Modelado didáctico con el lenguaje de programación MateFun

Manuela Cabezas[1] and Sylvia da Rosa[2]

[1] Facultad de Ciencias de la Educación, Universidad de la Empresa,
mcabezas@ude.edu.uy
[2] Instituto de Computación, Facultad de Ingeniería, Universidad de la República
darosa@fing.edu.uy

Resumen En este artículo se presentan los principales resultados del trabajo de operacionalización de nuestro modelo didáctico para ciencias computacionales. El modelo fue elaborado durante el proyecto de investigación **El paradigma de las ciencias computacionales y la educación** entre los años 2021 y 2023, donde se realizaron estudios de aula junto a profesores y estudiantes de física e informática de educación media y bachillerato para identificar las bases didácticas para la integración de la computación en la enseñanza de ciencias en secundaria. Aquí se discute la etapa de operacionalización del modelo, centrada en la relación entre el modelo matemático y el modelo computacional para fenómenos/sistemas abordados en clases de ciencias experimentales. En esta etapa fueron diseñadas y testeadas varias aplicaciones concretas del modelo y durante el análisis se identificaron algunos aportes específicos: la alfabetización computacional de profesores y estudiantes desarrollada como alfabetización del área de dominio específico (física) y el vínculo entre el pensamiento computacional y las áreas de dominio con fundamento científico-matemático. Se propone una etapa futura de aplicación con instrumentos de evaluación de aprendizajes para validar resultados.

Palabras clave: didáctica, modelado, ciencias computacionales.

Abstract This article presents the key findings from the operationalisation stage of our didactic model for computational science. The model was developed during the research project **The Paradigm of Computational Science and Education** between 2021 and 2023. Classroom studies were conducted with secondary and upper secondary school physics and informatics teachers and students to identify the foundations for the didactic integration of computing into science classrooms. Here, we discuss the operationalisation stage of the model, which focuses on the relationship between the mathematical model and the computational model of phenomena/systems studied in science classrooms. During this stage, several applications of the model were designed and tested, and during the analysis, specific benefits were identified: the computational literacy of teachers and students developed as domain-specific literacy (physics) and the connections between computational thinking and domains with a scientific-mathematical foundation. A future implementation stage with learning assessment instruments to validate results is proposed.

Key words: didactics, modelling, computational sciences.

1. Introducción

Durante el proyecto de investigación 'El paradigma de las ciencias computacionales y la educación', llevado a cabo entre los años 2021 y 2023, se realizó un análisis didáctico del pensamiento computacional que tuvo como uno de sus resultados más interesantes la elaboración de un modelo didáctico para ciencias computacionales. Es decir, desarrollamos teoría didáctica para la introducción de la computación en otros dominios, fuera del área informática, desde las bases de la computación. En este paper, describimos cómo el trabajo conjunto con profesores de física y de informática nos permitió entender cómo se relacionan y se complementan los aspectos didácticos de cada disciplina dando lugar a soluciones para los desafíos que enfrentan las didáctica de las ciencias computacionales (Cabezas y da Rosa, 2022; Denning y Tedre, 2019).

El grupo director del proyecto estaba formado por investigadores en informática, una investigadora en didácticas específicas, un profesor de informática y una profesora de matemática (ambos de Enseñanza Secundaria). El grupo de profesores de aula estuvo conformado por tres profesores de Informática, dos con grupos del ciclo básico de liceos públicos del departamento de Canelones y una profesora de Montevideo con un grupo del ciclo superior opción educación técnica profecional (público); y dos profesores de física de ciclo superior opción bachillerato, del liceo público del departamento de Colonia. En Uruguay la educación secundaria se organiza en dos ciclos: básico y superior, cada uno con tres años de duración. Comienza luego de la etapa de educación primaria, cuando los estudiantes tienen alrededor de 12 años de edad. El ciclo básico provee una formación general con materias comunes a todos los estudiantes, mientras que en el ciclo superior los estudiantes pueden elegir entre bachillerato que los orienta y prepara para distintas carreras universitarias o educación técnica profesional que los orienta para el mercado laboral (aunque también pueden ingresar a carreras universitarias).

El proceso de modelado se basó en el análisis didáctico y elementos teóricos tanto los que aportamos (modelo epistemológico) como los que tomamos de la revisión bibliográfica (ideas fundamentales de la computación, diseño centrado en la teoría) se describen detalladamente en (Cabezas y da Rosa, 2022). Uno de los principales resultados de dicho proceso es un modelo didáctico para ciencias computacionales, que se describe en este artículo, ejemplificando los elementos del modelo con un ejercicio de física en la Sección 2.

Se discute la etapa de operacionalización del modelo, centrada en definición de la relación entre el modelo matemático y el modelo computacional para fenómenos/sistemas abordados en clases de ciencias experimentales. En esta etapa fueron diseñadas y testeadas varias aplicaciones concretas del modelo, de las cuales se presenta una en la Sección 3.1. Durante el análisis se identificaron algunos aportes específicos: la alfabetización computacional de profesores y estudiantes desarrollada como alfabetización del área de dominio específico y el vínculo entre el pensamiento computacional y las áreas de dominio con fundamento científico-matemático.

Uno de los problemas que intentamos atender y al que apuntan otros autores, es la falta de relación entre investigación didáctica y práctica, que en parte surge del sub-valorado rol de las investigaciones didácticas dentro de la academia (Ametller y col., 2007). Teniendo en cuenta esa carencia, el equipo pasó a integrar profesores de secundaria a fines de 2022 y nos basamos en la tradición de la investigación acción, que lleva muchos varios años desarrollando metodologías centradas en un profundo intercambio académico entre investigadores y profesores de aula (Somekh y Noffke, 2009). El proceso de modelado didáctico consistió en combinar el aporte de los profesores, sus ideas y experiencias con las elaboraciones teóricas y articularlo con la metodología de investigación didáctica que el grupo director del proyecto venía elaborando (Cabezas y da Rosa, 2022).

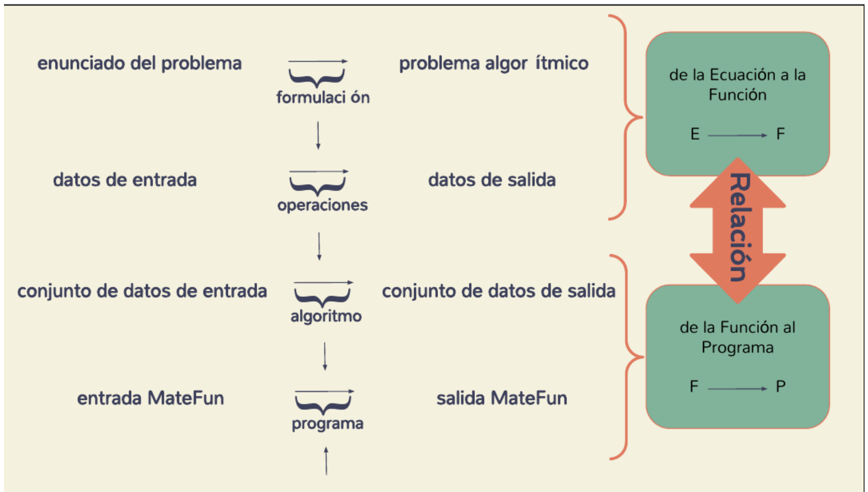
Para alcanzar los objetivos generales del proyecto (básicamente identificar y desarrollar las competencias y destrezas computacionales (pensamiento computacional) en la didáctica de cada disciplina científica), era necesario que cada profesor/a elaborase una secuencia didáctica para trabajar en sus aulas, de acuerdo a los resultados de las investigaciones teóricas y sus resultados que se describen en (Cabezas y da Rosa, 2022). El diseño de las secuencias didácticas se trabajó en conjunto con los profesores de física y de informática de Enseñanza Secundaria y los investigadores de la Universidad. Se propone una etapa futura de aplicación con instrumentos de evaluación de aprendizajes para validar resultados.

Las secciones de este artículo se organizan como sigue: en la Sección 2 se describe el modelo didáctico y se ejemplifica su aplicación con la solución detallada de un ejercicio de física. En la Sección 4 se describen las características principales del lenguaje de programación funcional MateFun usado en el modelo, en la Sección 3 se describe un ejemplo de operalización del modelo, realizada en una pasantía en 2024 con estudiantes de física de Bachillerato. Finalmente se incluye algunas conclusiones en la Sección 5.

2. El modelo didáctico

En esta sección describimos el modelo didáctico, que se ilustra en la Figura 1. El modelo provee a los profesores de ciencias de una herramienta para introducir en la didáctica de su disciplina ideas fundamentales de computación y desarrollar en los estudiantes competencias computacionales consideradas transversales en la formación de los jóvenes, tanto por el documento de la Comisión Europea (Commission y col., 2022 (anexo 2, página 103)) como por el (Marco Curricular Nacional ANEP, 2022, (página 42)). El punto de partida de la investigación fue un análisis de varias propuestas para definir las ideas fundamentales de la

Figura 1: Modelo didáctico.



computación (Cabezas y da Rosa, 2022) pudiendo abordar en los estudios de aula desde las siguentes tres ideas:

1. La información se representa en forma digital.
2. Los algoritmos interactúan con los datos para resolver problemas computacionales.
3. Los programas expresan algoritmos y datos en una forma que se puede implementar en una computadora.

Estas ideas también se vieron reflejadas en los marcos de competencias computacionales básicas y necesarias para las ciencias computacionales de estudios comparados internacionales (Commission y col., 2022). En particular, para el proceso de modelado se consideró:

- Datos e Información
- Modelado y Simulación
- Algoritmos y Programas

Explicamos cada uno de los elementos que aparece en la Figura 1 usando un ejemplo de física trabajado con los profesores de Enseñanza Secundaria que participaron del proyecto *El paradigma de las ciencias computacionales y la educación.*

2.1. De la ecuación a la función

La primera competencia computacional tiene que ver con los enunciados de los problemas donde la información debe ser transformada en datos, que permitan formularlos como problemas algorítmicos, como se describe en (Cabezas y da

Rosa, 2022). Partimos del enunciado de un ejercicio planteado por los profesores de física para el tema Campo Eléctrico:

> Una pequeña esfera posee carga positiva `q=3,0nC`. Calcula y representa el campo eléctrico que crea en un punto a `2,0cm` hacia su derecha.

El primer renglón de la Figura 1 sintetiza el siguiente análisis: el enunciado contiene información sobre valores de carga y distancia específicos, que deben ser usados para obtener las soluciones de dos problemas.

El primer problema es el cálculo del valor del campo eléctrico para la carga y la distancia dada, donde los datos de entrada son la carga (`q` de `3,0nC`), la distancia de (`2 cm`) y la constante de Coulomb `k`. El dato de salida es el par formado por el **valor** del campo eléctrico y la correspondiente unidad de medida. Acá aparece un subproblema que es determinar la **unidad** del campo elećtrico. Del intercambio con los profesores surgió que según el sistema internacional de medidas, la constante `k` utiliza Coulomb como unidad de carga y metros para la distancia, por lo tanto es necesario convertir las unidades dadas (nC y cm).

El segundo problema es la representación del vector del campo en el punto especificado. Los datos de entrada son el campo eléctrico calculado en el primer problema y la distancia de `2 cm` **hacia su derecha**. El dato de salida es la representación del vector, para lo cual se plantea el subproblema de determinar su módulo, dirección y sentido. Por razones de espacio, incluimos solamente el modelado computacional de la solución del primer problema (cálculo del valor del campo eléctrico). El segundo problema será presentado en otra publicación destinada a la *18th International Conference on Informatics in Schools (ISSEP)* que tendrá lugar en setiembre de 2025.

La solución planteada por la profesora para el cálculo del campo eléctrico consiste en sustituir los valores dados de la carga `q`, la distancia `r` y la constante de Coulomb `k` en la ecuación para el campo eléctrico E, como se muestra en la Figura 2. La ecuación se resuelve aplicando las operaciones matemáticas correspondientes a los datos de entrada `q, r y k` para obtener el valor deseado de 67500. La solución al subproblema de determinar las unidades del campo eléctrico, está implícito en la solución de la profesora: el dato de entrada `q 3,0nC` aparece en la ecuación como 3×10^{-9} `C` y la distancia de `2cm` aparece como `0,02m` como se ve en la Figura 2.

En este punto surgió una de las bases didácticas identificadas para la integración de la computación en la enseñanza de las ciencias a nivel de Enseñanza Secundaria: la necesidad de volver explícitos los elementos implícitos de los problemas, ya que si una solución debe ser expresada como un programa a ejecutar es indispensable resolver rigurosmante los subproblemas que surgen del análisis del enunciado. Esto contribuye a comprender que los conceptos de la física son n-uplas formadas por el valor real y las unidades correspondientes. Por lo tanto, es necesario diseñar un algoritmo que toma las unidades de los datos de entrada del enunciado y las devuelve convertidas en las unidades correspondientes para producir el valor de salida `E` en la unidad de carga Coulomb `(C)`. En la Figura 3 se muestra la definición de los conjuntos de unidades y las funciones de conversión de unidades (por la similitud con la notación matemática, usamos

$$E = \frac{k.q}{r^2}$$

$$k = 9 \cdot 10^9 \frac{N \cdot m^2}{C^2}$$

(a) Ecuación (b) Constante

$$E = \frac{k.q}{r^2} = \frac{9 \times 10^9 \, N.m^2/C^2 \, . \, 3 \times 10^{-9} C}{(0{,}02m)^2} = 67500 \, N/C$$

(c) Caso Concreto

Figura 2: Cálculo del campo eléctrico para el caso concreto

el lenguaje MateFun). La ecuación soluciona el caso concreto planteado en el ejercicio así como cualquier otro caso concreto que se plantee, como se muestra en el Cuadro 1. Si agrupamos los datos de entrada en un conjunto de un número (potencialmente infinito) de n-uplas (primera columna del cuadro) y aplicamos la ecuación para cada una obtenemos un conjunto (potencialmente infinito) de datos de salida (tercera columna del cuadro). La columna del medio que es el conjunto de operaciones de la ecuación, converge hacia una función que implementa el algoritmo solución del problema **para cualquier dato de entrada**, como se describe en el último renglón del Cuadro 1.

Cuadro 1: De las ecuaciones a la función

datos de entrada	**algoritmo**	**datos de salida**
((3,NanoCoulomb), (2,Cm))	ecuación	(67500,Newton,Coulomb)
((3,NanoCoulomb), (0.3 Mts))	ecuación	(300,Newton,Coulomb)
((10,MicroCoulomb), (1, Mts))	ecuación	(90000,Newton,Coulomb)
((5,MicroCoulomb), (10,Cm))	ecuación	(4500000,Newton,Coulomb)
((0.5, Coulomb), (800, Cm))	ecuación	(70312500,Newton,Coulomb)
...	ecuación	...
para toda (carga, distancia)	**función**	(CElectrico, Unidades)

2.2. De la función al programa

El potencial de la solución computacional se manifiesta cuando transformamos el algoritmo en un programa y lo ejecutamos para cualquiera de las n-uplas

```
conj UCarga = { Coulomb , MicroCoulomb, NanoCoulomb }
conj UDistancia = { Km, Mts, Millas, Cm }
conj UFuerza = { Newton }

convertUCarga :: R X UCarga -> (R X UCarga)
convertUCarga (c,u) = (c * 0.000000001, Coulomb)
                       si u == NanoCoulomb
                    { (c * 0.000001, Coulomb)
                       si u == MicroCoulomb
                    { (c, Coulomb)

convertUDistancia :: R X UDistancia ->(R X UDistancia)
convertUDistancia (d,u) = (d * 0.01, Mts)
                              si u == Cm
                        { (d * 1000, Mts)
                              si u == Km
                        { (d * 1609.34, Mts)
                              si u == Millas
                        { (d, Mts)
```

Figura 3: Definiciones de los tipos de datos para las unidades de carga, de distancia y de fuerza. Definiciones de las funciones de conversión de unidades.

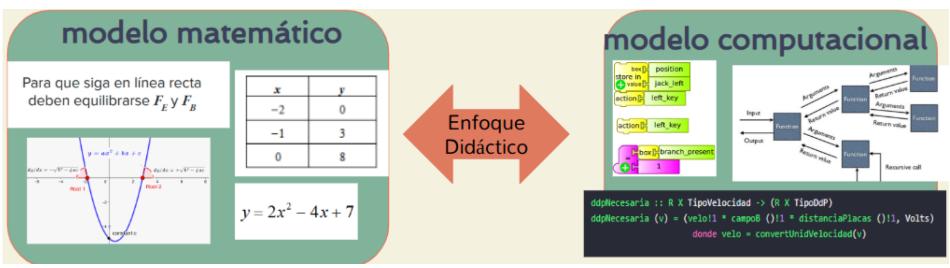
del conjunto de datos entrada obteniendo las soluciones de cada caso concreto (cuarto renglón de la Figura 1). De esta manera los estudiantes pueden experimentar su solución en "acción", lo que les permite obtener resultados en forma rápida, corregir eventuales errores y/o mejorar alguna parte del programa. La implementación en MateFun se muestra en la Figura 4.

Como puede apreciarse, el enfoque didáctico está centrado en la relación entre el modelado matemático (el algoritmo) y el modelado computacional (el programa) para fenómenos estudiados en las aulas de disciplinas científicas, como se ilustra en la Figura 5. Esta relación entre ambos modelos puso al descubierto que varias dificultades en el aprendizaje de matemática constituyen también obstáculos para el desarrollo de competencias computacionales. Algunos tópicos en particular están estrechamente asociados a representaciones tanto matematicas como computacionales. Uno de ellos es el concepto de función que admite varias representaciones, por ejemplo, expresiones algebraicas, tablas y gráficas cartesianas. El enfoque del modelo didáctico que describimos en este trabajo se basa en la relación entre las representaciones mencionadas y la representación computacional del concepto de función como solución de un problema algorítmico.Si bien el algoritmo está presente en el modelo matemático, lo que lo convierte en computacional es su implementación en un programa, donde aparecen aspectos críticos propios, por ejemplo la discretización de objetos infinitos que

```
{- Defino la constante electrica como funcion de dominio
 vacio. -}

constElec :: ()->(R X UFuerza X UDistancia X UCarga)
constElec() = (9000000000, Newton, Mts, Coulomb)

{- Defino la funcion para calcular el campo electrico -}

campoelec :: ((R X UCarga) X (R X UDistancia))
          -> (R X UFuerza X UCarga)

campoelec (carga,distancia) =
 (constElec()!1 * cargaN!1/(distanciaN!1^2, Newton,
                                        Coulomb)
       donde cargaN = convertUCarga (carga)
         distanciaN = convertUDistancia (distancia)
```

Figura 4: Definiciones de la constante de Coulomb y la función campo eléctrico.

Figura 5: Esquema del enfoque didáctico.



ha tenido un gran impacto tanto en el trabajo científico en física como en la educación en la disciplina. Describimos este punto en la siguiente sección.

3. Operacionalizando el modelo

Una vez finalizado el proyecto nos hemos dedicado a la etapa de operacionalización del modelo. En esta etapa fueron diseñadas varias aplicaciones concretas del modelo y durante el análisis didáctico se identificaron aportes específicos del modelo: contribución a la alfabetización computacional de profesores y estudiantes y el abordaje explícito de aspectos críticos que representan dificultades en la enseñanza de disciplinas científicas.

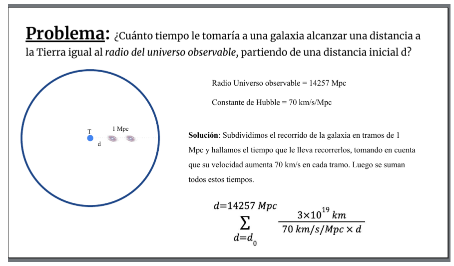
Estos aportes permiten abarcar nuevos problemas que eran inabordables en los cursos de Enseñanza Secundaria por su alta complejidad matemática. Un ejemplo clásico es el estudio del movimiento de proyectiles al que se suele dedicar tiempo en los cursos de Enseñanza Secundaria o los introductorios de física universitaria. Los movimientos de proyectil se modelan como la composición de dos movimientos, uno con velocidad constante en la dirección horizontal y otro con aceleración constante en la dirección vertical donde tanto la posición del proyectil como su velocidad en cada instante de tiempo se pueden describir con funciones de primero o de segundo grado, accesibles a estudiantes de Enseñanza Secundaria. Pero esta forma de modelar los movimientos parte de la premisa de que se desprecia la fuerza de rozamiento con el aire lo cual no es válido en la gran mayoría de las situaciones reales. Obtener una expresión analítica que describa la posición o la velocidad en función del tiempo incluyendo a la fuerza de rozamiento en el modelo, es un problema complejo y fuera del alcance de los cursos de Enseñanza Secundaria o universitarios iniciales. Los cursos avanzados de física, requieren el estudio de temas de matemática continua (como ecuaciones diferenciales e integrales) para dominar los conceptos necesarios para estudiar los movimientos considerando el rozamiento. Sin embargo, en el paradigma de las ciencias computacionales, este tipo de problemas y soluciones se puede modelar de una manera sencilla usando matemática discreta y programación lo cual permite incluirlos en cursos de Enseñanza Secundaria. Para el caso del estudio del movimiento de los proyectiles existen varios algoritmos que se pueden utilizar, por ejemplo Mark Newman en Newman, 2013 (cap.8, pag 332) utiliza el más sencillo (pero no más eficiente) que es el método de Euler. Podemos decir que la idea principal básica consiste en dividir los intervalos de tiempo en porciones suficientemente pequeñas y programar un algoritmo para que el computador realice los cálculos correspondientes.

En la Sección 3.1 mostramos un ejemplo desarrollado en una de las instancias de operalización se nuestro modelo que evidencia el impacto del uso de la discretización en la resolución de ejercicios en cursos de Enseñanza Secundaria.

3.1. Ley de Hubble

En esta sección se describe mediante un ejemplo sobre la ley de Hubble, el impacto del uso de matemática discreta y programación en el contenido de cursos de física de Enseñanza Secundaria. El ejemplo fue propuesto y desarrollado por una profesora de Física y un grupo de estudiantes de bachillerato que participaron en 2024 de un taller que propuismos para la etapa de operalización del modelo didáctico usando MateFun. La Ley de Hubble establece una relación entre los corrimientos al rojo (`z`) de astros lejanos y su distancia a la Tierra (`d`). Como `z` determina a su vez la velocidad relativa entre fuente y observador, una expresión alternativa de la Ley es la ecuación que se muestra en la parte (a) de la Figura 6, donde `v` está en km por segundo, `d` en Mpc (Mpc, un millón de pársecs) y la constante de Hubble (`Ho`) es de 70 km por segundo por Mpc. Esta relación entre entre `v` y `d` es lineal para valores de $z \ll 1$ (astros relativamente cercanos cuyas distancias y velocidades no cambian mucho con el tiempo), apareciendo para $z > 1$ términos de segundo orden por variación de `Ho` y la expansión acelerada del Universo. En este trabajo asumimos una relación lineal entre `v` y `d` para todos los valores de distancia posibles (desde 0 hasta el radio

(a) Ley de Hubble



(b) Problema

Figura 6: Expansión del Universo

del Universo observable (Robs), -cuyo aumento en el tiempo se desprecia-), para obtener una primera estimación del tiempo que le tomaría a una galaxia alcanzar Robs partiendo de una distancia inicial dada. Como se describe en la Sección 3 en física computacional se puede elaborar soluciones a problemas que o bien no puedan expresarse analíticamente o bien se pueda hacer pero recurriendo a conceptos complejos que los hace inaccesibles para cursos de Enseñanza Secundaria. En este caso la solución computacional consiste en dividir un intervalo entre dos distancias `[d1, d2]`, en tramos de, por ejemplo un Mpc, obteniendo una secuencia de tramos (línea 36 en Figura 7). Si consideramos que un Mpc es una distancia de $3e19\ km$, el tiempo en segundos que lleva recorrer cada tramo es $3e19/(Ho * d)$, donde el denominador es la velocidad según la ley de Hubble (parte (a) de la Figura 6). La sumatoria de la parte (b) de la Figura 6 se define como una función `tiempo` que toma la secuencia de tramos del intervalo `[d1, d2]` y suma los tiempos de recorrida de cada tramo, convirtiéndolos en años (línea 48 en Figura 7), donde el factor de conversión se define localmente como la constante `anios`. Con ayuda de la profesora, los estudiantes elaboraron la solución analítica del problema, (que utiliza integrales tema que no está incluido en el currículo), para comprobar que los resultados matemático y computacional coinciden dentro de un margen de error aceptable (recuadrado en la última línea de la Figura 8 y línea 50 de la Figura 7). Cabe señalar que durante la etapa de operalización del modelo en 2024 tuvimos la oportunidad de validarlo con profesores de Física, Matemática, Informática y Química, produciendo numerosos ejemplos que por falta de espacio no incluimos aquí.

4. El lenguaje MateFun

En nuestros ejemplos usamos MateFun, un lenguaje de programación funcional pura diseñado para introducir el modelado computacional en aulas de Enseñanza Secundaria. A diferencia de otros lenguajes (como Python por ejemplo), MateFun permite definir tipos de datos como conjuntos, implementar las funciones incluyendo dominio y co dominio, graficar y construir figuras y animaciones. Estas funcionalidades hacen que la relación entre el modelado matemático y el

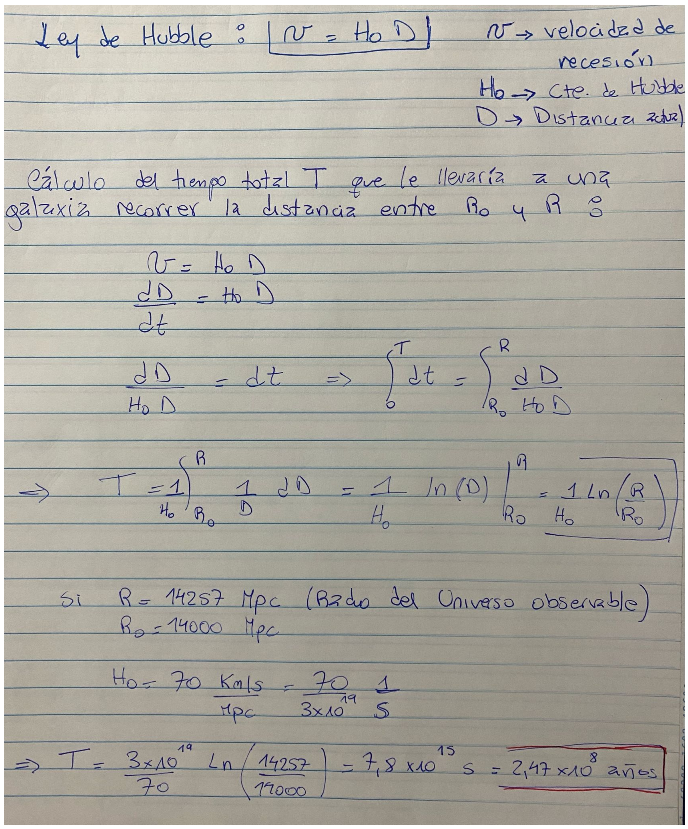
```
{- La funcion predefinida rango, toma el comienzo, el
fin y la distancia entre los elementos de una secuencia.
 -}

secuenciaDist :: Z X Z -> Z*
secuenciaDist (d1,d2) = rango (d1,d2,1)

{- Definimos la funcion tiempo para la sumatoria de los
tiempos transcurridos en cada tramo del intervalo de
distancias, contenido en la secuencia secD.
-}

conj UTiempo = { Anios }

tiempo :: Z* -> R X UTiempo
tiempo (secD)
    = (0, Anios) si secD==[]
    {((3e19/(constHubble() * (primero(secD))))/anios
                          + tiempo(resto(secD))!1, Anios)
                                donde anios = 365*24*60*60

{- Prueba para una distancia inicial de 14000 Mpc
Hubble>tiempo(secuenciaDist(14000,14257))
(248171740.4484,Anios)
-}
```

Figura 7: Funciones para calcular el tiempo que le tomaría a una galaxia alcanzar una distancia a la Tierra igual al radio del universo observable, partiendo de una distancia inicial.

modelado computacional sea explícita y fácilmente reconocida y comprendida por los estudiantes, como se puede apreciar en los fragmentos de soluciones en MateFun incluidos en este artículo. También resaltamos que, si bien los ambientes de visualización matemática como Geogebra sí se basan en los fundamentos y estructuras de la matemática al igual que MateFun, éstos no permiten discernir los conceptos computacionales claves que entendemos son la base de la alfabetización científico-computacional que presentamos en este artículo. Queremos destacar la diferencia de este concepto con el concepto de alfabetización digital-computacional que a veces se asocia con los ambientes de visualización matemática.

Otros autores han destacado la importancia de contar con lenguajes de programación de propósito específico (TSL por su sigla en inglés) (Guzdial, 1994), pudiendo atender los objetivos y requisitos del análisis y modelo didáctico. Entendemos que MateFun ofrece un lenguage particulatmente apto para la introducción del modelado computacional para poblaciones sin dominio informático.

Figura 8: Solución analítica del problema de la expansión del Universo.



Resulta sencillo introducir MateFun en las clases de ciencias desde una concepción amplia de la alfabetización científico-computacional de docentes y estudiantes de secundaria (diSessa, 2000, 2018), como hemos podido comprobar en las pasantías, cursos y talleres que venimos realizando.

MateFun fue diseñado en 2017 por investigadores del Instituto de Computación (InCo) de la Facultad de Ingeniería (FING) de la Udelar (Uruguay) y sus aplicaciones didácticas en cursos de matemáticas datan de 2018 (da Rosa y col., 2020). Se puede acceder al lenguaje MateFun a través de un entorno de programación web integrado[3] que permite la gestión, construcción y ejecución de programas, así como la visualización de gráficos, figuras y animaciones.

Considerando que el público objetivo está compuesto por estudiantes de Enseñanza Secundaria de habla hispana, las palabras clave se definieron en español, pero es posible configurarlo para otros idiomas. Actualmente se cuenta con una versión en inglés.

[3] https://www.fing.edu.uy/proyectos/matefun/#/es/login

5. Conclusiones

Los aportes de las intervenciones de aula que se realizaron durante el proyecto fueron fundamentales y resaltaron la importancia del intercambio entre profesores e investigadores. Especialmente en el caso de física, comprobamos que el impacto de la computación en la didáctica de las ciencias, no sólo aporta rigurosidad, orden y comprensión profunda de los conceptos, sino que deja en evidencia la necesidad de transformar los contenidos para la era actual de la ciencia computacional. Este tema no es menor porque en el contexto actual de reformas educativas, muchas veces el análisis de los contenidos, sus bases y especifidad epistemológica, así como los avances desde los estudios de investigaciones didácticas, quedan al margen de otras necesidades y agendas de políticas educativas. Sin embargo, es un error pensar que podemos desconectar los contenidos y su naturaleza del propio proceso de enseñarlos y aprenderlos. Las ciencias experimentales-computacionales tienen características propias que las enriquecen. La revolución computacional en la ciencia, ha ampliado los límites del conocimiento y también obliga a reformular los contenidos educativos y adaptarlos para abarcar las fronteras de ese conocimiento. El problema es complejo y su solución debe abordarse desde la perspectiva del trabajo conjunto entre la academia y la práctica profesional (Cabezas, 2021). El trabajo que venimos desarrollando desde 2020 en torno a la didáctica de las ciencias computacionales aspira a constribuir a superar las dificultades en la introducción de la computación como disciplina científica básica en la educación. El mayor potencial de nuestra metodología es la colaboración entre la academia y la profesión donde el diálogo permite profundizar la discusión disciplinar, sobre dominios, métodos y prácticas. En este sentido, entendemos que la base de la mejora de la enseñanza son los procesos colaborativos que reúnen a profesores e investigadores desde el análisis didáctico. Éstos procesos permiten desarrollar y validar constructos (teoría didáctica) desde la práctica, como el modelo que presentamos aquí, pero también productos, como el lenguaje de programación funcional pura MateFun que fue desarrollado específicamente desde el propósito didáctico. En conclusión, si vamos a considerar la computación como ciencia básica en el curriculo, debemos construir juntos las bases científicas del conocimiento didáctico que pueda sustentar de la práctica de los profesores.

Un análisis más profundo y exhaustivo de los datos sobre las actividades de aula recabados en las visitas a los profesores nos permitirá continuar desarrollando y validando el modelo didáctico. Aspiramos a trabajar con profesores de otras ciencias, en particular este año comenzaremos el proyecto “Espacios de investigación didáctica en el área de matemática y computación” aprobado en 2024 por la Comisión Sectorial de Investigación Científica de la Udelar.

Hemos registrado un dominio web y hosting y comenzado a elaborar una plataforma que provea un recurso abierto que oficiará de espacio de intercambio para todos los profesores e investigadores del país. La usaremos para divulgar resultados de investigaciones en didáctica de las ciencias computacionales.

Por último queremos agregar una reflexión sobre el trabajo que trasciende las instituciones y áreas de desempeño académico - profesional. En ese sentido hemos comprobado una vez más que la colaboración entre docentes e investigadores nos aporta mucho a todos, en especial cuando trabajamos en un nuevo paradigma como el de las ciencias computacionales donde la interdisciplina juega un rol muy importante y las nociones preconcebidas sobre cómo enseñamos (y cómo aprendimos en el pasado) deben ser reconocidas y corregidas.

Referencias

Ametller, J., Leach, J. & Scott, P. (2007). Using perspectives on subject learning to inform the design of subject teaching: an example from science education. *The curriculum journal, 18(4)*, 479-492. https://doi.org/10.1145/2632320.2632358.

ANEP. (2022). *Marco Curricular Nacional.* (Reporte de la Transformación Educativa Administración Nacional de Educación Pública).

Cabezas, M. (2021). Pensamiento Computacional, Educación STEM y la Educación Informática: Cuestiones Pendientes. *"Revista sudamericana de educación, universidad y sociedad (RSEUS)."*, *9(1)*, 45-59. https://plataformas.ude.edu.uy/revistas/rifedu/index.php/RSEUS.

Cabezas, M. & da Rosa, S. (2022). Modelado Didático para Ideas Fundamentales en Computación. *Proceedings of The 51 SADIO Conference, Simposio Argentino de Educación en Informática (SAEI 2022).* https://revistas.unlp.edu.ar/JAIIO/article/view/18293.

Commission, E., Education, E. & Agency, C. E. (2022). *Informatics education at school in Europe.* Publications Office of the European Union, Luxemburgo.

da Rosa, S., Viera, M. & García-Garland, J. (2020). A case of teaching practice founded on a theoretical model. *Lecture Notes in Computer Science 12518 from proceedings of the International Conference on Informatics in School: Situation, Evaluation, Problems*, 146-157. https://doi.org/10.1007/978-3-030-63212-0.

Denning, P. & Tedre, M. (2019). *Computational Thinking.* Cambridge, MA : The MIT Press.

diSessa, A. (2000). *Changing Minds: Computers, Learning, and Literacy.* MIT Press.

diSessa, A. (2018). Computational literacy and "the big picture" concerning computers in mathematics education. *Mathematical Thinking and Learning*, *20*(1), 3-31.

Guzdial, M. (1994). Approaches to Classroom-Based Computational Science.

Newman, M. (2013). Computational Physics. https://archive.org/details/university-of-michigan-mark-newman-%20computational-physics-university-of-michigan-2013

Somekh, B. & Noffke, S. E. (2009). *The SAGE Handbook of Educational Action Research.* SAGE Publications.