore della funzione obiettivo artificiale per il vettore \mathbf{b} , così definiti:

$$a_{m+2,1} = -\sum_{i=1}^m a_{i,1} \quad i \neq v$$

$$a_{m+2,2} = -\sum_{i=1}^m a_{i,2} \quad i \neq v$$

⋮

$$a_{m+2,n} = -\sum_{i=1}^m a_{i,n} \quad i \neq v$$

$$b_{m+2} = -\sum_{i=1}^m b_i \quad i \neq v$$

ove con $i \neq v$ si indica che la somma va effettuata solo rispetto a quelle righe che contengono una variabile artificiale.

Il metodo del simplexso prevederà quindi due fasi:

- Fase I: vengono eseguite operazioni di pivoting, con gli stessi criteri visti fino ad ora, fino a quando $b_{m+2} > 0$; in questo modo viene ottenuta una soluzione di base accettabile per il problema di partenza.
- Fase II: a questo punto, avendo una soluzione di base di partenza, è possibile far partire il metodo del simplexso come precedentemente descritto.

Si esemplifica quanto finora illustrato.

Esempio

massimizzare $5x_1 + 6x_2$

soggetta a: $2x_1 + 4x_2 \leq 8$

$3x_1 + 2x_2 \geq 6$

$x_1 - x_2 \leq 1$

$x_1 \geq 0, x_2 \geq 0$

Come primo passo il problema viene trasformato in forma standard introducendo le variabili ausiliarie x_3, x_4, x_5 e cambiando segno alla funzione obiettivo:

minimizzare $-5x_1 - 6x_2$

soggetta a: $2x_1 + 4x_2 + x_3 = 8$

$3x_1 + 2x_2 - x_4 = 6$

$x_1 - x_2 + x_5 = 1$

$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0$

Come si può notare, in questo caso l'origine ($x_1 = 0, x_2 = 0$) non è soluzione di base accettabile, in quanto si ottiene $x_4 = -6$. A questo punto viene introdotta la variabile artificiale x_6 , in modo da poter ottenere una soluzione di base, anche se ancora non accettabile per il sistema di partenza:

minimizzare $-5x_1 - 6x_2$

soggetta a: $2x_1 + 4x_2 + x_3 = 8$

$3x_1 + 2x_2 - x_4 + x_6 = 6$

$x_1 - x_2 + x_5 = 1$

$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0, x_4 \geq 0, x_5 \geq 0, x_6 \geq 0$

Viene quindi creata la tabella del simplesso, costituita da (n) colonne e (m+2) righe, e il vettore dei termini noti come illustrato in precedenza:

Tabella 1: tableau del Simplex, iterazione 0

| x_1 | x_2 | x_3 | x_4 | x_5 | b |
|-------|-------|-------|-------|-------|----|
| 2 | 4 | 1 | 0 | 0 | 8 |
| 3 | 2 | 0 | -1 | 0 | 6 |
| 1 | -1 | 0 | 0 | 1 | 1 |
| -5 | -6 | 0 | 0 | 0 | 0 |
| -3 | -2 | 0 | 1 | 0 | -6 |

FASE I

Viene utilizzata come funzione obiettivo l'ultima riga. Il valore negativo maggiore in valore e minore in valore assoluto è (-3) quindi x_1 entra nella base.

Vengono ora valutati i rapporti $\theta_1 = 8/2$, $\theta_2 = 6/3$, $\theta_3 = 1/1$; il minimo è θ_3 , quindi l'elemento di pivoting è 1 (colonna 1, riga 3). A seguito dell'operazione la

Tabella 1 risulta trasformata nella Tabella 2

Tabella 2: tableau del Simplex, iterazione 1

| x_1 | x_2 | x_3 | x_4 | x_5 | b |
|-------|-------|-------|-------|-------|----|
| 0 | 6 | 1 | 0 | -2 | 6 |
| 0 | 5 | 0 | -1 | -3 | 3 |
| 1 | -1 | 0 | 0 | 1 | 1 |
| 0 | -11 | 0 | 0 | 5 | 5 |
| 0 | -5 | 0 | 1 | 3 | -3 |

Il nuovo pivot è 5 (colonna 2, riga 2). Eseguendo il pivoting rispetto a questo elemento si ottiene:

Tabella 3: tableau del Simplex, iterazione 2

| x_1 | x_2 | x_3 | x_4 | x_5 | b |
|-------|-------|-------|-------|-------|------|
| 0 | 0 | 1 | 1.2 | 1.6 | 2.4 |
| 0 | 1 | 0 | -0.2 | -0.6 | 0.6 |
| 1 | 0 | 0 | -1.2 | 0.4 | 1.6 |
| 0 | 0 | 0 | -2.2 | -1.6 | 11.6 |
| 0 | 0 | 0 | 0 | 0 | 0 |

A questo punto si è ottenuta una soluzione di base accettabile per il sistema di partenza ed è quindi possibile passare alla fase successiva.

FASE II

Il pivot è 1,2 (colonna 4, riga 1). Eseguendo il pivoting rispetto a questo elemento si ottiene:

Tabella 4: tableau del Simplex, iterazione 3

| x_1 | x_2 | x_3 | x_4 | x_5 | b |
|-------|-------|-------|-------|--------|----|
| 0 | 0 | 0.83 | 1 | 1.33 | 2 |
| 0 | 1 | 0.166 | 0 | -0.334 | 1 |
| 1 | 0 | 0.166 | 0 | 0.666 | 2 |
| 0 | 0 | 1.82 | 0 | 1.32 | 16 |
| 0 | 0 | 0 | 0 | 0 | 0 |

La soluzione trovata è quella di ottimo, essendo tutti i coefficienti della funzione obiettivo (penultima riga) positivi.

La soluzione ottimale di base del problema è quindi:

$$x_1 = 2; x_2 = 1; x_3 = 0; x_4 = 2; x_5 = 0$$

Così come presentato però il metodo del simplesso presenta un problema: ad ogni iterazione bisogna modificare *tutti* i coefficienti della matrice **A**, anche se in realtà vengono utilizzati soltanto i coefficienti delle variabili che prima poi diventano variabili di base. Dal momento che normalmente molte variabili non diventano mai di base con tale procedimento si spreca un gran numero di calcoli. Inoltre, la matrice **A** presenta spesso una elevata sparsità, ma questa caratteristica viene meno modificando i coefficienti ad ogni iterazione.

2.2.2 Metodo del Simpleso – Algoritmo modificato

Per ovviare a questi problemi è stato studiato un *metodo del simplesso modificato*, che prevede l'aggiornamento di una matrice ausiliaria lasciando così inalterati i coefficienti della matrice **A**. Anche per questo metodo il punto di partenza è la matrice **A** modificata in modo da tenere conto delle variabili ausiliarie e artificiali, della funzione obiettivo artificiale e della funzione obiettivo reale. La matrice ausiliaria **U** ha dimensioni $(m+2) \times (m+2)$ ed è inizializzata come matrice unitaria.

Il procedimento di calcolo si articola ancora una volta in due fasi distinte: ottimizzazione della funzione obiettivo artificiale e ottimizzazione della funzione obiettivo reale.

FASE I

- i. se $b_{m+2} < 0$ vengono calcolati (per le variabili *non di base*)

$$\delta_j = \sum_{i=1}^{m+2} U_{m+2} \cdot a_{i,j} \quad (j = 1 \dots n)$$

Se $b_{m+2} \geq 0$ si passa alla fase II.

ii. Se almeno uno dei $\bar{\delta}_j < 0$ viene calcolato il minimo tra essi. Il suo indice è quello della colonna in cui si trova l'elemento pilota (che viene indicato con k).

iii. Vengono calcolati

$$x_{i,k} = \sum_{j=1}^{m+2} U_{i,j} \cdot a_{j,k} \quad (i=1 \dots m+2)$$

e, considerando solo gli $x_{i,k}$ positivi vengono calcolati

$$\theta_i = \frac{b_i}{x_{i,k}} \quad (i=1 \dots m)$$

Viene identificato il minimo tra essi ed il suo indice è quello della riga dell'elemento pilota (che viene indicato con l).

iv. Una volta così individuato l'elemento pilota vengono aggiornati la matrice ausiliaria ed il vettore dei termini noti:

$$U'_{i,j} = U_{i,j} - U_{l,j} \cdot \frac{x_{i,k}}{x_{l,k}} \quad i \neq l \quad \text{e} \quad b'_i = b_i - b_l \cdot \frac{x_{i,k}}{x_{l,k}} \quad i \neq l$$

$$U'_{l,j} = \frac{U_{l,j}}{x_{l,k}} \quad b'_l = \frac{b_l}{x_{l,k}}$$

v. Si torna al punto i.

FASE II

1. Vengono calcolati

$$\gamma_j = \sum_{i=1}^{m+2} U_{m+1,i} \cdot a_{i,j} \quad (j=1 \dots n)$$

2. Se tutti i γ_j sono positivi o nulli è stata trovata la soluzione di ottimo. Se invece almeno un γ_j è negativo viene cercato il minimo tra essi; l'indice di tale minimo è quello della colonna dell'elemento di pivot (k).

3. Vengono calcolati

$$x_{i,k} = \sum_{j=1}^{m+2} U_{i,j} \cdot a_{j,k} \quad (i=1 \dots m+2)$$

e, considerando solo gli $x_{i,k}$ positivi, vengono calcolati

$$\theta_i = \frac{b_i}{x_{i,k}} \quad (i=1 \dots m)$$

L'indice del minimo di tali valori è quello della riga dell'elemento pilota (l).

4. Una volta individuato l'elemento pilota vengono aggiornati la matrice ausiliaria ed il vettore dei termini noti:

$$U'_{i,j} = U_{i,j} - U_{l,j} \cdot \frac{x_{i,k}}{x_{l,k}} \quad (i \neq l) \qquad b'_i = b_i - b_l \cdot \frac{x_{i,k}}{x_{l,k}} \quad (i \neq l)$$

e

$$U'_{l,j} = \frac{U_{l,j}}{x_{l,k}} \qquad b'_l = \frac{b_l}{x_{l,k}}$$

5. Si torna al punto 1.

2.2.3 Metodo del Simplexso – Aspetti avanzati

Nella sezione precedente sono stati presentati gli aspetti fondamentali della programmazione lineare e del metodo del Simplexso; sono stati trascurati alcuni aspetti avanzati che possono tuttavia essere rilevanti in determinati problemi.

Nello specifico nella prossima sezione verranno presentati gli aspetti relativi alla teoria della dualità, della fattorizzazione delle matrici (le cui applicazioni si riflettono sulla programmazione lineare in generale) e alla degenerazione delle variabili di base (problematica tipica del metodo del Simplexso).

2.2.3.1 Teoria della dualità

Un problema di programmazione lineare è detto *problema primitivo* nella sua formulazione originale, ma possiede sempre un *problema duale*. La soluzione del problema primitivo permette di ottenere facilmente quella del problema duale e viceversa.

Problema primitivo e problema duale possono quindi essere definiti nel seguente modo:

| primitivo | duale |
|---|---|
| minimizzare $\mathbf{c}^T \mathbf{x}$ | minimizzare $\boldsymbol{\lambda}^T \mathbf{b}$ |
| soggetta a: $\mathbf{Ax} \geq \mathbf{b}$ | soggetta a: $\boldsymbol{\lambda}^T \mathbf{A} \leq \mathbf{c}^T$ |
| $\mathbf{x} \geq \mathbf{0}$ | $\boldsymbol{\lambda} \geq \mathbf{0}$ |

che è detta *forma simmetrica della dualità*.

È possibile ottenere una forma analoga anche partendo da un problema primitivo posto in forma standard. Infatti il problema

$$\begin{aligned} &\text{minimizzare } \mathbf{c}^T \mathbf{x} \\ &\text{soggetta a: } \mathbf{Ax} = \mathbf{b} \\ &\quad \mathbf{x} \geq \mathbf{0} \end{aligned}$$

può essere scritto nella forma equivalente

$$\begin{aligned} &\text{minimizzare } \mathbf{c}^T \mathbf{x} \\ &\text{soggetta a: } \mathbf{Ax} \geq \mathbf{b} \\ &\quad -\mathbf{Ax} \geq -\mathbf{b} \\ &\quad \mathbf{x} \geq \mathbf{0} \end{aligned}$$

Definendo un vettore suddiviso in (\mathbf{u}, \mathbf{v}) , la forma del problema duale corrispondente risulterà:

$$\begin{aligned} &\text{massimizzare } \mathbf{u}^T \mathbf{b} - \mathbf{v}^T \mathbf{b} \\ &\text{soggetta a: } \mathbf{u}^T \mathbf{A} - \mathbf{v}^T \mathbf{A} \leq \mathbf{c}^T \\ &\quad \mathbf{u} \geq \mathbf{0} \\ &\quad \mathbf{v} \geq \mathbf{0} \end{aligned}$$

Definendo $\boldsymbol{\lambda} = \mathbf{u} - \mathbf{v}$ il problema sarà dunque definito nel seguente modo:

| primitivo | duale |
|--|---|
| minimizzare $\mathbf{c}^T \mathbf{x}$ | minimizzare $\boldsymbol{\lambda}^T \mathbf{b}$ |
| soggetta a: $\mathbf{Ax} = \mathbf{b}$ | soggetta a: $\boldsymbol{\lambda}^T \mathbf{A} \leq \mathbf{c}^T$ |
| $\mathbf{x} \geq \mathbf{0}$ | $\boldsymbol{\lambda} \geq \mathbf{0}$ |

che è detta *forma asimmetrica della dualità*.

Tra il problema primitivo e il problema suo corrispondente duale esiste la seguente relazione: se \mathbf{x} è un punto accettabile del problema primitivo e \mathbf{y} un punto accettabile del problema duale, allora

$$-\mathbf{y}^T \mathbf{b} \leq \mathbf{c}^T \mathbf{x}$$

In corrispondenza della soluzione di entrambi i problemi si avrà che

$$-\mathbf{y}^T \mathbf{b} = \mathbf{c}^T \mathbf{x}$$

2.2.3.2 La Fattorizzazione LU

Utilizzando un approccio matriciale ed iterativo per l'applicazione del metodo del Simplex, questo può essere riscritto come:

1. Calcolare la soluzione attuale $\mathbf{x}_b = \mathbf{y}_0$ che soddisfi $\mathbf{B}\mathbf{y}_0 = \mathbf{b}$
2. Risolvere $\lambda^T \mathbf{B} = \mathbf{c}^T_B$ e porre $\mathbf{r}^T_D = \mathbf{c}^T_D - \lambda^T_D$; se $\mathbf{r}_D \geq \mathbf{0}$ la soluzione è quella di ottimo.
3. Determinare quale vettore \mathbf{a}_q entra nella base scegliendo quello con minor costo relativo e risolvere $\mathbf{B}\mathbf{y}_q = \mathbf{a}_q$.
4. Calcolare y_{i0}/y_{iq} per $y_{iq} > 0$; scegliere il più piccolo non negativo per determinare quale vettore lascia la base.
5. Aggiornare \mathbf{B} e tornare allo step 1.

Ove \mathbf{B} è la base, \mathbf{D} la matrice degli elementi non di base, \mathbf{C}_D sono i coefficienti della funzione obiettivo associati ad elementi non di base, \mathbf{r}_D il vettore dei costi relativi.

Dal punto di vista numerico, è molto più stabile risolvere i sistemi lineari presenti attraverso il metodo di Eliminazione di Gauss piuttosto che attraverso il pivoting. Pertanto si suppone che \mathbf{B} sia stata scomposta nella forma $\mathbf{B} = \mathbf{L}\mathbf{U}$ ove \mathbf{L} è una matrice triangolare inferiore e \mathbf{U} una matrice triangolare superiore. Ciascuno dei tre sistemi lineari ($\mathbf{B}\mathbf{y}_0 = \mathbf{b}$; $\lambda^T \mathbf{B} = \mathbf{c}^T_B$; $\mathbf{B}\mathbf{y}_q = \mathbf{a}_q$) può quindi essere risolto risolvendo due sistemi triangolari.

Viene quindi mostrato come aggiornare la forma **LU** quando cambia un vettore della base.
All'inizio del ciclo **B** assume la seguente forma

$$\mathbf{B} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m]$$

Mentre alla fine del ciclo la nuova base sarà

$$\mathbf{B} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{k-1}, \mathbf{a}_{k+1}, \dots, \mathbf{a}_m, \mathbf{a}_q]$$

ove \mathbf{a}_k ha lasciato la base, tutti gli elementi successivi sono stati spostati a sinistra e \mathbf{a}_q è stato aggiunto come ultimo elemento.

Si ha dunque che

$$\mathbf{L}^{-1}\mathbf{B} = [\mathbf{L}^{-1}\mathbf{a}_1, \mathbf{L}^{-1}\mathbf{a}_2, \dots, \mathbf{L}^{-1}\mathbf{a}_{k-1}, \mathbf{L}^{-1}\mathbf{a}_{k+1}, \dots, \mathbf{L}^{-1}\mathbf{a}_m, \mathbf{L}^{-1}\mathbf{a}_q] = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_{k-1}, \dots, \mathbf{u}_m, \mathbf{L}^{-1}\mathbf{a}_q] = \bar{\mathbf{H}}$$

Ove **H** avrà la seguente struttura:

$$\bar{\mathbf{H}} = \begin{bmatrix} / & & & & \\ & / & & & \\ & & / & & \\ & & & / & \\ & & & & / \end{bmatrix}$$

Da questa matrice è possibile ricavare $\bar{\mathbf{U}}$ applicando una serie di trasformazioni nella forma:

$$M_i = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & m_i 1 & \\ & & & & \ddots \\ & & & & & 1 \end{bmatrix}$$

per $i = k, k+1, \dots, m-1$.

La matrice \bar{U} può essere quindi espressa come

$$\bar{U} = \mathbf{M}_{m-1} \mathbf{M}_{m-2} \dots \mathbf{M}_k \bar{\mathbf{H}}$$

e di conseguenza

$$\bar{\mathbf{B}} = \mathbf{L} \bar{\mathbf{H}} = \mathbf{L} \mathbf{M}_k^{-1} \mathbf{M}_{k+1}^{-1} \dots \mathbf{M}_{m-1}^{-1} \bar{\mathbf{U}}.$$

Calcolando

$$\bar{\mathbf{L}} = \mathbf{L} \mathbf{M}_k^{-1} \dots \mathbf{M}_{m-1}^{-1}$$

si ottiene quindi la nuova base scomposta:

$$\bar{\mathbf{B}} = \bar{\mathbf{L}} \bar{\mathbf{U}}$$

2.2.3.3 Degenerazione

Le variabili di base all'interno di una soluzione di base non sono necessariamente tutte non-negative. Si possono cioè avere delle soluzioni di base degeneri. Per definizione:

“Se una o più variabili di base in una soluzione di base hanno valore zero, tale soluzione è detta *soluzione di base degenera*”.

Durante la procedura del metodo del Simplex è possibile incontrare soluzioni di base accettabili degeneri. Spesso è possibile trattarle come soluzioni di base accettabili non degeneri. Tuttavia, è possibile che dopo che una nuova colonna q è stata selezionata per entrare nella base, il minimo dei rapporti y_{i0}/y_{iq} sia zero, facendo sì che sia la variabile di base che ha valore zero a lasciare la base. Questo significa che la nuova variabile x_q entrerà con valore zero, la funzione obiettivo non diminuirà e la nuova soluzione di base sarà anch'essa degenera. È possibile che questo processo continui per una serie di cicli fino a che, finalmente, la soluzione degenera di partenza viene ottenuta di nuovo. Il risultato è un ciclo che può essere ripetuto all'infinito ed assume il nome di *cycling loop problem*, identificato e definito da Beale (1955)). Sono stati sviluppati metodi per evitare tali cicli.

Nella pratica, comunque, il problema di *cycling loop* è considerato piuttosto raro da alcuni autori, i quali ritengono tali procedure spesso non necessarie. Tuttavia, è importante sottolineare uno dei maggiori caveat dell'analisi numerica:

“È un errore credere che un evento molto raro sia al tempo stesso molto importante.”

Essendo, quindi, le procedure anti-ciclo relativamente semplici, molti codici le incorporano per garantire la sicurezza dei programmi di calcolo. È importante capire come il fatto di entrare in un ciclo infinito sia strettamente correlato al criterio di pivoting descritto all'interno della procedura del Simplex. Modificando, infatti, tale criterio è possibile garantire che il metodo del Simplex arrivi sempre ad una soluzione senza cadere in cicli infiniti. Uno dei criteri più semplici per raggiungere tale scopo è la *Regola di Bland*, che stabilisce che la variabile che entra nella base e quella che esce dalla base devono essere selezionate, all'interno dei loro set di scelta, in maniera tale da avere l'indice più piccolo possibile. E', infatti, possibile dimostrare la validità del seguente teorema:

“Il metodo del Simplex giunge sempre a termine a patto che sia la variabile che lascia la base sia quella che vi entra siano scelte seguendo la regola di Bland”.

2.3 Il metodo dell'Interior Point

Uno dei problemi principali che presenta il metodo del Simplex è quello dei tempi di calcolo: è stato dimostrato che, nei casi peggiori, tale algoritmo richiede un numero di iterazioni che è esponenziale rispetto al numero di variabili.

Caso emblematico in questo senso è la classe di esempi di Klee-Minty:

$$\begin{aligned} &\text{massimizzare } \sum_{j=1}^n 10^{n-j} x_j \\ &\text{soggetta a: } 2 \sum_{j=1}^{i-1} 10^{i-j} x_j + x_i \leq 100^{i-j} \quad i = 1, \dots, n \\ & \quad \quad \quad x_j \geq 0 \quad j = 1, \dots, n \end{aligned}$$

Questo tipo di problema richiede $2^n - 1$ operazioni di pivoting; nel caso a 50 variabili un PC che esegua 1e6 pivot al secondo impiegherebbe 33 anni a risolverlo.

La ricerca di un algoritmo capace di risolvere ogni tipo di problema con un numero massimo di iterazioni polinomiale rispetto al numero di variabili è proseguita cercando di adattare i metodi della programmazione non lineare alla programmazione lineare. Ne è un esempio il **Metodo dell'Ellissoide**.

2.3.1 Il metodo dell'Ellissoide

L'idea che sta dietro a tale metodo è la seguente: risolvere un problema lineare può essere considerato equivalente al trovare un punto dello spazio poliedrico Ω così definito:

$$\Omega = \{\mathbf{y} \in E^m : \mathbf{y}^T \mathbf{a}_j \leq c_j, j = 1, \dots, n\}$$

con le seguenti assunzioni:

- Esistono un vettore $\mathbf{y}_0 \in E^m$ ed uno scalare $R > 0$ tali che la palla $S(\mathbf{y}_0, R)$ di centro \mathbf{y}_0 e raggio R contiene Ω , ovvero $\{\mathbf{y} \in E^m : |\mathbf{y} - \mathbf{y}_0| \leq R\}$ contiene Ω .
- Se Ω è non-vuoto, esso contiene una palla $S(\mathbf{y}^*, r)$.

In E^m può quindi essere definito un *ellissoide* nel seguente modo:

$$E = \{\mathbf{y} \in E^m : (\mathbf{y} - \mathbf{z})^T \mathbf{Q} (\mathbf{y} - \mathbf{z}) \leq 1\}$$

ove $\mathbf{z} \in E^m$ è un punto noto detto *centro* e \mathbf{Q} una matrice definita positiva. Gli autovalori di \mathbf{Q} sono gli assi dell'ellissoide.

Ad ogni iterazione dell'algoritmo, si avrà che $\Omega \subset E^k$. Se $\mathbf{y}_k \in \Omega$ il problema è risolto; in caso contrario significa che almeno un vincolo è stato violato. In tal caso si procede dividendo in due l'ellissoide attraverso il centro:

$$\Omega \subset \frac{1}{2} E_k = \{\mathbf{y} \in E_k : \mathbf{a}_j^T \mathbf{y} \leq \mathbf{a}_j^T \mathbf{y}_k\}$$

L'ellissoide successivo E_{k+1} sarà quello di minimo volume tale che $\frac{1}{2} E_k \subset E_{k+1}$.

2.3.2 Interior Point

Nel 1984 Karmakar propone l' algoritmo dell'Interior Point: anziché basarsi sui vertici del semplice, tale algoritmo si muove all'interno di esso convergendo verso il vertice corrispondente alla soluzione.

Come detto tale algoritmo sfrutta le tecniche della programmazione non lineare, introducendo il concetto di *barriera* o *potenziale* per definire la convergenza alla soluzione.

Il primo concetto fondamentale, associato a quello di barriera, è quello di *centro analitico*.

Viene considerato uno spazio S in un sottospazio \mathcal{X} di E^n definito da un gruppo di disuguaglianze:

$$S = \{x \in \mathcal{X} : g_j(\mathbf{x}) \geq 0, j = 1, 2, \dots, m\}$$

Le funzioni g_i sono continue e S ha un interno:

$$S^0 = \{x \in \mathcal{X} : g_j(\mathbf{x}) > 0, j = 1, 2, \dots, m\}.$$

Ad esso è associata una funzione potenziale, definita su S^0 :

$$\psi(\mathbf{x}) = -\sum_{j=1}^m \log g_j(\mathbf{x}).$$

Il centro analitico è quindi il vettore (o insieme di vettori) che minimizza il potenziale.

Tutti questi concetti possono quindi essere applicati alla programmazione lineare nel seguente modo.

Considerato un programma lineare primitivo in forma standard:

$$\begin{aligned} &\text{minimizzare } \mathbf{c}^T \mathbf{x} \\ &\text{soggetta a: } \mathbf{Ax} = \mathbf{b} \\ &\quad \mathbf{x} \geq 0 \end{aligned}$$

può essere definito un *barrier problem* associato ad esso ($\mu \geq 0$):

$$\text{minimizzare } \mathbf{c}^T \mathbf{x} - \mu \sum_{j=1}^n \log x_j$$

$$\text{soggetta a: } \mathbf{Ax} = \mathbf{b}$$

$$\mathbf{x} \geq 0$$

Teoricamente sarebbe sufficiente risolvere tale problema per un valore molto piccolo di μ . Tuttavia, ciò potrebbe essere difficoltoso a causa del mal condizionamento del sistema. Una strategia è, quindi, quella di risolvere il problema per un valore abbastanza elevato di μ (per esempio $\mu = 100$) ed utilizzare la soluzione così ottenuta come punto di partenza per la nuova iterazione, che presenta un valore di μ minore.

Ad ogni iterazione μ viene diminuito di un fattore γ costante ($\gamma < 1$); in questo modo per ridurre $\frac{\mu_k}{\mu_0}$ al di sotto del valore voluto ε saranno necessari $k = \frac{\log \varepsilon}{\log \gamma}$ passi.

Per ogni valore di μ viene quindi trovato un punto sempre più vicino alla soluzione del problema originale. L'insieme di tali punti prende il nome di *central path*.

Per risolvere ciascun *barrier problem* viene generalmente utilizzato il Metodo di Newton: ad ogni iterazione viene calcolata la direzione di ricerca ottimale \mathbf{d}_x in maniera tale che ad ogni iterazione del metodo si avrà un nuovo punto $\mathbf{x}_{\text{new}} = \mathbf{x}_{\text{old}} + \mathbf{d}_x$. Ogni iterazione richiede il calcolo di tre parametri ($\mathbf{d}_x, \mathbf{d}_y, \mathbf{d}_z$) attraverso la risoluzione di altrettanti sistemi lineari.

Per prima cosa viene ottenuto \mathbf{d}_y risolvendo il seguente sistema:

$$(\mathbf{AX}^2\mathbf{A}^T)\mathbf{d}_y = -\mu\mathbf{AX}\mathbf{I} + \mathbf{AX}^2\mathbf{c}$$

A questo punto viene calcolato \mathbf{d}_s risolvendo un secondo sistema:

$$-\mathbf{A}^T\mathbf{d}_y - \mathbf{d}_s = 0$$

Infine viene calcolato \mathbf{d}_x dalla risoluzione di un terzo sistema:

$$\mu\mathbf{X}^{-2}\mathbf{d}_x + \mathbf{d}_s = \mu\mathbf{X}^{-1}\mathbf{I} - \mathbf{c}$$

Il procedimento viene iterato fino a che $\mathbf{d} \equiv (\mathbf{d}_x, \mathbf{d}_y, \mathbf{d}_z) = \mathbf{0}$, ovvero \mathbf{x} è la soluzione lungo il *central path* per il valore di μ dato.

2.3.2.1 Interior Point e dualità

Definito il problema primitivo

$$\begin{aligned} &\text{minimizzare } \mathbf{c}^T \mathbf{x} \\ &\text{soggetta a: } \mathbf{Ax} = \mathbf{b} \\ &\quad \mathbf{x} \geq 0 \end{aligned}$$

ed il suo problema duale

$$\begin{aligned} &\text{minimizzare } \mathbf{y}^T \mathbf{b} \\ &\text{soggetta a: } \mathbf{y}^T \mathbf{A} \geq \mathbf{c}^T \\ &\quad \mathbf{y} \geq 0 \end{aligned}$$

in ogni istante è possibile sapere quanto ci si trova lontano dalla soluzione di ottimo calcolando quello che viene definito *duality gap*:

$$\mathbf{c}^T \mathbf{x} - \mathbf{y}^T \mathbf{b} = \mathbf{s}^T \mathbf{x}$$

Ricordando che per la teoria della dualità nella soluzione di ottimo del problema il valore della funzione obiettivo del problema primitivo coincide con il valore obiettivo del problema duale, nel punto di ottimo si avrà che $\mathbf{s}^T \mathbf{x} = 0$.

Dal momento che $\mathbf{s}^T \mathbf{x} = n\mu$, ancora una volta è evidente come a mano a mano che $\mu \rightarrow 0$ ci si avvicini sempre di più alla soluzione di ottimo.

Capitolo 3 – Il metodo dell’Attico

In questo capitolo verrà illustrato il metodo dell’Attico, partendo ancora una volta dalle intuizioni geometriche che hanno portato al suo sviluppo per poi definire più in dettaglio dal punto di vista matematico il suo funzionamento.

3.1 Introduzione

Il nome di questo metodo deriva da un’interpretazione geometrica della procedura; per semplicità, consideriamo un problema dove sono presenti solo disuguaglianze. Ogni vincolo è un iperpiano nello spazio delle variabili e, se esiste una soluzione accettabile, questi vincoli limitano in qualche modo la regione accettabile creando una sorta di cupola formata da diverse facce, in maniera molto simile alle tegole di un tetto. Nei problemi di massimizzazione la soluzione è un vertice e corrisponde alla connessione di 3 tegole (vincoli) adiacenti.

L’idea su cui si basa il metodo dell’Attico è che qualunque punto accettabile (vertice o meno) vede il vertice su cui si trova la soluzione, ovvero può essere congiunto alla soluzione da una linea grazie alla convessità dello spazio (o attico), garantita dalla completa linearità di funzione obiettivo e vincoli. Spesso, tale linea deve allontanarsi da uno o più vincoli su cui si trova il punto di lavoro e passare attraverso l’attico.

Nel metodo dell’Attico viene dunque usata una direzione che si muove all’interno della regione accettabile e lungo la quale la funzione migliora e la ricerca viene fermata quando si incontra un nuovo vincolo. La direzione di ricerca è selezionata cercando il massimo miglioramento della funzione. Il nuovo punto generalmente non è un vertice e il numero di vincoli attivi per ogni iterazione è generalmente più piccolo del numero di vincoli necessario per un vertice. Iterando la procedura, il numero di vincoli attivi solo qualche volta, ma comunque di rado, eguaglia le dimensioni del LP n_V , riducendo drasticamente il rischio di degenerazione.

3.2 Concetti base

Il LP consiste nella ricerca del minimo della funzione $F(\mathbf{x})=\mathbf{s}^T\mathbf{x}$ con n_V variabili soggette a n_E vincoli di uguaglianza linearmente indipendenti $\mathbf{E}\mathbf{x}=\mathbf{e}$ e n_D vincoli di disuguaglianza $\mathbf{D}\mathbf{x}\leq\mathbf{c}$. Alcuni di questi vincoli di disuguaglianza possono essere vincoli semplici su una singola variabile: per esempio limiti superiori ($x_i\leq u_i$) e limiti inferiori ($-x_i\leq -u_i$). Il numero di limiti inferiori n_L è importante per definire:

- Il numero di variabili senza limite inferiore: $n_F = n_V - n_L$.
- Il numero di vincoli di disuguaglianza che non sono limiti inferiori: $k_L = n_D - n_L$.
- Il numero totale di vincoli che non sono vincoli inferiori: $k_V = n_E + k_L$.

Questa valutazione è molto importante dal momento che permette di definire se risolvere il problema primitivo (se $n_V < k_V$) oppure il problema duale ($n_V > k_V$). Infatti in questo modo viene scelto il problema soggetto al minor numero di vincoli e, di conseguenza, che presenta la matrice di lavoro più piccola, diminuendo ulteriormente lo sforzo computazionale del metodo.

3.3 Il metodo dell'Attico

Supponendo di avere n_E vincoli di uguaglianza e almeno $n_Q = n_V - n_E$ vincoli di disuguaglianza soddisfatti nel punto di lavoro \mathbf{x} , le condizioni di Karush, Kuhn e Tucker (KKT) prevedono che:

$$\mathbf{J}^T \boldsymbol{\lambda} = -\mathbf{s}$$

$$\mathbf{J}\mathbf{x} = \mathbf{b}$$

$$\lambda_i \geq 0 \quad (i = 1, \dots, n_Q)$$

Ove la matrice non singolare \mathbf{J} è dimensionata ($n_V \times n_V$) ed è formata da tutti i vincoli di equazione linearmente indipendenti e una selezione di vincoli di disuguaglianza che riempiono le rimanenti $n_Q = n_V - n_E$ righe; \mathbf{s} è il vettore dei coefficienti della funzione obiettivo e \mathbf{b} è il vettore dei termini noti dei vincoli presenti nella matrice \mathbf{J} . Se risulta che $\lambda_i < 0$ per qualche vincolo di disuguaglianza attivo, le righe corrispondenti possono essere rimosse dal sistema, rendendo tali vincoli passivi, dal momento che la funzione obiettivo migliora abbandonando questi vincoli.

All'interno del metodo dell'Attico, è fondamentale distinguere questi tipi di variabili:

- Vincoli di uguaglianza: sono sempre presenti nella matrice **J**, devono sempre essere soddisfatti.
- Vincoli di disuguaglianza passivi: non sono mai presenti in **J** dal momento che il punto di lavoro è interno a questi vincoli.
- Vincoli di disuguaglianza soddisfatti: il punto di lavoro giace su questi vincoli e alcuni di questi possono essere presenti all'interno della matrice **J**.
- Vincoli di disuguaglianza attivi: i vincoli di disuguaglianza presenti in **J**.
- Vincoli di disuguaglianza non soddisfatti: il punto di lavoro li viola.
- Vincoli di disuguaglianza artificiali: sono vincoli su una singola variabile; dal momento che non rappresentano vincoli reali devono essere trattati diversamente quando sono presenti in **J**.

È inoltre fondamentale distinguere fra vertici dell'Attico e vertici tradizionali: il vertice dell'Attico può coinvolgere un numero più piccolo di variabili e soddisfa n vincoli con $n \geq n_v$ e tutte le righe di **J** sono vincoli reali. Questo è possibile solo se il rango m della matrice **J** è pari a n_v .

È inoltre importante distinguere fra vincoli cosiddetti "di pavimento" e vincoli "di tetto" per ciascuna variabile. Un vincolo di disuguaglianza dove una variabile ha un coefficiente di segno opposto al corrispondente nella funzione obiettivo è un vincolo di tetto per quella variabile; in caso contrario, essa è una variabile di pavimento. Tutte le variabili presenti nei vincoli di uguaglianza hanno un vincolo di tetto mentre un vincolo artificiale è un vincolo di pavimento per tutte le variabili.

È ora il momento di esaminare la matrice **J** e la sua fattorizzazione. In un punto accettabile tale matrice è formata da n_E righe di vincoli di uguaglianza e n_W righe di vincoli di disuguaglianza attivi. Le rimanenti $n_v - (n_E + n_W)$ righe sono formate da vincoli artificiali appositamente selezionati. Tali righe hanno un singolo coefficiente non-nullo pari a 1, posizionato in modo tale da rendere la matrice **J** non singolare e ben condizionata.

In un'iterazione generica, solo la i -esima riga della matrice \mathbf{J} , con $1 \leq i \leq n_v$, deve essere fattorizzata; tutte le righe con un indice più piccolo di i sono già state fattorizzate in iterazioni precedenti e non sono modificate ulteriormente nella iterazione corrente, le righe con indice più grande di i non sono usate né modificate. Quando una riga è fattorizzata, la variabile che rende la fattorizzazione più stabile è selezionata come pivot.

Esaminiamo più in dettaglio quello che accade nella generica iterazione del metodo.

In un punto accettabile si risolve il seguente sistema: $\mathbf{J}^T \lambda = -\mathbf{s}$. L'iterazione continua se λ_a è non-nulla per qualche vincolo artificiale o se qualche vincolo di disuguaglianza attivo presenta $\lambda_i < 0$. In base ai valori di λ vengono selezionati gli elementi del vettore \mathbf{h} e viene risolto il seguente sistema $\mathbf{J}\mathbf{d} = \mathbf{h}$ al fine di trovare la direzione di ricerca \mathbf{d} . La scelta di \mathbf{h} comporta i seguenti effetti:

- Nessuno degli n_w vincoli di disuguaglianza attivi viene rimosso. Questo è possibile solo se il punto di lavoro non è un vertice dell'Attico. Un nuovo vincolo è inserito alla riga $n_E + n_w + 1$.
- $n_z \geq 1$ vincoli di disuguaglianza attivi sono rimossi: perché sono diventati passivi, al fine di entrare nello spazio delle soluzioni accettabili, per sfruttare la sparsità delle righe da fattorizzare o prevenire problemi di degenerazione. Le n_z righe selezionate sono le ultime delle n_w righe fattorizzate. Sono rimpiazzate da un nuovo vincolo di disuguaglianza attivo e $n_z - 1$ vincoli artificiali.

Una volta risolto il sistema $\mathbf{J}\mathbf{d} = \mathbf{h}$ e calcolata la direzione di ricerca \mathbf{d} , è possibile ottenere sia il nuovo punto di lavoro accettabile che il vincolo di disuguaglianza da rendere attivo.

Vengono inizialmente valutati i vettori \mathbf{p} e \mathbf{q} :

$$\mathbf{p} = \mathbf{D}\mathbf{d}$$

$$\mathbf{q} = \mathbf{c} - \mathbf{D}\mathbf{x}_i$$

Viene poi calcolato il valore t_{\min} , ovvero il minimo valore di $t_j = \frac{q_j}{p_j}$ con $p_j > 0$ e j non

presente in \mathbf{J} . Il vincolo per cui viene ottenuto t_{\min} è inserito come nuovo vincolo e il corrispondente valore di $\mathbf{x} = \mathbf{x}_i + t\mathbf{d}$ diventa il nuovo punto di lavoro accettabile.

È ora il momento di esaminare più in dettaglio come selezionare il vettore \mathbf{h} . Se $h_i=0$ il corrispondente vincolo rimane soddisfatto; pertanto tutti i vincoli di uguaglianza sono caratterizzati da $h_e = 0$. Se $h_i < 0$ viene imposto per un vincolo di disuguaglianza+, questo viene reso passivo. Non è tuttavia possibile assegnare un valore $h_i > 0$ per un vincolo di disuguaglianza attivo perché in tal caso la direzione di ricerca violerebbe tale vincolo.

La direzione \mathbf{d} permette un miglioramento della funzione obiettivo se risulta che $\mathbf{g}^T \mathbf{d} < 0$. Dal momento che già sappiamo che $\mathbf{J}^T \boldsymbol{\lambda} = -\mathbf{s}$ e $\mathbf{J}\mathbf{d} = \mathbf{h}$ ne segue che $\mathbf{g}^T \mathbf{d} = \mathbf{s}^T \mathbf{d} = -\boldsymbol{\lambda}^T \mathbf{J}\mathbf{J}^{-1} \mathbf{h} = -\boldsymbol{\lambda}^T \mathbf{h}$.

L'ultimo prodotto viene quindi diviso in tre parti: $\sum_a \lambda_a h_a + \sum_j \lambda_j h_j + \sum_k \lambda_k h_k$.

- La prima parte è quella che raccoglie i vincoli artificiali; è possibile scegliere il valore di h_a pari a una costante con lo stesso segno di λ_a -
- La seconda parte raccoglie i vincoli di disuguaglianza con un valore negativo di λ_j ; è possibile scegliere il valore di h_j pari a una costante con lo stesso segno di λ_j .
- La terza parte raccoglie i vincoli di disuguaglianza con un valore positivo di λ_k ; è possibile scegliere il valore di h_k pari a una costante con lo stesso segno di λ_k .

Il vettore \mathbf{h} può quindi essere costruito semplicemente ponendo $h_e = 0$, $h_a = \lambda_a$, $h_j = \lambda_j$ e $h_k = \lambda_k$ in modo tale che la direzione di ricerca massimizzi la diminuzione della funzione obiettivo e soddisfi tutti i vincoli di uguaglianza e disuguaglianza attivi.

Al fine di prevenire movimenti troppo piccoli nella procedura di ottimizzazione è possibile valutare un valore soglia per cui la direzione \mathbf{d} renda passivi anche quei vincoli con $\lambda_k > 0$.

Questo è possibile imponendo $\mathbf{g}^T \mathbf{d} = \mathbf{s}^T \mathbf{d}$ e quindi selezionando $h_a = \lambda_a$, $h_j = -1$, $h_k = -\alpha_{max}$

$$\text{ove } \alpha_{max} = \frac{\sum_a \lambda_a^2 + \sum_j |\lambda_j|}{\sum_k |\lambda_k|}.$$

La scelta di \mathbf{h} deve tenere in considerazione due contributi connessi ma opposti: il costo di ogni singola iterazione e il numero totale di iterazioni. Il metodo dell'Attico tiene in considerazione tre diverse situazioni:

- **Il punto di lavoro non è quello iniziale e ci sono alcuni vincoli artificiali con $\lambda_a \neq 0$ nella matrice \mathbf{J} .** Non viene rimosso alcun vincolo di disuguaglianza e \mathbf{h} è formato da $h_a = \lambda_a$, $h_e = 0$, $h_j = 0$, $h_k = 0$. Non è fatto alcun movimento ma si ricade nella situazione successiva.
- **Il punto di lavoro non è quello iniziale e non ci sono vincoli artificiali o tutti i vincoli artificiali hanno $\lambda_a = 0$.** Sono effettuate le seguenti ricerche:
 - Massimo miglioramento locale: $h_j = \lambda_j$, $h_e = h_a = h_k = 0$.
 - Seconda ricerca: $h_j = -1$, $h_k = 0.8 \alpha_{max}$ (per tutti i vincoli di disuguaglianza con $\lambda_k > 0$ presenti nella matrice \mathbf{J} , che si trovino sotto il primo vincolo con $\lambda_j < 0$ e che non siano mai usciti e rientrati nella matrice \mathbf{J}), mentre tutti gli altri valori di \mathbf{h} sono nulli.

Vengono rimossi tutti i vincoli attivi partendo da quelli con indice più basso nella matrice \mathbf{J} tra quelli che sono diventati passivi. I vincoli rimossi sono sostituiti dall'unico vincolo risultato la migliore opzione tra le due ricerche precedenti. Tutti i rimanenti vincoli diventano vincoli artificiali.

- **Il punto di lavoro è il punto iniziale.** Bisogna verificare se alcune variabili che non appaiono in nessun vincolo di uguaglianza abbiano lo stesso segno nella funzione obiettivo e in tutti i vincoli di disuguaglianza soddisfatti: tali variabili giacciono solo su vincoli pavimento. In questo caso, una ricerca monodimensionale è effettuata sulla variabile con tale caratteristica e iterata per tutte le altre possibili variabili utilizzando lo stesso punto iniziale. Viene selezionato il nuovo punto che presenti la miglior funzione obiettivo. Una volta effettuate queste ricerche, \mathbf{J} contiene solo vincoli di uguaglianza e la direzione \mathbf{d} viene ottenuta imponendo $h_e = 0$ e $h_a = \lambda_a$.

Dal punto di vista dello sforzo computazionale, ogni iterazione richiederà:

- Il prodotto di una matrice sparsa \mathbf{D} e di un vettore \mathbf{d} ; il prodotto deve sfruttare la sparsità della matrice.
- La ricerca del valore t_{min} .
- La fattorizzazione dell'ultima riga inserita nell'iterazione specifica.
- La soluzione dei sistemi $\mathbf{J}^T \lambda = -\mathbf{s}$ e $\mathbf{J}\mathbf{d} = \mathbf{h}$, sfruttando la fattorizzazione della matrice \mathbf{J} .

3.4 Differenze rispetto agli approcci tradizionali.

Il metodo dell'attico si differenzia rispetto ai metodi tradizionali in 5 caratteristiche principali:

1. Il metodo dell'Attico non richiede che il problema sia scritto in forma standard e pertanto non richiede l'aggiunta di nuove variabili. Le dimensioni della matrice \mathbf{J} sono in ogni iterazioni uguali a quelle del problema originale n_v .
2. Tutte le variabili e i termini noti dei vincoli non devono più essere obbligatoriamente non-negativi.

Queste due differenze portano ai seguenti vantaggi:

- La matrice \mathbf{J} utilizzata nel metodo dell'Attico può essere significativamente più piccola della matrice utilizzata nei metodi tradizionali. Ciò comporta una performance di soluzione superiore non solo per il ridotto numero delle iterazioni, ma anche per il tempo computazionale richiesto dalla singola iterazione.
- Alcuni vincoli di disuguaglianza possono essere sempre esclusi dalla matrice \mathbf{J} . Pertanto, anche se il problema presenta un elevato numero di vincoli di disuguaglianza solo una piccola parte verrà utilizzata per la ricerca dell'ottimo.
- La formulazione del problema è più immediata e naturale.

3. Alcune delle righe della matrice \mathbf{J} possono essere artificiali. Questa è la vera e più importante differenza rispetto ai metodi tradizionali e rende il metodo dell'Attico più flessibile e potenzialmente più efficiente. Alcune delle principali conseguenze di tale caratteristica sono:
- Il punto di lavoro può non essere un vertice e quindi non è necessario muoversi da un vertice all'altro.
 - Diversi vincoli di disuguaglianza attivi possono essere rimossi contemporaneamente.
 - Vi sono diverse tecniche alternative per cercare il miglior punto successivo.
 - Nella matrice \mathbf{J} , possono essere inclusi solo i vincoli di uguaglianza e solo alcuni (se non nessuno) vincoli di disuguaglianza attivi, selezionati tra i vincoli di disuguaglianza soddisfatti.
 - Nel metodo del simplesso, un vertice è considerato solo come punto di lavoro; nell'Attico, un vertice è l'intersezione di n_V vincoli.
 - In uno spazio convesso, ogni punto accettabile può essere connesso alla soluzione mediante una linea retta. Spesso, tale linea deve allontanarsi da alcuni vincoli su cui si trova il punto di lavoro e passare attraverso l'attico per raggiungere la soluzione: il metodo dell'Attico può farlo quando è vantaggioso, il metodo del Simplexso no.
4. Come precedentemente menzionato, nel metodo dell'Attico è importante distinguere tra vertici dell'attico e punti interni, tra diversi tipi di vincolo, tra variabili di tetto o pavimento.
5. Il metodo dell'Attico non richiede alcuna discriminazione tra variabili di base o non di base come nel metodo del Simplexso; non richiede l'utilizzo di un sentiero centrale come per il metodo dell'Interior Point.

È inoltre possibile evidenziare alcuni vantaggi aggiuntivi rispetto al metodo del Simplexso:

- Dal momento che il punto di lavoro può essere un non vertice è possibile rimuovere più vincoli simultaneamente. Questo può aiutare a trovare un nuovo punto migliore del vertice adiacente.
- Dal momento che è possibile muoversi su punti non vertici, il rischio di ottenere una lunga serie di vertici adiacenti può essere spesso evitato o in qualche modo ridotto.
- La possibilità di includere nella matrice \mathbf{J} solo alcuni vincoli di disuguaglianza può aiutare a prevenire problemi di degenerazione.
- Il numero di iterazioni richieste è sempre minore o uguale a quelle del metodo del Simplexso.
- Alcuni punti che il metodo del Simplexso potrebbe considerare vengono invece rimossi a priori dal metodo dell'Attico.
- Il punto iniziale può essere qualunque punto, incluso un non vertice dell'attico. Se è accettabile e interno, è possibile effettuare ricerche lungo gli assi. Molti punti che giacciono sulla frontiera dello spazio accettabile possono sfruttare tale caratteristica.
- Nel metodo del Simplexso, una colonna è inserita nella matrice di base ed un'altra esce ad ogni iterazione. Questo può condurre a problemi nell'aggiornamento della matrice in termine di memoria e stabilità. Nel metodo dell'Attico invece viene inserita una riga per volta ma immediatamente dopo le righe già fattorizzate, non in mezzo ad esse. La fattorizzazione della nuova riga può pertanto essere efficacemente effettuata sfruttando la fattorizzazione già esistente delle righe già presenti nella matrice e la sparsità delle righe sia nuove che esistenti.
- Ad ogni iterazione, se il punto di lavoro non è un vertice dell'attico non vengono rimossi vincoli; viceversa possono essere rimossi diversi vincoli quando utile e vengono rimpiazzati da vincoli artificiali. Questo limita l'esplosione dei vertici intermedi e i problemi di degenerazione.

- È più facile ottenere un punto iniziale accettabile dal momento che può non essere un vertice.
- Il punto iniziale può essere interno. Questo risulta vantaggioso in due situazioni: quando vogliamo sfruttare un punto come primo tentativo per la soluzione di un problema leggermente diverso; quando il punto di partenza è un vertice degenere, è possibile allontanarsi facilmente.

Capitolo 4 - Caso studio

Ottimizzazione di una Thermal Cracking Refinery

In questo capitolo verrà illustrato un caso studio, utilizzato al fine di confrontare le prestazioni dei diversi algoritmi attualmente disponibili con quelle del metodo dell'Attico.

Vista l'importanza di convalidare il nuovo metodo dell'Attico e di confrontarlo con i metodi esistenti, e quindi l'importanza di evitare ogni tipo di problema numerico non inerente alla mera programmazione lineare, per semplicità dello schema e chiarezza e completezza dei dati forniti, l'esempio selezionato è quello illustrato da Manne nel suo libro "Scheduling of petroleum refinery operations" (1963).

Lo schema del processo descritto è rappresentato in Figura 3.

La raffineria considerata ha come obiettivo la produzione di diversi tipi di prodotti, ovvero:

1. Benzina premium (Alto numero di ottani, Research)
2. Benzina (basso numero di ottani, Research)
3. Propano e propilene (taglio C3)
4. Butano e butilene (taglio C4)
5. Kerosene
6. Olio combustibile 1
7. Olio combustibile 2
8. Combustibile Diesel premium (numero di cetano 55)
9. Olio combustibile 6
10. Gas combustibili (C2 e leggeri)

L'alimentazione è costituita da un grezzo Mid-Continent di densità 38°API. Tale grezzo ha le proprietà illustrate in Tabella 5.

Tabella 5: Tagli del greggio Mid-Continent a 38.0° API

| Taglio | Temperatura di distillazione del 10% (°F) | Punto finale (°F) | Percentuale massiva del grezzo | Percentuale massiva cumulata |
|--------------------|---|-------------------|--------------------------------|------------------------------|
| C2 e leggeri | - | - | 0.02 | 0.02 |
| Gas C3 | - | - | 0.22 | 0.24 |
| Gas C4 | - | - | 1.46 | 1.70 |
| Benzina | 150 | 400 | 24.8 | 26.5 |
| Distillati leggeri | 400 | 520 | 14.6 | 41.1 |
| Distillati medi | 510 | 630 | 9.5 | 50.6 |
| Distillati pesanti | 620 | 720 | 12.3 | 62.9 |
| Residuo | - | - | 37.1 | 100.0 |

La colonna di distillazione primaria è una colonna di distillazione standard, per cui non è necessaria l'introduzione di variabili arbitrarie, necessità che invece si presenta nelle successive sezioni di cracking, riciclo e blending come indicato nel seguito nella formulazione del modello matematico. La sezione di cracking è costituita da quattro forni e una colonna di frazionamento secondaria. È predisposto il riciclo di porzioni delle tre correnti di distillato. Dalla testa della colonna si ottengono benzina, gas e vapore usato per lo stripping. Tali prodotti vengono inviati ad un condensatore parziale dove benzina e vapore vengono condensati. Proseguendo con l'illustrazione delle proprietà delle diverse correnti dell'impianto, quelle prima del blending sono presentate in Tabella 6:

Tabella 6: Prodotti, prima del blending

| Materiale | Temperatura di distillazione del 10% (°F) | Punto finale (°F) | Densità specifica | NO (benzine) e numero di cetano (distillati) | Viscosità | | | |
|--------------------|---|-------------------|-------------------|--|--------------|--------|--------------|-----------------|
| | | | | | a 100 °F | | A 122 °F | |
| | | | | | centi stokes | μi | centi stokes | μi |
| Benzine | | | | | | | | |
| Straight run | 150 | 400 | 0.735 | NO 58, clear | - | - | - | - |
| Cracked | 150 | 400 | 0.755 | NO 75, clear | - | - | - | - |
| Distillati leggeri | | | | | | | | |
| Straight run | 400 | 520 | 0.816 | NC 53 | 1.73 | -0.422 | 1.44 | - 0.52 14 |
| Cracked | 400 | 520 | 0.840 | NC 35 | 1.75 | -0.419 | 1.45 | - 0.49 19 |
| Distillati medi | | | | | | | | |
| Straight run | 510 | 630 | 0.830 | NC 58 | 3.43 | -0.215 | 2.6 | - 0.29 65 |
| Cracked | 510 | 630 | 0.865 | NC 35 | 3.37 | -0.220 | 2.65 | - 0.28 53 |
| Distillati pesanti | | | | | | | | |
| Straight run | 620 | 720 | 0.855 | NC 61 | 7.4 | -0.044 | 5.1 | - 0.12 15 |
| Cracked | 620 | 720 | 0.916 | NC 40 | 11.2 | 0.030 | 7.5 | - 0.04 17 |
| Residuo | | | | | | | | |
| Straight run | - | - | 0.944 | - | - | - | 88 | 0.28 95 |
| Cracked | - | - | 1.022 | - | - | - | 3.3 | 0.54 64 |

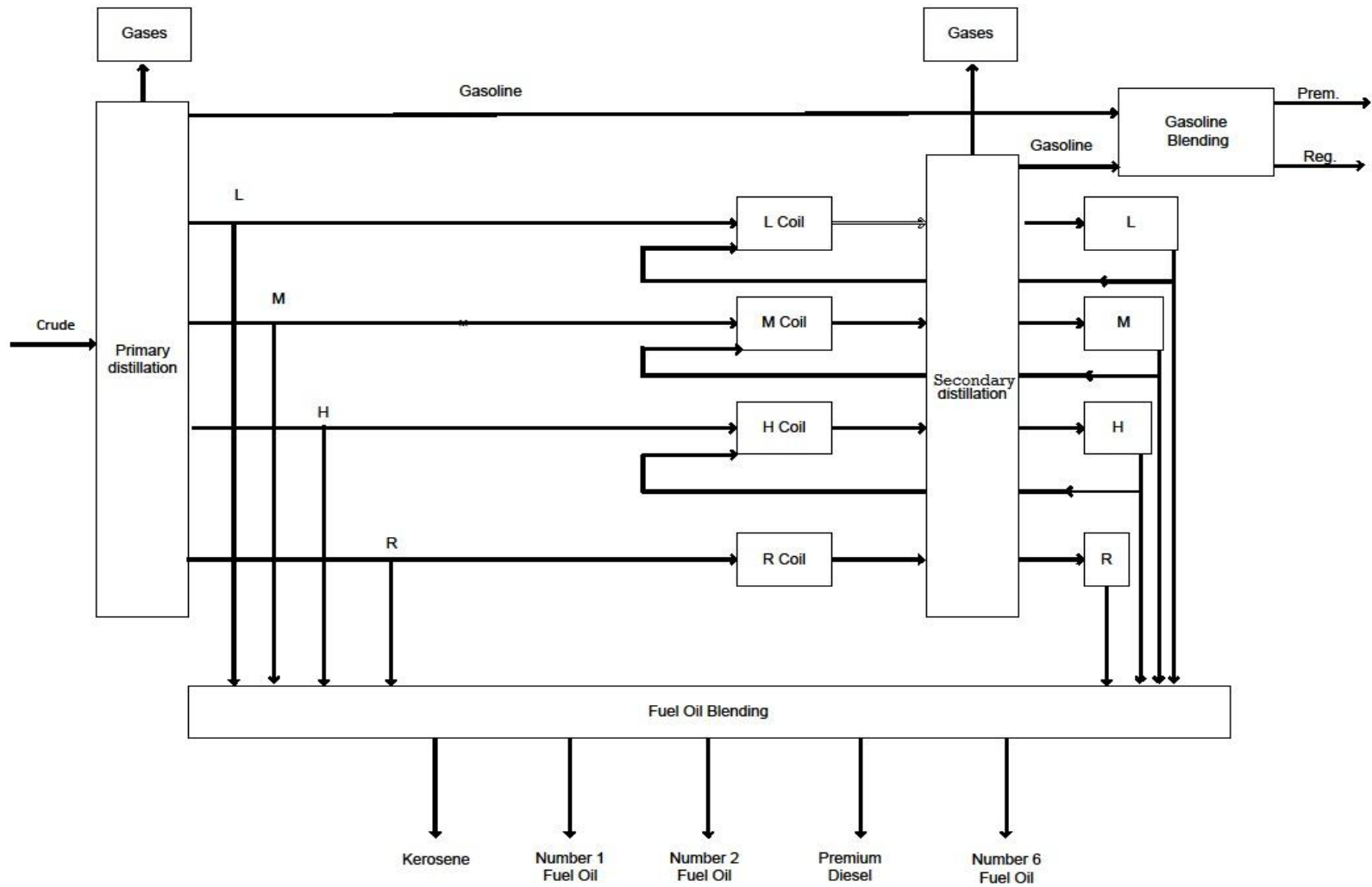


Figura 3: Flowsheet della raffineria

Entrando più nel dettaglio della sezione di cracking, si assume che la frazione crackizzata per passaggio sia pari a 0.3 per i distillati e 0.5 per il residuo (senza considerare il riciclo). In Tabella 7 vengono quindi riportate le rese per passaggio di tale sezione:

Tabella 7: Rese del cracking a singolo passaggio

| Senza riciclo | | | | |
|--|---------|-------|---------|---------|
| Rese | Leggero | Medio | Pesante | Residuo |
| Gas | 15 | 15 | 15 | 5 |
| Benzina | 76 | 67 | 67 | 34 |
| Olio leggero | - | 1 | 9 | 19 |
| Olio medio | 4 | - | 5 | 21 |
| Olio pesante | 3 | 4 | - | 21 |
| Residuo | 2 | 3 | 4 | - |
| Totale | 100 | 100 | 100 | 100 |
| Totale convertito (percentuale massiva della carica) | 30 | 30 | 30 | 30 |
| Con riciclo | | | | |
| Rese | Leggero | Medio | Pesante | |
| Gas | 18 | 16 | 16 | |
| Benzina | 51 | 48 | 48 | |
| Olio leggero | - | 8 | 8 | |
| Olio medio | 13 | - | 12 | |
| Olio pesante | 10 | 14 | - | |
| Residuo | 8 | 14 | 16 | |
| Totale | 100 | 100 | 100 | |
| Totale convertito (percentuale massiva della carica) | 30 | 30 | 30 | |

In seguito si passa quindi alla sezione di blending, ove i diversi prodotti intermedi vengono miscelati per formare i prodotti finali desiderati, rispettando le specifiche riportate in Tabella 8.

Tabella 8: Prodotti finali, specifiche

| | | Specifiche | | | | | | |
|---------------------|--|---------------------------|---------------------------------------|-------------------|---------------------|-------------|---------------------|---------------------------------------|
| | | Massima densità specifica | Minimo NO (benzine) e NC (distillati) | Viscosità massima | | | | Prezzo di mercato (cents per gallone) |
| | | | | A 100 °F | | A 122 °F | | |
| Prodotti finali | Componenti permessi | | | centistokes | μ_{spec} | centistokes | μ_{spec} | |
| Benzina premium | benzina da SR e cracking, C4, PTE | - | 88 NO | - | - | - | - | 11.56 |
| Benzina | benzina C4, PTE | - | 82 NO | - | - | - | - | 10.44 |
| Propano propilene | Taglio C3 | - | - | - | - | - | - | 4.00 |
| Butano butilene | Taglio C4 | - | - | - | - | - | - | 9.00 |
| Kerosene | Distillato leggero da SR | - | - | - | - | - | - | 8.88 |
| Olio combustibile 1 | Distillati leggeri e medi da SR e cracking | 0.850 | - | 1.9 | -0.400 | - | - | 8.44 |
| Olio combustibile 2 | Distillati leggeri, medi e pesanti da SR e cracking | 0.882 | 40 NC | 4.3 | -0.160 | - | - | 7.88 |
| Premium diesel | Distillati leggeri, medi e pesanti da SR e cracking | 0.840 | 55 NC | 2.6 | -0.290 | - | - | 8.63 |
| olio combustibile 6 | Distillati leggeri e medi da SR e cracking; residuo da SR e cracking | 1.014 | - | - | - | 375 | 0.411 | 2.26 |
| Gas leggeri | Taglio C2 | - | - | - | - | - | - | 0.2 cents per pound |

Avendo definito tutte le specifiche per le diverse sezioni dell'impianto è quindi possibile formulare il modello matematico che dovrà descrivere il funzionamento dello stesso. A tal fine è necessario compiere una scelta arbitraria: definire le porzioni delle correnti uscenti dalla colonna di distillazione primaria (indicate con S_L , S_M , S_H , S_R) da mandare al cracking (indicate con F_L , F_M , F_H , F_R). Ne segue che rimarranno disponibili per il blending le seguenti porzioni: $(1-F_L)S_L$, $(1-F_M)S_M$, $(1-F_H)S_H$ e $(1-F_R)S_R$.

Indicando con Z_i la quantità di ciascun prodotto uscente dalla colonna di distillazione secondaria e con R_i la quantità riciclata, i forni di cracking tratteranno le seguenti quantità di materiale: $R_L Z_L$, $R_M Z_M$ e $R_H Z_H$. La quantità netta uscente verrà invece indicata con C_i .

Le rese andranno quindi calcolate con la seguente procedura:

- Scelta arbitraria di F_i .
- Scelta arbitraria di R_i .
- Risoluzione rispetto a Z_i delle seguenti equazioni:

$$Z_L = 0.1022F_L + 0.00314F_M + 0.00322F_H + 0.0352F_R + 0.7R_L Z_L + 0.024R_M Z_M + 0.024R_H Z_H$$

$$Z_M = 0.00175F_L + 0.0665F_M + 0.00185F_H + 0.039F_R + 0.039R_L Z_L + 0.7R_M Z_M + 0.036R_H Z_H$$

$$Z_H = 0.00131F_L + 0.00114F_M + 0.0861F_H + 0.0390F_R + 0.03R_L Z_L + 0.042R_M Z_M + 0.7R_H Z_H$$

- Calcolo dei bilanci materiali netti:

$$C_L = Z_L (1 - R_L)$$

$$C_M = Z_M (1 - R_M)$$

$$C_H = Z_H (1 - R_H)$$

$$C_G = 0.00657F_L + 0.00428F_M + 0.00554F_H + 0.00928F_R + 0.054R_L Z_L + 0.048R_M Z_M + 0.048R_H Z_H$$

$$C_{G_{aso}} = 0.0333F_L + 0.0191F_M + 0.0247F_H + 0.0631F_R + 0.1503R_L Z_L + 0.144R_M Z_M + 0.144R_H Z_H$$

$$C_R = 0.0009F_L + 0.009F_M + 0.0015F_H + 0.185F_R + 0.024R_L Z_L + 0.042R_M Z_M + 0.048R_H Z_H$$

Le varie correnti devono quindi essere miscelate per soddisfare le specifiche dei singoli prodotti, illustrate nei successivi paragrafi.

Benzine: vengono prodotti due tipi di benzina commercializzabile derivate dalla miscelazione di benzina "straight-run", benzina da cracking, butano e piombo tetraetile. Le specifiche da rispettare sono: tensione di vapore Reid massima per entrambe (10 pounds per square inch) e numero di ottano minimo rispettivamente di 82 e 88.0

Dal momento che il butano è disponibile a basso costo, si stima che un 10% in butano consenta di raggiungere la specifica sulla tensione di vapore. Si cerca di rispettare esattamente anche il limite sul numero di ottano. Sebbene i prodotti finali, oggi, abbiano caratteristiche e processamenti significativamente diversi, il concetto di programmazione lineare non cambia.

Kerosene: Il kerosene è costituito da olio leggero “straight-run” senza ulteriori trattamenti.

Olio combustibile 1: Questo prodotto si ottiene per miscelazione di distillati leggeri e medi sia “straight-run” che da cracking. Le specifiche da rispettare sono viscosità massima di 1.9 centiStokes a 100°F e una specific gravity di 0.850.

Olio combustibile 2: Questo prodotto si ottiene per miscelazione di distillati leggeri medi e pesanti, sia “straight-run” che da cracking. Le specifiche da rispettare sono le seguenti: viscosità inferiore a 4.3 centiStokes a 100 °F; specific gravity massima di 0.882; numero di cetano minimo di 40; flash point superiore a 100 °F; punto di fine distillazione ASTM massimo a 675 °F.

Premium Diesel: Questo prodotto è costituito da una miscela di qualsiasi dei sei distillati e deve rispettare le seguenti specifiche: viscosità inferiore a 2.6 centiStokes a 100°F, specific gravity inferiore a 0.840, numero di cetano superiore a 55, 90% della curva di distillazione ASTM ad un massimo di 585 °F, punto finale ad un massimo di 646 °F.

Olio combustibile 6: Questo prodotto è costituito dalla miscela di residui “straight-run” e da cracking. Deve soddisfare le seguenti specifiche: viscosità inferiore a 375 centiStokes a 122 °F, specific gravity inferiore a 1.014.

4.1 Il modello matematico

Questo paragrafo è dedicato alla descrizione completa del modello matematico. Il sistema, per come è stato definito, è riconducibile ad un problema di programmazione lineare in forma standard del tipo:

$$\text{massimizzare } \sum_{j=1}^n p_j x_j \quad (j = 1, 2, \dots, n)$$

$$\text{soggetto a } \sum_{j=1}^n a_{ij} x_j = q_i \quad (i = 1, 2, \dots, n)$$

$$x_j \geq 0 \quad (j = 1, 2, \dots, n)$$

I coefficienti di tale modello verranno descritti in maniera più dettagliata nel seguito.

Il problema risulta quindi descritto da 26 equazioni con 61 incognite. A tali incognite x_j vengono assegnate delle etichette numeriche in modo da rendere riconoscibile il loro significato fisico all'interno della raffineria (come illustrato in Tabella 9) e risultano suddivise in tre classi generali:

- 1,2,...,7 sono le quantità di materiale "straight-run" e di riciclo caricate nei singoli forni di cracking.
- 16,21,...,99 sono le componenti dei vari prodotti di miscelazione.
- 101,102,...,126 sono variabili di bilancio.

Anche le 26 equazioni sono raggruppate in tre gruppi:

- 1,...,10 disponibilità e utilizzo di materie prime "straight-run" e da cracking.
- 11,...,14 limitazioni di capacità delle apparecchiature.
- 15,...,26 specifiche sui vari prodotti di miscelazione.

Una volta che le variabili x_j sono state così definite, i coefficienti p_j saranno connessi ai costi di processo e ai ricavi della vendite dei prodotti.

Definizione delle variabili e dei coefficienti del sistema

Alimentazione ai forni di cracking: le prime sette variabili derivano dalle definizioni e assunzioni già precedentemente adottate e sono pari a rispettivamente F_L , F_M , F_H , F_R , $R_L Z_L$, $R_M Z_M$, $R_H Z_H$.

Per esempio, se F_L è posto uguale a 1 si ha una resa di 0.0333 pound di benzina da cracking per pound di grezzo alla raffineria. Il coefficiente a_{16} della matrice **A** del modello sarà quindi posto pari a -0.0333 e così via per le righe 7,8,9,10, come illustrato in Tabella 9.

Per quanto riguarda le variabili x_5 , x_6 , x_7 (ovvero i ricicli), è noto dai dati che i ricicli consumano il 30% della carica, pertanto i coefficienti a_{75} , a_{86} , a_{97} sono posti pari a +0.3.

Componenti delle miscele: A ogni componente è associato un pedice a due cifre. La prima indica il prodotto finale, la seconda il materiale intermedio utilizzato per la miscelazione. Per esempio il pedice 63 indica che il componente 3 (medio "straight-run") viene miscelato per ottenere il prodotto 6 (olio combustibile 1). Dal momento che i prodotti sono venduti su base volumetrica, ciascuno dei componenti deve essere espresso come "galloni di prodotto per libbra di grezzo".

Variabili di bilancio: Le variabili da 101 a 126 rappresentano l'eccesso (se presente) di ogni componente rispetto a quanto utilizzato nei prodotti finali. Tali variabili formano una sotto-matrice identità particolarmente utile come punto di partenza per la risoluzione del problema. Per semplicità a queste variabili non è associata una penalità nella funzione obiettivo.

Definizione delle equazioni

Disponibilità dei materiali: Ognuna delle prime dieci righe della tabella è connessa con la produzione e il consumo di un particolare olio intermedio (sono quindi bilanci materiali). L'unità di misura è "libbre di olio intermedio per libbra di grezzo alimentato alla raffineria". Per le righe 6-10 il termine noto è posto pari a 0 in quanto per i materiali da cracking non vi è una alimentazione dall'esterno.

Limitazioni di capacità delle apparecchiature: I forni di cracking sono dimensionati ciascuno in modo tale da poter gestire una quantità di greggio fino a 2.5 volte maggiore di quella del corrispettivo prodotto straight-run. Tali capacità sono date e costituiscono i termini noti delle equazioni 11-13. I coefficienti sono dati dalle quantità di greggio che entra nei forni. Per ottenere invece i coefficienti della riga 14 sarebbe necessario introdurre le definizioni esplicite delle variabili. I coefficienti riportati in tabella sono quindi ricavati dalla risoluzione di un sistema lineare, quindi rappresentano l'uso della capacità del frazionatore secondario associato a ciascuna delle sette operazioni di cracking.

Specifiche dei prodotti: le equazioni 15-26 servono a garantire che le specifiche di ciascun prodotto finale siano rispettate.

Per esempio, nel caso della benzina si possono associare i valori dei coefficienti p_{21} e p_{26} alla quantità di piombo tetraetile utilizzata k_1 . Nel caso di $k_1 = 3$ i coefficienti assumono il valore 9.7666.

La riga 16 rappresenta la specifica sulla viscosità dell'olio combustibile 1. Le colonne 62,63,67,68 rappresentano gli eccessi di viscosità rispetto alla specifica per ogni componente.

La riga 17 rappresenta invece la specifica sulla viscosità dell'olio combustibile 2. I coefficienti hanno un significato simile al caso precedente.

L'equazione 18 rappresenta la specifica sul numero di cetano; i coefficienti rappresentano quindi gli eccessi rispetto a tale specifica.

L'equazione 19 garantisce il rispetto del flash point limitando il distillato leggero al 50% mentre l'equazione 20 rappresenta la specifica sul punto finale di distillazione limitando il distillato pesante al 50% in volume.

L'equazione 21 rappresenta la specifica sulla viscosità del Premium Diesel, mentre quella 22 la specifica sul numero di cetano. In questo caso è necessario controllare anche la specific gravity, attraverso l'equazione 23.

L'equazione 24 rappresenta il rispetto del punto a 90% della curva di distillazione ASTM mantenendo gli oli leggeri ad almeno 20%; allo stesso modo l'equazione 25 rappresenta il rispetto del punto finale limitando il contenuto di distillato pesante al 33%.

L'ultima equazione rappresenta il controllo della viscosità nell'olio combustibile 6.

Definizione dei coefficienti della funzione obiettivo

p_1, \dots, p_7 rappresentano i ricavi sull'eccesso di gas da cracking rispetto alle spese per le utilities.

p_{16}, p_{21}, p_{26} rappresentano il valore netto di un gallone di componente da cracking nella benzina premium. Per calcolare tali valori è necessario tenere in considerazione che all'interno della miscela si hanno anche butano e piombo tetraetile.

$p_{52}, p_{62}, p_{63}, \dots, p_{99}$ misurano i ricavi della raffineria per kerosene, olio combustibile 1, olio combustibile 2, premium diesel, olio combustibile 6.

I coefficienti $p_{101} - p_{126}$ sono posti uguali a 0 come precedentemente detto.

In Tabella 9 sono riportati quindi i coefficienti di tutte le equazioni così come devono essere risolte dall'algoritmo LP:

Tabella 9: Coefficienti del LP

| xi | Quantità caricate in ogni forno di cracking | | | | | | |
|---|---|----------|----------|----------|---------------|---------------|---------------|
| | FL | FM | FH | FR | RLZL (libbre) | RMZM (libbre) | RHZH (libbre) |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Coefficienti p,i | 0,006648 | 0,004278 | 0,005456 | 0,009165 | 0,055313 | 0,04827 | 0,047597 |
| Disponibilità delle materie prime (libbre) | | | | | | | |
| 1. benzina "straight-run" | | | | | | | |
| 2. Olio leggero "straight-run" | 0,146 | | | | | | |
| 3. Olio medio "straight-run" | | 0,095 | | | | | |
| 4. Olio pesante "straight-run" | | | 0,123 | | | | |
| 5. Residuo "straight-run" | | | | 0,371 | | | |
| 6. Benzina da cracking | -0,0333 | -0,0191 | -0,0247 | -0,0631 | -0,1503 | -0,144 | -0,144 |
| 7. Olio leggero da cracking | -0,1022 | -0,00314 | -0,00332 | -0,0352 | 0,3 | -0,024 | -0,024 |
| 8. Olio medio da cracking | -0,00175 | -0,0665 | -0,00185 | -0,039 | -0,039 | 0,3 | -0,036 |
| 9. Olio pesante da cracking | -0,00131 | -0,00114 | -0,0861 | -0,039 | -0,03 | -0,042 | 0,3 |
| 10. Residuo da cracking | -0,0009 | -0,0009 | -0,0015 | -0,185 | -0,024 | -0,042 | -0,048 |
| Limitazioni sulla capacità delle apparecchiature | | | | | | | |
| 11. Forno di cracking per olio leggero (libbre) | 0,146 | | | | 1 | | |
| 12. Forno di cracking per olio medio (libbre) | | 0,095 | | | | 1 | |
| 13. Forno di cracking per olio pesante (libbre) | | | 0,123 | | | | 1 |
| 14. Colonna di frazionamento secondario (10 ⁻⁴ moli) | 18,206 | 10,371 | 13,431 | 10,021 | 125,93 | 110,35 | 110,41 |
| Specifiche sui prodotti | | | | | | | |
| 15. Benzina (NO) | | | | | | | |
| 16. Olio combustibile 1 (viscosità - galloni) | | | | | | | |
| 17. Olio combustibile 2 (viscosità-galloni) | | | | | | | |
| 18. Olio combustibile 2 (numero di cetano - galloni) | | | | | | | |
| 19. Olio combustibile 2 (flashpoint - galloni) | | | | | | | |
| 20. Olio combustibile 2 (endpoint - galloni) | | | | | | | |
| 21. premium diesel (viscosità - galloni) | | | | | | | |
| 22. premium diesel (numero di cetano - galloni) | | | | | | | |
| 23. premium diesel (specific gravity - galloni) | | | | | | | |
| 24. premium diesel (punto a 90% - galloni) | | | | | | | |
| 25. premium diesel (endpoint - galloni) | | | | | | | |
| 26. Olio combustibile 6 (viscosità - galloni) | | | | | | | |

| xi | Componenti nei vari prodotti i miscelazione (galloni) | | | | | | | |
|---|---|------------------|--------|----------|---------------------|-------|--------|-------|
| | Premium gasoline | Regular gasoline | | Kerosene | Olio combustibile 1 | | | |
| | 16 | 21 | 26 | 52 | 62 | 63 | 67 | 68 |
| Coefficienti p,i | 11,0666 | 9,7666 | 9,7666 | 8,88 | 8,44 | 8,44 | 8,44 | 8,44 |
| Disponibilità delle materie prime (libbre) | | | | | | | | |
| 1. benzina "straight-run" | | 6,123 | | | | | | |
| 2. Olio leggero "straight-run" | | | | 6,797 | 6,797 | | | |
| 3. Olio medio "straight-run" | | | | | | 6,914 | | |
| 4. Olio pesante "straight-run" | | | | | | | | |
| 5. Residuo "straight-run" | | | | | | | | |
| 6. Benzina da cracking | 6,289 | | 6,289 | | | | | |
| 7. Olio leggero da cracking | | | | | | | 6,997 | |
| 8. Olio medio da cracking | | | | | | | | 7,205 |
| 9. Olio pesante da cracking | | | | | | | | |
| 10. Residuo da cracking | | | | | | | | |
| Limitazioni sulla capacità delle apparecchiature | | | | | | | | |
| 11. Forno di cracking per olio leggero (libbre) | | | | | | | | |
| 12. Forno di cracking per olio medio (libbre) | | | | | | | | |
| 13. Forno di cracking per olio pesante (libbre) | | | | | | | | |
| 14. Colonna di frazionamento secondario (10 ⁻⁴ moli) | | | | | | | | |
| Specifiche sui prodotti | | | | | | | | |
| 15. Benzina (NO) | | 1 | -1,727 | | | | | |
| 16. Olio combustibile 1 (viscosità - galloni) | | | | | -0,022 | 0,185 | -0,019 | 0,18 |
| 17. Olio combustibile 2 (viscosità-galloni) | | | | | | | | |
| 18. Olio combustibile 2 (numero di cetano - galloni) | | | | | | | | |
| 19. Olio combustibile 2 (flashpoint - galloni) | | | | | | | | |
| 20. Olio combustibile 2 (endpoint - galloni) | | | | | | | | |
| 21. premium diesel (viscosità - galloni) | | | | | | | | |
| 22. premium diesel (numero di cetano - galloni) | | | | | | | | |
| 23. premium diesel (specific gravity - galloni) | | | | | | | | |
| 24. premium diesel (punto a 90% - galloni) | | | | | | | | |
| 25. premium diesel (endpoint - galloni) | | | | | | | | |
| 26. Olio combustibile 6 (viscosità - galloni) | | | | | | | | |

| xi | Componenti nei vari prodotti i miscelazione (galloni) | | | | | | | | | | | |
|---|---|--------|-------|--------|-------|------|----------------|-------|-------|--------|-------|-------|
| | Olio combustibile 2 | | | | | | Premium diesel | | | | | |
| | 72 | 73 | 74 | 77 | 78 | 79 | 82 | 83 | 84 | 87 | 88 | 89 |
| Coefficienti p,i | 7,88 | 7,88 | 7,88 | 7,88 | 7,88 | 7,88 | 8,63 | 8,63 | 8,63 | 8,63 | 8,63 | 8,63 |
| Disponibilità delle materie prime (libbre) | | | | | | | | | | | | |
| 1. benzina "straight-run" | | | | | | | | | | | | |
| 2. Olio leggero "straight-run" | 6,797 | | | | | | 6,797 | | | | | |
| 3. Olio medio "straight-run" | | 6,914 | | | | | | 6,914 | | | | |
| 4. Olio pesante "straight-run" | | | 7,122 | | | | | | 7,122 | | | |
| 5. Residuo "straight-run" | | | | | | | | | | | | |
| 6. Benzina da cracking | | | | | | | | | | | | |
| 7. Olio leggero da cracking | | | | 6,997 | | | | | | 6,997 | | |
| 8. Olio medio da cracking | | | | | 7,205 | | | | | | 7,205 | |
| 9. Olio pesante da cracking | | | | | | 7,63 | | | | | | 7,63 |
| 10. Residuo da cracking | | | | | | | | | | | | |
| Limitazioni sulla capacità delle apparecchiature | | | | | | | | | | | | |
| 11. Forno di cracking per olio leggero (libbre) | | | | | | | | | | | | |
| 12. Forno di cracking per olio medio (libbre) | | | | | | | | | | | | |
| 13. Forno di cracking per olio pesante (libbre) | | | | | | | | | | | | |
| 14. Colonna di frazionamento secondario (10 ⁻⁴ moli) | | | | | | | | | | | | |
| Specifiche sui prodotti | | | | | | | | | | | | |
| 15. Benzina (NO) | | | | | | | | | | | | |
| 16. Olio combustibile 1 (viscosità - galloni) | | | | | | | | | | | | |
| 17. Olio combustibile 2 (viscosità-galloni) | -0,262 | -0,055 | 0,116 | -0,259 | -0,06 | 0,19 | | | | | | |
| 18. Olio combustibile 2 (numero di cetano - galloni) | -13 | -18 | -21 | 5 | 5 | | | | | | | |
| 19. Olio combustibile 2 (flashpoint - galloni) | 1 | -1 | -1 | 1 | -1 | -1 | | | | | | |
| 20. Olio combustibile 2 (endpoint - galloni) | -1 | -1 | 1 | -1 | -1 | 1 | | | | | | |
| 21. premium diesel (viscosità - galloni) | | | | | | | -0,132 | 0,075 | 0,246 | -0,129 | 0,07 | 0,32 |
| 22. premium diesel (numero di cetano - galloni) | | | | | | | 2 | -3 | -6 | 20 | 20 | 15 |
| 23. premium diesel (specific gravity - galloni) | | | | | | | -0,024 | -0,01 | 0,015 | | 0,025 | 0,076 |
| 24. premium diesel (punto a 90% - galloni) | | | | | | | -0,8 | 0,2 | 0,2 | -0,8 | 0,2 | 0,2 |
| 25. premium diesel (endpoint - galloni) | | | | | | | -0,33 | -0,33 | 0,67 | -0,33 | -0,33 | 0,67 |
| 26. Olio combustibile 6 (viscosità - galloni) | | | | | | | | | | | | |

| xi | Componenti nei vari prodotti i miscelazione (galloni) | | | | | | | |
|---|---|---------|---------|---------|---------|---------|---------|---------|
| | olio combustibile 6 | | | | | | | |
| | 90 | 92 | 93 | 94 | 95 | 97 | 98 | 99 |
| Coefficienti p,i | 2,26 | 2,26 | 2,26 | 2,26 | 2,26 | 2,26 | 2,26 | 2,26 |
| Disponibilità delle materie prime (libbre) | | | | | | | | |
| 1. benzina "straight-run" | | | | | | | | |
| 2. Olio leggero "straight-run" | | 6,797 | | | | | | |
| 3. Olio medio "straight-run" | | | 6,914 | | | | | |
| 4. Olio pesante "straight-run" | | | | 7,122 | | | | |
| 5. Residuo "straight-run" | | | | | 7,864 | | | |
| 6. Benzina da cracking | | | | | | | | |
| 7. Olio leggero da cracking | | | | | | 6,997 | | |
| 8. Olio medio da cracking | | | | | | | 7,205 | |
| 9. Olio pesante da cracking | | | | | | | | 7,63 |
| 10. Residuo da cracking | 8,513 | | | | | | | |
| Limitazioni sulla capacità delle apparecchiature | | | | | | | | |
| 11. Forno di cracking per olio leggero (libbre) | | | | | | | | |
| 12. Forno di cracking per olio medio (libbre) | | | | | | | | |
| 13. Forno di cracking per olio pesante (libbre) | | | | | | | | |
| 14. Colonna di frazionamento secondario (10 ⁻⁴ moli) | | | | | | | | |
| Specifiche sui prodotti | | | | | | | | |
| 15. Benzina (NO) | | | | | | | | |
| 16. Olio combustibile 1 (viscosità - galloni) | | | | | | | | |
| 17. Olio combustibile 2 (viscosità-galloni) | | | | | | | | |
| 18. Olio combustibile 2 (numero di cetano - galloni) | | | | | | | | |
| 19. Olio combustibile 2 (flashpoint - galloni) | | | | | | | | |
| 20. Olio combustibile 2 (endpoint - galloni) | | | | | | | | |
| 21. premium diesel (viscosità - galloni) | | | | | | | | |
| 22. premium diesel (numero di cetano - galloni) | | | | | | | | |
| 23. premium diesel (specific gravity - galloni) | | | | | | | | |
| 24. premium diesel (punto a 90% - galloni) | | | | | | | | |
| 25. premium diesel (endpoint - galloni) | | | | | | | | |
| 26. Olio combustibile 6 (viscosità - galloni) | 0,1354 | -0,9324 | -0,7075 | -0,5325 | -0,1215 | -0,9029 | -0,6963 | -0,4527 |

| xi | Variabili di bilancio | | | | | | | | | | | | | | | | | | | | | | | | termini noti q,i | | |
|---|-----------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------------------------|-----|--------|
| | 101 | 102 | 103 | 104 | 105 | 106 | 107 | 108 | 109 | 110 | 111 | 112 | 113 | 114 | 115 | 116 | 117 | 118 | 119 | 120 | 121 | 122 | 123 | 124 | | 125 | 126 |
| Coefficienti p _i | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Disponibilità delle materie prime (libbre) | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1. benzina "straight-run" | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | 0,248 |
| 2. Olio leggero "straight-run" | | 1 | | | | | | | | | | | | | | | | | | | | | | | | | 0,146 |
| 3. Olio medio "straight-run" | | | 1 | | | | | | | | | | | | | | | | | | | | | | | | 0,095 |
| 4. Olio pesante "straight-run" | | | | 1 | | | | | | | | | | | | | | | | | | | | | | | 0,123 |
| 5. Residuo "straight-run" | | | | | 1 | | | | | | | | | | | | | | | | | | | | | | 0,371 |
| 6. Benzina da cracking | | | | | | 1 | | | | | | | | | | | | | | | | | | | | | 0 |
| 7. Olio leggero da cracking | | | | | | | 1 | | | | | | | | | | | | | | | | | | | | 0 |
| 8. Olio medio da cracking | | | | | | | | 1 | | | | | | | | | | | | | | | | | | | 0 |
| 9. Olio pesante da cracking | | | | | | | | | 1 | | | | | | | | | | | | | | | | | | 0 |
| 10. Residuo da cracking | | | | | | | | | | 1 | | | | | | | | | | | | | | | | | 0 |
| Limitazioni sulla capacità delle apparecchiature | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 11. Forno di cracking per olio leggero (libbre) | | | | | | | | | | | 1 | | | | | | | | | | | | | | | | 0,365 |
| 12. Forno di cracking per olio medio (libbre) | | | | | | | | | | | | 1 | | | | | | | | | | | | | | | 0,2375 |
| 13. Forno di cracking per olio pesante (libbre) | | | | | | | | | | | | | 1 | | | | | | | | | | | | | | 0,3075 |
| 14. Colonna di frazionamento secondario (10 ⁴ -4 moli) | | | | | | | | | | | | | | 1 | | | | | | | | | | | | | 98 |
| Specifiche sui prodotti | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 15. Benzina (NO) | | | | | | | | | | | | | | | 1 | | | | | | | | | | | | 0 |
| 16. Olio combustibile 1 (viscosità - galloni) | | | | | | | | | | | | | | | | 1 | | | | | | | | | | | 0 |
| 17. Olio combustibile 2 (viscosità-galloni) | | | | | | | | | | | | | | | | | 1 | | | | | | | | | | 0 |
| 18. Olio combustibile 2 (numero di cetano - galloni) | | | | | | | | | | | | | | | | | | 1 | | | | | | | | | 0 |
| 19. Olio combustibile 2 (flashpoint - galloni) | | | | | | | | | | | | | | | | | | | 1 | | | | | | | | 0 |
| 20. Olio combustibile 2 (endpoint - galloni) | | | | | | | | | | | | | | | | | | | | 1 | | | | | | | 0 |
| 21. premium diesel (viscosità - galloni) | | | | | | | | | | | | | | | | | | | | | 1 | | | | | | 0 |
| 22. premium diesel (numero di cetano - galloni) | | | | | | | | | | | | | | | | | | | | | | 1 | | | | | 0 |
| 23. premium diesel (specific gravity - galloni) | | | | | | | | | | | | | | | | | | | | | | | 1 | | | | 0 |
| 24. premium diesel (punto a 90% - galloni) | | | | | | | | | | | | | | | | | | | | | | | | 1 | | | 0 |
| 25. premium diesel (endpoint - galloni) | | | | | | | | | | | | | | | | | | | | | | | | | | 1 | 0 |
| 26. Olio combustibile 6 (viscosità - galloni) | | | | | | | | | | | | | | | | | | | | | | | | | | 1 | 0 |

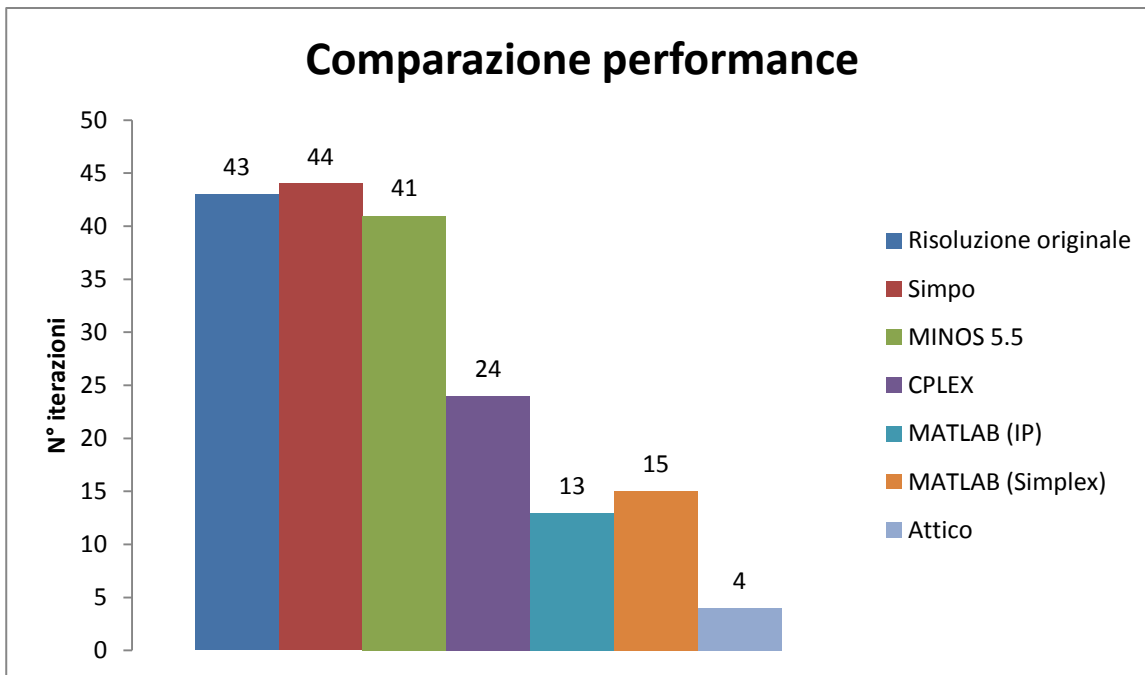
Il problema è stato risolto utilizzando i seguenti risolutori “commerciali”:

- Simpo, basato sul metodo del simplesso (Vanderbei, 2001);
- CPLEX, che implementa al suo interno sia il metodo del simplesso che quello dell’interior point (IBM);
- MINOS 5.5 (AMPL)
- Matlab, attraverso la funzione linprog, impostando le opzioni in modo tale da risolvere il problema una volta attraverso simplesso e una volta attraverso interior point (MathWorks)
- Successivamente è stato utilizzato il nuovo metodo dell’Attico.

I risultati ottenuti attraverso i diversi risolutori sono identici, a conferma della unicità del punto di ottimo e della affidabilità dei risolutori utilizzati.

Come è possibile notare, il numero di iterazioni dichiarato nella risoluzione originale del problema è molto simile a quello necessario ai risolutori Simpo e MINOS 5.5 per risolvere il sistema. Il risolutore CPLEX, sfruttando le tecniche della programmazione non lineare, riesce a risolvere il problema quasi dimezzando il numero di iterazioni necessarie, a testimonianza di come, da questo punto di vista, il metodo dell’Interior Point costituisca una interessante alternativa. I risolutori forniti con Matlab presentano invece prestazioni decisamente più interessanti dal punto di vista del semplice numero di iterazioni: essi sono stati infatti in grado di risolvere il problema attraverso 15 (Simplex) e 13 (Interior Point) iterazioni. Il metodo dell’Attico ha però effettuato una prestazione ancora migliore: il problema ha richiesto la valutazione di soli 4 vertici.

Per quanto riguarda invece i tempi di calcolo non è stato invece possibile effettuare alcun confronto: date le piccole dimensioni del problema i tempi di calcolo sono inferiori al secondo e non forniscono dunque alcuna informazione significativa.



Dal momento che, come già sottolineato, quest'ultimo parametro è quello maggiormente preso in considerazione durante lo sviluppo di algoritmi per uso industriale per poter definire le effettive prestazioni del metodo dell'Attico sarà necessario effettuare nuovi test, studiando un problema di maggiori dimensioni o sviluppando un programma MILP (Mixed-Integer Linear Programming), ove si presenta la necessità di risolvere un elevato numero di problemi LP e quindi si potrebbero mettere in evidenza le prestazioni in termini di tempo di calcolo dei diversi risolutori.

In aggiunta all'utilizzo dei risolutori commerciali, si sono anche sviluppati dei semplici algoritmi in linguaggio C++ basati sul metodo del semplice come descritto nei capitoli 2.2.1 e 2.2.2. I tableau lì riportati sono, infatti, i risultati prodotti da tali algoritmi. Per la risoluzione del caso studio si è però preferito utilizzare esclusivamente i risolutori commerciali e l'Attico in quanto l'obiettivo principale era quello di ottenere un confronto con lo stato dell'arte dei risolutori disponibili: il numero di iterazioni e i tempi di calcolo sono dipendenti, infatti, non solo dal metodo di risoluzione stesso ma anche dalla struttura dell'algoritmo risolutivo. Dal momento che i risolutori commerciali presentano caratteristiche non facilmente riproducibili (per esempio il pre-processamento del problema) si è preferito utilizzare soltanto questi risolutori ai fini del confronto.

| | Originale | Simplo | MINOS 5.5 | CPLEX | Matlab (IP) | Matlab (simplex) | Attico | | | | | | | | |
|---------|-----------|--------|-----------|--------|-------------|------------------|--------|--------|--|--|--|--|--|--|--|
| n° iter | 43 | 44 | 41 | 24 | 13 | 15 | 4 | | | | | | | | |
| F.O. | 1,1204 | 1,1204 | 1,1204 | 1,1204 | 1,1204 | 1,1204 | 1,1204 | 1,1204 | | | | | | | |
| 1 | 0,3555 | 0,3555 | 0,35546 | 0,3555 | 0,3555 | 0,3555 | 0,3555 | 0,3555 | Porzione di distillato L mandato al cracking | | | | | | |
| 2 | 0,8040 | 0,8040 | 0,8040 | 0,8040 | 0,8040 | 0,8040 | 0,8040 | 0,8040 | Porzione di distillato M mandato al cracking | | | | | | |
| 3 | 0,3251 | 0,3251 | 0,3251 | 0,3251 | 0,3251 | 0,3251 | 0,3251 | 0,3251 | Porzione di distillato H mandato al cracking | | | | | | |
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | Porzione di residuo mandato al cracking | | | | | | |
| 5 | 0,1706 | 0,1706 | 0,1706 | 0,1706 | 0,1706 | 0,1706 | 0,1706 | 0,1706 | RLZL (contenuto del coil) | | | | | | |
| 6 | 0,1611 | 0,1611 | 0,1611 | 0,1611 | 0,1611 | 0,1611 | 0,1611 | 0,1611 | RMZM | | | | | | |
| 7 | 0,2675 | 0,2675 | 0,2675 | 0,2675 | 0,2675 | 0,2675 | 0,2675 | 0,2675 | RHZH | | | | | | |
| 8 | 0,0061 | 0,0061 | 0,0061 | 0,0061 | 0,0061 | 0,0061 | 0,0061 | 0,0061 | Premium gasoline 16 | | | | | | |
| 9 | 0,0405 | 0,0405 | 0,0405 | 0,0405 | 0,0405 | 0,0405 | 0,0405 | 0,0405 | Regular gasoline 26 | | | | | | |
| 10 | 0,0235 | 0,0235 | 0,0235 | 0,0235 | 0,0235 | 0,0235 | 0,0235 | 0,0235 | Regular gasoline 26 | | | | | | |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Kerosene 52 | | | | | | |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 1 fuel oil 62 | | | | | | |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 1 fuel oil 63 | | | | | | |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 1 fuel oil 67 | | | | | | |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 1 fuel oil 68 | | | | | | |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 2 fuel oil 72 | | | | | | |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 2 fuel oil 73 | | | | | | |
| 18 | 0,0044 | 0,0044 | 0,0044 | 0,0044 | 0,0044 | 0,0044 | 0,0044 | 0,0044 | Number 2 fuel oil 74 | | | | | | |
| 19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 2 fuel oil 77 | | | | | | |
| 20 | 0,0086 | 0,0086 | 0,0086 | 0,0086 | 0,0086 | 0,0086 | 0,0086 | 0,0086 | Number 2 fuel oil 78 | | | | | | |
| 21 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 2 fuel oil 79 | | | | | | |
| 22 | 0,0138 | 0,0138 | 0,0138 | 0,0138 | 0,0138 | 0,0138 | 0,0138 | 0,0138 | Premium diesel fuel oil 82 | | | | | | |
| 23 | 0,0027 | 0,0027 | 0,0027 | 0,0027 | 0,0027 | 0,0027 | 0,0027 | 0,0027 | Premium diesel fuel oil 83 | | | | | | |
| 24 | 0,0072 | 0,0072 | 0,0072 | 0,0072 | 0,0072 | 0,0072 | 0,0072 | 0,0072 | Premium diesel fuel oil 84 | | | | | | |
| 25 | 0,0012 | 0,0012 | 0,0012 | 0,0012 | 0,0012 | 0,0012 | 0,0012 | 0,0012 | Premium diesel fuel oil 87 | | | | | | |
| 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Premium diesel fuel oil 88 | | | | | | |
| 27 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Premium diesel fuel oil 89 | | | | | | |
| 28 | 0,0247 | 0,0247 | 0,0247 | 0,0247 | 0,0247 | 0,0247 | 0,0247 | 0,0247 | Number 6 fuel oil 90 | | | | | | |
| 29 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 6 fuel oil 92 | | | | | | |
| 30 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 6 fuel oil 93 | | | | | | |
| 31 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 6 fuel oil 94 | | | | | | |
| 32 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 6 fuel oil 95 | | | | | | |
| 33 | 0,0037 | 0,0037 | 0,0037 | 0,0037 | 0,0037 | 0,0037 | 0,0037 | 0,0037 | Number 6 fuel oil 97 | | | | | | |
| 34 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 6 fuel oil 98 | | | | | | |
| 35 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Number 6 fuel oil 99 | | | | | | |
| 36 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 101 | | | | | | |
| 37 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 102 | | | | | | |
| 38 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 103 | | | | | | |
| 39 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 104 | | | | | | |
| 40 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 105 | | | | | | |
| 41 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 106 | | | | | | |
| 42 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 107 | | | | | | |
| 43 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 108 | | | | | | |
| 44 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 109 | | | | | | |
| 45 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 110 | | | | | | |
| 46 | 0,1425 | 0,1425 | 0,1425 | 0,1425 | 0,1425 | 0,1425 | 0,1425 | 0,1425 | Balance account 111 | | | | | | |
| 47 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 112 | | | | | | |
| 48 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 113 | | | | | | |
| 49 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 114 | | | | | | |
| 50 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 115 | | | | | | |
| 51 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 116 | | | | | | |
| 52 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 117 | | | | | | |
| 53 | 0,0502 | 0,0502 | 0,0502 | 0,0502 | 0,0502 | 0,0502 | 0,0502 | 0,0502 | Balance account 118 | | | | | | |
| 54 | 0,0130 | 0,0130 | 0,0130 | 0,0130 | 0,0130 | 0,0130 | 0,0130 | 0,0130 | Balance account 119 | | | | | | |
| 55 | 0,0041 | 0,0041 | 0,0041 | 0,0041 | 0,0041 | 0,0041 | 0,0041 | 0,0041 | Balance account 120 | | | | | | |
| 56 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 121 | | | | | | |
| 57 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 122 | | | | | | |
| 58 | 0,0003 | 0,0003 | 0,0003 | 0,0003 | 0,0003 | 0,0003 | 0,0003 | 0,0003 | Balance account 123 | | | | | | |
| 59 | 0,0100 | 0,0100 | 0,0100 | 0,0100 | 0,0100 | 0,0100 | 0,0100 | 0,0100 | Balance account 124 | | | | | | |
| 60 | 0,0010 | 0,0010 | 0,0010 | 0,0010 | 0,0010 | 0,0010 | 0,0010 | 0,0010 | Balance account 125 | | | | | | |
| 61 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Balance account 126 | | | | | | |

Capitolo 5 - Futuri sviluppi Mixed Integer Linear Programming

5.1 Mixed-Integer Linear Programming (MILP)

Come suggerisce il nome, un problema MILP è un problema LP al cui interno non siano presenti solo variabili continue ma anche variabili discrete. Nell'ambito dell'Ingegneria Chimica, questo tipo di problemi sono risultati particolarmente importanti nelle aree di ricerca e sviluppo, di design e controllo di processo per citarne alcuni.

Tra le principali applicazioni possiamo trovare

- Il numero minimo di accoppiamenti nel progetto di una rete di scambiatori.
- Integrazione termica in sequenze intelligenti di distillazioni.
- Progetto di colonne di distillazione multicomponente e multi prodotto.
- Progetto di reti di scambiatori di calore multi periodo e sistemi di distillazione.
- Analisi di sensitività di processi chimici.
- Proprietà strutturali di sistemi di controllo.
- Pianificazione e programmazione di processi Batch.

Per descrivere questa tipologia di problemi verrà utilizzata, senza perdere in generalità, la sotto classe di problemi in cui le variabili intere sono binarie.

5.1.1 MILP - formulazione

Il problema MILP, utilizzando variabili binarie, presenta la seguente forma:

minimizzare $\mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y}$

soggetta a: $\mathbf{Ax} + \mathbf{By} \leq \mathbf{b}$

$$\mathbf{x} \geq 0, \mathbf{x} \in \mathbf{X} \subseteq R^N$$

$$\mathbf{y} \in \{0,1\}^q$$

Ove:

- \mathbf{x} è un vettore di n variabili continue
- \mathbf{y} è un vettore di q variabili 0-1
- \mathbf{c}, \mathbf{d} sono vettori di parametri dimensionati rispettivamente $(n \times 1)$ e $(q \times 1)$
- \mathbf{A}, \mathbf{B} sono matrici di dimensioni appropriate
- \mathbf{b} è un vettore di p disuguaglianze.

Le principali difficoltà riscontrabili nella risoluzione di un problema MILP nella forma proposta consistono nella natura combinatoria del dominio delle \mathbf{y} variabili. Qualsiasi scelta 0 o 1 per gli elementi del vettore \mathbf{y} risulta in un diverso LP nelle \mathbf{x} variabili che deve essere risolto per trovare la soluzione ottimale.

Dal momento che risulterebbe proibitivo risolvere a forza bruta tutte le possibili combinazioni per un singolo problema MILP (basti pensare che con 100 variabili binarie le combinazioni sarebbero 2^{100}) esistono diverse tipologie di algoritmi sviluppati appositamente per risolvere questa classe di problemi, ovvero:

- Branch and Bound (che verrà esaminato più in dettaglio nel seguito)
- Cutting plane
- Decomposizione
- Basati sulla logica

Nell'algoritmo branch and bound, un albero binario viene utilizzato per rappresentare le diverse combinazioni delle variabili intere, la regione accettabile viene suddivisa sistematicamente in sottodomini e vincoli inferiori e superiori vengono imposti a diversi livelli dell'albero binario.

Nel metodo cutting plane, la regione accettabile non viene suddivisa in sottodomini ma vengono generati e aggiunti dei nuovi vincoli (chiamati tagli) che restringono la regione accettabile fino a che non viene trovata la soluzione ottimale.

Nei metodi di decomposizione viene sfruttata la struttura matematica del modello attraverso partizionamento delle variabili, dualità metodi di rilassamento.

Nei metodi basati sulla logica vengono utilizzati vincoli disgiuntivi o tecniche di inferenza simbolica che possono essere espressi in termini di variabili binarie.

5.2 Il metodo Branch and Bound

Per poter definire e comprendere il funzionamento del metodo Branch and Bound è necessario per prima cosa definire e comprendere i concetti base di separazione, rilassamento, fathoming.

5.2.1 Separazione

Denominando un problema MILP nella forma precedentemente indicata come (P) e il suo insieme di soluzioni accettabili $FS(P)$, un insieme di sottoproblemi $(P_1), (P_2), (P_3) \dots (P_n)$ di (P) è denominato separazione di (P) se le seguenti condizioni vengono rispettate:

1. Una soluzione accettabile di ciascun sottoproblema $(P_1), (P_2), (P_3) \dots (P_n)$ è soluzione accettabile di (P)
2. Ogni soluzione accettabile di (P) è soluzione accettabile di uno e solo un sottoproblema.

Uno dei problemi principali nello sviluppo di algoritmi Branch&Bound risulta quindi quello di decidere come effettuare la separazione. Uno dei metodi utilizzati più frequentemente è quello di considerare due vincoli contraddittori su una singola variabile (per esempio imporre che, data una variabile binaria y_1 , il sottoproblema 1 abbia come vincolo aggiuntivo $y_1 = 0$ mentre il sottoproblema 2 abbia come vincolo aggiuntivo $y_1 = 1$).

Un altro metodo per generare una separazione è applicabile a problemi che presentano vincoli nella forma chiamata vincolo superiore generalizzato:

$$\sum_{i \in S} y_i = 1$$
$$y_i = 0-1, i \in S$$

In questo caso, viene selezionata una parte della sommatoria che viene posta pari a zero per generare soluzioni mutuamente esclusive. Dato per esempio il vincolo $y_1+y_2+y_3+y_4 = 1$ è possibile costruire la separazione imponendo $y_1+y_2 = 0$ nel sottoproblema 1 e $y_3+y_4 = 0$ nel sottoproblema 2.

5.2.2 Rilassamento

Un problema di ottimizzazione denominato (RP) è rilassamento di un problema (P) se l'insieme delle soluzioni accettabili di (P) è un sottoinsieme dell'insieme delle soluzioni accettabili di (RP), ovvero $FS(P) \subseteq FS(RP)$.

Tale definizione implica le seguenti relazioni:

1. Se (RP) non ha soluzioni accettabili, (P) non ha soluzioni accettabili.
2. Definendo la soluzione ottimale di (P) z_P e la soluzione ottimale di (RP) z_{RP} allora $z_{RP} \leq z_P$ ovvero la soluzione del problema rilassato costituisce un vincolo inferiore per la soluzione del problema originale.
3. Se la soluzione ottimale di (RP) è accettabile per il problema (P) allora questa è la soluzione ottimale di (P)

Per effettuare il rilassamento del problema all'interno dell'algoritmo, la soluzione più adottata è quella di sostituire il vincolo di integralità delle variabili \mathbf{y} con una condizione di continuità (nel caso di variabili binarie sarà $0 \leq \mathbf{y} \leq 1$). In questo modo il problema assume la forma di un LP. Resta però da sottolineare come quanto più un rilassamento è facile da risolvere tanto più grande è l'intervallo tra la soluzione ottimale di (P) e il vincolo inferiore fornito da (RP).

5.2.3 Fathoming

Sia (CS) un sottoproblema candidato per risolvere (P). Bisogna determinare se la regione accettabile di (CS), $F(\text{CS})$, contiene la soluzione ottimale di (P) e trovarla in caso affermativo.

Un problema candidato (CS) verrà considerato sondato se è verificata una delle seguenti condizioni:

1. Può essere verificato che la regione accettabile $F(\text{CS})$ non può contenere una soluzione migliore della migliore soluzione trovata finora (ovvero la soluzione incombente).
2. Viene trovata una soluzione ottimale di (CS).

In ogni caso il problema viene considerato esaminato e non necessita di ulteriori separazioni.

Vi sono tre criteri generali di fathoming basati sul rilassamento e sulle seguenti assunzioni:

- (RCS) sia un rilassamento del problema candidato (CS).
- z_{RCS} sia il valore ottimale di (RCS).
- z_{CS} sia il valore ottimale di (CS).
- z^* sia il valore dell'incombente ($+\infty$ se nessun valore è ancora stato trovato).

Tali criteri sono dunque:

1. Se (RCS) non ha soluzioni accettabili, CS non ha soluzioni accettabili e può quindi essere considerato sondato.
2. Se la soluzione ottimale di (RCS) è maggiore o uguale all'incombente, $z_{\text{RCS}} \geq z^*$, allora (CS) può essere considerato sondato.

3. Se la soluzione ottimale di (RCS) risulta accettabile per (CS), questa è anche soluzione ottimale per (CS). Per questo (CS) può essere considerato sondato. Tale soluzione sarà anche accettabile per (P) e per questo diventerà il nuovo incumbente se il suo valore è inferiore a z^* .

5.2.4 Struttura del Branch&Bound

La generica struttura di un algoritmo Branch&Bound si compone dei seguenti passi:

1. **Inizializzazione:** Inizializzare la lista dei sottoproblemi candidati inserendo il problema MILP stesso e porre $z^* = +\infty$.
2. **Terminazione:** Se la lista dei problemi candidati è vuota, terminare con soluzione ottimale l'incumbente. Se l'incumbente non esiste il problema è impossibile.
3. **Selezione del sottoproblema candidato:** selezionare uno dei sottoproblemi nella lista e renderlo il problema candidato attuale (CS).
4. **Rilassamento:** selezionare un rilassamento (RCS) del problema attuale (CS), risolverlo e chiamare la soluzione z_{RCS} .
5. **Fathoming:** Applicare i tre criteri di fathoming:
 - Se (RCS) è impossibile, (CS) non ha soluzioni accettabili. Tornare allo step 2.
 - Se $z_{RCS} \geq z^*$ il (CS) corrente non ha soluzioni accettabili migliori dell'incumbente. Tornare allo step 2.
 - Se la soluzione ottimale di (RCS) è accettabile per (CS) è una soluzione ottimale di (CS). Se $z_{RCS} \leq z^*$ porre $z^* = z_{RCS}$. Tornare allo step 2.
6. **Separazione:** Eseguire una separazione del sottoproblema candidato corrente (CS) e aggiungere i suoi nodi figli alla lista dei sottoproblemi candidati. Tornare allo step 2.

Esiste ancora un grado di libertà nella scrittura dell'algoritmo, ovvero la scelta di quali candidati esaminare prima degli altri. Anche in questo caso esistono tre alternative principali:

- 1. Last In First Out (LIFO):** viene considerato come candidato corrente l'ultimo aggiunto alla lista dei sottoproblemi candidati. Quando questo viene sondato si ritorna indietro attraverso il nodo genitore fino a trovare un nodo che abbia un nodo figlio non ancora sondato. Tale approccio presenta come vantaggi principali la facile ri-ottimizzazione e il fatto che spesso la soluzione ottimale viene ottenuta in profondità nell'albero. Potrebbe però richiedere un elevato numero di sottocandidati per riuscire ad arrivare all'ottimo.
- 2. Ricerca breadth-first:** tutti i nodi ad un certo livello vengono considerati prima di passare al livello successivo. Viene utilizzato soprattutto come base 'per la selezione euristica dei nodi o per creare stime o miglioramenti veloci dei limiti.
- 3. Ricerca best bound:** il candidato successivo è quello nella lista dei candidati con il limite inferiore minore. Tale tipo di ricerca è quella che presenta il minor numero di problemi candidati.

Non bisogna inoltre dimenticare che un altro punto cruciale di tali algoritmi è la selezione della variabile rispetto alla quale effettuare la separazione. Dal momento che non sono ancora disponibili metodi robusti, una pratica comune è quella di stabilire una lista di priorità fornita dall'utente, creare degli pseudo - costi e utilizzare dei pesi.

Capitolo 6 – conclusioni

Dal punto di vista matematico esistono due metodi principali per la risoluzione di problemi LP: il metodo del Simplex sviluppato da Dantzig e il metodo dell'Interior Point sviluppato da Karmakar. Entrambi si sono rivelati fondamentali nella risoluzione di un numero sempre maggiore di problemi, soprattutto da parte di algoritmi che utilizzano entrambi i metodi. Tali metodi hanno peculiarità specifiche che li rendono più o meno efficaci a seconda delle situazioni. Qui di seguito si riportano i principali vantaggi e svantaggi.

Metodo del semplice

Pro:

- Garantisce il raggiungimento della soluzione ottimale (a patto che vengano fornite procedure anti degenerazione).
- Utilizza tecniche proprie della programmazione lineare.

Contro:

- Nei casi peggiori richiede un numero di iterazioni esponenziale rispetto la numero delle variabili. I problemi della famiglia di Klee-Minty, per esempio, richiederebbero anni di calcoli su computer estremamente potenti.

Metodo dell'Interior Point

Pro:

- Richiede un numero di iterazioni polinomiale rispetto al numero delle variabili.

Contro:

- Utilizza un approccio proprio dei problemi non lineari, pertanto lo sforzo computazionale per la singola iterazione potrebbe essere maggiore di quello necessario nel metodo del Simplex.

Il metodo dell'Attico presenta invece un approccio innovativo rispetto ad entrambi i metodi, a partire dalla sua ossatura matematica: esso è la diretta applicazione delle condizioni di Karush-Kuhn-Tucker e non è più necessario muoversi da un vertice ad uno adiacente e gli spostamenti interni alla regione accettabile sono dettati da ricerche lineari senza quindi appesantire la risoluzione con tecniche derivate dalla programmazione non lineare. Le matrici di lavoro sono inoltre generalmente più piccole: il metodo dell'Attico non richiede di trasformare il problema in forma standard e risulta quindi più immediato anche dal punto di vista dell'utilizzo in quanto non richiede ulteriori modifiche del LP originale. È inoltre sviluppato per poter garantire la massima stabilità della fattorizzazione della matrice di lavoro, con l'introduzione di alcune caratteristiche (utilizzo di vincoli artificiali in primis) non presenti nei precedenti metodi risolutivi.

I test finora effettuati utilizzando questo metodo sono risultati incoraggianti: come illustrato in questo lavoro di tesi il metodo dell'Attico effettua molti meno spostamenti per raggiungere la soluzione di quanto non facciano Simplex e Interior Point, a conferma della bontà della base matematica del modello e dell'efficacia delle sue caratteristiche innovative. Per poter però pensare ad un suo utilizzo applicativo saranno necessarie ancora ulteriori verifiche: dal punto di vista industriale il parametro che influenza maggiormente la scelta di un solutore rispetto ad un altro sono spesso i tempi di calcolo e un confronto su tale parametro non è ancora stato sviluppato. Per poterlo fare bisognerà effettuare ulteriori test, soprattutto sulla classe di problemi MILP in quanto la particolare struttura della matrice di lavoro del metodo dell'Attico dovrebbe portare i più grandi vantaggi soprattutto in tale tipologia di problemi vista la necessità di risolvere un elevato numero di problemi LP simili fra loro.

Resta inoltre da sottolineare un altro punto: se dal punto di vista prettamente matematico sono stati dimostrati e verificati i vantaggi (effettivi e potenziali) del metodo dell'Attico rispetto ai metodi esistenti, dal punto di vista numerico la situazione cambia radicalmente: dal momento che Simplex e Interior Point esistono da molti anni, è anche stato sviluppato un numero enorme di algoritmi diversi basati su tali metodi, ognuno con le sue peculiarità e i suoi accorgimenti. Gli algoritmi oggi più efficienti sono quindi il risultato di una evoluzione che ha portato all'implementazione di accorgimenti e tecniche (non ultima un pre-processamento della matrice di lavoro) che hanno portato tali algoritmi ad un livello di efficienza altissimo.

Lo sviluppo di algoritmi basati sul metodo dell'Attico è invece ad oggi ancora a livello embrionale e risulta quindi ancora privo di tutto quell'insieme di caratteristiche (per esempio il già citato preprocessore) che lo renderebbero numericamente competitivo e adatto a risolvere anche classi particolari di problemi (per esempio quelli con punto di partenza non accettabile, per i quali l'algoritmo attuale risulta ancora abbastanza lento nel trovare il primo vertice e far quindi partire la procedura risolutiva).

Le caratteristiche matematiche peculiari e i primi test e confronti effettuati in questo lavoro di tesi fanno tuttavia ipotizzare che, una volta sviluppato adeguatamente l'algoritmo, attraverso il metodo dell'Attico sarà possibile avere uno strumento efficace e performante per la risoluzione di problemi LP e MILP di dimensioni maggiori rispetto a quanto oggi possibile e permetterà dunque di aumentare ulteriormente l'importanza e l'impatto di questa già fondamentale classe di problemi.

Bibliografia

AMPL. (s.d.). *Solvers that works with AMPL.* Tratto da <http://www.ampl.com/solvers.html#Table>

Buzzi Ferraris, G., & Manenti, F. (2010). *Fundamentals and Linear Algebra for the Chemical Engineer: solving numericals problems.* Wiley Vch Verlag GmbH.

Buzzi-Ferraris, G. (2011). Linear Programming with the Attic Method. *I&EC research* , 50, 4858-4878.

Dantzig, G. B. (1963). *Linear Programming and Extensions.* RAND Corporation.

Floudas, C. A. (1995). *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications.* Oxford Univ Pr.

IBM. (s.d.). *IBM ILOGCPLEX Optimizer.* Tratto da <http://www-01.ibm.com/software/integration/optimization/cplex-optimizer/>

Karmakar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica* , 373-395.

Luenberger, D. G., & Ye, Y. (2010). *Linear and Nonlinear Programming.* Springer.

Manne, A. S. (1963). *Scheduling of petroleum refinery operations.* Cambridge: Harvard University Press.

Martin, R. K. (1998). *Large Scale Linear and Integer Optimization: A Unified Approach.* Springer.

MathWorks. (s.d.). *Matlab.* Tratto da <http://www.mathworks.it/products/matlab/index.html>

Vanderbei, R. (2001). *Linear Programming: foundations and extension.*