Knapsack in a Monadic Setting

Shin-Cheng Mu

Institute of Information Science, Academia Sinica

1 DEFINITIONS

Refer to other documents for definitions of *foldR*.

The function *subseq* non-deterministically computes a sub-sequence of the given list. It can be defined inductively:

$$\begin{aligned}
&\mathit{subseq} :: \mathsf{List}\ a \rightarrow \mathsf{P}\ (\mathsf{List}\ a)\\
&\mathit{subseq}\ [\,] \qquad\ = \mathit{return}\ [\,]\\
&\mathit{subseq}\ (x : xs) = \mathit{subseq}\ xs \mathbin{[\!]} ((x:) \mathbin{\langle\$\rangle} \mathit{subseq}\ xs) \quad ,
\end{aligned}$$

but we will use a *foldR*-based definition here:

$$\begin{aligned}
&\mathit{subseq} = \mathit{foldR}\ \mathit{subs}\ (\mathit{return}\ [\,]) \quad ,\\
&\quad \mathbf{where}\ \mathit{subs}\ x\ ys = \mathit{return}\ ys \mathbin{[\!]} \mathit{return}\ (x : ys) \quad .
\end{aligned}$$

An item is specified by its value and weight:

$$\begin{aligned}
&\mathbf{type}\ \mathsf{Val} = \mathsf{Int}\\
&\mathbf{type}\ \mathsf{Wgt} = \mathsf{Int}\\
&\mathbf{type}\ \mathsf{Item} = (\mathsf{Val}, \mathsf{Wgt}) \quad .\\
&\mathit{val} :: \mathsf{List}\ \mathsf{Item} \rightarrow \mathsf{Val}\\
&\mathit{val} = \mathit{sum} \cdot \mathit{map}\ \mathit{fst} \quad .\\
&\mathit{wgt} :: \mathsf{List}\ \mathsf{Item} \rightarrow \mathsf{Wgt}\\
&\mathit{wgt} = \mathit{sum} \cdot \mathit{map}\ \mathit{snd} \quad .
\end{aligned}$$

Let $(\leqslant_v)$ be defined by $xs \leqslant_v ys \equiv \mathit{val}\ xs \leqslant \mathit{val}\ ys$, thus $\mathit{max}_{\leqslant_v} :: \mathsf{P}\ (\mathsf{List}\ \mathsf{Item}) \rightarrow \mathsf{P}\ (\mathsf{List}\ \mathsf{Item})$ choose those lists having the largest value. The *knapsack* problem can be defined by:

$$\begin{aligned}
&\mathit{knapsack} :: \mathsf{List}\ \mathsf{Item} \rightarrow \mathsf{P}\ (\mathsf{List}\ \mathsf{Item})\\
&\mathit{knapsack} = \mathit{max}_{\leqslant_v} \cdot (\mathit{filt}\ ((w>) \cdot \mathit{wgt}) \lll \mathit{subseq}) \quad .
\end{aligned}$$

2 FUSION

Recall the *foldR* fusion rule:

$$\mathit{foldR}\ g\ (h\ e) \subseteq h \cdot \mathit{foldR}\ f\ e \;\Leftarrow\; g\ x \mathbin{=\!\!\lll} h\ m \subseteq h\ (f\ x \mathbin{=\!\!\lll} m) \quad . \tag{1}$$

The task is to fuse $\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll \mathit{subseq}$ into $\mathit{foldR}\ \mathit{subsw}\ (\mathit{return}\ [\,])$ for some subsw. For the base case, we assume that w is non-negative, therefore $\mathit{filt}\ ((w{>})\cdot \mathit{wgt})\ [\,] = \mathit{return}\ [\,]$ holds. The function subsw should satisfy the fusion condition:

$$\mathit{subsw}\ x \ggg (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \ggg m) \subseteq \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll (\mathit{subs}\ x \ggg m) \quad .$$

To construct subsw we reason:

$$\begin{aligned}
& \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll (\mathit{subs}\ x \ggg m) \\
= \quad & \{ \text{ definition of } \mathit{subs} \ \} \\
& \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll ((\lambda ys \rightarrow \mathit{return}\ ys \mathbin{[\!]} \mathit{return}\ (x:ys)) \ggg m) \\
= \quad & \{ \text{ distributivity, definition of } (\langle\$\rangle) \ \} \\
& \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll (m \mathbin{[\!]} (x{:}) \langle\$\rangle\ m) \\
= \quad & \{ \text{ distributivity } \} \\
& \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m \mathbin{[\!]} (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll (x{:}) \langle\$\rangle\ m) \\
\supseteq \quad & \{ \text{ since } (\mathit{filt}\ p{\lll}) \subseteq \mathit{id} \ \} \\
& \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m \mathbin{[\!]} (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll ((x{:}) \langle\$\rangle\ (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m))) \\
= \quad & \{ \text{ definition of } (\langle\$\rangle) \text{ and monad laws, to factor out } \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m \ \} \\
& \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m \mathbin{[\!]} ((\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \cdot (x{:})) \lll (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m)) \\
= \quad & \{ \text{ distributivity, definition of } (\langle\$\rangle) \ \} \\
& (\lambda ys \rightarrow \mathit{return}\ ys \mathbin{[\!]} \mathit{filt}\ ((w{>})\cdot \mathit{wgt})\ (x:ys)) \lll (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll m) \quad .
\end{aligned}$$

Therefore we have

$$\begin{aligned}
& \mathit{foldR}\ \mathit{subsw}\ (\mathit{return}\ [\,]) \ \supseteq\ \mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll \mathit{subseq} \quad , \\
& \quad \mathbf{where}\ \mathit{subsw}\ x\ ys = \mathit{return}\ ys \mathbin{[\!]} \mathit{filt}\ ((w{>})\cdot \mathit{wgt})\ (x:ys) \quad .
\end{aligned}$$

Curiously, in the step using $(\mathit{filt}\ p{\lll}) \subseteq \mathit{id}$ we need only one side of the inclusion, therefore we have not yet demanded that $(w{>})\cdot \mathit{wgt}$ being suffix-closed.

3 INTRODUCING THINNING

$$\begin{aligned}
& \mathit{max}_{\leqslant_v} \cdot (\mathit{filt}\ ((w{>})\cdot \mathit{wgt}) \lll \mathit{subseq}) \\
\supseteq \quad & \{ \ \mathit{foldR}\text{-fusion } \} \\
& \mathit{max}_{\leqslant_v} \cdot \mathit{foldR}\ \mathit{subsw}\ (\mathit{return}\ [\,]) \\
\supseteq \quad & \{ \text{ introducing } \mathit{thin} \ \} \\
& \mathit{max}_{\leqslant_v} \cdot \mathit{thin}_{\preceq} \cdot \mathit{foldR}\ \mathit{subsw}\ (\mathit{return}\ [\,]) \quad . \\
\supseteq \quad & \{ \text{ thinning theorem } \} \\
& \mathit{max}_{\leqslant_v} \cdot \mathit{foldR}\ (\lambda x \rightarrow \mathit{thin}_{\preceq} \cdot \mathit{collect} \cdot (\mathit{subsw}\ x \lll \in))\ (\mathit{thin}_{\preceq}\ (\mathit{collect}\ e))
\end{aligned}$$