

# Tecniche di decomposizione e rilassamenti Lagrangiani

Antonio Frangioni \*

## Sommario

Una tecnica molto diffusa per costruire rilassamenti di problemi di Programmazione Lineare Intera (PLI) o mista, ed anche per risolvere problemi di Programmazione Lineare (PL) di grande dimensione, è quella del rilassamento Lagrangiano. È ben noto che il duale Lagrangiano di un problema di PL è equivalente al duale lineare del problema, e quindi al problema stesso, mentre il duale Lagrangiano di un problema di PLI è equivalente al “rilassamento convessificato” del problema in cui appare l’involuppo convesso dei punti interi che soddisfano i vincoli non rilassati, ossia una rappresentazione poliedrale della regione ammissibile del rilassamento Lagrangiano. È forse meno noto che gli algoritmi utilizzati per risolvere duali Lagrangiani calcolano automaticamente, o possono farlo con modifiche minori, una soluzione (approssimata) del rilassamento convessificato, fornendo quindi informazione primale che può risultare molto utile quando queste tecniche sono utilizzate all’interno di approcci, esatti o euristici, per risolvere il problema di PLI originale. Mostriamo come questo sia il caso per l’algoritmo dei piani di taglio, noto nel contesto della PL come metodo di decomposizione di Dantzig-Wolfe, e per molte sue varianti, così come per i più recenti algoritmi del tipo subgradiente. Discuteremo poi il possibile uso dell’informazione primale così ottenuta all’interno di approcci per la PLI.

**Keywords:** *Tecniche di decomposizione, rilassamenti Lagrangiani, Programmazione Lineare Intera.*

## 1 Rilassamento Lagrangiano e duale Lagrangiano

Moltissimi problemi di Programmazione Lineare Intera (PLI) posseggono un qualche tipo di “struttura” che può essere sfruttata per progettare approcci risolutivi efficienti. In questo articolo ci concentreremo su un tipo molto generale e diffuso di struttura, quella dei problemi di PLI nella forma

$$(P) \quad \min_x \{ cx : Ax = b, Ex = d, x \in \mathbb{N}^n \}$$

---

\*Dipartimento di Informatica, Via Buonarroti 2, 56127 Pisa (ITALY), e-mail: frangio@di.unipi.it

in cui i vincoli  $Ex = d$  sono “facili” – ( $P$ ) sarebbe un problema “facile” se i *vincoli complicanti*  $Ax = b$  fossero rimossi. In altre parole, l’introduzione dei vincoli  $Ax = b$  distrugge la struttura presente nei vincoli  $Ex = d$ , e che permetterebbe di usare algoritmi specifici, più efficienti, per la soluzione del problema. Alternativamente, o in aggiunta a questo, i vincoli  $Ex = d$  possono essere separabili, ossia il problema si decomporrebbe in sottoproblemi indipendenti di minore dimensione se non fosse per i vincoli  $Ax = b$ . Lo stesso tipo di struttura si trova in molti problemi in cui alcune delle variabili (o anche tutte) non sono sottoposte a vincoli di integralità, ma eviteremo di trattare esplicitamente questo caso per non appesantire la notazione.

Un tipico esempio di problema con questa struttura è il problema di Flusso Multicommodity “indivisibile” (unsplittable) di Costo Minimo. È dato un grafo  $G = (N, A)$ , che rappresenta ad esempio una rete di comunicazione, con costi di routing  $c = [c_{ij}] \geq 0$  e capacità  $u = [u_{ij}] > 0$  associate agli archi. Su questo grafo è individuato un insieme  $K$  di coppie origine/destinazione  $(o_h, d_h)$ , ciascuna con associata una domanda di comunicazione  $\delta_h$ . Si vuole selezionare per ciascuna coppia  $(o_h, d_h)$  un singolo cammino lungo il quale instradare tutte le  $\delta_h$  unità di flusso che rappresentano la comunicazione tra l’origine e la destinazione, col vincolo rappresentato dalla capacità degli archi, e minimizzando il costo di routing complessivo dato dalla somma pesata dei costi dei cammini utilizzati. Introducendo variabili di flusso  $x^h = [x_{ij}^h]$  per ciascuna coppia  $h \in K$  (detta *commodity*), il problema può essere formulato come

$$\begin{aligned} \min \sum_{h \in K} \delta_h \sum_{(i,j) \in A} c_{ij} x_{ij}^h \\ \sum_{(j,i) \in BS(i)} x_{ji}^h - \sum_{(i,j) \in FS(i)} x_{ij}^h &= b_i^h \quad (i,j) \in A, h \in K & (1) \\ \sum_{h \in K} \delta_h x_{ij}^h + s_{ij} &= u_{ij} \quad h \in K & (2) \\ x_{ij}^h \in \{0, 1\}, s_{ij} \geq 0 & & (i,j) \in A, h \in K \end{aligned}$$

dove  $b_i^h$  vale  $-1$  se  $i = o_h$ ,  $1$  se  $i = d_h$  e  $0$  altrimenti,  $FS(i)$  e  $BS(i)$  indicano rispettivamente la stella uscente e la stella entrante nel nodo  $i$ , e le  $s_{ij}$  sono variabili di scarto.

I vincoli di capacità (2) risultano “complicanti” per il problema in quanto se fossero rimossi il problema si decomporrebbe in  $|K|$  sottoproblemi indipendenti di cammino minimo, uno per ciascuna coppia origine/destinazione, e quindi potrebbe essere risolto molto efficientemente. La presenza dei vincoli (2) rende invece il problema “difficile”, sia per le grandi dimensioni sia per il fatto che la totale unimodularità dei vincoli di flusso (1) viene perduta, rendendo il problema  $\mathcal{NP}$ -arduo.

Un possibile modo per utilizzare questo tipo di struttura è quello di effettuare un *rilassamento Lagrangiano* di ( $P$ ) rispetto ai vincoli complicanti  $Ax = b$ , ossia considerare il problema

$$(P_y) \quad \min_x \{ cx + y(b - Ax) : Ex = d, x \in \mathbb{N}^n \}$$

per un fissato vettore  $y$  di *moltiplicatori Lagrangiani*. Per le ipotesi fatte ( $P_y$ ) è “facile”, ossia significativamente più facile da risolvere in pratica di ( $P$ ). Nell’esempio precedente è fissato un moltiplicatore Lagrangiano  $y_{ij}$  per ciascun vincolo di capacità, e quindi per ciascun arco; il rilassamento Lagrangiano si decompone in  $|K|$  problemi di cammino minimo, uno per ciascuna coppia  $(o_h, d_h)$ , rispetto

ai costi Lagrangiani  $c_{ij} - y_{ij}$ , più  $m$  problemi della forma  $\min_s \{-ys : s \geq 0\}$  che hanno soluzione finita se e solo se  $y_{ij} \leq 0$  per ogni  $(i, j) \in A$ .

Indichiamo con  $m$  è il numero di vincoli complicanti  $Ax = b$  e con  $v(\cdot)$  il valore ottimo della funzione obiettivo di un problema; è immediato verificare che, comunque scelto  $y \in \mathbb{R}^m$ ,  $(P_y)$  è un rilassamento di  $(P)$ , ossia risulta  $v(P_y) \leq v(P)$ . Infatti la regione ammissibile di  $(P_y)$  contiene quella di  $(P)$  e le due funzioni obiettivo coincidono sulla regione ammissibile di  $(P)$ , quindi la soluzione ottima di  $(P)$  è ammissibile per  $(P_y)$  ed ha lo stesso valore della funzione obiettivo nei due problemi. Si pone quindi il problema di determinare il migliore dei rilassamenti Lagrangiani, ossia di risolvere il *duale Lagrangiano*

$$(D) \quad \max_y \{ v(P_y) \} .$$

Siccome vale  $v(P_y) \leq v(P) \quad \forall y \in \mathbb{R}^m$ ,  $(D)$  è ancora un rilassamento di  $(P)$ , ossia vale  $v(D) \leq v(P)$ . È quindi interessante cercare di valutare la qualità della valutazione inferiore fornita da  $(D)$  confrontandola con quella delle valutazioni fornite da rilassamenti diversi. Nel seguito richiameremo alcuni noti risultati a tale proposito.

Nel caso della PL la dualità Lagrangiana coincide con quella lineare. Si consideri infatti il rilassamento continuo di  $(P)$

$$(\bar{P}) \quad \min_x \{ cx : Ax = b, Ex = d, x \geq 0 \} ,$$

il rilassamento Lagrangiano di  $(\bar{P})$  rispetto ad  $Ax = b$

$$(\bar{P}_y) \quad \min_x \{ cx + y(b - Ax) : Ex = d, x \geq 0 \}$$

ed il corrispondente duale Lagrangiano

$$(\bar{D}) \quad \max_y \{ v(\bar{P}_y) \} .$$

Poiché  $(\bar{P}_y)$  è un problema di PL si ha

$$v(\bar{P}_y) = \max_w \{ wd : wE \leq c - yA \} + yb ,$$

e dei conseguenza

$$v(\bar{D}) = \max_{y,w} \{ yb + wd : wE + yA \leq c \} ;$$

è immediato riconoscere nella precedente espressione il duale lineare di  $(\bar{P})$ .

Quindi, nel caso della PL il duale Lagrangiano è in effetti il duale lineare (si può dire che sia un *duale parziale*). Difatti il duale Lagrangiano segue regole analoghe a quelle del duale lineare per quanto riguarda i segni dei moltiplicatori: se i vincoli complicanti hanno la forma  $Ax \leq b$  i moltiplicatori Lagrangiani sono vincolati ad essere non positivi ( $y \leq 0$ ), come abbiamo visto nell'esempio, mentre se i vincoli complicanti hanno la forma  $Ax \geq b$  i moltiplicatori Lagrangiani sono vincolati ad essere non negativi ( $y \geq 0$ ).

Nel caso della PLI vale una generalizzazione del risultato precedente. Conviene esprimere il problema originale come

$$(P) \quad \min_x \{ cx : Ax = b, x \in X \}$$

dove  $X = \{ x : Ex = d, x \in \mathbb{N}^n \}$ . È ben noto che, nell'ambito della PLI, l'insieme discreto  $X$  può essere rimpiazzato dal suo *inviluppo convesso*  $\text{conv}(X)$ , ossia

$$v(P_y) = \min_x \{ cx + y(b - Ax) : x \in \text{conv}(X) \}$$

per ogni  $y \in \mathbb{R}^m$ . È altrettanto noto che (se la matrice  $E$  ed il vettore  $d$  hanno tutte componenti razionali)  $\text{conv}(X)$  è un poliedro convesso, ossia esiste un insieme finito di vincoli lineari tali che

$$\text{conv}(X) = \{ x : \tilde{A}x \geq \tilde{b} \} .$$

Definiamo quindi il *rilassamento convessificato* di  $(P)$  come

$$(\tilde{P}) \quad \min_x \{ cx : Ax = b, x \in \text{conv}(X) \} .$$

$(\tilde{P})$  è un problema di PL, per quanto in generale la descrizione poliedrale di  $\text{conv}(X)$  non sia disponibile. È facile dimostrare il seguente risultato fondamentale:

**Teorema 1.1**  $v(D) = v(\tilde{P})$ .

*Dimostrazione:* Dalla definizione si ha

$$v(\tilde{P}) = \min_x \{ cx : Ax = b, \tilde{A}x \geq \tilde{b} \} .$$

Dalla dualità lineare si ottiene quindi

$$v(\tilde{P}) = \max_{y,w} \{ yb + w\tilde{b} : yA + w\tilde{A} = c, w \geq 0 \}$$

che può essere riscritto come

$$\max_y \{ yb + \max_w \{ w\tilde{b} : w\tilde{A} = c - yA, w \geq 0 \} \} .$$

Ancora per dualità lineare, applicata al problema interno, si ha

$$v(\tilde{P}) = \max_y \{ yb + \min_x \{ (c - yA)x : \tilde{A}x \geq \tilde{b} \} \}$$

e quindi il teorema è dimostrato.  $\square$

Il Teorema 1.1 ha alcune importanti conseguenze, tra le quali vengono di solito sottolineate le seguenti:

- Poiché la regione ammissibile di  $(\tilde{P})$  è contenuta in quella di  $(\bar{P})$ , e le funzioni obiettivo dei due problemi coincidono,  $(\tilde{P})$  è un rilassamento di  $(P)$  non peggiore di  $(\bar{P})$ , e quindi  $v(D) \geq v(\bar{P})$ .
- Se i vincoli  $Ex = d$  hanno la *proprietà di integralità*, ossia  $v(P_y) = v(\bar{P}_y)$  per ogni  $y \in \mathbb{R}^m$ , allora si ha che  $v(D) = v(\tilde{P}) = v(\bar{P})$ ; si ha cioè un “principio di conservazione della difficoltà”, per cui se il sottoproblema Lagrangiano è “facile” (il vincolo di integralità è soddisfatto automaticamente, come avviene ad esempio nei problemi di flusso) allora il duale Lagrangiano è equivalente al rilassamento continuo, mentre per ottenere una valutazione inferiore strettamente migliore di quella fornita dal rilassamento continuo è necessario che il sottoproblema Lagrangiano sia “difficile”, ossia che i vincoli  $Ex = d$  non forniscano una descrizione “esatta” di  $\text{conv}(X)$ .

- Le osservazioni precedenti consentono in alcuni casi di valutare le prestazioni relative di rilassamenti Lagrangiani diversi dello stesso problema. Ad esempio, si consideri il duale Lagrangiano di  $(P)$  rispetto al secondo blocco di vincoli

$$(D') \quad \max_w \{ \min_x \{ cx + y(d - Ex) : Ax = b, x \in \mathbb{N}^n \} \} .$$

Se sia i vincoli  $Ax = b$  che i vincoli  $Ex = d$  posseggono la proprietà di integralità allora  $v(D') = v(D) = v(\bar{P})$ ; se invece i vincoli  $Ex = d$  posseggono la proprietà di integralità ed i vincoli  $Ax = b$  non la posseggono allora  $v(D') \geq v(D) = v(\bar{P})$ , e la disuguaglianza può essere stretta. Questo accade nell'esempio del problema di Flusso Multicommodity indivisibile di Costo Minimo: il duale Lagrangiano rispetto ai vincoli di capacità (2) si decompone in  $m$  problemi indipendenti di zaino binario, che non posseggono la proprietà di integralità, e quindi il corrispondente duale Lagrangiano può fornire una valutazione inferiore di  $v(P)$  strettamente migliore di quella del rilassamento continuo.

- Un diverso modo di sfruttare il rilassamento Lagrangiano è la cosiddetta *decomposizione Lagrangiana* [GK87], che consiste nel riscrivere  $(P)$  nella forma equivalente

$$(P) \quad \min_{x', x''} \begin{cases} c(x' + x'')/2 \\ Ax' = b, \quad x' \in \mathbb{N}^n \\ Ex'' = d, \quad x'' \in \mathbb{N}^n \\ x' = x'' \end{cases}$$

ed a risolvere il duale Lagrangiano rispetto ai vincoli  $x' = x''$ , ossia

$$(D'') \quad \max_w \begin{cases} \min_x \{ (c/2 - w)x : Ex = d, x \in \mathbb{N}^n \} + \\ \min_x \{ (c/2 + w)x : Ax = b, x \in \mathbb{N}^n \} \end{cases} .$$

Dal Teorema 1.1 segue facilmente che

$$v(D'') = \min_x \{ cx : x \in \text{conv}(X) \cap \text{conv}(\{ Ax = b, x \in \mathbb{N}^n \}) \} ,$$

e quindi che  $v(D'') \geq \max\{v(D'), v(D)\} \geq v(\bar{P})$ . In altri termini, la decomposizione Lagrangiana fornisce una valutazione inferiore di  $v(P)$  non peggiore di quella fornita da ciascuno dei due rilassamenti Lagrangiani, e quindi non peggiore di quella fornita dal rilassamento continuo. In particolare, è facile verificare che  $v(D'')$  coincide con  $v(P)$  se sia i vincoli  $Ax = b$  che i vincoli  $Ex = d$  posseggono la proprietà di integralità, che  $v(D'')$  coincide con la valutazione fornita dal migliore dei due rilassamenti Lagrangiani se uno solo dei due insiemi di vincoli possiede la proprietà di integralità, e che  $v(D'')$  può essere strettamente maggiore di  $\max\{v(D'), v(D)\}$  se nessuno dei due insiemi di vincoli possiede la proprietà di integralità.

Le proprietà precedentemente enunciate derivano dal fatto che i valori ottimi delle funzioni obiettivo di  $(D)$  e  $(\tilde{P})$  coincidono. Quello che non sempre viene messo in luce, nella descrizione dei rilassamenti Lagrangiani per problemi di PLI, sono le seguenti ulteriori conseguenze del Teorema 1.1:

- $(D)$  e  $(\tilde{P})$  non solo hanno lo stesso valore della funzione obiettivo, ma sono il duale (lineare) l'uno dell'altro;

- di conseguenza, gli algoritmi che risolvono  $(D)$  e che hanno una regola di terminazione effettiva (ossia in grado di *dimostrare* l'ottimalità della soluzione ottenuta) tipicamente devono produrre una *soluzione ottima*  $\tilde{x}$  di  $(\tilde{P})$  come *certificato di ottimalità*;
- quindi, la soluzione di  $(D)$  mediante uno di tali algoritmi fornisce non solo una valutazione inferiore su  $v(P)$ , ma anche l'*informazione primale*  $\tilde{x}$  ad essa associata che può risultare utile per lo sviluppo di approcci, esatti o euristici, per  $(P)$ .

Nella prossima sezione vedremo come un semplice algoritmo che risolve  $(D)$ , il metodo dei piani di taglio di Kelley, produca a terminazione una soluzione ottima  $\tilde{x}$  di  $(\tilde{P})$ . Questo algoritmo è l'interpretazione duale del ben noto *metodo di decomposizione di Dantzig-Wolfe* per la Programmazione Lineare strutturata, ed è strettamente correlato con i metodi di generazione di colonne per la PL con un numero elevato di variabili.

## 2 Piani di taglio, decomposizione di Dantzig-Wolfe e generazione di colonne

Per semplificare la trattazione assumeremo temporaneamente che l'insieme  $X$  sia *compatto*; nel seguito mostreremo come l'assunzione possa essere eliminata al costo di complicare leggermente la notazione e gli algoritmi. Possiamo inoltre assumere che  $X$  non sia vuoto: se  $X$  è vuoto allora  $(P)$  non ha soluzione, e questo può essere determinato semplicemente risolvendo  $(P_y)$  con un qualsiasi vettore di moltiplicatori  $y$ .

L'immediata conseguenza di queste assunzioni è che la *funzione Lagrangiana*

$$\phi(y) = \min_x \{ cx + y(b - Ax) : x \in X \}$$

è finita ovunque, ossia ha per *dominio effettivo* (l'insieme dei punti in cui assume valore finito) l'intero  $\mathbb{R}^m$ , ed è *propria*, ossia  $\phi(y) < +\infty \forall y$ . Essendo il minimo puntuale di (un numero finito di) funzioni lineari,  $\phi$  è concava; in particolare, dato che  $X$  è un insieme discreto,  $\phi$  è *poliedrale*, ossia il suo *epigrafo*

$$\text{Epi}(\phi) = \{ (v, y) : v \leq \phi(y) \}$$

è un poliedro. In altri termini, ad ogni elemento  $x$  di  $X$  che risolve  $(P_y)$  per un qualche  $y$  è associato un *iperpiano di supporto*  $b - Ax$  dell'epigrafo di  $\phi$ , noto come *subgradiente* di  $\phi$  in  $y^1$ . Chiaramente  $\phi$  non è differenziabile ovunque, per quanto l'insieme dei punti in cui non è differenziabile abbia misura nulla, ed in particolare in ogni punto  $y$  in cui  $\phi$  non è differenziabile  $(P_y)$  ha soluzioni ottime multiple. La nondifferenziabilità di  $\phi$  non consente di usare, per la soluzione di  $(D)$ , la maggior parte degli algoritmi efficienti di ottimizzazione nonlineare, che sono basati sull'esistenza (e la computazione) di derivate prime ed eventualmente seconde della funzione.

<sup>1</sup>In realtà il nome subgradiente sarebbe appropriato per una funzione convessa, mentre per la funzione concava  $\phi$  si dovrebbe parlare di "supergradiente", così come di "ipografo".

## 2.1 L'Algoritmo dei piani di taglio

Il duale Lagrangiano  $(D)$  corrisponde alla massimizzazione della funzione concava nondifferenziabile  $\phi$ ; è però facile riscrivere  $(D)$  come un problema di PL (questo non dovrebbe stupire, in quanto  $(D)$  è il duale lineare di  $(\tilde{P})$ ). Infatti, è immediato verificare che un modo equivalente di formulare  $(D)$  è

$$(D) \quad \max_{v,y} \{ v : v \leq cx + y(b - Ax), x \in X \}. \quad (3)$$

In (3), ciascun elemento di  $X$  definisce un vincolo del problema, ossia possibilmente definisce una faccia della regione ammissibile, che altro non è che  $Epi(\phi)$ . Il fatto che sia possibile scrivere  $(D)$  come un problema di PL non implica direttamente che  $(D)$  sia di “facile” risoluzione: il numero di vincoli del problema, corrispondente al numero di elementi di  $X$ , può essere enorme. È però vero che non tutti i vincoli di  $(D)$  sono necessari per la determinazione della soluzione ottima; al limite, sarebbero sufficienti gli  $m + 1$  vincoli corrispondenti ad una base ottima del problema. Ciò suggerisce un approccio di *generazione di vincoli*, in cui ad ogni iterazione si risolve il *Master Problem* (Duale)

$$(D_{\mathcal{B}}) \quad \max_{v,y} \{ v : v \leq cx + y(b - Ax), x \in \mathcal{B} \}.$$

ove  $\mathcal{B} \subset X$  è un “piccolo” sottoinsieme delle soluzioni ammissibili di  $(P_y)$ . Ciò corrisponde a risolvere

$$\max_y \{ \phi_{\mathcal{B}}(y) = \min_x \{ cx + y(b - Ax) : x \in \mathcal{B} \} \},$$

ossia a massimizzare la funzione concava poliedrale  $\phi_{\mathcal{B}}$ , tale che  $\phi_{\mathcal{B}}(y) \geq \phi(y) \forall y$ , al posto di  $\phi$ ;  $\phi_{\mathcal{B}}$  è detta *modello cutting plane* di  $\phi$ . La soluzione ottima  $(v^*, y^*)$  di  $(D_{\mathcal{B}})$  può quindi essere usata per generare un ulteriore vincolo, se necessario, semplicemente risolvendo il rilassamento Lagrangiano  $(P_{y^*})$  ad esso relativa, ossia calcolando  $\phi(y^*)$ ; in questo contesto  $(P_{y^*})$  viene detto *problema di separazione*. Se infatti si ha  $v^* > \phi(y^*)$ , allora una qualsiasi soluzione ottima  $\bar{x}$  di  $(P_{y^*})$  fornisce un vincolo di  $(D)$  violato dalla soluzione corrente  $(v^*, y^*)$ , che può quindi essere aggiunto a  $\mathcal{B}$ . Altrimenti, ed è facile verificare che in questo caso risulta  $v^* = \phi(y^*)$ ,  $(v^*, y^*)$  rispetta tutti i vincoli in  $(D)$ , anche quelli non esplicitamente rappresentati in  $\mathcal{B}$ , e quindi è ottima per  $(D)$ .

```

< inizializza  $\mathcal{B}$  >;
do
   $(v^*, y^*) = \operatorname{argmax}_{v,y} \{ v : v \leq cx + y(b - Ax), x \in \mathcal{B} \};$  /*  $(D_{\mathcal{B}})$  */
   $\bar{x} = \operatorname{argmin}_x \{ (c - y^*A)x : x \in X \};$  /*  $(P_{y^*})$  */
   $\phi(y^*) = c\bar{x} + y^*(b - A\bar{x}); \mathcal{B} = \mathcal{B} \cup \{\bar{x}\};$ 
while(  $v^* > \phi(y^*)$  );

```

Figura 1: L'algoritmo dei piani di taglio

L'*algoritmo dei piani di taglio di Kelley* [Ke60], sintetizzato in Figura 1, determina quindi ad ogni passo una valutazione inferiore ed una superiore su  $v(D)$ , in quanto  $v^* \geq v(D) \geq \phi(y^*)$ , e termina in un numero finito di passi (al limite  $\mathcal{B} = X$ ) quando le due coincidono. Occorre solamente assicurarsi che

l'insieme di vincoli  $\mathcal{B}$  determinato dalla fase di inizializzazione sia sufficiente ad assicurare che  $(D_{\mathcal{B}})$  abbia soluzione ottima finita; un modo in cui questo può essere ottenuto è ponendo  $\mathcal{B} = \{\hat{x}\}$  dove  $\hat{x}$  è una soluzione ammissibile per  $(P)$ , il che corrisponde ad inserire in  $(D_{\mathcal{B}})$  il vincolo  $v \leq c\hat{x}$ .

## 2.2 Il metodo di decomposizione di Dantzig-Wolfe

L'algoritmo dei piani di taglio può essere "rivisitato" in notazione primale, nel qual caso prende il nome di *metodo di decomposizione di Dantzig-Wolfe*. Per questo occorre semplicemente notare che  $(D_{\mathcal{B}})$  ha un duale (lineare), il *Master Problem Primale*

$$(P_{\mathcal{B}}) \quad \min_{\theta} \{ c(\sum_{x \in \mathcal{B}} x\theta_x) : A(\sum_{x \in \mathcal{B}} x\theta_x) = b, \theta \in \Theta \} \quad (4)$$

dove  $\Theta = \{ \theta \geq 0 : \sum_{x \in \mathcal{B}} \theta_x = 1 \}$  è il simpleso unitario di dimensione opportuna.  $(P_{\mathcal{B}})$  ha una variabile per ogni riga di  $(D_{\mathcal{B}})$ , ossia per ciascun elemento di  $\mathcal{B}$ . È interessante notare che questa "forma esplicita" del problema è equivalente alla "forma implicita"

$$(P_{\mathcal{B}}) \quad \min_x \{ cx : Ax = b, x \in X_{\mathcal{B}} = \text{conv}(\mathcal{B}) \}. \quad (5)$$

Questo chiarisce la relazione tra l'algoritmo dei piani di taglio e  $(\tilde{P})$ ; infatti, (4) con  $\mathcal{B} = X$  è una formulazione di  $(\tilde{P})$ , in cui sono esplicitamente rappresentati i moltiplicatori convessi  $\theta$ . In particolare, (4) con  $\mathcal{B} = X$  è il duale lineare di (3), come anticipato dal Teorema 1.1; solamente, in questo caso  $\text{conv}(X)$  è espresso mediante una rappresentazione *per punti*, piuttosto che mediante la più usuale rappresentazione *per facce* utilizzata nel teorema.

L'algoritmo dei piani di taglio può quindi essere "rivisitato" dal punto di vista primale, notando che la "forma esplicita" di  $(\tilde{P})$  è un problema di PL con "molte" colonne, una per ciascun elemento di  $X$ . Questo suggerisce un approccio di generazione di colonne nel quale si risolve la restrizione di  $(\tilde{P})$  rispetto al sottoinsieme di colonne  $\mathcal{B}$ , ottenendo una soluzione  $x^*$  ammissibile per  $(\tilde{P})$  ed un vettore di variabili duali  $y^*$  dei vincoli  $Ax = b$ . Le variabili duali determinano il *costo ridotto*  $(c - yA)x$  della variabile  $\theta_x$ , ossia della colonna corrispondente. Siccome i costi ridotti di tutte le colonne in  $\mathcal{B}$  sono non negativi all'ottimo di  $(P_{\mathcal{B}})$ , per verificare se  $x^*$  è ottima per  $(\tilde{P})$  è sufficiente verificare se esiste oppure no una colonna  $x \in X \setminus \mathcal{B}$  di costo ridotto negativo. Questo viene fatto risolvendo  $(P_{y^*})$ , che determina la colonna di costo ridotto minimo; in questo contesto  $(P_{y^*})$  viene detto *problema di pricing*. Se il costo ridotto minimo tra le colonne di  $X$  è negativo allora è stata generata una colonna "promettente" che può essere inserita in  $\mathcal{B}$  per migliorare la soluzione  $x^*$ ; altrimenti  $x^*$  è ottima per  $(\tilde{P})$ . Questi passi sono esattamente quelli effettuati dall'algoritmo in Figura 1, in cui semplicemente  $x^*$  non era stata esplicitata. Quindi, l'algoritmo dei piani di taglio fornisce, a terminazione, una soluzione  $x^*$  ottima per  $(\tilde{P})$ .

L'analisi appena svolta dimostra che, almeno nel caso dell'algoritmo dei piani di taglio, risolvere  $(D)$  è in effetti equivalente a risolvere  $(\tilde{P})$ , e suggerisce le seguenti considerazioni:

- Dal punto di vista primale il Master Problem utilizza l'*approssimazione interna*  $X_{\mathcal{B}}$  di  $\text{conv}(X)$ , che viene espansa aggiungendo i punti generati dal problema di pricing  $(P_{y^*})$  finché non contiene l'ottimo di  $(\tilde{P})$ ; dal



punto di vista duale il Master Problem utilizza l'*approssimazione esterna*  $\phi_{\mathcal{B}}$  di  $\phi$ , e quindi della regione ammissibile di  $(D)$ , che viene raffinata inserendo i subgradienti generati dal problema di separazione finché non risulta "esatta" nell'ottimo di  $(D)$ .

- La struttura del problema viene utilizzata per *generare efficientemente punti* (estremi) di  $\text{conv}(X)$ ; in altri termini, questo procedimento è particolarmente attraente nel caso in cui ottenere una rappresentazione esplicita di  $\text{conv}(X)$  sia significativamente più difficile che ottimizzare su  $X$ . In effetti, come mostrato nel paragrafo precedente, la soluzione del duale Lagrangiano di un problema di PLI è particolarmente attraente nel caso in cui i vincoli  $Ex = d$  non rilassati *non* hanno la proprietà di integralità, ossia non rappresentano esattamente  $\text{conv}(X)$ , ma si dispone comunque di un algoritmo "ragionevolmente efficiente" per risolvere il rilassamento Lagrangiano.
- Gli algoritmi per la soluzione del duale Lagrangiano devono in effetti risolvere, dal punto di vista primale, un problema di PL con "molte" colonne avendo a disposizione un modo efficiente per generare "nuove" colonne; quindi, esiste una stretta relazione tra le tecniche Lagrangiane e gli algoritmi di generazione di colonne, tanto che l'algoritmo dei piani di taglio è *essattamente* il classico algoritmo di generazione di colonne applicato alla "forma esplicita" di  $(\tilde{P})$ . Questa relazione può essere sfruttata per costruire algoritmi di generazione di colonne più efficienti, come vedremo nel seguito.

### 2.3 Estensioni: $X$ non compatto e/o decomponibile

L'algoritmo nei piani di taglio può essere facilmente esteso al caso in cui  $X$  non sia compatto, ossia il dominio effettivo di  $\phi$  non sia tutto  $\mathbb{R}^m$ . Il poliedro  $\text{conv}(X)$  può essere in generale espresso come la somma di un politopo  $P$  e di un cono finitamente generato  $C$ , detto *cono delle direzioni* di  $\text{conv}(X)$  (quando  $X$  è compatto si ha  $C = \{0\}$ ). Ogni vettore  $\nu \in C$  costituisce una direzione ammissibile illimitata per  $\text{conv}(X)$ ; ne consegue che se per un qualche  $y$  si ha  $(c - yA)\nu < 0$  allora il rilassamento Lagrangiano  $(P_y)$  è inferiormente illimitato, ossia  $\phi(y) = -\infty$ . In altri termini, a qualunque vettore  $\nu \in C$  è associato un vincolo lineare  $(c - yA)\nu \geq 0$  che è rispettato da tutti i punti  $y$  nel dominio effettivo di  $\phi$ . Quindi,  $C$  – ed in particolare i suoi *generatori* o *raggi estremi* – caratterizza il dominio effettivo di  $\phi$ .

L'algoritmo dei piani di taglio si estende dunque al caso in cui  $X$  non sia compatto semplicemente mantenendo l'insieme  $\mathcal{B} = \mathcal{B}^0 \cup \mathcal{B}^1$  in cui  $\mathcal{B}^0 \subset C$  e  $\mathcal{B}^1 \subset X$ . Il Master Problem Primale e Duale divengono rispettivamente

$$(P_{\mathcal{B}}) \quad \min_{\theta} \begin{cases} c \left( \sum_{x \in \mathcal{B}^0} x\theta_x + \sum_{\nu \in \mathcal{B}^0} \nu\theta_{\nu} \right) \\ A \left( \sum_{x \in \mathcal{B}^1} x\theta_x + \sum_{\nu \in \mathcal{B}^0} \nu\theta_{\nu} \right) = b \\ \sum_{x \in \mathcal{B}^1} \theta_x = 1 \\ \theta \geq 0 \end{cases}, \quad (6)$$

$$(D_{\mathcal{B}}) \quad \max_{v,y} \begin{cases} yb + v \\ v \leq (c - yA)x \quad x \in \mathcal{B}^1 \\ 0 \leq (c - yA)\nu \quad \nu \in \mathcal{B}^0 \end{cases}.$$

Un modo equivalente di riscrivere i problemi è

$$(P_{\mathcal{B}}) \quad \min_x \{ cx : Ax = b, x \in \text{conv}(\mathcal{B}^1) + \text{cone}(\mathcal{B}^0) \},$$

$$(D_{\mathcal{B}}) \quad \max_y \{ \phi_{\mathcal{B}^1}(y) : y \in Y_{\mathcal{B}} \},$$

dove  $Y_{\mathcal{B}} = \{ y : (c - yA)v \geq 0, v \in \mathcal{B}^0 \}$  è un'approssimazione esterna del dominio effettivo di  $\phi$ . Ad ogni iterazione dell'algoritmo, la soluzione del rilassamento Lagrangiano riporta o una soluzione ottima  $\bar{x} \in X$ , che viene quindi aggiunta a  $\mathcal{B}^1$ , oppure una direzione di decrescita illimitata  $\bar{v} \in C$ , che viene quindi aggiunta a  $\mathcal{B}^0$ . Si noti che nel secondo caso si ha  $\phi(y^*) = -\infty$ , e di conseguenza in questo tipo di iterazioni non si ha a disposizione una nuova valutazione inferiore su  $z(D)$ .

È interessante rilevare che al limite si può avere  $P = \{0\}$ , ossia  $\text{conv}(X)$  è un cono; in questo caso in *nessuna* iterazione, tranne l'ultima, si dispone di una valutazione inferiore su  $z(D)$ , ossia in tutte le iterazioni, tranne l'ultima, si ha  $\phi(y^*) = -\infty$ . Questo è in effetti il caso normalmente considerato nella descrizione degli algoritmi di generazione di colonne, in cui si ha un problema nella forma (6) con  $\mathcal{B}_1 = \{0\}$ , ossia non si ha un vincolo di convessità sulle colonne; non è quindi possibile stimare il gap tra la soluzione primale corrente  $x^*$  e quella ottima, poiché non si ha a disposizione una soluzione duale ammissibile che fornisca una valutazione inferiore. Si noti che per qualsiasi problema si può assumere  $P = \{0\}$  semplicemente considerando il vincolo di convessità alla pari degli altri, ossia lavorando nello spazio  $\mathbb{R}^{m+1}$  in cui  $v$  viene considerato alla stregua degli altri moltiplicatori Lagrangiani; di conseguenza, si potrebbe dire che la descrizione in termini di generazione di colonne è più generale di quella in termini della funzione Lagrangiana. Quest'ultima però risulta, per le ragioni precedentemente discusse, più conveniente da considerare; in effetti, negli approcci di generazione di colonne la presenza di vincoli di convessità (non infrequente nelle applicazioni) viene sfruttata per calcolare approssimazioni inferiori [CGN95, BA02].

Una diversa estensione dell'algoritmo dei piani di taglio si ha nel caso, molto frequente nelle applicazioni, in cui  $X$  è il prodotto cartesiano di  $k$  di insiemi (che assumiamo temporaneamente compatti)  $X = X_1 \times X_2 \times \dots \times X_k$ , ossia il rilassamento Lagrangiano si decompone in  $k$  problemi indipendenti ed una soluzione ottima  $\bar{x} = [\bar{x}^1, \bar{x}^2, \dots, \bar{x}^k]$  del rilassamento Lagrangiano si ottiene semplicemente giustapponendo le soluzioni ottime dei  $k$  sottoproblemi; questo è ad esempio il caso del problema di Flusso Multicommodity. In altri termini, la funzione Lagrangiana

$$\phi(y) = yb + \sum_{h \in K} \phi^h(y)$$

( $K = \{1, 2, \dots, k\}$ ) si decompone nella somma di  $k$  funzioni

$$\phi^h(y) = \min_{x^h} \{ (c^h - yA^h)x^h : x^h \in X^h \},$$

e  $-A^h \bar{x}^h$  è un subgradiente di  $\phi^h$  (si può pensare che la funzione lineare  $yb$  sia la  $(k+1)$ -esima componente). Si può allora risolvere ad ogni iterazione, al posto di (4), il *Master Problem Primale disaggregato*

$$(P_{\mathcal{B}}) \quad \min_{\theta} \left\{ \begin{array}{l} \sum_{h \in K} c^h \sum_{x \in \mathcal{B}^h} x \theta_x \\ \sum_{h \in K} A^h \sum_{x \in \mathcal{B}^h} x \theta_x = b \\ \sum_{x \in \mathcal{B}^h} \theta_x = 1 \quad h \in K \\ \theta \geq 0 \end{array} \right. , \quad (7)$$

in cui tutte le componenti  $h$ -esime delle soluzioni generate sono immagazzinate nell'insieme  $\mathcal{B}^h$  ed hanno un moltiplicatore  $\theta_x$  indipendente dalle altre componenti della stessa soluzione. Il corrispondente *Master Problem Duale disaggregato* è

$$(D_{\mathcal{B}}) \quad \max_{v,y} \left\{ \begin{array}{l} yb + \sum_{h \in K} v^h \\ v^h \leq (c^h - yA^h)x^h \quad x^h \in \mathcal{B}^h \quad h \in K \end{array} \right. .$$

I due problemi possono essere scritti rispettivamente come

$$(P_{\mathcal{B}}) \quad \min_x \left\{ \begin{array}{l} \sum_{h \in K} c^h x^h \\ \sum_{h \in K} A^h x^h = b \\ x^h \in \text{conv}(\mathcal{B}^h) \quad h \in K \end{array} \right. \quad (8)$$

$$(D_{\mathcal{B}}) \quad \max_y \left\{ \sum_{h \in K} \phi_{\mathcal{B}^h}(y) \right\}$$

in cui  $\phi_{\mathcal{B}^h}$  è il modello cutting plane dell' $h$ -esima componente  $\phi^h$  di  $\phi$ . Dal punto di vista della generazione di colonne, ciò corrisponde al caso, non infrequente nelle applicazioni, in cui non si ha un solo vincolo di convessità che riguarda tutte le colonne, ma si hanno  $k$  vincoli di convessità che riguardano sottoinsiemi disgiunti di colonne; le variabili  $v^h$  sono le variabili duali dei vincoli di convessità, e la loro somma fornisce una valutazione inferiore sull'ottimo di  $(\tilde{P})$ .

È facile verificare come, dato uno stesso insieme di soluzioni  $\mathcal{B} \subset X$ , l'insieme ammissibile di (8) contenga strettamente quello di (5); infatti, (4) è la restrizione di (7) in cui tutte le componenti  $x^h$  corrispondenti ad una stessa soluzione  $x$  sono forzate ad avere lo stesso moltiplicatore. In altri termini,  $\text{conv}(\mathcal{B}^1) \times \text{conv}(\mathcal{B}^2) \times \dots \times \text{conv}(\mathcal{B}^k)$  è una migliore approssimazione di  $\text{conv}(X)$  rispetto a  $\text{conv}(\mathcal{B})$ ; alternativamente, si può dire che la somma dei  $k$  modelli cutting plane  $\phi_{\mathcal{B}^h}$  è una migliore approssimazione di  $\phi$  rispetto al modello cutting plane "aggregato"  $\phi_{\mathcal{B}}$ . Il Master Problem disaggregato ha dimensione maggiore di un fattore  $k$  rispetto al Master Problem aggregato (a parità di informazione raccolta), ed è quindi più costoso da risolvere; il miglior uso dell'informazione disponibile determina però spesso una convergenza sensibilmente più rapida dell'algoritmo [JLFP93, BFLN02], che può abbondantemente controbilanciare il maggior costo della soluzione del Master Problem.

Nel caso in cui alcuni degli insiemi  $X^h$  non siano compatti l'algoritmo può essere esteso analogamente a quanto visto in precedenza. Si noti che una direzione  $\nu^h$  di un qualsiasi insieme  $\text{conv}(X^h)$  non è soggetta ad alcun vincolo di convessità, per cui non è necessario partizionare l'insieme  $\mathcal{B}^0$  delle direzioni secondo le componenti; in altre parole, è sufficiente che sia  $\phi^h(y) = -\infty$  per una sola delle componenti di  $\phi$  affinché risulti  $\phi(y) = -\infty$ , e quindi  $\nu^h$  definisce un vincolo  $(c^h - yA^h)\nu^h \geq 0$  che è rispettato da tutti i punti del dominio effettivo di  $\phi$ .

### 3 Altri algoritmi per il duale Lagrangiano

Per quanto l'algoritmo dei piani di taglio sia facile da comprendere e da implementare, in pratica si rivela spesso poco efficiente; nel seguito discuteremo brevemente le cause fondamentali della scarsa efficienza dell'algoritmo e presenteremo altri approcci che non soffrono degli stessi inconvenienti, illustrando per ciascuno di essi se e come permetta di determinare una soluzione ottima di  $(\tilde{P})$ .

### 3.1 Instabilità dell'algoritmo dei piani di taglio

Un primo problema dell'algoritmo dei piani di taglio è che, qualora non si disponga di una soluzione ammissibile di  $(P)$ , o, più in generale, di una valutazione superiore dell'ottimo di  $(D)$ , non è facile costruire un insieme  $\mathcal{B}$  tale che  $(D_{\mathcal{B}})$  abbia ottimo finito, o equivalentemente  $(P_{\mathcal{B}})$  possieda soluzioni ammissibili.

Una possibile soluzione a questo problema consiste nel dividere l'algoritmo dei piani di taglio in due fasi, la Fase 0 e la Fase 1. Durante la Fase 0, al posto di  $(P_{\mathcal{B}})$  si risolve il problema di PL

$$\min_x \{ \|b - Ax\|_{\infty} : x \in X_{\mathcal{B}} \}$$

che possiede sicuramente una soluzione ammissibile, e si usa la soluzione ottima  $y^*$  del suo duale per generare nuove soluzioni  $x \in X$  fino a quando  $Ax^* = b$ ; a quel punto la Fase 0 termina ed inizia la Fase 1 in cui si applica il normale algoritmo dei piani di taglio. Questo approccio si dimostra scarsamente efficiente in quanto la Fase 0 può richiedere molte iterazioni e, non tenendo in alcun conto il costo delle soluzioni generate, normalmente produce un insieme  $\mathcal{B}$  di partenza per la Fase 1 di scarsa qualità. Una Fase 0 alternativa prevede di risolvere la coppia duale di problemi di PL

$$\min_x \{ cx + t\|b - Ax\|_{\infty} : x \in X_{\mathcal{B}} \} \quad (9)$$

$$\max_y \{ \phi_{\mathcal{B}}(y) : \|y\|_1 \leq t \} \quad (10)$$

al posto di  $(P_{\mathcal{B}})$  e  $(D_{\mathcal{B}})$ , in cui  $t$  è un parametro opportunamente fissato, fino a quando  $Ax^* = b$ ; questo aiuta a produrre un insieme  $\mathcal{B}$  di migliore qualità, ma può comunque richiedere un numero considerevole di iterazioni.

Il principale problema dell'algoritmo del piano di taglio è la bassa velocità di convergenza; in molti casi pratici, l'algoritmo esegue una lunga sequenza di passi in cui il valore della soluzione ottima del Master Problem cambia pochissimo (effetto "tailing off"), sia nella Fase 0 che nella Fase 1. Questo comportamento è fondamentalmente attribuibile a quella che viene denominata *instabilità* dell'algoritmo, e che consiste nel fatto che la sequenza delle soluzioni ottime  $y^*$  del Master Problem duale non mostra alcuna proprietà di località; detto in termini più pittoreschi,  $y^*$  "varia selvaggiamente" nel corso dell'algoritmo.

La ragione di questo comportamento risiede semplicemente nel fatto che nell'algoritmo non esiste nessun meccanismo che induca proprietà di località nella sequenza delle  $y^*$ . Il punto fondamentale è che si utilizza, per determinare il nuovo punto  $y^*$ , il modello  $\phi_{\mathcal{B}}$  della funzione Lagrangiana reale  $\phi$  senza avere alcun controllo sull'"affidabilità" del modello in  $y^*$ . Ciò porta a valutare  $\phi$  in punti molto distanti tra loro in cui il valore della funzione è molto diverso da quello previsto dal modello, e quindi ad una scarsa efficienza complessiva dell'approccio.

Un caso rivelatore è quello in cui si supponga di utilizzare, come primo punto in cui valutare  $\phi$ , un'approssimazione di buona qualità della soluzione ottima  $\tilde{y}$  di  $(D)$ , o al limite la stessa  $\tilde{y}$ ; la disponibilità di informazione di questo tipo non è infrequente nelle applicazioni [BFLN02]. Si dispone allora fin dalla prima iterazione di una valutazione molto buona di  $v(D)$ , al limite persino del valore ottimo; ciononostante, il numero di iterazioni necessarie all'algoritmo per convergere non è sostanzialmente diverso da quello richiesto qualora si utilizzi un

punto iniziale arbitrariamente lontano da  $\tilde{y}$  [BA02]. Questo perchè le successive iterate duali sono localizzate in punti dello spazio duale arbitrariamente lontani da  $\tilde{y}$ , in cui il valore di  $\phi$  è molto distante da  $v(D)$ .

L'algoritmo dei piani di taglio non è quindi in grado di sfruttare il fatto di essere giunto in prossimità dell'ottimo per fare in modo che le successive iterate siano anch'esse vicine (possibilmente più vicine) all'ottimo. È interessante notare che questa incapacità riguarda l'informazione duale: se l'algoritmo avesse a disposizione, in  $\mathcal{B}$ , informazione primale che identifichi correttamente l'ottimo  $\tilde{y}$  di  $(D)$  allora terminerebbe in una sola iterazione. Questo fatto illustra chiaramente un'importante distinzione, che spesso non viene appropriatamente sottolineata, che deve essere fatta nella descrizione del processo di convergenza di un algoritmo risolutivo per il duale Lagrangiano  $(D)$ : ottenere una buona approssimazione di  $v(D)$ , ossia un vettore di moltiplicatori  $\tilde{y}$  tale che  $\phi(\tilde{y}) \approx v(D)$ , è solamente *una* delle condizioni necessarie per risolvere  $(D)$ , l'altra essendo la disponibilità di informazione primale  $(x^*)$  che *dimostr*i la bontà dell'approssimazione ottenuta. Il motivo per cui questa distinzione non viene sottolineata è che nelle applicazioni spesso ci si concentra solamente sull'ottenere una valutazione inferiore di  $v(P)$ , senza cercare di stimare quanto questa valutazione sia distante da  $v(D)$ ; ciò può risultare poco lungimirante, in quanto l'informazione necessaria per valutare una soluzione duale è informazione primale che, come discuteremo nel seguito, può risultare molto utile in un algoritmo risolutivo per il problema originale  $(P)$ .

Può essere interessante rilevare che l'instabilità dell'algoritmo dei piani di taglio, oltre a rallentarne la convergenza, può impattare negativamente anche sul costo computazionale in pratica della soluzione sia del rilassamento Lagrangiano  $(P_y)$  che del Master Problem  $(D_B)$ . Infatti, in entrambe i casi si risolvono, all'interno dell'algoritmo, sequenze di problemi di ottimizzazione correlati in cui cambiano soltanto alcuni dati (i costi per il rilassamento Lagrangiano, i vincoli per il Master Problem duale); è quindi chiaro che le tecniche di riottimizzazione possono giocare un ruolo decisivo per diminuire il costo computazionale complessivo richiesto per risolvere tutta la sequenza di problemi [FG99]. Normalmente, le tecniche di riottimizzazione sono tanto più efficienti quanto meno differiscono le soluzioni ottime di due problemi successivi nella sequenza; il fatto che  $y^*$  “vari selvaggiamente” implica che le soluzioni ottime del rilassamento Lagrangiano e del Master Problem duale possono essere anche molto diverse da un'iterazione alla successiva, e quindi che le tecniche di riottimizzazione siano presumibilmente meno efficienti.

Un ulteriore problema dell'algoritmo dei piani di taglio consiste poi nel fatto che, almeno in teoria, per garantire la convergenza dell'algoritmo occorre mantenere in  $\mathcal{B}$  *tutte* le soluzioni determinate; data la lenta convergenza dell'algoritmo, ciò implica che il Master Problem possa giungere ad avere grandi dimensioni, e quindi risulti costoso da risolvere. Questo problema è, nella pratica, secondario per due ragioni: la prima è che eliminare da  $\mathcal{B}$  elementi  $x$  che hanno avuto moltiplicatore  $\theta_x$  nullo per un congruo numero di iterazioni consecutive di solito non compromette la convergenza, anche se in teoria ciò potrebbe accadere, la seconda è che *acquire* la giusta informazione è ciò che fa funzionare l'algoritmo, ed infatti una rimozione eccessivamente aggressiva ha di solito un impatto negativo sulla velocità di convergenza del metodo. È comunque vero che la gestione dell'insieme  $\mathcal{B}$  è potenzialmente critica nell'algoritmo dei piani di taglio,

nel senso che non esistono regole precise e completamente affidabili per determinare la giusta dimensione di  $\mathcal{B}$  e le politiche di eliminazione dell'informazione obsoleta.

### 3.2 Algoritmi dei piani di taglio stabilizzati

L'instabilità dell'algoritmo dei piani di taglio ha ispirato lo sviluppo di un buon numero di varianti che cercano, in diversi modi, di porre rimedio al problema. L'idea alla base di molti di questi approcci è quella di modificare il Master Problem duale introducendo uno "stabilizzatore" che aumenti la località della sequenza delle soluzioni  $y^*$ . Ad esempio, in [Fr02] è analizzata un'ampia classe di approcci nei quali, al posto di  $(D_{\mathcal{B}})$ , viene risolto il *Master Problem Duale stabilizzato*

$$(D_{\mathcal{B},\bar{y},\mathcal{D}}) \max\{ \phi_{\mathcal{B}}(y) - \mathcal{D}(y - \bar{y}) : y \in Y_{\mathcal{B}} \} \quad (11)$$

dove  $\bar{y}$  è il *punto corrente* e  $\mathcal{D} : \mathbb{R}^m \rightarrow \mathbb{R} \cup \{+\infty\}$  è il *termine di stabilizzazione* che assicura che la soluzione  $y^*$  di (11) non sia "troppo distante" da  $\bar{y}$ . Risolvere (11) è equivalente a risolvere il suo *duale di Fenchel*

$$(P_{\mathcal{B},\bar{y},\mathcal{D}}) \max_x \begin{cases} cx + \bar{y}(b - Ax) + \mathcal{D}^*(Ax - b) \\ x \in \text{conv}(\mathcal{B}^1) + \text{cone}(\mathcal{B}^0) \end{cases} \quad (12)$$

in cui

$$\mathcal{D}^*(w) = \sup_y \{yw - \mathcal{D}(y)\}$$

è la *coniugata di Fenchel* di  $\mathcal{D}$  [HL93]; usualmente, una soluzione ottima  $x^*$  di (12) è disponibile come "sottoprodotto" del calcolo della soluzione ottima  $y^*$  di (11) (o viceversa). È possibile mostrare [Fr02] che se  $\mathcal{D}$  è una funzione convessa non negativa tale che  $\mathcal{D}(0) = 0$  e le sue curve di livello corrispondenti a qualsiasi valore  $\delta > 0$  sono compatte e di dimensione piena, allora  $\mathcal{D}^*$  ha le stesse caratteristiche: ad esempio, si ha

- la coniugata di  $\mathcal{D} = \frac{1}{2t} \|\cdot\|_2^2$  è  $\mathcal{D}^* = \frac{1}{2}t \|\cdot\|_2$ ;
- la coniugata di  $\mathcal{D} = I_{1,t}$  è  $\mathcal{D}^* = t \|\cdot\|_{\infty}$ ;
- la coniugata di  $\mathcal{D} = \frac{1}{t} \|\cdot\|_1$  è  $\mathcal{D}^* = I_{\infty,(1/t)}$ .

dove  $I_{p,t}$  è la funzione caratteristica della sfera di centro zero e raggio  $t$  nella norma  $p$ , ossia  $I_{p,t}(w) = 0$  se  $\|w\|_p \leq t$  e  $I_{p,t}(w) = \infty$  altrimenti (si noti che ponendo  $\bar{y} = 0$  e  $\mathcal{D} = I_{1,t}$  in (11) e (12) si ottengono (10) e (9)). Quindi, stabilizzare il Master Problem duale corrisponde a risolvere un *Lagrangiano Aumentato generalizzato* del Master Problem primale  $(P_{\mathcal{B}})$  in cui il termine "al primo ordine" corrisponde al punto corrente  $\bar{y}$  e quello "al secondo ordine" corrisponde al termine di stabilizzazione  $\mathcal{D}$ . Utilizzando  $(D_{\mathcal{B},\bar{y},\mathcal{D}})$  e  $(P_{\mathcal{B},\bar{y},\mathcal{D}})$  si possono realizzare approcci per la soluzione di  $(D)$  e  $(\tilde{P})$  che modificano in modo opportuno, ad ogni iterazione, il punto corrente  $\bar{y}$ , il termine di stabilizzazione  $\mathcal{D}$  e l'insieme delle soluzioni  $\mathcal{B}$ . Tali approcci hanno diverse proprietà a seconda di come, tra le molte opzioni possibili, sono effettuate alcune delle scelte fondamentali: ad esempio, se  $\mathcal{D}$  è strettamente convessa (o, equivalentemente,  $\mathcal{D}^*$  è differenziabile) è possibile garantire la convergenza degli algoritmi anche se ad ogni passo viene eliminato un insieme arbitrario di elementi da  $\mathcal{B}$ , purchè la soluzione ottima  $x^*$  di  $(P_{\mathcal{B},\bar{y},\mathcal{D}})$  sia inserita in  $\mathcal{B}$  per fungere da loro

“rappresentante”, il che permette di mantenere la cardinalità di  $\mathcal{B}$  limitata a qualsiasi valore ( $\geq 2$ ) fissato. Approcci di questa classe, o molto simili, sono stati utilizzati più volte con successo sia per risolvere problemi di ottimizzazione convessa strutturata che nell’ambito degli algoritmi di generazione di colonne, utilizzando termini di stabilizzazione quadratici [FG99, CGN95], esponenziali [GK95], lineari-quadratici [PZ94], lineari a tratti [BA02] e di altro tipo ancora.

Molte altre varianti di metodi del piano di taglio stabilizzati, basate su “stabilizzatori” differenti, sono state proposte in letteratura: tali sono ad esempio i metodi *proximal level* [HL93, Algorithm XV.2.3.1], i metodi di  $\varepsilon$ -discesa, [HL93, Algorithm XIV.3.4.2], il metodo della traiettoria prossimale [Fu98], il metodo delle funzioni barriera di Bregman [Ki99] ed altri. Altri algoritmi basati su concetti simili sono quelli in cui il punto  $y^*$  viene scelto calcolando un opportuno “centro” dell’insieme di localizzazione

$$\{ (v, y) : \phi_{\mathcal{B}}(y) \geq v \geq \bar{\phi} \},$$

dove  $\bar{\phi}$  è la miglior valutazione inferiore nota di  $v(D)$ ; risultano ad esempio efficienti i metodi basati sul *centro analitico*, che può essere (approssimativamente) calcolato in modo efficiente utilizzando tecniche sviluppate per gli algoritmi del punto interno [DGV98], e che garantisce buone proprietà di convergenza [Ne95].

In tutti i casi citati in precedenza, viene ad ogni passo risolto (eventualmente in modo approssimato) un Master Problem duale che coinvolge il modello cutting plane  $\phi_{\mathcal{B}}$ , e quindi un Master Problem primale nel quale appare una combinazione lineare delle soluzioni  $x \in \mathcal{B}$ . Tali algoritmi sono in grado di risolvere  $(D)$ , o quantomeno di ottenere una soluzione duale  $\varepsilon$ -ottima per qualsiasi  $\varepsilon$  fissato, in un numero finito di passi; al termine del processo tutti gli algoritmi producono una soluzione ottima  $\tilde{x}$  di  $(\tilde{P})$ , o almeno una sua approssimazione con precisione arbitraria.

Essendo basati su idee molto simili, ed in particolare sull’utilizzo del modello cutting plane  $\phi_{\mathcal{B}}$  di  $\phi$ , quasi tutti gli algoritmi precedentemente menzionati condividono un certo numero di caratteristiche. In particolare, vale la pena di rilevare che gli algoritmi si adattano in modo naturale all’uso di ottimizzazione approssimata al posto di quella esatta, sia nella soluzione del Master Problem che in quella del rilassamento Lagrangiano. In altre parole, è in generale possibile aumentare l’efficienza dell’algoritmo risolvendo il Master Problem solamente in modo approssimato, soprattutto nelle iterazioni iniziali, purchè la precisione sia portata, a terminazione, ad essere comparabile con quella richiesta alla soluzione di  $(D)$ . Questo è altrettanto vero per la soluzione del rilassamento Lagrangiano, che potrebbe persino essere effettuata con algoritmi diversi (euristiche di diversa qualità ed efficienza) nel corso dell’esecuzione dell’algoritmo, purchè la precisione sia incrementata in modo opportuno durante il processo. In questo caso occorre però rilevare che solo soluzioni ottime (o con errore valutabile) di  $(P_y)$  permettono di determinare valutazioni inferiori di  $v(P)$ . Inoltre, a seconda del costo relativo della soluzione del Master Problem rispetto a quella del rilassamento Lagrangiano, può risultare conveniente non solo risolvere all’ottimalità  $(P_y)$ , ma anche inserire in  $\mathcal{B}$  diverse soluzioni ottime alternative, ove disponibili, e persino soluzioni subottime (possibilmente di buona qualità). Questa pratica è ad esempio ampiamente utilizzata nell’ambito della generazione di colonne, dove è ben noto come sia critico per l’efficienza degli algoritmi inserire

in  $\mathcal{B}$  ad ogni iterazione non solo una colonna di costo ridotto minimo, ma un insieme di cardinalità consistente di colonne a costo ridotto negativo. Ciò è meno comune nell'ambito della soluzione dei duali Lagrangiani, forse perchè i metodi derivati dall'algoritmo del piano di taglio sono meno utilizzati, nonostante il fatto che molti degli algoritmi utilizzati per risolvere i rilassamenti Lagrangiani (metodi primali, programmazione dinamica, metodi enumerativi) producano naturalmente molte soluzioni primali che potrebbero essere sfruttate a questo scopo.

Un'altra caratteristica comune a molti degli approcci derivati dall'algoritmo dei piani di taglio è quella di trattare in modo naturale il caso in cui il dominio effettivo di  $\phi$  non sia tutto  $\mathbb{R}^m$ , ossia  $X$  non sia compatto; come vedremo, questo non è il caso per un'altra classe di algoritmi molto noti, i metodi del subgradiente.

### 3.3 Algoritmi del subgradiente

I metodi del subgradiente sono una classe di approcci per l'ottimizzazione di funzioni nondifferenziabili caratterizzati da una notevole semplicità di implementazione e dal fatto di non richiedere la soluzione di problemi di ottimizzazione per la determinazione del nuovo punto in cui valutare la funzione. Dato il punto corrente  $\bar{y}$ , una direzione  $d$  viene calcolata utilizzando il subgradiente  $g = b - A\bar{x}$ , dove  $\bar{x}$  è una soluzione ottima di  $(P_{\bar{y}})$ , ed eventualmente altra informazione proveniente dalle precedenti iterazioni; il successivo punto corrente, in cui  $\phi$  viene valutata, è semplicemente ottenuto come  $\bar{y} + td$ , dove il passo  $t$  è calcolato mediante opportune regole. Nelle versioni iniziali del metodo la direzione  $d$  era presa uguale al subgradiente  $g$ ; le versioni attuali utilizzano invece la formula generale  $d = g + \theta d^-$ , ove  $d^-$  è la direzione all'iterazione precedente ( $d^- = 0$  alla prima iterazione) e  $\theta \geq 0$  è un moltiplicatore scelto in modo opportuno. Esistono molte varianti dell'algoritmo che si differenziano proprio per la scelta di  $\theta$ : ad esempio, la *regola di Crowder* prevede che  $\theta$  sia semplicemente un valore fissato ( $< 1$ ), mentre in [CFM75] è proposta la regola

$$\theta = \begin{cases} \|g\|/\|d^-\|, & \text{se } gd^- < 0, \\ 0, & \text{altrimenti.} \end{cases}$$

basata sia sull'esperienza computazionale che su considerazioni geometriche. Per quanto riguarda la scelta del passo, la formula più frequentemente utilizzata è  $t = \lambda(\bar{\phi} - \phi(\bar{y}))/gd$ , dove  $\bar{\phi}$  (il "bersaglio") è una valutazione dell'ottimo  $v(D)$  e  $\lambda$  è un fattore di scala che varia nel corso dell'algoritmo. Gli algoritmi del subgradiente possono essere utilizzati nel caso in cui il dominio effettivo di  $\phi$  non sia tutto lo spazio  $\mathbb{R}^m$  utilizzando come direzione la proiezione  $\bar{d}$  di  $d$  sul cono dei vincoli attivi in  $\bar{y}$ ; in questo caso il denominatore della formula di selezione del passo diviene  $g\bar{d}$ . La direzione proiettata  $\bar{d}$  può essere computata efficientemente se i vincoli sono "semplici", quali i vincoli di non negatività corrispondenti al rilassamento di vincoli di disequaglianza in  $(P)$ ; in presenza di vincoli più generali la necessità di effettuare una proiezione rende meno attraente l'algoritmo, che ha tra i suoi pregi proprio quello dell'estrema semplicità.

La semplicità degli algoritmi del subgradiente è in qualche modo controbilanciata dalla necessità di determinare buoni valori per un certo numero di parametri fondamentali, tra cui principalmente quelli per determinare il "bersaglio"  $\bar{\phi}$  e quelli per modificare il parametro di scala  $\lambda$ , e quindi il passo  $t$ , nel



corso dell'algoritmo. Molte diverse formule sono state proposte per il calcolo di  $\bar{\phi}$  e  $\lambda$  tenendo in conto la storia passata dell'algoritmo, ad esempio utilizzando informazione come il numero di iterazioni consecutive nelle quali il valore di  $\phi(\bar{y})$  non è migliorato; tutte queste implementazioni dipendono comunque da un certo numero di parametri che devono essere determinati sperimentalmente. In molti casi la regolazione di questi parametri si rivela un compito non semplice: l'algoritmo non è in generale robusto e può convergere ad un punto arbitrariamente distante dall'ottimo, o persino divergere, se i parametri non sono scelti in modo opportuno, e i parametri che risultano corretti per un'istanza possono risultare altamente inappropriati anche per istanze molto simili [CFG01].

In effetti, le dimostrazioni di convergenza per il metodo del subgradiente [CL93] richiedono normalmente che la successione dei passi  $t$  utilizzati nel corso dell'algoritmo converga a zero "abbastanza lentamente" (ossia la serie sia divergente), in modo da permettere all'algoritmo di raggiungere regioni dello spazio arbitrariamente lontane dal punto di partenza; in pratica, regole di questo tipo risultano in una convergenza del metodo molto lenta, per cui vengono preferite nelle implementazioni regole di decrescita esponenziale di  $t$ , che non garantiscono l'ottimalità della soluzione determinata. In ogni caso, le dimostrazioni classiche di convergenza dell'algoritmo non sono costruttive, ossia non esibiscono nessun "certificato" dell'ottimalità della soluzione determinata, al contrario di quanto avviene per gli algoritmi derivati da quello del piano di taglio.

Per contro, una volta individuati parametri opportuni gli algoritmi del subgradiente possono fornire valutazioni abbastanza accurate di  $v(D)$  in un numero relativamente basso di iterazioni, e soprattutto con un costo computazionale – se la soluzione di  $(P_y)$  è molto efficiente – contenuto. Per questo, gli algoritmi del subgradiente sono stati principalmente utilizzati per determinare velocemente valutazioni inferiori di  $v(P)$ , senza produrre alcun tipo di informazione primale (se non le singole soluzioni  $x \in X$  dei rilassamenti Lagrangiani) durante il processo.

Recentemente è stato però mostrato [LPS99, BA00] come metodi di tipo subgradiente possano essere utilizzati per produrre soluzioni (approssimate) di  $(\tilde{P})$ . Questo in realtà non è sorprendente perché la direzione  $d$  utilizzata nei metodi del subgradiente è di fatto il subgradiente corrispondente ad una soluzione  $\hat{x} \in \text{conv}(X)$  ottenuta da quelle generate nel corso dell'algoritmo, eventualmente scalato di un fattore  $\alpha \geq 0$  (che viene poi "assorbito" dal passo  $t$ ). Infatti è  $d = g = (b - A\hat{x})$  alla prima iterazione, e nelle iterazioni successive si ha

$$d = g + \theta d^- = b - A\bar{x} + \theta\alpha(b - A\hat{x}) ,$$

e quindi

$$d = (1 + \theta\alpha) \left( \frac{1}{1 + \theta\alpha}(b - A\bar{x}) + \frac{\theta\alpha}{1 + \theta\alpha}(b - A\hat{x}) \right) .$$

Siccome  $1/(1 + \theta\alpha) + \theta\alpha/(1 + \theta\alpha) = 1$ , il punto  $1/(1 + \theta\alpha)\bar{x} + \theta\alpha/(1 + \theta\alpha)\hat{x}$  appartiene a  $\text{conv}(X)$ . Quindi, anche negli algoritmi del subgradiente di fatto si computano combinazioni convesse di punti di  $X$ ; operando opportunamente è possibile generare una sequenza di punti in  $\text{conv}(X)$  che converge ad un ottimo  $\tilde{x}$  di  $(\tilde{P})$  (ciò peraltro permette di implementare regole di trminazione effettive per l'algoritmo).

Ad esempio, nell'algoritmo di [BA00] la direzione utilizzata ad ogni passo ha la forma  $\theta\bar{x} + (1 - \theta)\hat{x}$ , dove  $\theta \in [0, 1]$  è ottenuto risolvendo un problema

quadratico in una singola variabile, e  $\hat{x}$  converge a  $\tilde{x}$ . In effetti, questo algoritmo, almeno nelle implementazioni più comunemente utilizzate, è di fatto un metodo del piano di taglio stabilizzato (con  $\mathcal{D} = \|\cdot\|_2^2$ ) nel quale ad ogni passo  $\mathcal{B} = \{\hat{x}, \bar{x}\}$  e  $\hat{x}$  è una soluzione (approssimata) di  $(P_{\mathcal{B}, \bar{y}, \mathcal{D}})$ .

Quali algoritmi risultino più efficienti, tra i metodi del subgradiente e gli algoritmi di tipo piano di taglio, nelle diverse applicazioni è ancora soggetto di ricerca. In prima approssimazione, ove il rilassamento Lagrangiano  $(P_y)$  sia molto costoso da risolvere i metodi di tipo piano di taglio, che utilizzano in modo più efficace l'informazione ottenuta durante il processo di massimizzazione, sono preferibili a quelli del subgradiente [BFLN02]. Quando invece il rilassamento Lagrangiano sia poco costoso i metodi del subgradiente, che non necessitano della soluzione di un Master Problem, possono risultare competitivi. In sostanza, l'efficienza relativa dei due tipi di approcci dipende dall'esatto rapporto tra il costo di risolvere  $(P_y)$  ed il costo di risolvere il Master Problem, e quindi anche dall'efficienza con cui è implementato il metodo del piano di taglio, dallo specifico “stabilizzatore” utilizzato, dall'efficienza dello specifico solutore utilizzato per risolvere il Master Problem e così via; un metodo del piano di taglio appropriatamente implementato può risultare competitivo, a seconda dell'accuratezza desiderata, anche in casi in cui il rilassamento Lagrangiano è poco costoso [CFG01]. Inoltre, i metodi di tipo subgradiente sono meno adatti al caso in cui il dominio di  $\phi$  non sia l'intero spazio  $\mathbb{R}^m$ , poichè calcolano una direzione ammissibile mediante proiezione sul cono dei vincoli attivi, e quindi in modo molto più “miope” rispetto agli algoritmi di tipo piano di taglio che tengono esplicitamente in conto dei vincoli nel calcolare il successivo punto tentativo  $y^*$ ; questo rende i metodi del subgradiente più soggetti a fenomeni di tipo “zig-zagging”, anche nel caso di vincoli semplici quali quelli di non-negatività corrispondenti al rilassamento di vincoli di disequaglianza in  $(P)$ .

Comunque, la distinzione tra le due classi di metodi appare ad un'attenta analisi essere più di tipo quantitativo che qualitativo: i metodi del subgradiente sono sostanzialmente analoghi a metodi del tipo piano di taglio in cui l'insieme  $\mathcal{B}$  viene mantenuto di dimensioni estremamente limitate. Questo è possibile anche in molti (anche se non in tutti gli) algoritmi di tipo piano di taglio, con il vantaggio di disporre di un più ampio ventaglio di scelte per quanto riguarda la dimensione di  $\mathcal{B}$  e lo svantaggio di dover in ogni caso risolvere un problema di ottimizzazione, per quanto di dimensioni limitate, ad ogni passo. Le ragioni per preferire una classe di approcci rispetto all'altra appaiono quindi più legate alla effettiva disponibilità di codici efficienti in grado di essere facilmente adattati alle diverse necessità dei vari ambiti applicativi – più alta nel caso dei metodi del subgradiente, semplici da implementare e che non richiedono la soluzione di un Master Problem – piuttosto che a fondamentali differenze metodologiche degli approcci stessi.

Quale che sia la classe di approcci utilizzati, è possibile ottenere, come sottoprodotto del processo di soluzione (approssimata) di  $(D)$ , una soluzione (approssimata)  $\tilde{x}$  di  $(\tilde{P})$ . Nel prossimo paragrafo discuteremo come questa informazione – o i moltiplicatori convessi  $\theta_x$  che la realizzano a partire dalle soluzioni  $x$  del rilassamento Lagrangiano – possa essere utilizzata all'interno di approcci esatti o euristici per il problema originale  $(P)$ .

## 4 Uso dell'informazione primale

Gli algoritmi per la soluzione del duale Lagrangiano discussi nei paragrafi precedenti costruiscono un insieme  $\mathcal{B} \subset X$  di soluzioni di  $(P_y)$  (quindi intere ma che violano i vincoli  $Ax = b$ ) e, ad ogni iterazione, associano a ciascuna soluzione  $x$  un moltiplicatore  $\theta_x$  per costruire una soluzione  $x^* \in \text{conv}(X)$ . A seconda del particolare algoritmo utilizzato  $x^*$  sarà ammissibile per  $(\tilde{P})$ , ossia rispetterà i vincoli  $Ax = b$ , oppure no; in generale però le soluzioni  $x^*$  diventano “rapidamente quasi ammissibili”, nel senso che la violazione dei vincoli  $Ax = b$  si riduce rapidamente rispetto a quella tipica delle soluzioni  $\bar{x}$  dei rilassamenti Lagrangiani, per cui per semplicità assumeremo che  $x^*$  sia ammissibile. Inoltre, in molti approcci  $\mathcal{B}$  non contiene solamente elementi di  $X$  ma anche elementi di  $\text{conv}(X)$  ottenuti per combinazione convessa da soluzioni che possono nel frattempo essere state scartate; limitandoci in questa sede a considerazioni molto generali eviteremo per quanto possibile di differenziare tra i due tipi di soluzioni. Infine, lo specifico tipo di algoritmo utilizzato influisce sulla cardinalità di  $\mathcal{B}$  (molto ristretta per gli algoritmi del subgradiente) e sul formato dei suoi elementi (aggregato o disaggregato qualora  $X$  sia decomponibile), ma anche su questi dettagli eviteremo per quanto possibile di soffermarci.

### 4.1 Euristiche Lagrangiane

Le Euristiche Lagrangiane [BS93, BFLN02] sfruttano il processo di soluzione del duale Lagrangiano per produrre soluzioni ammissibili di  $(P)$ . In particolare, nella maggior parte dei casi si utilizza la soluzione  $\bar{x}$  del rilassamento Lagrangiano all'iterazione corrente, intera ma che viola i vincoli  $Ax = b$ , e la si rende ammissibile mediante una procedura euristica, spesso di tipo “greedy”; alla soluzione ammissibile così ottenuta possono poi essere applicate euristiche di raffinamento, tipicamente di ricerca locale, per migliorarne la qualità. Il processo iterativo per la soluzione di  $(D)$  funge quindi da “multistart” per normali euristiche, che possono eventualmente utilizzare anche i costi Lagrangiani per guidare la costruzione della soluzione ammissibile. Le procedure euristiche possono essere invocate ad ogni iterazione, oppure ad intervalli prestabiliti, oppure solamente quando viene prodotta una soluzione  $\bar{x}$  con determinate caratteristiche. Le procedure euristiche possono essere invocate in modo uniforme durante il processo iterativo oppure essere invocate più spesso in determinate fasi, tipicamente verso il termine in quanto i moltiplicatori Lagrangiani  $y$  sono di “migliore qualità”; al limite è possibile invocare le procedure euristiche solamente in corrispondenza al miglior vettore di moltiplicatori Lagrangiani trovato, per quanto ciò non garantisca di determinare la migliore tra le soluzioni ottenibili. Le soluzioni ammissibili prodotte dall'euristica Lagrangiana possono poi essere usate, ad esempio, come popolazione di partenza per algoritmi di tipo genetico.

In questo contesto può risultare molto utile anche l'altra informazione prodotta dal processo risolutivo, ossia la soluzione  $x^* \in \text{conv}(X)$  e/o i moltiplicatori  $\theta_x$  ad essa associati. Senza entrare nei dettagli, che richiederebbero lo studio di casi specifici, possiamo fare le seguenti considerazioni generali:

- Essendo  $x^*$  una soluzione continua (quasi) ammissibile si possono applicare le tecniche di arrotondamento sviluppate, ad esempio, per le soluzioni del rilassamento continuo di  $(P)$ ; se i vincoli  $Ex = b$  non hanno la pro-

prietà di integralità  $x^*$  può essere una soluzione migliore di quella prodotta dal rilassamento continuo, e quindi le tecniche di arrotondamento potrebbero rivelarsi ancora più efficaci in questo contesto. In particolare, gli elementi di  $x^*$  corrispondenti a variabili binarie sono valori nell'intervallo  $[0, 1]$  che possono essere interpretati come “probabilità che la variabile sia al valore 1 nella soluzione ottima”, suggerendo euristiche randomizzate o che comunque assegnino priorità alle variabili utilizzando questi valori di probabilità. È evidente che l'interpretazione “probabilistica” di  $x^*$  è, dal punto di vista matematico, assolutamente arbitraria; ciononostante è stata usata con successo per costruire algoritmi approssimati per molti problemi utilizzando la soluzione del rilassamento continuo [Va01], e si è dimostrata efficace, in alcune applicazioni, anche in questo contesto [BFLN02].

- Indipendentemente dalla struttura delle soluzioni  $x$ , i moltiplicatori  $\theta_x$  hanno essi stessi la forma di una distribuzione di probabilità sugli elementi di  $\mathcal{B}$ , il che può suggerire approcci in le soluzioni  $x \in \mathcal{B}$  vengono combinate per costruire una soluzione ammissibile sfruttando queste indicazioni. Nel caso in cui  $X$  sia decomponibile e venga utilizzato un algoritmo di tipo disaggregato si hanno “probabilità” diverse per componenti  $x^h$  diverse provenienti dalla stessa soluzione del rilassamento Lagrangiano, e quindi si ottiene naturalmente un effetto di “ibridazione” in cui la soluzione complessiva viene costruita sfruttando componenti provenienti da soluzioni diverse del rilassamento Lagrangiano. Nel caso del Flusso Multicommodity indivisibile, ad esempio, ad ogni passo ciascun insieme  $\mathcal{B}^h$  contiene cammini tra  $o_h$  e  $d_h$ , ed i moltiplicatori  $\theta$  possono essere (arbitrariamente) interpretati come “probabilità che il cammino faccia parte di una soluzione ottima di  $(P)$ ”.
- Se il rilassamento Lagrangiano si decompone in problemi di tipo diverso, le varie componenti di  $x^*$  possono essere usate in modi diversi. Un caso interessante è quello in cui solo in alcune delle componenti le variabili siano vincolate ad assumere valori interi: per le componenti corrispondenti a problemi continui  $x^*$  fornisce una soluzione ammissibile rispetto ai vincoli non rilassati, che può quindi, ad esempio, essere utilizzata nell'euristica Lagrangiana senza necessità di alcuna modifica [BFLN02].
- Sfruttare la soluzione continua  $x^*$  non esclude di utilizzare anche l'informazione tradizionalmente utilizzata nelle euristiche Lagrangiane, ossia la soluzione  $\bar{x}$  del rilassamento Lagrangiano ed i costi Lagrangiani  $c - yA$ . Questo viene fatto ad esempio in [BFLN02], dove  $x^*$  ed i costi Lagrangiani vengono utilizzati per costruire una lista di priorità sulla base della quale la soluzione  $\bar{x}$  viene modificata fino a renderla ammissibile; almeno in quella particolare applicazione, l'approccio combinato si è dimostrato essere significativamente più efficace di approcci che utilizzassero solamente  $\bar{x}$  oppure solamente  $x^*$ .

## 4.2 Algoritmi enumerativi

L'utilizzo di tecniche Lagrangiane per calcolare valutazioni inferiori all'interno di approcci enumerativi per problemi di PLI è discretamente diffuso e si è rivelato molto efficace in un buon numero di casi. Qualora però il duale Lagrangiano sia

risolto con tecniche che non producono informazione primale, quali le varianti “classiche” del metodo del subgradiente, questo approccio ha due inconvenienti:

- In assenza di informazione primale risulta più difficoltoso sviluppare regole di separazione (branching) generali ed efficienti per l’algoritmo enumerativo; possono ovviamente essere implementate regole ad-hoc per il problema, sfruttando la soluzione primale  $\bar{x}$  del rilassamento Lagrangiano all’ultima iterazione ed i corrispondenti costi Lagrangiani, ma tali regole risultano più complesse da realizzare e spesso meno efficienti.
- In assenza di informazione primale risulta difficile, se non impossibile, utilizzare gli algoritmi di separazione eventualmente disponibili che producono diseuguaglianze valide per l’involuppo convesso dell’insieme ammissibile di  $(P)$ .

Questi inconvenienti non sono intrinsecamente legati all’uso di tecniche Lagrangiane, ma dipendono dal fatto che gli algoritmi utilizzati per risolvere  $(D)$  non producono informazione primale. Ove l’informazione primale sia disponibile gli inconvenienti scompaiono ed è possibile, in linea di principio, utilizzare le tecniche Lagrangiane congiuntamente alle regole di separazione ed agli algoritmi di separazione sviluppati per gli approcci basati sulla Programmazione Lineare, come ad esempio la separazione basata sulla variabile più frazionaria, gli “pseudocosti ridotti” e così via.

L’affermazione precedente va in qualche modo qualificata. I vincoli generati dagli algoritmi di separazione non possono normalmente essere inclusi nel rilassamento Lagrangiano, e quindi devono essere considerati come ulteriori vincoli “complicanti” e rilassati. Questo significa che ad ogni nodo dell’albero di enumerazione la funzione Lagrangiana  $\phi$  cambia, ed in particolare cambia (aumenta) il numero di variabili. Occorre inoltre che le regole di separazione non alterino la struttura del rilassamento Lagrangiano in modo tale da renderne sostanzialmente più difficile la soluzione. Si consideri ad esempio il problema del Flusso Multicommodity indivisibile, per il quale si intende usare la normale regola di separazione che seleziona una variabile  $x_{ij}^h$  frazionaria e la fissa a 0 o ad 1 nei due sottoproblemi. Mentre fissare a 0 un numero arbitrario di variabili nel rilassamento Lagrangiano non crea alcun problema, in quanto basta cancellare gli archi relativi dal grafo su cui si computa il cammino minimo, fissare ad 1 un numero arbitrario di variabili trasforma il rilassamento Lagrangiano nel problema di determinare un cammino minimo che passa per un insieme fissato di archi, che è  $\mathcal{NP}$ -arduo. È quindi necessario sviluppare regole di separazione “compatibili” col rilassamento Lagrangiano utilizzato; per il problema del Flusso Multicommodity nonseparabile si può ad esempio selezionare un nodo “divergenza” nella soluzione continua  $x^*$ , ossia un nodo  $i$  tale che almeno due archi nella sua stella uscenta abbiano flusso positivo in  $x^*$ , partizionare la stella uscente del nodo in due sottoinsiemi disgiunti che contengano ciascuno almeno un arco con flusso positivo ed imporre nei due sottoproblemi che il flusso su tutti gli archi di uno dei due insiemi sia nullo. Si noti comunque come questo sia necessario non solamente se si utilizza un algoritmo per il duale Lagrangiano che produce informazione primale, ma anche se si utilizza un algoritmo che non la produce; al contrario, la regola di separazione precedentemente illustrata non potrebbe essere implementata in assenza di informazione primale. Si noti inoltre che un’alternativa alla separazione esplicita è quella di introdurre (e rilassare)

vincoli nel problema che compiano la stessa funzione. Ad esempio, per la normale regola di separazione su una variabile frazionaria  $x_i^*$  è possibile introdurre in  $(P)$  alternativamente i due vincoli  $x_i \geq \lceil x_i^* \rceil$  e  $x_i \leq \lfloor x_i^* \rfloor$ , e poi considerarli come ulteriori vincoli “complicanti”; ciò consente di evitare qualsiasi problema con la regola di separazione al costo di aumentare il numero di moltiplicatori Lagrangiani.

Può essere interessante rilevare che gli algoritmi di tipo piano di taglio hanno un ulteriore potenziale vantaggio su quelli di tipo subgradiente quando usati nel contesto di metodi enumerativi, dato che conservano una maggiore quantità dell’informazione generata durante il processo risolutivo. Infatti, gli algoritmi del subgradiente conservano – e comunque utilizzano – ad ogni passo solamente una soluzione aggregata  $\hat{x}$  ottenuta per combinazione convessa da tutte quelle generate durante il processo risolutivo, mentre gli algoritmi di tipo piano di taglio conservano normalmente un numero maggiore di soluzioni, eventualmente aggregate. Quindi, al momento di risolvere il nuovo duale Lagrangiano, corrispondente ad un diverso sottoproblema nell’albero di enumerazione, gli algoritmi di tipo piano di taglio hanno a disposizione una quantità possibilmente rilevante di informazione sulla precedente funzione Lagrangiana  $\phi$ , che può essere tramutata in informazione sulla nuova funzione Lagrangiana. Quindi, gli algoritmi di tipo piano di taglio sono potenzialmente in grado di sfruttare maggiormente l’informazione complessiva generata durante tutta la visita dell’albero di enumerazione; questo può tradursi in una convergenza media più rapida, e quindi in una maggiore efficienza computazionale.

Le tecniche Lagrangiane possono quindi offrire un’alternativa alle più usuali tecniche di Programmazione Lineare per determinare valutazioni inferiori e soluzioni continue all’interno di approcci enumerativi per algoritmi di PLI. È interessante rilevare che l’uso di tecniche Lagrangiane, all’interno di algoritmi enumerativi, è fondamentalmente equivalente all’uso di tecniche di generazione di colonne per risolvere il rilassamento di PL, ossia agli algoritmi di “Branch & Price” o “Branch & Cut & Price”; in effetti, le due tecniche coincidono quando per risolvere il duale Lagrangiano venga utilizzato un algoritmo classico del piano di taglio. Ad esempio, la regola di separazione illustrata per il caso del Flusso Multicommodity indivisibile è stata proposta in [BHV98] per un algoritmo “Branch & Price” basato su una formulazione per cammini del problema; ciò è completamente equivalente ad un algoritmo enumerativo in cui si risolve ad ogni passo il duale Lagrangiano rispetto ai vincoli (2) con l’algoritmo di Figura 1. L’uso di tecniche Lagrangiane all’interno di approcci enumerativi generalizza quindi gli algoritmi di “Branch & Price”, e potrebbe risultare più conveniente in quanto la variante “stabilizzate” degli algoritmi del piano di taglio possono rivelarsi significativamente più efficienti dell’algoritmo non stabilizzato [FG99].

## 5 Conclusioni

Risolvere il duale Lagrangiano di un problema di Programmazione Lineare Intera  $(P)$  è equivalente a risolvere il rilassamento convessificato  $(\tilde{P})$  in cui appare l’involuppo convesso delle soluzioni ammissibili per i vincoli non rilassati; questa equivalenza è effettiva, ossia una soluzione (approssimata) di  $(\tilde{P})$  viene costruita dalla maggior parte degli algoritmi utilizzati per risolvere il duale Lagrangiano.

Le tecniche Lagrangiane offrono quindi un'alternativa alle tecniche di Programmazione Lineare per la costruzione di approcci esatti o euristici a problemi di PLI, e possono sfruttare, in linea di principio, le stesse strategie (regole di separazione, separatori, tecniche di arrotondamento e così via) sviluppate per la PL. In certi casi le tecniche Lagrangiane offrono anzi un'informazione "più ricca" di quella fornita dalle delle normali tecniche di PL: non solo una soluzione continua  $x^*$ , ma un insieme di soluzioni  $\mathcal{B} \subset X$  ed i relativi moltiplicatori convessi  $\theta_x$  che producono  $x^*$ . Per contro le soluzioni  $x^*$  fornite dai metodi Lagrangiani tipicamente non sono di base, e quindi possono, ad esempio, avere una percentuale di variabili frazionarie più alta rispetto a quelle fornite da un normale rilassamento continuo.

Nonostante questo le tecniche Lagrangiane sono relativamente molto meno utilizzate della PL nell'ambito di approcci esatti o euristici per la PLI; questo avviene per almeno tre motivi:

- Le tecniche Lagrangiane richiedono un'attenta analisi del modello e la soluzione di problemi di ottimizzazione; sono quindi più complesse da applicare, meno resistenti a cambiamenti del modello (che possono rendere il rilassamento Lagrangiano "difficile") e richiedono una qualche forma di programmazione. La PL può invece essere utilizzata, attraverso i solutori e gli ambienti di sviluppo, senza nessuno sforzo di programmazione e con grande flessibilità; per questo è preferibile in molte applicazioni ed in diversi ambienti.
- I codici disponibili per la soluzione di duali Lagrangiani non hanno ancora raggiunto l'elevato livello di efficienza, facilità d'uso e standardizzazione ormai consueto nei codici per la PL.
- I duali Lagrangiani possono risultare costosi da risolvere in pratica con l'elevata precisione tipicamente ottenibile per i problemi di PL, sia per la relativa carenza di solutori efficienti che per l'intrinseca difficoltà del problema.

La diffusione delle tecniche Lagrangiane è quindi condizionata sia da caratteristiche intrinseche dell'approccio che da fattori tecnologici, principalmente dalla disponibilità di codici efficienti e di facile uso per la soluzione di duali Lagrangiani. Miglioramenti nella tecnologia disponibile potrebbero permettere di applicare con successo queste tecniche a molti problemi di rilevanza pratica, sia in ambito accademico che industriale, fornendo un ulteriore, utile strumento per la soluzione di problemi "difficili" di ottimizzazione.

## Riferimenti bibliografici

- [BA00] F. Barahona, R. Anbil, *The Volume Algorithm: Producing Primal Solutions with a Subgradient Method*, Math. Prog. 87(3) (2000), pp. 385–400.
- [BS93] C. Barnhart, Y. Sheffi, *Dual-Ascent Methods for Large-Scale Multi-Commodity Flow Problems*, Naval Research Logistics 40 (1993), pp. 305–324.

- [BHV98] C. Barnhart, C. A. Hane, P. Vance, *Using Branch-and-Price-and-Cut to Solve Origin-Destination Integer Multicommodity Flow Problems*, Op. Res. 32(3) (1998), pp. 208–220.
- [BA02] H. Ben Amor, *Stabilisation de l'Algorithme de Génération de Colonnes*, Ph.D. Thesis, GERAD, Université de Montréal, Montreal, QC, Canada, 2002.
- [BFLN02] A. Borghetti, A. Frangioni, F. Lacalandra, C. A. Nucci, *Lagrangian Heuristics Based on Disaggregated Bundle Methods for Hydrothermal Unit Commitment*, IEEE Transactions on Power Systems, to appear (2002).
- [CFM75] P. M. Camerini, L. Fratta, F. Maffioli, *On Improving Relaxation Methods by Modified Gradient Techniques*, Math. Prog. Study 3 (1975), pp. 26–34.
- [CGN95] P. Carraresi, L. Girardi, M. Nonato, *Network Models, Lagrangean Relaxations and Subgradient Bundle Approach in Crew Scheduling Problems*, in Computer Aided Scheduling of Public Transport (J. Pao-xiao ed.), Lecture Notes in Economical and Mathematical Systems, Springer-Verlag, 1995.
- [CL93] R. Correa, C. Lemaréchal, *Convergence of some algorithms for convex minimization*, Math. Prog. 62 (1993), pp. 261–275.
- [CFG01] T. G. CRAINIC, A. FRANGIONI, AND B. GENDRON, *Bundle-based relaxation methods for multicommodity capacitated fixed charge network design problems*, Discrete Appl. Math. 112 (2001), pp. 73–99.
- [DW60] G. B. Dantzig, P. Wolfe, *The Decomposition Principle for Linear Programs*, Op. Res. 8 (1960), pp. 101–111.
- [DGV98] O. du Merle, J.-L. Goffin, J.-P. Vial, *On Improvements to the Analytic Center Cutting Plane Method*, Computational Optimization and Applications 11 (1998), pp. 37–52.
- [Fr02] A. Frangioni, *Generalized Bundle Methods*, SIAM J.-Opt., to appear, 2002.
- [FG99] A. Frangioni and G. Gallo, *A Bundle Type Dual-Ascent Approach to Linear Multicommodity Min Cost Flow Problems*, INFORMS J. on Comp. 11(4) (1999), pp. 370–393.
- [Fu98] A. Fuduli, *Metodi Numerici per la Minimizzazione di Funzioni Convesse NonDifferenziabili*, Ph.D. Thesis, DEIS, Università della Calabria, Calabria, Italy, 1998.
- [GK95] M. D. Grigoriadis and L. G. Kahchian, *An exponential-function reduction method for block-angular convex programs*, Networks 26 (1995), pp. 59–68.
- [GK87] M. Guignard, *Lagrangean Decomposition: a Model Yelding Stronger Lagrangean Bounds bounds*, Math. Prog. 39 (1987), pp. 215–228.



- [HWC74] M. Held, P. Wolfe, H. P. Crowder, *Validation of Subgradient Optimization*, Math. Prog. 6 (1974), pp. 62–88.
- [HL93] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms II—Advanced Theory and Bundle Methods*, Grundlehren Math. Wiss. 306, Springer-Verlag, New York, 1993.
- [JLFP93] K. L. Jones, I. J. Lustig, J. M. Farwolden, W. B. Powell, *Multicommodity Network Flows: The Impact of Formulation on Decomposition*, Math. Prog. 62 (1993), pp. 95–117.
- [Ki99] K. Kiwiel, *A bundle Bregman proximal method for convex nondifferentiable optimization*, Math. Program., 85 (1999), pp. 241–258.
- [Ke60] J. E. Keley, *The Cutting-Plane Method for Solving Convex Programs*, Journal of the SIAM 8 (1960), pp. 703–712.
- [LPS99] T. Larsson, M. Patriksson, A.-B. Strömberg, *Ergodic, Primal Convergence in Dual Subgradient Schemes for Convex Programming*, Math. Prog. 86 (1999), pp. 283–312.
- [Ne95] Y. Nesterov, *Complexity estimates of some cutting plane methods based on the analytic barrier*, Math. Prog. 69 (1995), pp. 149–176.
- [PZ94] M. C. Pinar and S. A. Zenios, *On smoothing exact penalty functions for convex constrained optimization*, SIAM J. Opt. 4 (1994), pp. 486–511.
- [Va01] V. V. Vazirani, *Approximation Algorithms*, Springer-Verlag, New York, 2001