Constructive Algorithmics (Part IV)

Zhenjiang HU
University of Tokyo

Arrays

- Formalization of arrays as a binoid
- Formalization of well-defined functions on arrays as homomorphisms
- General rules: promotion/fusiton
- Special rules: Horner's rule
- Program derivation by calculation

Though you do not need to understand the details in this part, you should get feeling of how the theory of lists can be naturally extended to other data structures such as arrays and trees.

A Problem

Given is an array x with elements in the set $\{0,1\}$. Required is an efficient algorithm for computing the area of the largest rectangle (i.e., contiguous subarray) of x, all of whose elements are 1.

Binoid

Suppose α is a set of values closed under two partial operations $+$ and $\times$ such that:

(i) $+$ and $\times$ are associative, in the sense that each of the equations

$$\begin{array}{rcl} (a+b)+c & = & a+(b+c) \\ (a\times b)\times c & = & a\times(b\times c) \end{array}$$

holds whenever both sides of the equation are defined;

(ii) $+$ and $\times$ satisfy the further equation

$$(a+b)\times(c+d) = (a\times c)+(b\times d)$$

whenever both sides are defined. (we shall refer to this property by saying that $+$ *abides* with $\times$.)

Examples

- Let $\oplus$ be associative and commutative. Then $(\alpha, \oplus, \oplus)$ is a binoid.
- $(\alpha, \lessdot, \lessdot)$ is a binoid.
- $(\alpha, \gtrdot, \gtrdot)$ is a binoid.
- $(\alpha, \gtrdot, \lessdot)$ is a binoid.

Exercise: Let $\oplus$ is some associative operator, and $\bullet$ is a partial operator defined by the equation:

$$a \bullet b = a \quad \text{provided } a = b$$

Prove that $(\alpha, \oplus, \bullet)$ is a binoid.

Arrays

The type of arrays with elements from α will be denoted by $|\alpha|$.

- $|\cdot|$ maps elements of α to singleton arrays.
- ϕ (pronounced *beside*) puts two arrays with the same height one beside the other.
- $\ominus$ (pronounced *above*) puts two arrays with the same width one above the other.

$(|\alpha|, \phi, \ominus)$ forms a binoid.

$$(x \phi y) \ominus (u \phi v) \;=\; (x \ominus u) \phi (y \ominus v)$$

For instance, the array

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix}.$$

is described by the formula

$$(|1| \obar |2| \obar |3|) \ominus (|4| \obar |5| \obar |6|) \ominus (|7| \obar |8| \obar |9|)$$

as well as many others.

Exercise: Give another formula for the above array.

Two Functions: *height* and *width*

$$
\begin{aligned}
\mathit{height}\ |a| &= 1\\
\mathit{height}\ (x \ominus y) &= \mathit{height}\ x + \mathit{height}\ y\\
\mathit{height}\ (x \obar y) &= \mathit{height}\ x \bullet \mathit{height}\ y
\end{aligned}
$$

$$
\begin{aligned}
\mathit{width}\ |a| &= 1\\
\mathit{width}\ (x \ominus y) &= \mathit{width}\ x \bullet \mathit{width}\ y\\
\mathit{height}\ (x \obar y) &= \mathit{width}\ x + \mathit{width}\ y
\end{aligned}
$$

Map

We shall use the same symbol $*$ for mapping over arrrays as for mapping over lists.

$$
\begin{aligned}
f * |a| \quad &= \quad |f \ a| \\
f * (x \ominus y) \quad &= \quad (f * x) \ominus (f * y) \\
f * (x \phi y) \quad &= \quad (f * x) \phi (f * y)
\end{aligned}
$$

Map Distributivity

$$(f \cdot g) * \ = \ f * \cdot g*$$

Exercise: Prove the above map distributivity.

Reduce

Given two operators $\oplus, \otimes : \alpha \to \alpha \to \alpha$, we can define a reduction operator $(\oplus, \otimes)/$ for arrays by three equations:

$$
\begin{array}{lcl}
(\oplus, \otimes)/|a| & = & a \\
(\oplus, \otimes)/(x \ominus y) & = & ((\oplus, \otimes)/x) \ominus ((\oplus, \otimes)/y) \\
(\oplus, \otimes)/(x \phi y) & = & ((\oplus, \otimes)/x) \phi ((\oplus, \otimes)/y)
\end{array}
$$

For these equation to be consistent, we require that $(\alpha, \oplus, \otimes)$ forms a binoid.

Examples

- $(+,+)/$: sums the elements in an array of numbers.
- $(\wedge,\wedge)/$: determines whether there exists an entry in an array of booleans.
- $\mathit{height} = (+,\bullet)/\cdot K_1*$
- $\mathit{width} = (\bullet,+)/\cdot K_1*$
- $\mathit{area} = (+,+)/\cdot K_1*$
- $\mathit{topleft} = (\lessdot,\lessdot)/$
- $\mathit{id} = (\ominus,\phi)/\cdot|\cdot|*$
- $\mathit{tr} = (\phi,\ominus)/\cdot|\cdot|*$

Exercise: Define *topright* and *bottomleft*.

Promotion

The one-point and join rules for lists have counterparts in the theory of arrays.

One-point Rules:

$$\begin{array}{rcl} f * \cdot |\cdot| & = & |\cdot| \cdot f \\ (\oplus, \otimes)/ \cdot |\cdot| & = & id \end{array}$$

Join Rules:

$$\begin{array}{rcl} f * \cdot (\ominus\ , \ \phi\ )/ & = & (\ominus\ , \ \phi\ )/ \cdot f ** \\ (\oplus, \otimes)/ \cdot (\ominus\ , \ \phi\ )/ & = & (\oplus, \otimes)/ \cdot (\oplus, \otimes)/* \end{array}$$

Transpose Rules:

$$\begin{array}{rcl} f * \cdot (\ \phi, \ominus\ )/ & = & (\ \phi, \ominus\ ) \cdot f ** \\ (\oplus, \otimes)/ \cdot (\ \phi, \ominus\ ) & = & (\otimes, \oplus)/ \cdot (\oplus, \otimes)/* \end{array}$$

Exercise: Prove the transpose rules.

Exercise: Prove that $tr \cdot tr = id$.

Zip

Zip on Lists

We define a partial operator $\Upsilon_\oplus$ (pronounced *zip with* $\oplus$) informally by the equation:

$$[a_1, a_2, \ldots, a_n] \Upsilon_\oplus [b_1, b_2, \ldots, b_n] = [a_1 \oplus b_1, a_2 \oplus b_2, \ldots, a_n \oplus b_n]$$

or formally by

$$\begin{array}{lcl} [\,] \Upsilon_\oplus [\,] & = & [\,] \\ [a] \Upsilon_\oplus [b] & = & [a \oplus b] \\ (x +\!\!\!+\, y) \Upsilon_\oplus (u +\!\!\!+\, v) & = & (x \Upsilon_\oplus u) +\!\!\!+ (y \Upsilon_\oplus v) \end{array}$$

The third equation is asserted only under the conditions that $\#x = \#u$ and $\#y = \#v$.

Zip on Arrays

The same zip operator can be defined on arrays.

$$
\begin{array}{lcl}
|a| \Upsilon_{\oplus} |b| & = & |a \oplus b| \\
(x \ominus y) \Upsilon_{\oplus} (u \ominus v) & = & (x \Upsilon_{\oplus} u) \ominus (y \Upsilon_{\oplus} v) \\
(x \phi y) \Upsilon_{\oplus} (u \phi v) & = & (x \Upsilon_{\oplus} u) \phi (y \Upsilon_{\oplus} v)
\end{array}
$$

Examples

The function

$$rows = (\ominus\ ,\ \Upsilon_{\otimes})/ \cdot | \cdot |*$$

converts an array ointo a column vector whose entries are rwo vectors.

$$rows \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = \begin{pmatrix} (& 1 & 2 & 3 &) \\ (& 4 & 5 & 6 &) \\ (& 7 & 8 & 9 &) \end{pmatrix}$$

Exercise: Define the function *cols* which converts an array into a row vector whose entries are column vectors.

Array to List of Lists

The function *listrows* turns an array into a list of rows, each row being a list of entries from a row of the array. The function *listcols* turns an array into a list of columns.

$$\begin{array}{lcl} \mathit{listrows} & = & (+\!\!+\,, \Upsilon_{+\!\!+}\,)/ \cdot [[\cdot]]* \\ \mathit{listcols} & = & (\,\Upsilon_{+\!\!+}\,, +\!\!+\,)/ \cdot [[\cdot]]* \end{array}$$

Properties:

$$\begin{array}{lcl} height & = & \# \cdot listrows \\ length & = & \# \cdot listcols \\ \\ listcols & = & listrows \cdot tr \\ listrows & = & listcol \cdot tr \\ \\ (\oplus, \otimes)/ & = & \oplus/ \cdot \otimes/ * \cdot listrows \\ (\oplus, \otimes)/ & = & \otimes/ \cdot \oplus/ * \cdot listcols \end{array}$$

Directed Reductions

- Top Reductions:

$$\oplus \mapsfromdown \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = \begin{pmatrix} (1 \oplus 4) \oplus 7 & (2 \oplus 5) \oplus 8 & (3 \oplus 6) \oplus 9 \end{pmatrix}$$

- Left Reductions:

$$\oplus \mapsto \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = \begin{pmatrix} (1 \oplus 2) \oplus 3 \\ (4 \oplus 5) \oplus 6 \\ (7 \oplus 8) \oplus 9 \end{pmatrix}$$

Identities

$$
\begin{aligned}
rows &= \phi \; {+\!\!\!\rightarrow} \cdot |\cdot|* \\
cols &= \ominus \; {-\!\!\!\!\downarrow} \cdot |\cdot|*
\end{aligned}
$$

$$
\begin{aligned}
(\oplus.\otimes)/ &= the \cdot (\oplus {-\!\!\!\!\downarrow}) \cdot (\otimes {+\!\!\!\rightarrow}) \\
(\oplus.\otimes)/ &= the \cdot (\otimes {+\!\!\!\rightarrow}) \cdot (\oplus {-\!\!\!\!\downarrow})
\end{aligned}
$$

Note the function *the* is to extract the value from a singleton matrix.

$$the\ |a| \;=\; a$$

Accumulations

- Top Accumulation:

$$\oplus \mathrel{\downarrow\!\!\!\!=} \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = \begin{pmatrix} 1 & 2 & 3 \\ 1 \oplus 4 & 2 \oplus 5 & 3 \oplus 6 \\ (1 \oplus 4) \oplus 7 & (2 \oplus 5) \oplus 8 & (3 \oplus 6) \oplus 9 \end{pmatrix}$$

- Left Reductions:

$$\oplus \mathrel{\Vert\!\!\!\rightarrow} \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = \begin{pmatrix} 1 & 1 \oplus 2 & (1 \oplus 2) \oplus 3 \\ 4 & 4 \oplus 5 & (4 \oplus 5) \oplus 6 \\ 7 & 7 \oplus 8 & (7 \oplus 8) \oplus 9 \end{pmatrix}$$

Tops and Bottoms

There are four reasonable way to dissecting an array: we shall call them *tops*, *bottoms*, *lefts*, and *rights*.

$$\mathit{lefts}\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = \left(\begin{pmatrix} 1 \\ 4 \\ 7 \end{pmatrix} \; \begin{pmatrix} 1 & 2 \\ 4 & 5 \\ 7 & 8 \end{pmatrix} \; \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \right)$$

$$\mathit{lefts} = (\,\phi\ \#\!\!\!\rightarrow) \cdot \mathit{cols}$$

Exercise: Give definitions for *tops*, *bottoms*, and *rights*.

Properties

$$
\begin{aligned}
(\oplus\!-\!\!\!\!\downarrow)*\cdot lefts &= lefts\cdot(\oplus\!-\!\!\!\!\downarrow)\\
(\oplus\!=\!\!\!\!\downarrow)*\cdot lefts &= lefts\cdot(\oplus\!=\!\!\!\!\downarrow)
\end{aligned}
$$

Accumulation Lemmas

$$
\begin{aligned}
(\oplus\!-\!\!\!\!\downarrow)*\cdot tops &= rows\cdot(\oplus\!=\!\!\!\!\downarrow)\\
(\oplus\mid\!\!\!\rightarrow)*\cdot lefts &= cols\cdot(\oplus\|\!\!\!\rightarrow)
\end{aligned}
$$

Horner's Rule

The equation

$$(\oplus, \oplus)/ \cdot (\otimes, \odot)/ * \cdot \mathit{bottoms} \;=\; (\odot, \odot)/ \cdot \circledast\!\!\not{\!\!\!\!\!\!\downarrow}$$

$$\text{where } a \circledast b = (a \otimes b) \oplus b$$

holds, provided that (i) $\otimes$ distributes (backwards) through $\oplus$; and (ii) $\oplus$ abides with $\odot$.

Exercise: Prove the Horner's rule.

Rectangles

A rectangle of an array x is a contiguous subarray of x.

$$
\begin{aligned}
\mathit{topls} &= (\ominus\,,\,\phi)/\cdot\,\mathit{tops} * \cdot\mathit{lefts}\\
\mathit{botrs} &= (\ominus\,,\,\phi)/\cdot\,\mathit{bottoms} * \cdot\mathit{rights}\\
\mathit{rects} &= (\ominus\,,\,\phi)/\cdot\,\mathit{botrs} * \cdot\mathit{topls}
\end{aligned}
$$

Exercise: What is the result of the following expression?

$$
\mathit{tops}\begin{pmatrix} 1 & 2 & 3\\ 4 & 5 & 6\\ 7 & 8 & 9 \end{pmatrix}
$$

Solving Our Problem

We can solve our problem (largest rectangle area) by

$$lra \;=\; \uparrow / \cdot area * \cdot filled \triangleleft \cdot a2l \cdot rects$$

where

$$\begin{array}{lcl} a2l & = & (+\!\!+\,, +\!\!+\,)/ \cdot [\cdot] \\ filled & = & (\wedge, \wedge)/ \cdot (= 1)* \end{array}$$

Exercise (Challenge): Calculate an efficient algorithm for the largest rectangle area problem from the above naive solution.