μ-operator

Canonical name	muoperator
Date of creation	2013-03-22 19:05:47
Last modified on	2013-03-22 19:05:47
Owner	CWoo (3771)
Last modified by	CWoo (3771)
Numerical id	12
Author	CWoo (3771)
Entry type	Definition
Classification	msc 03D20
Synonym	minimization operator
Synonym	minimization-operator
Synonym	search operator
Synonym	search-operator

μ on predicates

Let a property on non-negative integers be given. Informally, the μ-operator looks for the smallest number satisfying the given property. The μ-operator is also known as the (unbounded) minimization operator or the (unbounded) search operator. Formally,

Definition. Let $\Phi(\boldsymbol{x}, y)$ be an $(m+1)$-ary predicate (property) over $\mathbb{N}$, the set of natural numbers (0 included here), with m a non-negative integer ($\boldsymbol{x}$ is m-ary). Define

$$A_\Phi(\boldsymbol{x}) := \{y \in \mathbb{N} \mid \Phi(\boldsymbol{x}, y)\},$$

The *μ-operator* on Φ is given by

$$\mu y \Phi(\boldsymbol{x}, y) := \begin{cases} \min A_\Phi(\boldsymbol{x}) & \text{if } A_\Phi(\boldsymbol{x}) \neq \varnothing, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

The notation $\mu y \Phi(\boldsymbol{x}, y)$ reads "the smallest y such that $\Phi(\boldsymbol{x}, y)$ is satisfied". Note that y is used both as a variable and a number that the variable y represents.

Note that $\mu y \Phi(\boldsymbol{x}, y)$ is a partial function on $\boldsymbol{x}$ (y is a bounded variable). In other words, the μ-operator is a function that takes an $m+1$-ary predicate to an m-ary partial function. When $m = 0$, μ is either an integer, or $\varnothing$.

For example, suppose $\Phi(x, y)$ is the property $(x-y)(x+y) \geq 10$. Then $\mu y \Phi(x, y) = 0$ iff $x \geq 4$, and undefined otherwise.

The reason why the μ-operator is called the search operator comes from recursive function theory. The search for the smallest y such that $\Phi(\boldsymbol{x}, y)$ begins with testing for $\Phi(\boldsymbol{x}, 0)$. If the test fails ($\Phi(\boldsymbol{x}, 0)$ is false), then testing for $\Phi(\boldsymbol{x}, 1), \Phi(\boldsymbol{x}, 2), \Phi(\boldsymbol{x}, 3), \cdots$ are successively performed. The testing stops when a y with $\Phi(\boldsymbol{x}, y)$ is found. This y is also the smallest y satisfying Φ. Nevertheless, the testing can conceivably go on indefinitely, hence the name *unbounded*. There is also a bounded version of μ-operation:

Definition. Let $\Phi(\boldsymbol{x}, y)$ be given as above. Define

$$A_\Phi(\boldsymbol{x}, y) := \{z \in \mathbb{N} \mid \Phi(\boldsymbol{x}, z) \text{ and } z \leq y\}.$$

The *bounded μ-operator* on Φ is given by

$$\mu z \leq y \Phi(\boldsymbol{x}, z) := \begin{cases} \min A_\Phi(\boldsymbol{x}, y) & \text{if } A_\Phi(\boldsymbol{x}, y) \neq \varnothing, \\ y+1 & \text{otherwise.} \end{cases}$$

Thus the bounded μ-operator takes an $(m+1)$-ary predicate (on $(\boldsymbol{x}, z)$, where z is a free variable) to an $(m+1)$-ary total function (on $(\boldsymbol{x}, y)$, as z is now bounded by μ).

μ on total functions

The μ operator can also be defined on functions. We first discuss the case of μ on total functions.

Definition. Given a total $(m+1)$-ary function $f : \mathbb{N}^{m+1} \to \mathbb{N}$, define

$$A_f(\boldsymbol{x}) := \{y \in \mathbb{N} \mid f(\boldsymbol{x}, y) = 0\},$$

The *μ-operator* on f is given by

$$\mu y f(\boldsymbol{x}, y) := \begin{cases} \min A_f(\boldsymbol{x}) & \text{if } A_f(\boldsymbol{x}) \neq \varnothing, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

This definition is actually equivalent to the one regarding predicates, in the following sense: given a total function f with arity $m+1$, define predicate $\Phi_f(\boldsymbol{x}, y)$ of arity $m+1$, as "$f(\boldsymbol{x}, y) = 0$". Then

$$\mu y f(\boldsymbol{x}, y) = \mu y \Phi_f(\boldsymbol{x}, y).$$

Conversely, given an $(m+1)$-ary predicate $\Phi(\boldsymbol{x}, y)$, define an $(m+1)$-ary function $f_\Phi(\boldsymbol{x}, y) := 1 \dot{-} \chi_\Phi(\boldsymbol{x}, y)$, where χ_Φ is the characteristic function of Φ. Then

$$\mu y \Phi(\boldsymbol{x}, y) = \mu y f_\Phi(\boldsymbol{x}, y).$$

Note also that μf may be partial even if f is total, since it is possible $f(\boldsymbol{x}, y) \neq 0$ for all y, and the search will go on indefinitely. For example, let $f(x, z, y) = x^2 + z^2 - y^2$ if $x^2 + z^2 \geq y^2$, and 1 otherwise. Clearly, f is a total function. It is easy to see that $\mu y f(3, 4, y) = 5$, while $\mu y f(1, 2, y)$ is undefined.

μ on partial functions

The definition of the μ-operator on total functions can be generalized to partial functions. However, in recursive functions theory, an additional condition is imposed in order to make the generalization.

Definition. Given a partial $(m+1)$-ary function $f : \mathbb{N}^{m+1} \to \mathbb{N}$, define

$$A_f(\boldsymbol{x}) := \{y \in \mathbb{N} \mid f(\boldsymbol{x}, y) = 0 \text{ and } (\boldsymbol{x}, z) \in \operatorname{dom}(f) \text{ for all } z \leq y\},$$

The *μ-operator* on f is given by

$$\mu y f(\boldsymbol{x}, y) := \begin{cases} \min A_f(\boldsymbol{x}) & \text{if } A_f(\boldsymbol{x}) \neq \varnothing, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

The extra condition that $f(\boldsymbol{x}, z)$ be defined for all $z \leq y$ is needed in order to avoid situations where testing for $f(\boldsymbol{x}, z) = 0$ gets stuck in an infinite loop (in a Turing machine or a URM) because $f(\boldsymbol{x}, z)$ is undefined, before y is ever reached for testing. If we drop this extra condition, then it is possible to find a partial recursive function f such that μf is not recursive.

Remarks.

- Bounded μ-operator may also be defined on functions. In the case of partial functions, the definition can be given as follows: let $f(\boldsymbol{x}, y)$ be an $(m+1)$-ary partial function, with

  $$A_f(\boldsymbol{x}, y) := \{z \in \mathbb{N} \mid f(\boldsymbol{x}, z) = 0, z \leq y, \text{ and } (\boldsymbol{x}, t) \in \operatorname{dom}(f) \text{ for all } t \leq z\},$$

  then

  $$\mu z \leq y f(\boldsymbol{x}, z) := \begin{cases} \min A_f(\boldsymbol{x}, y) & \text{if } A_f(\boldsymbol{x}, y) \neq \varnothing, \\ y+1 & \text{if } A_f(\boldsymbol{x}, y) = \varnothing \text{ and } (\boldsymbol{x}, t) \in \operatorname{dom}(f) \text{ for all } t \leq y \\ \text{undefined} & \text{otherwise.} \end{cases}$$

- Given the set $\mathcal{PR}$ of primitive recursive functions, one obtains the set $\mathcal{R}$ of recursive functions by taking the closure of $\mathcal{PR}$ with respect to the application of the μ-operator. Furthermore, if $f \in \mathcal{R}$, so is $\mu f \in \mathcal{R}$, and it can be shown that any recursive function can be obtained from primitive recursive functions by no more than one application of the μ-operator. This is known as the Kleene normal form theorem.

- With respect to the bounded μ-operator, any primitive recursive function (or predicate) stays primitive recursive after an application of the bounded μ, and any total recursive function stays total after an application of the bounded μ.