Theoretical definitions

Immagine che contiene testo, Carattere, bianco

Descrizione generata automaticamente

The first part requires proving that set difference of recursive sets is recursive, while the second part examines if this property extends to recursively enumerable (r.e.) sets.

Part 1: Proving A \ B is recursive when A and B are recursive

Since A and B are recursive, their characteristic functions χA and χB are computable. We can construct the characteristic function for A \ B as:

χA\B(x) = χA(x) · sg(χB(x))

where sg is the complement sign function (sg(0) = 1 and sg(1) = 0), which we know is computable.

Since χA\B is constructed through composition of computable functions, it is computable. Therefore, A \ B is recursive.

Part 2: Examining if the property extends to r.e. sets

This property does not extend to r.e. sets. We can prove this with a counterexample:

Let A = K (the halting set) and B = N (the set of natural numbers). Both A and K are r.e.:

* K is known to be r.e. but not recursive
* N is recursive (thus also r.e.)

However, A \ B = K \ N = ∅

If K \ N were r.e., then K would be recursive, since: x ∈ K iff x ∉ (K \ N) = x ∈ K

This would mean both K and K are r.e., implying K is recursive. However, this contradicts the known fact that K is not recursive.

Therefore, A \ B is not r.e., providing a counterexample to show that set difference of r.e. sets is not necessarily r.e.

Immagine che contiene testo, Carattere, bianco, ricevuta

Descrizione generata automaticamente

Part 1: If f,g are computable → Qf,g is semidecidable

Since g is computable, there exists e ∈ N such that g = φe. We can express:

Qf,g(x) ≡ "f(x) = g(x)" ≡ "f(x) = φe(x)" ≡ ∃t. S(e,x,f(x),t)

where S(e,x,z,t) is the decidable predicate "φe(x) halts with output z in t steps or less".

Since f is computable and S is decidable, by the Structure Theorem, Qf,g is semidecidable.

Part 2: If Qf,g is semidecidable → f,g computable

The converse does not hold. Consider the predicate:

Qf,g(x) ≡ "φx(x)↑"

where g is not computable but the predicate "g(x)↑" is decidable. By the Projection Theorem, if Qf,g were semidecidable, its negation would also be semidecidable, which would make K = {x | φx(x)↓} recursive - a contradiction.

Therefore, the semidecidability of Qf,g does not imply that both f and g are computable

Immagine che contiene testo, Carattere, bianco, schermata

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, bianco, ricevuta

Descrizione generata automaticamente

Immagine che contiene testo, ricevuta, Carattere, algebra

Descrizione generata automaticamente

Exercise 7.11:

No, such an index x cannot exist. Let's prove this by contradiction.

Suppose there exists x ∈ N such that K̄ = {y² - 1 : y ∈ Ex}. Since K̄ is not recursively enumerable (r.e.), but {y² - 1 : y ∈ Ex} is r.e. (as it's the image of a computable function on an r.e. set), we immediately reach a contradiction.

To see why {y² - 1 : y ∈ Ex} is r.e., observe that:

1) Ex is r.e. by definition (it's the image of a partial recursive function)

2) The function f(y) = y² - 1 is computable

3) The image of an r.e. set under a computable function is r.e.

Therefore, such an index x cannot exist as it would imply K̄ is r.e., which contradicts a fundamental result in computability theory.

Exercise 7.12:

Let's prove both directions of the equivalence.

(⇒) Assume f is computable. Then:

- The function h(x) = π(x,f(x)) is computable as composition of computable functions

- Af = img(h) = {π(x,f(x)) : x ∈ N}

- Since Af is the image of a computable function, it is r.e.

(⇐) Assume Af is r.e. Then:

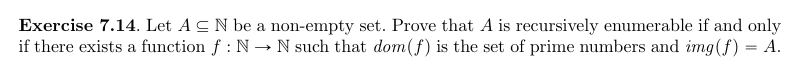
- Let sc\_A be the semicharacteristic function of Af

- We can define f(x) = (μz.∃w. π(x,z) ∈ Af)

- More formally: f(x) = (μz.sc\_A(π(x,z)))₁

- Since π is injective and Af is r.e., this function is computable

- This gives us exactly f(x), as π(x,y) ∈ Af if and only if y = f(x)

Therefore, f is computable if and only if Af is r.e.

(⇒) Assume A is recursively enumerable. Then by definition, there exists a computable function sc\_A that semi-characterizes A. Consider an enumeration of prime numbers P = {p₁, p₂, p₃, ...}. We can define g : N² → N as:

g(x,y) = y·1(sc\_A(x))

This function is computable since sc\_A is computable. By the smn theorem, there exists a total computable function s such that φₛ(ₓ)(y) = g(x,y) for all x,y ∈ N.

Now define f(x) = μy.H(s(y),x,y) where H is the halting predicate. The function f has domain P (the set of prime numbers) and img(f) = A because:

- If x ∈ A then sc\_A(x)↓ so there exists a prime p where f(p) = x

- If x ∈ img(f) then x must be in A since sc\_A(x)↓

(⇐) Assume there exists f with dom(f) = P and img(f) = A. We need to show A is recursively enumerable. Define:

sc\_A(x) = 1(μz.S(e,π₁(z),x,π₂(z)))

where e is an index for f, S is the step predicate, and π₁,π₂ are the projection functions. This function is computable since:

- The step predicate S is decidable

- The projections are primitive recursive

- Minimalization preserves computability

Moreover:

- If x ∈ A then x ∈ img(f) so there exists a prime p where f(p) = x, hence sc\_A(x)↓

- If sc\_A(x)↓ then x must be in img(f) = A

Therefore sc\_A semi-characterizes A, showing A is recursively enumerable.

Not computable functions

Immagine che contiene testo, Carattere, bianco, ricevuta

Descrizione generata automaticamente

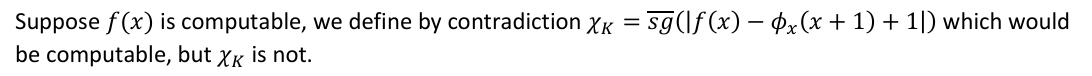


Immagine che contiene testo, ricevuta, Carattere, bianco

Descrizione generata automaticamente

Assume, for contradiction, that f is computable. Then we can define:

χ\_T(x) = sg(|f(x) - (φ\_x(x) + 1)|)

where T = {x | φ\_y(y)↓ for each y ≤ x} is the set of indices where all smaller indices halt.

However, χ\_T would correctly characterize T since:

* If x ∈ T, then φ\_y(y)↓ for all y ≤ x, so f(x) = φ\_x(x) + 1, thus χ\_T(x) = 1
* If x ∉ T, then there exists y ≤ x where φ\_y(y)↑, so f(x) = 0, and thus χ\_T(x) = 0

But T is not recursive (this can be proven by reduction from K). Therefore χ\_T cannot be computable, contradicting our assumption that f was computable.

Therefore, f is not computable.

Assume, for contradiction, that there exists a non-computable function f : N → N such that D = {x ∈ N | f(x) ≠ φ\_x(x)} is finite.

Let D = {x₁, x₂, ..., x\_n} be this finite set. Since D is finite, we can define a new function g : N → N as:

g(x) = {

φ\_x(x) + 1 if x ∈ D

f(x) if x ∉ D

}

This function g would be computable because:

1) For x ∈ D (which is finite), we can compute φ\_x(x) + 1 using the universal function

2) For x ∉ D, we know that f(x) = φ\_x(x), so g(x) = φ\_x(x)

3) Testing if x ∈ D is decidable since D is finite

However, by construction:

- For x ∈ D: g(x) = φ\_x(x) + 1 ≠ φ\_x(x)

- For x ∉ D: g(x) = f(x) = φ\_x(x)

Therefore g = f, which means f would be computable, contradicting our initial assumption.

Therefore, no such non-computable function f can exist where D is finite.

Immagine che contiene testo, Carattere, ricevuta, bianco

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, bianco, ricevuta

Descrizione generata automaticamenteSmn-theorem

First, let me state the s-m-n theorem (smn theorem):

For any m,n ≥ 1, there exists a total computable function s\_m,n : N^(m+1) → N such that for all e ∈ N, x⃗ ∈ N^m, y⃗ ∈ N^n:

φ^(m+n)\_e(x⃗,y⃗) = φ^(n)\_(s\_m,n(e,x⃗))(y⃗)

Now, for the proof that there exists a total computable function s: N² → N such that |W\_s(x,y)| = x\*y:

Let's define an auxiliary two-argument computable function g(x,y,z):

g(x,y,z) = {

0 if z < x\*y

↑ otherwise

}

This function is computable since it can be written as:

g(x,y,z) = 0 + μw.(z + 1 - x\*y)

By the smn theorem, there exists a total computable function s : N² → N such that for all x,y,z ∈ N:

φ\_s(x,y)(z) = g(x,y,z)

Therefore:

- For any x,y ∈ N, W\_s(x,y) = {z | g(x,y,z)↓} = {z | z < x\*y}

- Thus |W\_s(x,y)| = x\*y

This function s has the required properties:

1) It is total and computable (by the smn theorem)

2) For all x,y ∈ N, |W\_s(x,y)| = x\*y (by construction)

Therefore, we have proven the existence of the required function s.

Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamente

Immagine che contiene testo, schermata

Descrizione generata automaticamente

First, let me state the s-m-n theorem:

For any m,n ≥ 1, there exists a total computable function s\_m,n : N^(m+1) → N such that for all e ∈ N, x⃗ ∈ N^m, y⃗ ∈ N^n:

φ^(m+n)\_e(x⃗,y⃗) = φ^(n)\_(s\_m,n(e,x⃗))(y⃗)

Now, for the proof that there exists a total computable function k: N → N such that φ\_k(x)(y) = lcd(x,y):

Let's define a computable function g(x,y) that computes the least common multiple:

g(x,y) = μz ≤ x\*y.(x|z ∧ y|z)

This function is computable since:

1) The "divides" relation (x|z) is decidable

2) The bound x\*y ensures the minimalization is bounded

3) The composition of these operations preserves computability

By the smn theorem with m=1 and n=1, there exists a total computable function k : N → N such that for all x,y ∈ N:

φ\_k(x)(y) = g(x,y)

Therefore:

1) k is total and computable by the smn theorem

2) For all x,y ∈ N, φ\_k(x)(y) = g(x,y) = lcd(x,y) by construction

3) k satisfies the required property φ\_k(x)(y) = lcd(x,y)

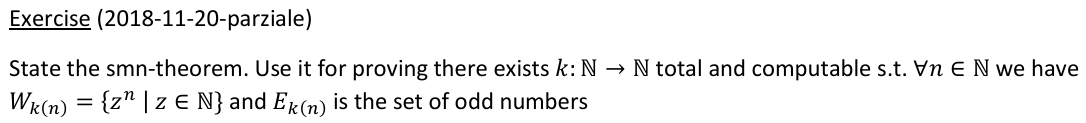
This completes the proof of the existence of the required function k.

Immagine che contiene testo, Carattere, ricevuta, bianco

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamentePrimitive recursion

Immagine che contiene testo, schermata, Carattere, algebra

Descrizione generata automaticamente

Immagine che contiene testo, schermata, Carattere, documento

Descrizione generata automaticamente

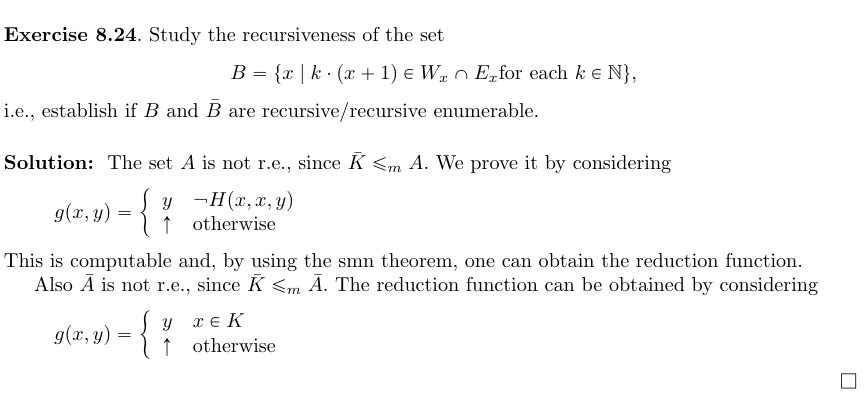
Recursiveness

Immagine che contiene testo, ricevuta, Carattere, bianco

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamente

Exercise 8.37:

First, let's prove that V is not recursive by showing that K ≤m V.

To establish this reduction, let's define function g(x,y):

g(x,y) = {

1 if x ∈ K

↑ otherwise

}

This function is computable since g(x,y) = scK(x). By the s-m-n theorem, there exists a total computable function s: ℕ → ℕ such that φs(x)(y) = g(x,y) for all x,y ∈ ℕ.

We can show s is a reduction function from K to V:

1) If x ∈ K, then φs(x)(y) = 1 for some y. Therefore Ws(x) = {1}, and 1 = 1·s(x), so s(x) ∈ V.

2) If x ∉ K, then φs(x)(y)↑ for all y. Therefore Ws(x) = ∅, and there are no y,k such that y = k·s(x), so s(x) ∉ V.

Since K ≤m V and K is not recursive, V is not recursive.

Additionally, V is recursively enumerable since:

scV(x) = 1(μw.(H(x,(w)1,(w)3) ∧ (w)2·x = (w)1))

Therefore V is r.e. but not recursive, so V̄ is not r.e.

Exercise 8.38:

This set is saturated since V = {x | φx ∈ A} where A = {f | |dom(f)| > 1}.

We can use Rice-Shapiro theorem to prove both V and V̄ are not r.e.:

1) V is not r.e.:

Consider id ∉ V and θ ⊆ id where θ(x) = x for x ≤ 1. Then θ ∈ V since |dom(θ)| = 2 > 1, and θ is finite. By Rice-Shapiro theorem, V is not r.e.

2) V̄ is not r.e.:

Consider the constant function 1 ∉ V̄ (since its domain is infinite). Let θ ⊆ 1 where θ(0) = 1. Then θ ∈ V̄ since |dom(θ)| = 1, and θ is finite. By Rice-Shapiro theorem, V̄ is not r.e.

Immagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteTherefore, V is neither recursive nor r.e., and V̄ is not r.e.

Immagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteImmagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamente

Immagine che contiene testo, Carattere, schermata, documento

Descrizione generata automaticamenteImmagine che contiene testo, ricevuta, Carattere, schermata

Descrizione generata automaticamenteSecond recursion theorem

Immagine che contiene testo, ricevuta, Carattere, algebra

Descrizione generata automaticamente