Resource Analysis: Problem Set 8

Jan Hoffmann
Carnegie Mellon University

March 28, 2016

Due before 1:30pm on Monday, April 11

8.1 (20 Points) Resource-Polymorphic Recursion

In this problem we are interested in the number of cons operations. We use a metric M with $M^{\mathsf{cons}} = 1$ and $M^K = 0$ for all $K \neq \mathrm{cons}$.

Consider the following OCaml functions.

```
let rec rev_append (l1,l2) =
  match l1 with
  | [] → l2
  | x::xs → rev_append (xs, x::l2)

let rec skip l =
  match l with
    | [] → []
    | x1::xs →
      match xs with
        | [] → []
        | x2::xs → x2::skip(xs)

let rec f1 l =
  match l with
    | [] → []
    | x::xs →
        let l' = skip l in
        rev_append (f1 l', l')
```

a) Give linear resource-annotated types for the functions *rev_append*, *skip*, and *f1*.

b) Provide annotated type derivations for the functions *skip* and *f1* that justify your types. Transform the functions to share-let-normal form for the type derivation.

c) Argue informally why the type inference algorithm that we discussed in class cannot derive a resource-annotated type for *f1*.

d) Now give a resource-annotated type derivation for *f2*, which is defined below. Why can the type inference algorithm derive this bound?

```
let rec f2 l =
  match l with
    | [] → []
    | x::xs →
          let l' = skip l in
          rev_append (l', f2 l')
```

e) Finally, consider again the original program in which we replace *skip* with the following implementation.

```
let rec skip l =
  match l with
    | [] → []
    | x1::xs →
      match xs with
        | [] → [x1]
        | x2::xs → x2::skip(xs)
```

Explain informally why your type derivation from (b) does not work for the new variant of *skip*. Can you informally derive a bound for *f1* with the new variant of *skip*?

8.2 (16 Points) Non-Negative Polynomials

The potential functions of *univariate polynomial amortized resource analysis* are a generalization of non-negative linear combinations of binomial coefficients (binomials)

$$\mathcal{B} = \left\{ \lambda\, n.\ \sum_{i=0,\ldots,k} q_i \binom{n}{i} \;\middle|\; k \in N, q_i \in \mathbb{Q}_{\geq 0} \right\}$$

Recall that $\binom{n}{0} = 1$ for every $n \in \mathbb{N}$.

For a function $f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$, the *discrete derivative* $\Delta f : \mathbb{N} \to \mathbb{Q}_{\geq 0}$ is defined by

$$(\Delta f)(n) = f(n+1) - f(n)\,.$$

As usual, we define $\Delta^0 f = f$ and $\Delta^k f = \Delta(\Delta^{k-1} f)$ if $k > 0$.

The set of polynomials is defined as

$$\mathcal{P} = \left\{ \lambda\, n.\ \sum_{i=0,\ldots,k} q_i n^i \;\middle|\; k \in N, q_i \in \mathbb{Q} \right\}$$

We call a function $f : \mathbb{N} \to \mathbb{Q}$ *hereditary non-negative* if $\Delta^i f \geq 0$ for all $i \in \mathbb{N}$.

a) Prove that if $f \in \mathcal{B}$ and $k \in N$ then $\Delta^k f \in \mathcal{B}$. Note that it follows that $\mathcal{B}$ is a set of hereditary non-negative polynomials.

b) Show that $\mathcal{B}$ is the largest set of hereditary non-negative polynomials: $\mathcal{P} \cap \{f \mid \forall i\ \Delta^i f \geq 0\} = \mathcal{B}$.

   Hint: If $p : \mathbb{N} \to \mathbb{Q}$ is a hereditary non-negative polynomial then $p(n) = \sum_i q_i \binom{n}{i}$ where $q_i = (\Delta^i p)(0)$.

c) Let $\mathcal{C}$ be a set of non-negative polynomials that is closed under discrete differentiation, that is, $p \in \mathcal{C} \implies \Delta p \in \mathcal{C}$. Show that $\mathcal{C} \subseteq \mathcal{B}$.

8.3 (12 Points) Resource Aware ML

Resource Aware ML (RAML) is an implementation of *multivariate polynomial amortized resource analysis* for OCaml. A web interface for RAML is available at

`http://raml.co`

1. Use the web interface of RAML to derive evaluation-step bounds on the functions defined in Problem 8.1 and report the derived bounds.

2. Use the template in the file *search.raml* and the functional queue from Problem 2.3 to implement a breadth-first search and a depth-first search in RAML. Derive and report evaluation-step bounds using the web interface.

$\Sigma;\Gamma \vdash^{q}_{q'} e : B$	Given resource metric M, expression e has annotated type A under signature Σ in context Γ.

$$\frac{q = q' + M^{\mathsf{var}}}{\Sigma; x : B \vdash^{q}_{q'} x : B}\ (\text{L:VAR}) \qquad \frac{A \xrightarrow{p/p'} B \in \Sigma(f) \qquad q = p + M^{\mathsf{app}}}{\Sigma; x{:}A \vdash^{q}_{p'} \text{app}(f, x) : B}\ (\text{L:APP})$$

$$\frac{\Sigma;\Gamma_1 \vdash^{p}_{p'} e_1 : A \qquad \Sigma;\Gamma_2, x{:}A \vdash^{p'}_{q'} e_2 : B \qquad q = p + M^{\mathsf{let}}}{\Sigma;\Gamma_1,\Gamma_2 \vdash^{q}_{q'} \text{let}(e_1, x.e_2) : B}\ (\text{L:LET})$$

$$\frac{e \in \{\text{true}, \text{false}\} \qquad q = M^{\mathsf{cbool}} + q'}{\Sigma; \cdot \vdash^{q}_{q'} b : \text{Bool}}\ (\text{L:BCONST})$$

$$\frac{\Sigma;\Gamma \vdash^{q_1}_{q'} e_1 : B \qquad \Sigma;\Gamma \vdash^{q_2}_{q} e_2 : B \qquad q = M_1^{\mathsf{cond}} + q_1 \qquad q = M_2^{\mathsf{cond}} + q_2}{\Sigma;\Gamma, x : \text{Bool} \vdash^{q}_{q'} \text{if}(x, e_1, e_2) : B}\ (\text{L:COND})$$

$$\frac{q = M^{\mathsf{pair}} + q'}{\Sigma; x_1 : A_1, x_2 : A_2 \vdash^{q}_{q'} \text{pair}(x_1, x_2) : A_1 * A_2}\ (\text{L:PAIR})$$

$$\frac{\Sigma;\Gamma, x_1{:}A_1, x_2{:}A_2 \vdash^{p}_{q'} e' : B \qquad q = M^{\mathsf{MatP}} + p}{\Sigma;\Gamma, x : A_1 * A_2 \vdash^{q}_{q'} \text{matP}(e, (x_1, x_2).e') : B}\ (\text{L:MATP}) \qquad \frac{q = M^{\mathsf{nil}} + q'}{\Sigma; \cdot \vdash^{q}_{q'} \text{nil} : L^p(A)}\ (\text{L:NIL})$$

$$\frac{q = M^{\mathsf{cons}} + p + q'}{\Sigma; x_1 : A, x_2 : L^p(A) \vdash^{q}_{q'} \text{cons}(x_1, x_2) : L^p(A)}\ (\text{L:CONS})$$

$$\frac{\begin{array}{c}\Sigma;\Gamma \vdash^{q_1}_{q'} e_1 : B \\ \Sigma;\Gamma, x_1 : A, x_2 : L^p(A) \vdash^{q_2}_{q'} e_2 : B \qquad q = M_1^{\mathsf{matL}} + q_1 \qquad q + p = M_2^{\mathsf{matL}} + q_2\end{array}}{\Sigma;\Gamma, x{:}L^p(A) \vdash^{q}_{q'} \text{matL}(x, e_1, (x_1, x_2).e_2) : B}\ (\text{L:MATL})$$

$$\frac{\Sigma;\Gamma, x_1 : A_1, x_2 : A_2 \vdash^{q}_{q'} e : B \qquad A \curlyvee (A_1, A_2)}{\Sigma;\Gamma, x : A \vdash^{q}_{q'} \text{share}(x, (x_1, x_2).e) : B}\ (\text{L:SHARE})$$

$$\frac{\Sigma;\Gamma, x{:}A \vdash^{q}_{q'} e : B \qquad A' <: A}{\Sigma;\Gamma, x{:}A' \vdash^{q}_{q'} e : B}\ (\text{L:SUPERTYPE}) \qquad \frac{\Sigma;\Gamma \vdash^{q}_{q'} e : B \qquad B <: B'}{\Sigma;\Gamma \vdash^{q}_{q'} e : B'}\ (\text{L:SUBTYPE})$$

$$\frac{\Sigma;\Gamma \vdash^{p}_{p'} e : B \qquad q \geq p \qquad q - p \geq q' - p'}{\Sigma;\Gamma \vdash^{q}_{q'} e : B}\ (\text{L:RELAX}) \qquad \frac{\Sigma;\Gamma \vdash^{q}_{q'} e : B}{\Sigma;\Gamma, x{:}A \vdash^{q}_{q'} e : B}\ (\text{L:WEAK})$$

Figure 1: Linear resource-annotated type rules.