Knuth's Generalization of Takeuchi's Tarai Function

by

Tom Bailey

John Cowles

University of Wyoming

Takeuchi's Tarai Function

For integer inputs, x, y, z,

$$t(x,y,z) \stackrel{\text{def}}{\Leftarrow} \text{ if } \ x \le y \ \text{ then } \ y \\ \qquad\qquad\qquad\qquad \text{else } \ t(\ t(x-1,y,z), \\ \qquad\qquad\qquad\qquad\qquad t(y-1,z,x), \\ \qquad\qquad\qquad\qquad\qquad t(z-1,x,y)\ ).$$

John McCarthy proved this recursion terminates and t can be computed without any recursion,

$$t(x,y,z) \ = \ \text{if } \ x \le y \ \text{ then } \ y \\ \qquad\qquad\qquad\qquad \text{else if } \ y \le z \ \text{ then } \ z \\ \qquad\qquad\qquad\qquad \text{else } \ x.$$

McCarthy's Measure

$$\text{measure}(x, y, z) \stackrel{\text{def}}{\Leftarrow} \text{measure}_1(x - y, z - y)$$

where

$$\begin{array}{l}
\text{measure}_1(m, n) \stackrel{\text{def}}{\Leftarrow} \\
\quad \text{if} \;\; m \leq 0 \;\; \text{then} \;\; 0 \\
\qquad \text{else if} \;\; n \geq 2 \\
\qquad\qquad \text{then} \;\; m + n(n-1)/2 - 1 \\
\qquad \text{else if} \;\; n \geq 0 \;\; \text{then} \;\; m \\
\qquad \text{else if} \;\; n = -1 \\
\qquad\qquad \text{then} \;\; (m+1)(m+2)/2 - 1 \\
\qquad \text{else} \;\; (m-n)(m-n+1)/2 - m - 1.
\end{array}$$

J Moore's Simpler Measure

Early Boyer-Moore theorem prover, THM, used to verify termination and to show t satisfies the simpler nonrecursive equation.

Lexicographical ordering on triples of nonnegative integers:

$$m(x,y,z) \stackrel{\text{def}}{\Leftarrow} \\ < m_1(x,y,z), m_2(x,y,z), m_3(x,y,z) >$$

where

$$\begin{aligned} m_1(x,y,z) &\stackrel{\text{def}}{\Leftarrow} \text{ if } x \leq y \text{ then } 0 \text{ else } 1, \\ m_2(x,y,z) &\stackrel{\text{def}}{\Leftarrow} \max(x,y,z) - \min(x,y,z), \\ m_3(x,y,z) &\stackrel{\text{def}}{\Leftarrow} x - \min(x,y,z). \end{aligned}$$

Knuth's Generalization

Generalize the tarai function to higher dimensions:

For integer inputs, $x_1, x_2, \ldots, x_m$,

$$
\begin{array}{l}
t(x_1, x_2, \ldots, x_m) \overset{\text{def}}{\Leftarrow} \\
\quad \text{if } \ x_1 \le x_2 \ \text{ then } \ x_2 \\
\quad\quad \text{else } \ t(\ t(x_1 - 1, x_2, \ldots, x_m), \\
\quad\quad\quad\quad\quad\quad t(x_2 - 1, x_3, \ldots, x_m, x_1), \\
\quad\quad\quad\quad\quad\quad\quad\quad \vdots \\
\quad\quad\quad\quad\quad\quad t(x_m - 1, x_1, \ldots, x_{m-1})\ ).
\end{array}
$$

Knuth's Two Questions

1. Are there total functions on the integers that satisfy the recursive equation based on the definition?

   That is, are there total functions $f(x_1, x_2, \ldots, x_m)$ on the integers that satisfy the equation

$$
\begin{array}{l}
f(x_1, x_2, \ldots, x_m) = \\
\quad \text{if } \; x_1 \le x_2 \; \text{ then } \; x_2 \\
\qquad \text{else } \; f(\; f(x_1 - 1, x_2, \ldots, x_m), \\
\qquad\qquad\qquad f(x_2 - 1, x_3, \ldots, x_m, x_1), \\
\qquad\qquad\qquad\qquad \vdots \\
\qquad\qquad\qquad f(x_m - 1, x_1, \ldots, x_{m-1}) \;)?
\end{array}
$$

2. Does the recursion terminate for all integer inputs?

Question 1: Can the recursive equation be satisfied?

McCarthy proved when $m = 3$:

The function

$$t(x_1, x_2, x_3) \stackrel{\text{def}}{\Leftarrow} \begin{array}{l} \text{if } \ x_1 \le x_2 \ \text{ then } \ x_2 \\ \quad \text{else if } \ x_2 \le x_3 \ \text{ then } \ x_3 \\ \qquad \text{else } \ x_1 \end{array}$$

satisfies the equation

$$t(x_1, x_2, x_3) \ = \ \begin{array}{l} \text{if } \ x_1 \le x_2 \ \text{ then } \ x_2 \\ \quad \text{else } \ \ t(\ t(x_1 - 1, x_2, x_3), \\ \qquad\qquad t(x_2 - 1, x_3, x_1), \\ \qquad\qquad t(x_3 - 1, x_1, x_2)\ ). \end{array}$$

Question 1: Can the recursive equation be satisfied?

Knuth notes when $m = 4$:

The function

$$
\begin{array}{l}
t(x_1, x_2, x_3, x_4) \stackrel{\text{def}}{\Leftarrow} \\
\quad \text{if } \; x_1 \le x_2 \; \text{ then } \; x_2 \\
\qquad \text{else if } \; x_2 \le x_3 \; \text{ then } \; x_3 \\
\qquad \text{else if } \; x_3 \le x_4 \; \text{ then } \; x_4 \\
\qquad\qquad \text{else } \; x_1
\end{array}
$$

satisfies the equation

$$
\begin{array}{l}
t(x_1, x_2, x_3, x_4) = \\
\quad \text{if } \; x_1 \le x_2 \; \text{ then } \; x_2 \\
\qquad \text{else } \quad t(\; t(x_1 - 1, x_2, x_3, x_4), \\
\qquad\qquad\qquad t(x_2 - 1, x_3, x_4, x_1), \\
\qquad\qquad\qquad t(x_3 - 1, x_4, x_1, x_2), \\
\qquad\qquad\qquad t(x_4 - 1, x_1, x_2, x_3) \;).
\end{array}
$$

Question 1: Can the recursive equation be satisfied?

Conjecture

The function

$$
\begin{aligned}
&t(x_1, x_2, \ldots, x_m) \stackrel{\text{def}}{\Leftarrow} \\
&\quad \text{if} \quad (\exists k < m)(x_1 > x_2 > \cdots > x_k \leq x_{k+1}) \\
&\qquad \text{then} \quad x_{k+1} \\
&\qquad \text{else} \quad x_1
\end{aligned}
$$

satisfies the equation

$$
\begin{aligned}
&t(x_1, x_2, \ldots, x_m) = \\
&\quad \text{if} \quad x_1 \leq x_2 \quad \text{then} \quad x_2 \\
&\qquad \text{else} \quad t(\ t(x_1 - 1, x_2, \ldots, x_m), \\
&\qquad\qquad\quad t(x_2 - 1, x_3, \ldots, x_m, x_1), \\
&\qquad\qquad\qquad \vdots \\
&\qquad\qquad\quad t(x_m - 1, x_1, \ldots, x_{m-1})\ ).
\end{aligned}
$$

Question 1: Can the recursive equation be satisfied?

Knuth's Counterexample, when $m = 5$

Left side

$$t(5,3,2,0,1) = 1$$

Right side

$$\begin{aligned} t(5,3,2,0,1) &= t(\ t(4,3,2,0,1), \\ &\qquad t(2,2,0,1,5), \\ &\qquad t(1,0,1,5,3), \\ &\qquad t(-1,1,5,3,2), \\ &\qquad t(0,5,3,2,0)\ ) \\ &= t(1,2,1,1,5) \\ &= 2 \end{aligned}$$

Question 1: Can the recursive equation be satisfied?

Knuth proposes a solution

$$
\begin{array}{l}
f(x_1, x_2, \ldots, x_m) \stackrel{\text{def}}{\Leftarrow} \\
\quad \text{if} \;\; (\exists k < m)(x_1 > x_2 > \cdots > x_k \leq x_{k+1}) \\
\qquad \text{then} \;\; g(x_1, x_2, \ldots, x_{k+1}) \\
\qquad \text{else} \;\; x_1
\end{array}
$$

Here the "function" g takes a variable number (at least two) of integer inputs.

$$
\begin{array}{l}
g(x_1, x_2, \ldots, x_j) \stackrel{\text{def}}{\Leftarrow} \\
\quad \text{if} \;\; j = 2 \;\; \text{then} \;\; x_2 \\
\qquad \text{else if} \;\; x_1 = x_2 + 1 \;\; \text{then} \;\; g(x_2, \ldots, x_j) \\
\qquad \text{else if} \;\; x_2 = x_3 + 1 \;\; \text{then} \;\; \max(x_3, x_j) \\
\qquad\qquad \text{else} \;\; x_j
\end{array}
$$

Question 1: Can the recursive equation be satisfied?

Knuth's Challenge

Theorem 4. The function $f(x_1, x_2, \ldots, x_m)$ satisfies the m-dimensional tarai recurrence.

That is,

$$
\begin{aligned}
&f(x_1, x_2, \ldots, x_m) = \\
&\quad \text{if } x_1 \le x_2 \text{ then } x_2 \\
&\qquad \text{else } f(\, f(x_1 - 1, x_2, \ldots, x_m), \\
&\qquad\qquad f(x_2 - 1, x_3, \ldots, x_m, x_1), \\
&\qquad\qquad\qquad \vdots \\
&\qquad\qquad f(x_m - 1, x_1, \ldots, x_{m-1})\,).
\end{aligned}
$$

Open Problem 4. Prove Theorem 4 by computer.

Question 1: Can the recursive equation be satisfied?

References

Textbook Examples of Recursion.
Chapter 22 in
D.E. Knuth,
Selected Papers on the Analysis of Algorithms,
CSLI Publications,
Distributed by Cambridge University Press,
2000.

Update of paper in
V. Lifschitz, Editor,
Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy,
Academic Press, 1991.

Question 1: Can the recursive equation be satisfied?

Bailey's Counterexample, when $m = 6$

Left side

$$f(8,6,4,3,1,2) = g(8,6,4,3,1,2) = 2$$

Right side *inner* evaluations

$$\begin{aligned}
f(7,6,4,3,1,2) &= g(7,6,4,3,1,2) \\
&= g(6,4,3,1,2) \\
&= \max(3,2) = 3 \\
f(5,4,3,1,2,8) &= g(5,4,3,1,2) \\
&= g(4,3,1,2) \\
&= g(3,1,2) = 2 \\
f(3,3,1,2,8,6) &= g(3,3) = 3 \\
f(2,1,2,8,6,4) &= g(2,1,2) = g(1,2) = 2 \\
f(0,2,8,6,4,3) &= g(0,2) = 2 \\
f(1,8,6,4,3,1) &= g(1,8) = 8
\end{aligned}$$

Right side outer evaluation

$$f(3,2,3,2,2,8) = g(3,2,3) = g(2,3) = 3$$

Question 1: Can the recursive equation be satisfied?

Knuth's Reaction to Counterexample

Date: Wed, 16 Feb 2000 13:42:50 -0800 (PST)

To: cowles
Subject: note from Prof Knuth
Dear Dr Cowles,

Wow!
or maybe I should say Ow!
or Whoa!
or Woe!
Thanks again for another crucial news flash.

. . .

This is certainly a great way to make me believe in mechanical verification. I presume that the case m=5 will still hold up to scrutiny by ACL2 . . .

Question 1: Can the recursive equation be satisfied?

Bailey Proposes a Correction

$$f(x_1, x_2, \ldots, x_m) \stackrel{\text{def}}{\Leftarrow}$$
$$\text{if } (\exists k < m)(x_1 > x_2 > \cdots > x_k \le x_{k+1})$$
$$\text{then } g_b(x_1, x_2, \ldots, x_{k+1})$$
$$\text{else } x_1$$

Here the "function" g_b, like g, takes a variable number (at least two) of integer inputs.

$$g_b(x_1, x_2, \ldots, x_j) \stackrel{\text{def}}{\Leftarrow}$$
$$\text{if } j \le 3 \text{ then } x_j$$
$$\text{else if } x_1 = x_2 + 1 \text{ or } x_2 > x_3 + 1$$
$$\text{then } g_b(x_2, \ldots, x_j)$$
$$\text{else } \max(x_3, x_j)$$

Question 1: Can the recursive equation be satisfied?

Machine Verification Pending

Conjecture 1. The function $f(x_1, x_2, \ldots, x_m)$, *with* g_b *replacing* g, satisfies the m-dimensional tarai recurrence.

That is,

$$
\begin{array}{l}
f(x_1, x_2, \ldots, x_m) = \\
\quad \text{if } \; x_1 \le x_2 \; \text{ then } \; x_2 \\
\qquad \text{else } \; f(\; f(x_1 - 1, x_2, \ldots, x_m), \\
\qquad\qquad\quad f(x_2 - 1, x_3, \ldots, x_m, x_1), \\
\qquad\qquad\qquad \vdots \\
\qquad\qquad\quad f(x_m - 1, x_1, \ldots, x_{m-1}) \;).
\end{array}
$$

Proof checked by hand.

Question 2: Does the recursion terminate?

May Depend on Evaluation Rule

Knuth points out,
"... a call-by-need technique *will* always terminate when applied to the recursive equation for $t(x_1, \ldots, x_m)$.

If $x_1 > x_2 > \cdots > x_k \leq x_{k+1}$, the values $y_i = t(x_i - 1, x_{i+1}, \ldots, x_{i-1})$ need be expanded only for $1 \leq i \leq k+1$, and this will be sufficient to determine the value of $t(y_1, \ldots, y_m) = t(x_1, \ldots, x_m)$ in a finite number of steps."

Knuth's argument depends on the faulty proof given for **Theorem 4.**

Question 2: Does the recursion terminate?

Machine Verification Pending

Conjecture 2. The recursion for computing $t(x_1, \ldots, x_m)$ always terminates using the following version of Knuth's call-by-need.

If $x_1 > x_2 > \cdots > x_k \leq x_{k+1}$, it is sufficient to expand the values $y_i = t(x_i - 1, x_{i+1}, \ldots, x_{i-1})$ only for $1 \leq i \leq k$, to determine the value of $t(y_1, \ldots, y_m) = t(x_1, \ldots, x_m)$.

Note the change from $k+1$ to k in this range for i.

Question 2: Does the recursion terminate?

This is an Interesting Question!

Knuth continues,
"Therefore we come to a final question, which will perhaps prove to be the most interesting aspect of the present investigation, particularly if it has a negative answer.

...If so, the tarai recurrence would be an extremely interesting example to include in *all* textbooks about recursion."

Open Problem 5. Does the m-dimensional tarai recursive equation define a total function, for all $m \geq 3$, if it is expanded fully (without call-by-need)?

Question 2: Does the recursion terminate?

An Interesting Example – of What?

Negative answer to **Open Problem 5:**

The m-dimensional tarai function shows that LISP's evaluation rules do *not* always compute *least fixed-points*.

Lisp's evaluation rules:

- innermost first, left to right
- call by value

Simpler examples are known.

Question 2: Does the recursion terminate?

A Simpler Example

LISP's evaluation rules do *not* always compute *least fixed-points*.

$$L(x,y) \stackrel{\text{def}}{\Leftarrow} \text{if } x = 0 \text{ then } 0 \text{ else } L(x-1, L(x,y)).$$

Only one function, over the nonnegative integers, satisfies the recursive equation.

$$L(x,y) \stackrel{\text{def}}{\Leftarrow} 0$$

Recursion does *not* terminate.
Assuming left-to-right, innermost-first evaluation order.

Question 2: Does the recursion terminate?

A Counterexample, when $m = 4$

$$
\begin{aligned}
t(3,2,1,5) &= t(\ t(2,2,1,5), \\
&\qquad t(1,1,5,3), \\
&\qquad t(0,5,3,2), \\
&\qquad t(4,3,2,1)\ ) \\
&= t(2,1,5,t(4,3,2,1)) \\
&= t(2,1,5,t(\ t(3,3,2,1), \\
&\qquad\qquad\qquad t(2,2,1,4), \\
&\qquad\qquad\qquad t(1,1,4,3), \\
&\qquad\qquad\qquad t(0,4,3,2)\ )) \\
&= t(2,1,5,t(3,2,1,4)) \\
&= t(2,1,5,t(\ t(2,2,1,4), \\
&\qquad\qquad\qquad t(1,1,4,3), \\
&\qquad\qquad\qquad t(0,4,3,2), \\
&\qquad\qquad\qquad t(3,3,2,1)\ )) \\
&= t(2,1,5,t(2,1,4,3))
\end{aligned}
$$

Question 2: Does the recursion terminate?

Counterexample continued

From last line on previous slide:

$$
\begin{aligned}
t(3,2,1,5) &= t(2,1,5,t(2,1,4,3)) \\
&= t(2,1,5,t(\ t(1,1,4,3), \\
&\qquad\qquad\quad t(0,4,3,2), \\
&\qquad\qquad\quad t(3,3,2,1), \\
&\qquad\qquad\quad t(2,2,1,4)\ )) \\
&= t(2,1,5,t(1,4,3,2)) \\
&= t(2,1,5,4) \\
&= t(\ t(1,1,5,4), \\
&\qquad t(0,5,4,2), \\
&\qquad t(4,4,2,1), \\
&\qquad t(3,2,1,5)\ )
\end{aligned}
$$

Initial inputs repeated.

Progress Using ACL2

Applying ACL2, in an inelegant way with brute force, verifies the following:

- For $2 \le m \le 7$, Bailey's function f of **Conjecture 1** satisfies the m-dimensional tarai recurrence.
  Thus ACL2 verifies **Conjecture 1** for $2 \le m \le 7$.

- For $2 \le m \le 5$, Knuth's version of f computes the same values as Bailey's version of f given in **Conjecture 1**.

Together, these items finish the mechanical verification that Knuth's f satisfies the recursive equation, for $m = 5$ (as well as for $2 \le m \le 4$).

Progress Using ACL2

Version 2.4 Linear Arithmetic Misbehavior

Fixed in Version 2.5.

Note the (INTEGERP FORTH).

```
(THM
 (IMPLIES (AND (INTEGERP THIRD)
               (INTEGERP FORTH)
               (INTEGERP FIFTH)
               (< FORTH THIRD)
               (< FIFTH FORTH)
               (<= (+ -1 THIRD) FORTH)
               (<= (+ -1 FORTH) FIFTH)
               ...

          (EQUAL FIRST
                 (GB (LIST (+ -1 FIRST)
                           SECOND))))
 ... )
```

Progress Using ACL2

Version 2.4 Linear Arithmetic Misbehavior

Now there is no (INTEGERP FORTH).

```
This simplifies, using linear arithmetic ...

Goal'
(IMPLIES (AND (INTEGERP (+ 1 FORTH))
              (INTEGERP (+ 1 FIFTH))
              (INTEGERP FIFTH)
              (< (+ 1 FORTH)(+ -1 FIRST))
              (< (+ 1 FIFTH)(+ 1 FORTH))
              (< (+ 1 FORTH)(+ -1 -1 FIRST))
              (<= (+ -1 1 FORTH)(+ 1 FIFTH))
              ...

         (EQUAL FIRST
                (GB (LIST (+ -1 FIRST)
                          (+ -1 FIRST))))).
```

Progress Using ACL2

Version 2.4 Linear Arithmetic Misbehavior

```
This forcibly simplifies, using linear
arithmetic ...
But simplification reduces this to T ...
q.e.d. (given one forced hypothesis)

Modulo the following forced goal ...
[1]Goal, below, will focus on
(ACL2-NUMBERP FORTH),
which was forced in
 Goal', above,
  by the linearization of
  (EQUAL FORTH (+ 1 FIFTH)).

We now undertake Forcing Round 1.
[1]Goal
(ACL2-NUMBERP FORTH).
******** FAILED ********
```

Progress Using ACL2

ACL2 verifies

- For $2 \le m \le 7$, Bailey's function f of **Conjecture 1** satisfies the m-dimensional *restricted* tarai recurrence:

$$
\begin{array}{l}
f(x_1, x_2, \ldots, x_m) = \\
\quad \text{if} \;\; x_1 \le x_2 \;\; \text{then} \;\; x_2 \\
\quad\quad \text{else} \;\; f(\; f(x_1 - 1, x_2, \ldots, x_m), \\
\quad\quad\quad\quad\quad f(x_2 - 1, x_3, \ldots, x_m, x_1), \\
\quad\quad\quad\quad\quad\quad \vdots \\
\quad\quad\quad\quad\quad f(x_k - 1, x_{k'}, \ldots, x_{k-1}) \;).
\end{array}
$$

- Note k and k' in the last line above.
  k satisfies $x_1 > x_2 > \cdots > x_k \le x_{k'}$.
  $k' = (k+1) \bmod' m$.
  $i \bmod' m = j \in \{1, \ldots, m\} \ni i \equiv j \bmod m$.

- f becomes a "function" with a variable number (at least two) of integer inputs.

Progress Using ACL2

Coping with a variable number of inputs

- The "functions" g, g_b, and the *restricted* tarai recurrence take a variable number of inputs

- Lisp provides an obvious implementation of functions with a variable number of inputs:

  Form the inputs into a list and use that list as the single input to the function.

Progress Using ACL2

ACL2 verifies

- For $2 \le m \le 7$, Bailey's function f of **Conjecture 1** is the *unique* total function on the integers that satisfies the m-dimensional tarai recurrence.

- For $2 \le m \le 7$, Bailey's function f of **Conjecture 1** is the *unique* total function on the integers that satisfies the m-dimensional *restricted* tarai recurrence.

- For $2 \le m \le 7$, the recursive calls on the right side of the m-dimensional *restricted* tarai recurrence, always terminate.
  Thus ACL2 verifies **Conjecture 2** for $2 \le m \le 7$.

Progress Using ACL2

Use `Encapsulate` to consistently axiomatize four functions `tarai`, `tarai-lst`, `rTarai`, and `rTarai-lst` so that for $2 \leq m \leq 7$

- `tarai` is a total function that satisfies the m-dimensional tarai recurrence.

- `tarai` returns an integer whenever the input is a list of integers of length 2 or more.

- `rTarai` is a total function that satisfies the m-dimensional *restricted* tarai recurrence.

The axioms specifically restrict their validity to input lists of lengths 2–7.

Bailey's f is used as the witness,

Progress Using ACL2

- *Uniqueness* is proved by cases.

  One case for each list length from 2–7.

  Induction is used to prove each case.

- The measure of lists of integers $(x_1, x_2, \ldots, x_m)$, used for the induction:

  Based on the lexicographical ordering on pairs $(k, \mathtt{nfix}(x_1 - x_2))$.

  Here k is the integer such that $x_1 > x_2 > \cdots > x_k \leq x_{k'}$.

  k' equals $(k+1) \bmod' m$.

- Use same measure to demonstrate termination of the *restricted* tarai recurrence.

Current Work

Use ACL2 to elegantly prove for all integers $m \geq 2$:

- Bailey's f *uniquely* satisfies the m-dimensional tarai and *restricted* tarai recurrences.

- The recursion always terminates for the *restricted* tarai recurrence.