Programación por Restricciones

Gerardo M. Sarria M.

Pontificia Universidad Javeriana

13 de agosto de 2008

Gerardo M. Sarria M.

SIMPLIFICACIÓN, OPTIMIZACIÓN E IMPLICACIÓ

Gerardo M.
Sarria M.

Contenido

Simplificación de Restricciones

Simplificación es el proceso de reemplazar una restricción por una restricción equivalente, la cual tiene una forma más "simple".

La siguiente lista de restricciones aritméticas son equivalentes:

$$
\begin{aligned}
& X \geq 3 \ \wedge \ 2 = Y + X \\
\leftrightarrow\ & 3 \leq X \ \wedge \ X = 2 - Y \\
\leftrightarrow\ & X = 2 - Y \ \wedge \ 3 \leq X \\
\leftrightarrow\ & X = 2 - Y \ \wedge \ 3 \leq 2 - Y \\
\leftrightarrow\ & X = 2 - Y \ \wedge \ Y \leq -1 \\
\leftrightarrow\ & X = 2 - Y \ \wedge \ Y \leq -1 \ \wedge \ Y \leq 2
\end{aligned}
$$

Implicación y Redundancia

Definición:
Una restricción C_1 implica otra C_2, escrito $C_1 \rightarrow C_2$, si las soluciones de C_1 son un subconjunto de las soluciones de C_2. Alternativamente, se dice que la restricción C_2 es redundante con respecto a la restricción C_1, si $C_1 \rightarrow C_2$.

Ejemplo:
La restricción $X \geq 1$ es redundante respecto a $Y \leq X + 2 \ \wedge \ Y \geq 4$.

Una restricción C de la forma $c_1 \wedge \ldots \wedge c_n$, donde $c_1, \ldots, c_n$ son restricciones primitivas, es redundante-libre, si ningún c_i es redundante con respecto a las demás restricciones primitivas.

Extensión

La utilidad de la simplificación de restricciones es más aparente cuando sólo se está interesado en algunas de las variables que están en la restricción. Dichas variables son denominadas variables de interés.

Limitar la consideración de una restricción a algunas de las variables de dicha restricción, significa que se examina las soluciones de la restricción, pero ignorando los valores asignados a las variables que no son de interés.

Definición:
Si θ es una valuación de la forma $\{x_1 \mapsto d_1, \ldots, x_m \mapsto d_m\}$ entonces α es una extensión de θ si es de la forma $\{x_1 \mapsto d_1, \ldots, x_m \mapsto d_m, x_{m+1} \mapsto d_{m+1}, \ldots, x_n \mapsto d_n\}$.

Gerardo M.
Sarria M.

Solución Parcial

Definición:
θ es una solución parcial de C si existe una extensión de θ que es solución de C.

Ejemplo:
$\{X \mapsto 3, Y \mapsto 4\}$ es una extensión de $\{Y \mapsto 4\}$.

Como $\{X \mapsto 3, Y \mapsto 4\}$ es solución de $X \leq Y$, $\{Y \mapsto 4\}$ es una solución parcial de $X \leq Y$.

Gerardo M.
Sarria M.

Proyección

Definición:
La proyección de una restricción C_1 en un conjunto de variables V, donde $V \subseteq vars(C_1)$, es una restricción C_2 asociada sólo con variables de V en las cuales:

- si θ es una solución de C_1, entonces es solución de C_2; y
- si θ es una valuación sobre V que es una solución de C_2, entonces es una solución parcial de C_1.

Ejemplo

La proyección de la restricción

$$X \geq Y \;\wedge\; Y \geq Z \;\wedge\; Z \geq T \;\wedge\; T \geq 0$$

sobre la variable X es la restricción $X \geq 0$.

Esto es porque:

Ejemplo

La proyección de la restricción

$$X \geq Y \;\wedge\; Y \geq Z \;\wedge\; Z \geq T \;\wedge\; T \geq 0$$

sobre la variable X es la restricción $X \geq 0$.

Esto es porque:

- Toda solución θ de $X \geq Y \;\wedge\; Y \geq Z \;\wedge\; Z \geq T \;\wedge\; T \geq 0$ es una solución de $X \geq 0$.

Ejemplo

La proyección de la restricción

$$X \geq Y \;\wedge\; Y \geq Z \;\wedge\; Z \geq T \;\wedge\; T \geq 0$$

sobre la variable X es la restricción $X \geq 0$.

Esto es porque:

- Toda solución θ de $X \geq Y \;\wedge\; Y \geq Z \;\wedge\; Z \geq T \;\wedge\; T \geq 0$ es una solución de $X \geq 0$.
- Toda solución de $X \geq 0$, como $\{X \mapsto n\}$ donde $n \geq 0$, puede ser extendida a una solución de $X \geq Y \;\wedge\; Y \geq Z \;\wedge\; Z \geq T \;\wedge\; T \geq 0$. Por ejemplo, $\{X \mapsto n, Y \mapsto n, Z \mapsto n, T \mapsto n\}$.

Eliminación de Fourier

El algoritmo de Fourier para eliminación de variables puede ser usado para proyectar una conjunción de desigualdades lineales sobre un conjunto de variables de interés.

Eliminación de Fourier

Considere una restricción C, compuesta de desigualdades lineales que se tomarán como *primitivas*(C). Eliminar una variable y de C sigue los siguientes pasos:

Eliminación de Fourier

Considere una restricción C, compuesta de desigualdades lineales que se tomarán como *primitivas*(C). Eliminar una variable y de C sigue los siguientes pasos:

1. C es particionado en tres subconjuntos: C^0, aquellas desigualdades que no contienen a y; C^+, aquellas desigualdades que son equivalentes a una desigualdad de la forma $y \leq t$, donde t no contiene a y; y C^-, aquellas desigualdades que son equivalentes a una desigualdad de la forma $t \leq y$, donde de nuevo t no contiene a y.

Eliminación de Fourier

Considere una restricción C, compuesta de desigualdades lineales que se tomarán como *primitivas*(C). Eliminar una variable y de C sigue los siguientes pasos:

1. C es particionado en tres subconjuntos: C^0, aquellas desigualdades que no contienen a y; C^+, aquellas desigualdades que son equivalentes a una desigualdad de la forma $y \leq t$, donde t no contiene a y; y C^-, aquellas desigualdades que son equivalentes a una desigualdad de la forma $t \leq y$, donde de nuevo t no contiene a y.
2. Para cada par de la forma $t_1 \leq y$ en C^- y $y \leq t_2$ en C^+, se forma una nueva desigualdad $t_1 \leq t_2$.

Eliminación de Fourier

Considere una restricción C, compuesta de desigualdades lineales que se tomarán como *primitivas*(C). Eliminar una variable y de C sigue los siguientes pasos:

1. C es particionado en tres subconjuntos: C^0, aquellas desigualdades que no contienen a y; C^+, aquellas desigualdades que son equivalentes a una desigualdad de la forma $y \leq t$, donde t no contiene a y; y C^-, aquellas desigualdades que son equivalentes a una desigualdad de la forma $t \leq y$, donde de nuevo t no contiene a y.
2. Para cada par de la forma $t_1 \leq y$ en C^- y $y \leq t_2$ en C^+, se forma una nueva desigualdad $t_1 \leq t_2$.
3. Este último conjunto de nuevas desigualdades junto con C^0 es la proyección de la restricción original C sobre las variables originales excepto y.

Gerardo M. Sarria M.

Algoritmo de Fourier

Algoritmo de Eliminación de Fourier

y es una variable;
t, t_1, t_2 son términos aritméticos lineales sin incluir y;
C, C^0, C^+, C^- son conjuntos de desigualdades lineales;
V es un conjunto de variables.

```
fourier_simplify(C, V)
   while vars(C) − V no sea vacío
      choose y ∈ vars(C) − V
      C := fourier_eliminate(C, y) endwhile
   return C
```

Gerardo M. Sarria M.

Algoritmo de Fourier

fourier_eliminate(C, y)
    sea C^0 aquellas desigualdades en C sin incluir y
    sea C^+ aquellas desigualdades en C que pueden ser escritas $t \leq y$
    sea C^- aquellas desigualdades en C que pueden ser escritas $y \leq t$

    for each $t_1 \leq y \in C^+$
        for each $y \leq t_2 \in C^-$
            $C^0 := C^0 \cup \{t_1 \leq t_2\}$
        endfor
    endfor
    return C^0

Simplificadores de Restricciones

Si existe una restricción C_1 en el dominio que contiene el conjunto de variables V (entre otros), es posible que no exista una restricción C_2 en el dominio, la cual sea la proyección de C_1 sobre V.

De allí que los algoritmos de proyección no pueden ser usados para simplificar una restricción en todos los dominios de restricciones.

Por lo anterior, se debe generalizar la noción de simplificación con respecto a un conjunto de variables para permitir la simplificación que contenga variables “locales”.

Gerardo M. Sarria M.

Equivalencia

Definición: Dos restricciones C_1 y C_2 son equivalentes con respecto a un conjunto de variables $\{x_1, \ldots, x_n\}$ si:

- para cada solución $\{x_1 \mapsto d_1, \ldots, x_n \mapsto d_n, \ldots\}$ de C_1, $\{x_1 \mapsto d_1, \ldots, x_n \mapsto d_n\}$ es una solución parcial de C_2; y
- para cada solución $\{x_1 \mapsto d_1, \ldots, x_n \mapsto d_n, \ldots\}$ de C_2, $\{x_1 \mapsto d_1, \ldots, x_n \mapsto d_n\}$ es una solución parcial de C_1

Gerardo M. Sarria M.

Simplificador

Definición: Un simplificador, para un dominio de restricciones D, toma una restricción C_1 en D y un conjunto V de variables de interés y retorna una restricción C_2 en D, tal que C_1 y C_2 son equivalentes con respecto a V.

Gerardo M. Sarria M.

Problemas de Optimización

En el caso en que se desea resolver el problema de la solución, rara vez se desea cualquier solución. Por el contrario, se quiere encontrar la mejor solución. Este es el denominado problema de optimización.

Definición:
Un problema de optimización, escrito (C, f), consiste en una restricción C y una función objetivo f la cual es una expresión sobre las variables de C y que evalúa a un número real.

Solución Óptima

Una valuación θ es preferente a una valuación θ', si el valor de la función objetivo f sobre θ es menor que el valor bajo θ'. En otras palabras, $\theta(f) < \theta'(f)$.

Una solución óptima, θ, de (C, f) es una solución de C tal que no hay otra solución de C que es preferente a θ.

Gerardo M. Sarria M.

Ejemplo

Considere la restricción para construir una casa vista anteriormente:

$$TS \geq 0 \;\wedge\; TA \geq TS + 7 \;\wedge\; TB \geq TA + 4 \;\wedge$$
$$TC \geq TA + 3 \;\wedge\; TD \geq TA + 3 \;\wedge\; TD \geq TC + 2 \;\wedge$$
$$TE \geq TB + 2 \;\wedge\; TE \geq TD + 3 \;\wedge\; TE \geq TC + 3$$

Si se tiene como función objetivo TE, una solución óptima es:

$$\{TS \mapsto 0, TA \mapsto 7, TB \mapsto 11, TC \mapsto 10, TD \mapsto 12, TE \mapsto 15\}$$

Ejemplo

Ahora, suponga que el inspector de construcción estará en el periodo comprendido entre la etapa B y la etapa E. Esto produce que se tenga que minimizar la duración entre dichas etapas, o sea, minimar la función objetivo $TE - TB$.

Una solución óptima a este problema es:

$$\{TS \mapsto 0, TA \mapsto 7, TB \mapsto 13, TC \mapsto 10, TD \mapsto 12, TE \mapsto 15\}$$

Problemas de Optimización

El problema de optimización no necesariamente tiene una sola solución. Por ejemplo, considere la restricción $X + Y \geq 4$ junto con la función objetivo $X + Y$. Cualquier solución de la restricción $X + Y = 4$ es una solución óptima de este problema.

Problemas de Optimización

El problema de optimización no necesariamente tiene una sola solución. Por ejemplo, considere la restricción $X + Y \geq 4$ junto con la función objetivo $X + Y$. Cualquier solución de la restricción $X + Y = 4$ es una solución óptima de este problema.

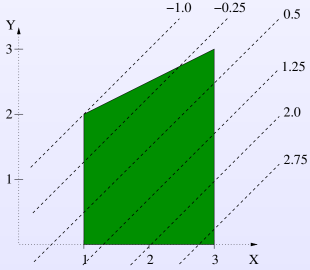
Por otro lado, un problema de optimización puede no tener solución óptima.
Dos posibles razones:

- Que la restricción no tenga solución.
  Ejemplo: $(X \geq 0 \;\wedge\; X \leq -1, X)$
- Que cada solución tiene una solución que es preferente a ella.
  Ejemplo: $(X \leq 0, X)$

Algoritmo Simplex

Este algoritmo da respuesta al problema de optimización para restricciones aritméticas lineales en la cual, la función objetivo es una expresión aritmética lineal.

Ejemplo:
Considere el problema de minimizar $X - Y$ bajo las restricciones $1 \leq X \ \wedge \ X \leq 3 \ \wedge \ 0 \leq Y \ \wedge \ 2Y - X \leq 3$.
La siguiente figura muestra el problema.



Algoritmo Simplex

Definición: Un problema de optimización (C, f) esta en forma simplex si la restricción C tiene la forma $C_E \wedge C_1$ donde C_E es una conjunción de ecuaciones aritméticas lineales y C_1 es $\bigwedge\{x \geq 0 \mid x \in vars(C)\}$ y f es una expresión lineal sobre las variables de C.

Ejemplo:
El siguiente es un problema de optimización en forma simplex:

minimizar $3X + 2Y - Z + 1$ sujeto a

$$\begin{aligned} & X + Y = 3 \\ \wedge \; & -X - 3Y + 2Z + T = 1 \\ \wedge \; & X \geq 0 \wedge Y \geq 0 \wedge Z \geq 0 \wedge T \geq 0 \end{aligned}$$

Algoritmo Simplex

Definición: Un problema de optimización en forma simplex está en forma básica factible resuelta si las ecuaciones son de la forma $x_0 = b + a_1 x_1 + \ldots + a_n x_n$ donde la variable x_0 no ocurre en ninguna otra ecuación o en la función objetivo, y la constante b es no-negativa.

La variable x_0 se dice que es básica y las otras variables son parámetros.

Una solución básica factible es obtenida fijando cada parámetro a 0 y cada variable básica al valor de la constante en lado derecho de su ecuación.

Gerardo M.
Sarria M.

Ejemplo

La siguiente restricción está en forma básica factible resuelta y es equivalente al problema del ejemplo anterior.

minimizar $10 - Y - Z$ sujeto a

$$\begin{aligned} & X = 3 - Y \\ \wedge\ & T = 4 + 2Y - 2Z \\ \wedge\ & X \geq 0 \ \wedge\ Y \geq 0 \ \wedge\ Z \geq 0 \ \wedge\ T \geq 0 \end{aligned}$$

X y T son variables básicas y Y y Z son parámetros.

Ejemplo

La siguiente restricción está en forma básica factible resuelta y es equivalente al problema del ejemplo anterior.

minimizar $10 - Y - Z$ sujeto a

$$
\begin{aligned}
& X = 3 - Y \\
\wedge\ & T = 4 + 2Y - 2Z \\
\wedge\ & X \geq 0 \ \wedge \ Y \geq 0 \ \wedge \ Z \geq 0 \ \wedge \ T \geq 0
\end{aligned}
$$

X y T son variables básicas y Y y Z son parámetros.

La solución básica factible correspondiente a dicha forma es $\{X \mapsto 3, Y \mapsto 0, Z \mapsto 0, T \mapsto 4\}$. El valor de la función objetivo con esta solución es 10.

Gerardo M. Sarria M.

Algoritmo Simplex

El algoritmo simplex encuentra el óptimo con una búsqueda repetida de una forma básica factible resuelta cuya solución básica factible disminuya el valor de la función objetivo.

Cuando no se pueda encontrar una forma básica factible resuelta, el óptimo ha sido hallado.

Algoritmo de Optimización Simplex

C es una conjunción de ecuaciones;
i, I, j, J, n, m son enteros;
a_{ij}, b_i, e, d_i son constantes reales;
f, t son expresiones lineales;
$c_1, \ldots, c_n$ son ecuaciones;
x_i, y_j son variables;

Algoritmo Simplex

simplex_opt(C, f)
    sea C de la forma $c_1 \ \wedge \ \ldots \ \wedge \ c_n$
    for each $i \in \{1, \ldots, n\}$
        sea c_i de la forma $x_i = b_i + \sum_{j=1}^{m} a_{ij} y_j$
    endfor
    sea f de la forma $e + \sum_{j=1}^{m} d_j y_j$
    % Escoja la variable y_J a ser básica
    if for all $j \in \{1, \ldots, m\}, d_j \geq 0$ then
        return $\langle true, C, f \rangle$
    endif
    escoja $J \in \{1, \ldots, m\}$ tal que $d_J < 0$
    % Escoja la variable x_I a ser no-básica
    if for all $i \in \{1, \ldots, n\}, a_{iJ} \geq 0$ then
        return $\langle false, C, f \rangle$
    endif
    escoja $I \in \{1, \ldots, m\}$ tal que
        $-b_I / a_{IJ} = \min\{-b_i / a_{iJ} \mid a_{iJ} < 0$ and $1 \leq i \leq n\}$
    $t :=$ $(x_I - b_I - \sum_{j=1, j \neq J}^{m} a_{Ij} y_j / a_{IJ})$
    $c_I :=$ $(y_J = t)$
    remplace y_J por t en f
    for each $i \in \{1, \ldots, n\}$
        if $i \neq I$ then remplace y_J por t en c_i endif
    endfor
    return simplex_opt$(\bigwedge_{i=1}^{n} c_i, f)$

Gerardo M. Sarria M.

Algoritmo Simplex

Encontrar una solución básica factible es exactamente un problema de satisfacción de restricciones. Luego se puede usar el algoritmo de optimización en el algoritmo de resolución del problema de satisfacción.

Algoritmo de Simplex Solver

C, C' son una conjunción de ecuaciones;
i, j, m son enteros;
$a_{ij}, b_i, e, d_i, a'_i$ son constantes reales;
f, f', f_i son expresiones lineales;
c_i, c'_i son ecuaciones;
x_i son variables;
z_i son nuevas variables;
flag es booleano;

Gerardo M. Sarria M.

Algoritmo Simplex

```
simplex_opt(C, f)
    sea C de la forma c_1 ∧ ... ∧ c_n
    for each i ∈ {1, ..., n}
        sea c_i de la forma b_i = Σ_{j=1}^{m} a_ij x_j donde b_i ≥ 0
        f_i := b_i − Σ_{j=1}^{m} a_ij x_j
        c'_i := (z_i = f_i)
    endfor
    f := Σ_{i=1}^{n} f_i
    ⟨flag, C', f'⟩ := simplex_opt(⋀_{i=1}^{n} c'_i, f)
    sea f' de la forma e + Σ_{j=1}^{n} d'_j z_j + Σ_{i=1}^{m} a'_i x_i
    if e ≡ 0 then
        return true
    else return false
    endif
```

Implicación

Dadas dos restricciones C_1 y C_2, se tiene el problema de determinar si $C_1 \rightarrow C_2$. Esto es, determinar si toda solución de C_1 es también solución de C_2.

Definición:
Un probador de implicación, *impl*, toma dos restricciones C_1 y C_2 retorna *true*, *false* o *unknown*. Cuando $impl(C_1, C_2)$ retorna *true* entonces $C_1 \rightarrow C_2$, mientras que cuando retorna *false*, $C_1 \nrightarrow C_2$.

Una aplicación importante de los probadores de implicación está en la simplificación, donde puede ser usado para eliminar restricciones primitivas redundantes.

Equivalencia

Otra operación sobre restricciones que está muy relacionada a la implicación es la equivalencia. Dadas dos restricciones C_1 y C_2, determinar si $C_1 \leftrightarrow C_2$. Esto es, si C_1 y C_2 tienen las mismas soluciones.

Definición:
Un probador de equivalencia, *equiv*, toma dos argumentos C_1 y C_2 y retorna *true*, *false* o *unknown*. Cuando $equiv(C1, C2) = true$, $C_1 \leftrightarrow C_2$ es válido y cuando $equiv(C1, C2) = false$, $C_1 \leftrightarrow C_2$ no lo es.

También se puede definir la equivalencia como

$$equiv(C1, C2) = impl(C1, C2) \;\wedge\; impl(C2, C1)$$

Fin de la Presentación