Нодельман Владимир Семенович

КОМПЬЮТЕР КАК СРЕДСТВО ОБУЧЕНИЯ МАТЕМАТИКЕ

В конце 60-х – начале 70-х годов, с созданием в НИИ Школьного оборудования и технических средств обучения Академии педагогических наук лаборатории математики, руководимой членом-корреспондентом АПН В.Г. Болтянским, разработка средств обучения математике (далее СО) приобрела плановый, научно обоснованный и профессиональный характер. В этот период сотрудниками лаборатории были опубликованы концептуальные материалы [1–3], определившие психолого-педагогический подход к созданию и использованию СО.

В основе этого подхода лежали релизация дидактического принципа наглядности и деятельностного подхода к восприятию учебного материала, основанного на психологической теории поэтапного формирования умственных действий П.Я. Гальперина. При этом наглядность понималась как представление знаний посредством изоморфных им моделей, которые просты в восприятии учащимися [1, 2]. Поскольку для формирования полноценных знаний одного лишь обеспечения наглядности недостаточно, предполагалась такая организация процесса усвоения, которая бы гарантировала активную и адекватную деятельность учащихся с учебным материалом (моделями), выполняемую по определенным правилам и в определенной последовательности [3].

Вплоть до 80-х, когда в школах стали появляться компьютеры, СО включали модели и приборы, учебные фильмы, диафильмы и диапозитивы, пленки для проецирования с помощью кодоскопа, тетради с печатной основой, дидактические материалы. Являясь воспитанником лаборатории, автор также принимал участие в их разработке.

Осознание педагогического потенциала компьютера как принципиально нового, интегрирующего СО пришло не сразу. Первые попытки состояли в слепом переносе сформировавшихся идей и решений с одного носителя на другой – компьютер[1]. Тем более, что это было легко сделать, так как традиционные СО, за исключением стереометрических моделей, представляли информацию в двумерном виде: тексты, чертежи, картинки, анимации. Даже приборы, несмотря на их трехмерную природу, в конечном счете служили цели формирования двумерных образов и манипулирования ими. Клавиатура и экран компьютера потенциально казались возможной заменой обычным средствам, а развитие компьтерной техники, разнообразие средств ввода, графические возможности современных ком-

[1] Вспоминается следующий факт. Где-то в первой половине 80-х, будучи в командировке в Москве, я был приглашен Г.Г. Левитасом – заведующим лабораторией математики НИИ ШОТСО на некое предприятие электронной промышленности. Предполагался сюрприз. Так и случилось – там я впервые увидел компьютер в его новой форме: клавиатура и дисплей. Но самым удивительным было то, что компьютерная программа моделировала разработанный мною и выпускавшийся к тому времени массовым тиражом «Полигон Логических Схем» – прибор, предназначенный для преподавания математики и развития логического мышления учащихся. Изображение на дисплее компьютера было точной копией передней панели прибора.

пьютеров далеко превзошли самые смелые ожидания педагогов 80-х.

Действительно ли компьютер может заменить традиционные средства обучения математике?

Ответ, в принципе, положительный. Исключение составляют случаи таких видов учебной деятельности, при которых реальное манипулирование трехмерными объектами, испытание тактильных ощущений являются педагогически значимыми. Более того, компьютер обогащает специфические функции традиционных СО новыми возможностями. Так, учебное кино, являясь мощным средством анимации, ограничено сценарием, режиссурой и монтажом: трудно или невозможно остановить фильм, изменить скорость показа, проиграть сначала конкретный его фрагмент. И уж, конечно, нельзя изменить содержание фильма: рассмотреть другой пример, изменить параметры задачи и т. д. Иное дело компьютер... Компьютерные программы свободны от указанных ограничений. Здесь поддержка возможности выбора предъявляемого материала, контроля над процессом представления информации, параметрического регулирования ее характеристиками является очевидной и общепринятой техникой.

Нужна ли подобная замена?

Пожалуй, нет! По крайней мере, не всего и не всегда. Компьютер может гораздо больше и это «компьютерово» ему и следует поручить в первую очередь.

Нужно ли при этом сохранить психолого-педагогические основы создания СО или им место лишь в литературе по истории педагогики?

Безусловно нужно! Покажем это на примерах.

Начнем с *Принципа Наглядности.* Самой популярной темой курса математики, для изучения которой разрабатываются и используются компьютерные программы, несомненно являются функции и их графики. Это «умеют делать» и любительские программы, написанные школьниками, и коммерческие продукты, созданные как специально для изучения математики («MATH», входящий в студенческий пакет Microsoft), так и для работы инженеров и математиков-профессионалов («MAPLE» Maplesoft, «MATLAB» Mathworks, «MATHEMATICA» Wolfram и др.), зачастую используемые в учебном процессе. Попробуем построить с их помощью график функции $y = \dfrac{\sin 2x}{\cos x}$ (рис. 1 *а*). Результат соответствует вполне «логичному» ответу учащихся:

$$y = \frac{\sin 2x}{\cos x} = \frac{2\sin x \cos x}{\cos x} = 2\sin x\,.$$

Разница лишь одна: программа не в состоянии «отловить» изолированные точки разрыва, а ученик просто забыл исключить случаи, когда знаменатель обращается в ноль. Еще «хуже» обстоит дело с графиком функции $y = [x]$ – целая часть числа (рис. 2 *а*). Налицо факт – коммерческие программы не умеют строить графики функций с конечными разрывами. Требование изоморфизма модели изучаемому материалу означает неприемлемость подобных построений. На рис. 1 *б* и 2 *б* показаны корректные графики, построенные авторской программой «ВижуМатика», которая легко справляется и с более содержательной задачей совместного построения графиков функций $y = \{x^3\}$ – дробная часть x^3 и $y = f_1(x)\sin x$ (рис. 3).

Следует ли любой ценой добиваться соблюдения требования изоморфизма модели изучаемому материалу?

Каждый педагог знает, насколько эффективны контрпримеры в процессе фор-

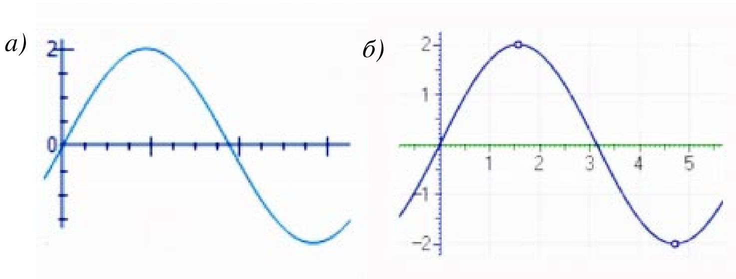

Рис. 1. *а)* «Графический Калькулятор», *б)* «ВижуМатика»

а) *б)*

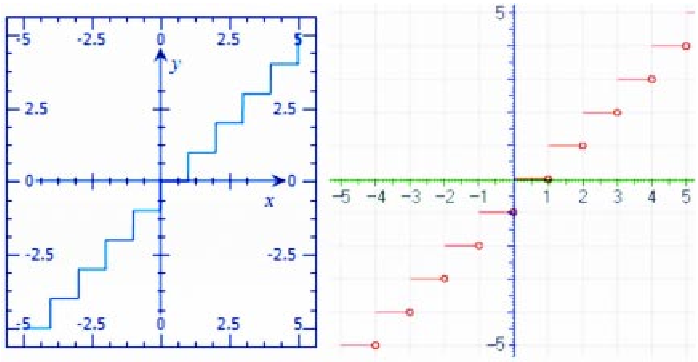

Рис. 2. *а)* «Графический Калькулятор», *б)* «ВижуМатика»

мирования знаний. Они позволяют акцентировать значимость наличия существенных признаков, создавая в результате в сознании учащегося адекватный образ (изоморфную модель) изучаемого понятия.[1] При этом сами контрпримеры оперируют с неизоморфными моделями!

Представим себе, что в нашем распоряжении есть только Excel, который уж точно не создавался для преподавания математики со всеми ее тонкостями. Попробуем с его помощью ввести понятие функции и ее графика. Построим таблицу (рис. 4 *а*), определив ячейки столбца B как удвоенные соседние ячейки столбца A. На основании этого табличного представления функции, построим ее график, который при подходящем определении будет выглядеть в виде последовательности изолированных точек (рис. 4 *б*). Обсудив с учащимися их взаимное расположение приходим к выводу о целесообразности соединить точки. Выбираем конфигурацию «Scatter with Straight Lines» и получаем подходящую картинку (рис. 4 *в*). Кликнем на ячейку B2 и переопределим ее: вместо "=A2*2" введем "=A2^2". После чего, ухватив за квадратик в нижнем правом углу выделенной ячейки B2, протянем его вниз до конца таблицы. В результате этих манипуляций наша таблица будет содержать значения функции $y = x^2$, но автоматически перестроившийся график выглядит в виде ломаной линии както несимпатично (рис. 4 *г*). Обсудив этот факт с классом и придя к выводу, что промежуточные значения строятся в Excel без учета функциональной зависимости, изменим конфигурацию на «Scatter with Smooth Lines», получаем гладкую параболу (рис. 4 *д*). Пойдем дальше и переопределим зависимость как "=A2-INT(A2)+3", то есть $y = \{x\} + 3$. В результате весь столбец B заполнился тройками, а наш гладкий график, соответственно, превратился в горизонтальную прямую (рис. 5 *а*).

Слишком просто для такой «непростой» функции! Для проверки наших подозрений добавим к графику три цветные точки, положением которых будем управлять с помощью несложного устройства (рис. 5 *б*)[2]. Здесь абсцисса красной точки[3] вводится с помощью скроллбара как вещественное число с точностью 0.1, а ее ордината высчитывается в Excel как "=абсцисса-INT(абсцисса)+3", тогда как желтая и зеленая точки определены как проекции красной на оси координат. Изменяя положение скроллбара об-

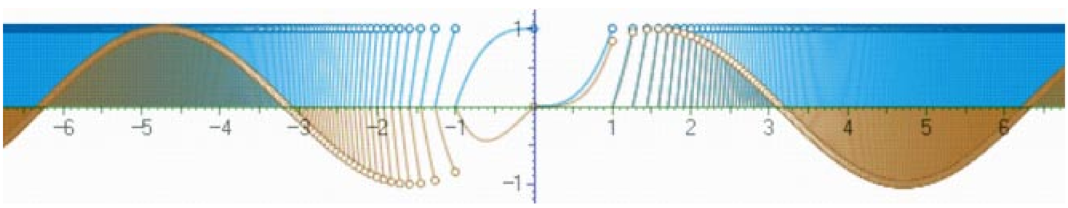

Рис. 3

[1] В [4] изложена система построения контрпримеров.

[2] В Израиле учащиеся изучают Excel в начале учебы в средних классах. Они могут сделать такое устройство сами.

[3] Точка внутри первой четверти, помеченная квадратиком) – *Прим. ред.*

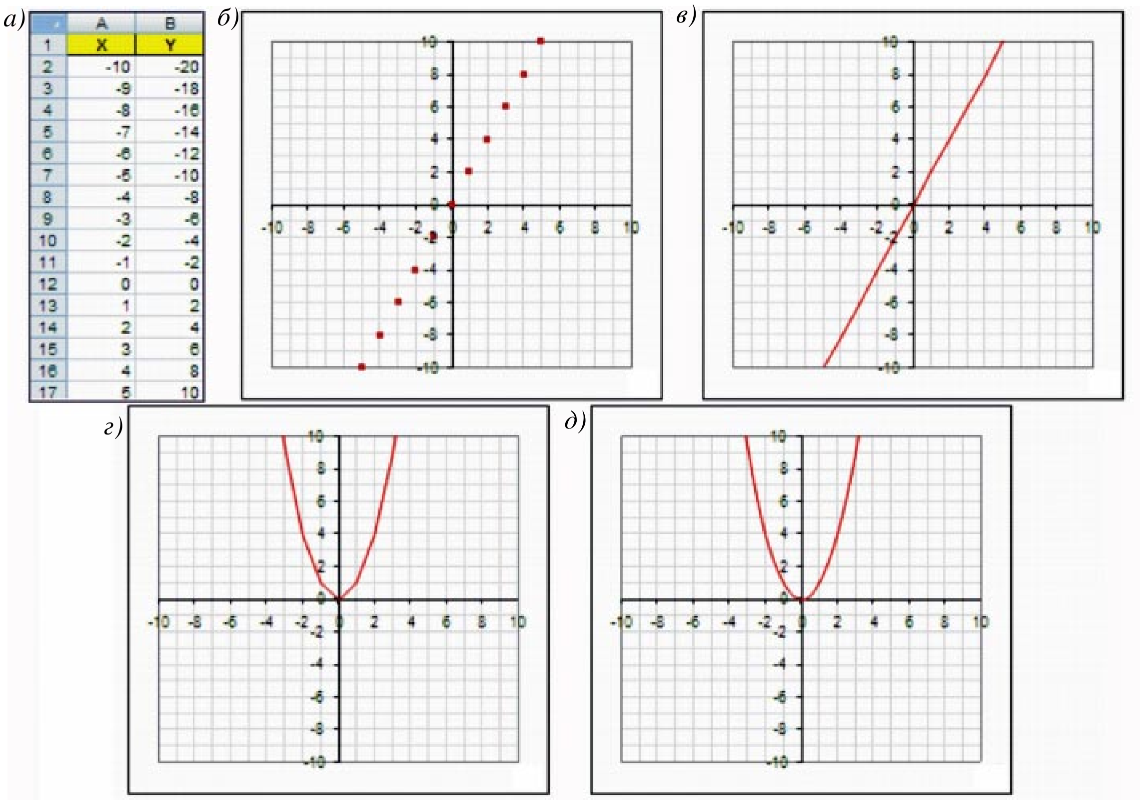

Рис. 4

наруживаем, что красная точка, будучи определена в соответствии с той же зависимостью, что и горизонтальная прямая, почти никогда на ней не оказывается! Обсуждение этой ситуации с классом позволяет придти к очень важным выводам с одной стороны о свойствах графического представления функции, а с другой – о необходимости критического отношения к результатам компьютерного моделирования.

В данном случае внутренняя ограниченность Excel послужила на пользу преподаванию.[1] Сейчас самая пора перейти к корректным моделям. Только после окончания

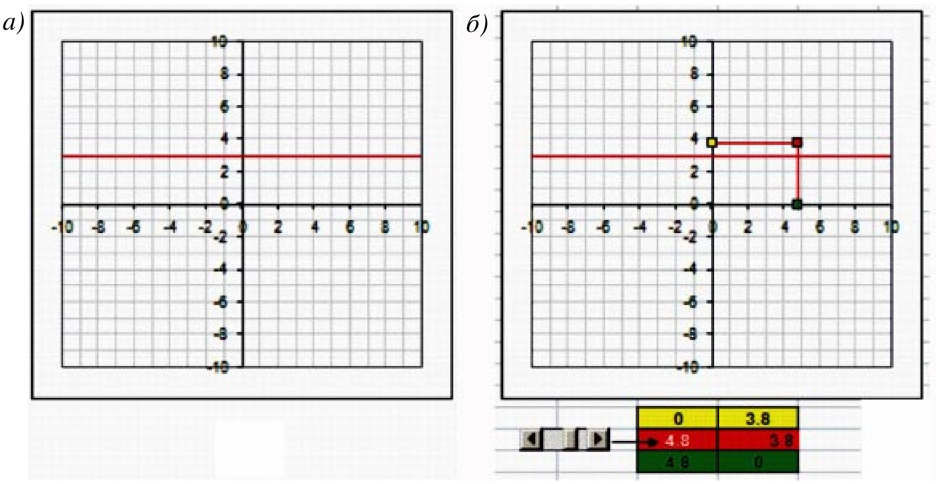

Рис. 5

[1] Учебные программы, в том числе и лишенные таких ограничений, должны позволять их симуляцию.

отработки темы в будущем можно будет порой игнорировать требование изоморфизма моделей, полагаясь на способность учащихся самостоятельно мысленно восполнять пробелы наблюдаемых графиков. Инженер или математик просто не заметит недостатков упомянутых выше профессиональных продуктов. Так и быть должно!

Понятие функции как соответствия находит широкое применение в геометрии. Интерфейс компьютерных программ динамической геометрии немыслим без интенсивной эксплуатации геометрических преобразований: базовая операция по передвижению захваченных мышкой объектов представляет собой параллельный перенос, большое количество трехмерных построений в «Cabri 3D» основано на поворотах. Эти преобразования *просты в восприятии* и очень быстро, без особых усилий, учащиеся начинают активно ими пользоваться. При этом работа по параллельному переносу и повороту выполняется как плавный процесс, на котором акцентируется внимание учащихся. *Какова цена этой простоты?*

Функциональная зависимость, соответствие (а геометрические преобразования являются примерами этих понятий) – это не процесс, а факт; хуже – «мгновенный» акт (прообраз $\to$ образ).

Рассматривая функцию одной переменной, не говорят о процессе постепенного превращения, «изгибания» оси абсцисс, до тех пор пока она не примет форму кривой – графика функции. Обсуждая методику преподавания графиков, выше мы обошлись без молотка, кусачек, плоско- и круглогубцев.

Сами термины «преобразование», «перенос», «поворот» семантически обозначают процесс, и это лишь усугубляет заблуждение. Чтобы избавиться от него, в какой-то момент нужно ввести соответствующие понятия в явном виде. Парадоксально, но программы динамической геометрии для этой цели не очень приспособлены! Дело в том, что в них имеется поддержка исключительно преобразований фигур, тогда как в математике наибольший интерес представляет рассмотрение этих понятий как преобразований пространства.

Сами преобразования фигур легитимны и являются традиционным элементом содержания курса геометрии. И если уж опираться на них, то они должны быть, по крайней мере, «распознаваемы» хотя бы как соответствия, а это означает, что программа должна поддерживать отработку как минимум двух умственных действий:

1) нахождение образа заданной фигуры,

2) нахождение прообраза по известному образу.

Попробуем, к примеру, подвинуть образ точки, полученный в результате поворота в «Cabri 2 Plus». Не тут то было, программа просто «не реагирует». Никаких «комментариев» о том, например, что положение этой точки всецело определяется другими элементами чертежа, какими и как... Стопор! Непедагогично как-то, хотя «математика» в порядке: в функционально зависимой паре свободен лишь аргумент-прообраз точки. «ВижуМатика», поддерживая режим преобразования фигур, имеет дружественный пользовательский интерфейс, который выражается, в частности, в следующем:

1. Выбранные объекты рисуются красным цветом.

2. При перемещении с помощью мышки выбранных объектов:

– положение мышки проецируется на оси координат (настройка по умолчанию);

– перемещаемые точки изображаются «полыми», тогда как точки, не участвующие в движении, закрашены внутри;

– образ мышки-курсора сопровождает только те точки, которые определяют перемещение (этих образов может быть несколько; не обязательно точка, которую выбрали последней, будет отмечена образом мышки-курсора);

– объекты, как правило, перемещаются даже в случае «объекта-функции». При этом зависимость между ними сохраняется.

Выделение независимых объектов-«аргументов», характеризующих движение и/или выбранный объект, является примером принципиальных отличий психолого-педагогической концепции, заложенной в разработку этого программного продукта,

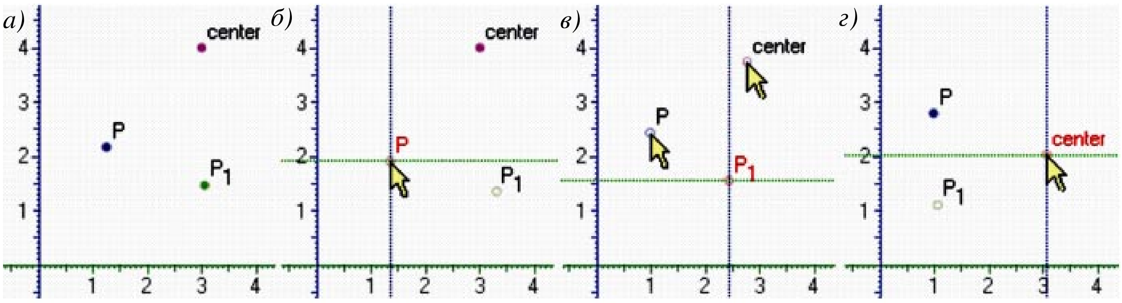

Рис. 6: *а)* конфигурация из трех точек, две из которых (P и *center*) независимы, а точка P_1 определена с помощью выражения «Rotate2D(P,45,center)» как результат поворота точки P вокруг точки center на 45°; *б)* перемещается прообраз – точка P; *в)* перемещается образ – точка P_1; *г)* движется *center*.

по сравнению с имеющимися средствами Динамической Геометрии.

Трудность наглядного рассмотрения геометрических преобразований как преобразований пространства состоит в «недостатке» размерности. Для изображения функции одной переменной нужно два измерения (мы строим графики на двумерной плоскости), для изображения функции двух переменных нужно три (мы строим поверхности в трехмерном пространстве). Преобразование плоскости требует четырех измерений, а преобразование пространства оперирует с шестью! Проблема визуализации кажется неразрешимой. Но так ли это?

Обратимся к традиционным СО. На уроках математики давно используются кодоскопы, проецирующие прозрачные объекты (чаще всего плоские пленки) на экран. Это создает возможность моделировать динамические изображения. Выпускаются наборы готовых пленок и среди них набор, содержащий пленки с изображениями систем координат, кривых и прямых, которые можно сдвигать по отношению друг к другу и относительно осей. Но ведь это готовый материал для работы с преобразованиями плоскости: координатные плоскости можно смещать по отношению друг к другу, моделируя *движения плоскости.* И самое главное: НЕВОЗМОЖНО ПРЕОБРАЗОВАТЬ ТОЛЬКО ФИГУРЫ – они *преобразуются вместе со своим носителем – всей плоскостью-пленкой их содержащей.* Идея простая – надо разделить пространства: отдельно показывать область определения и отдельно – множество значений. В компьютерной программе в роли пленок могут быть раздельные окна-«взгляды», и, конечно, их «наложение» можно поместить в дополнительное совместное окно.

Функции комплексной переменной также являются преобразованиями плоскости. Для их представления математическая литература содержит богатый арсенал исследовательских и педагогических средств, приемов и форм, в том числе всевозможных трехмерных поверхностей $f\colon C \to R$, где в качестве значений берутся отдельные характеристики функции (модуль, мнимая составляющая, аргумент и т. д.). Можно взять эти идеи на вооружение.

Осталось решить как дать компьютеру задание выполнить преобразование. Функциональное определение в двумерном случае выглядит так: $f(x, y) = (f_1(x, y), f_2(x, y))$. Например, $f(x, y) = (x + a, y + b)$ определяет параллельный перенос плоскости на вектор (a, b). А поскольку геометрические преобразования являются линейными, естественно привлечь для их изучения механизмы линейной алгебры, в том числе и матричное определение.

В итоге, двумерная модель параллельного переноса плоскости на вектор (b, c) с последующим поворотом на угол a вокруг начала координат выглядит как показано на рис. 7. Здесь в верхнем левом углу показана исходная плоскость, на которой построен треугольник PQR и график функции

$y = \sin x$. В верхнем правом углу – образ этой плоскости, полученный в результате преобразования, заданного в матричном виде. Внизу слева показано совместное изображение «содержимого» и его образа. Соответствие цвета точек плоскости и ее образа является кодом, облегчающим восприятие отображения в целом. При перемещении мышки в любом из этих подокон программа динамически изображает соответствующие образы, прообразы и проекции; так на рис. 7 мышка указывает на образ точки P, а программа выделяет ее прообраз на «исходной» плоскости и пару соответствующих точек на нижнем совместном изображении. Элементы матрицы преобразования зависят от параметров, изменяя значение которых с помощью элементов интерфейса программы, можно наблюдать характер их влияния на поведение преобразований данного семейства.

В более сложных случаях для создания по настоящему информативной, изоморфной и простой в восприятии модели могут понадобиться дополнительные окна и/или подходящие характеристики изображения. На рис. 8 приведены различные панели диалогового окна «Двумерное Преобразование / Функция Комплексного Переменного», служащего для настройки характеристик соответствующих моделей и режимов управления ими. Обратим внимание лишь на верхнюю часть второй панели, которая позволяет выбрать способ представления плоскости. По умолчанию, это цветная раскраска, показанная на рис. 7. Ее можно заменить на цветную или черно-белую сетку, как это принято в комплексном анализе, на картинку, например на фотографию, или убрать вовсе, оставив чистую плоскость с размещенными на ней математическими объектами. При всех таких заменах полученные модели – *а это совокупности всех используемых синхронизированных окон* – сохраняют свойство изоморфизма изучаемому материалу, которое, как указывалось выше, означает соответствие модели изучаемому с ее помощью преобразованию, поддержку прежде всего двух умственных действий: нахождение образа произвольной

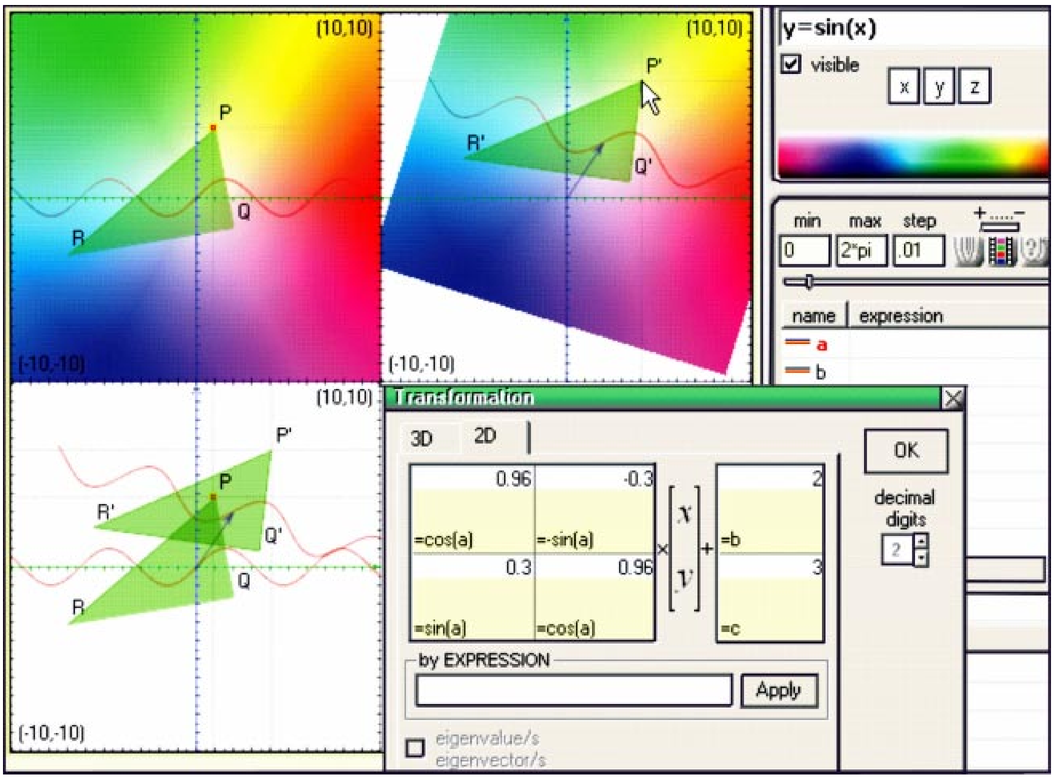

Рис. 7

точки *плоскости* и нахождение прообраза по известному образу в соответствии с определением изучаемого отображения[1].

Простота восприятия этих моделей зависит от аудитории: начинающему более «впечатляющей» покажется работа с фотографиями. Его внимание будет обращено исключительно на подтверждение соответствия, достигаемое простым манипулированием мышкой и сравнением изображений в верхних окнах (во время определения преобразования *плоскость-«картинка»* действительно *поворачивается* при изменении параметра *a* и *сдвигается* при изменении *b*, *c*). «Продвинутому» ученику гораздо более выразительной представится черно-белая сетка. Простота – характеристика субъективная.

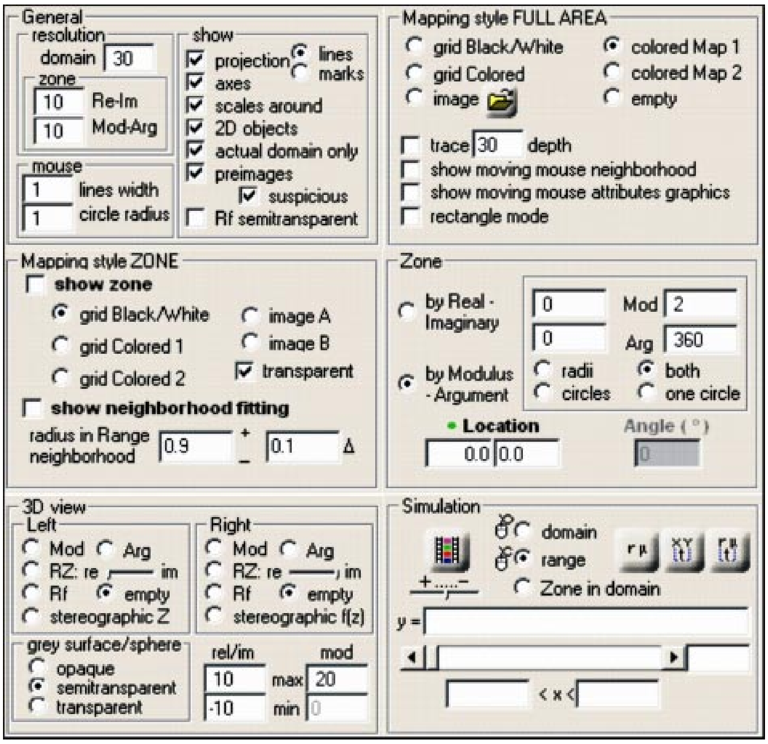

Рис. 8

Изоморфизм же, будучи *объективным свойством*, зависит однако от глубины изучения материала: на разных этапах учебного процесса рассматриваются различные, все более широкие наборы характеристик этого материала, и модели для сохранения соответствия содержанию становятся более детализированными и внутренне усложненными. Геометрические преобразования обладают двумя не зависящими от размерности пространства важными свойствами, лежащими в основе построений и доказательств:

1. Сохранение принадлежности прямой: для любой точки T, прямой PR и их образов T' и $P'R'$ если $T \in PR$, то $T' \in P'R'$.

2. Сохранение отношения расстояний: для любых отрезков PR и MN и их образов $P'R'$ и $M'N'$ $\dfrac{PR}{MN} = \dfrac{P'R'}{M'N'}$.

Объяснив их значение (для построения образа геометрической фигуры достаточно найти образы конечного множества ее точек, после чего можно просто соединить их отрезками), попробуем эти свойства «доказать».

На рис. 10 снова показана модель поворота вокруг точки (b, c) на угол α, предварительно «очищенная» от треугольника и синусоиды. Заменив режим отображения на цветную сетку, обнаруживаем, что составляющие ее горизонтальные и вертикальные прямые сохраняют свою «форму» – преоб-

[1] Даже отработка только этих действий может оказаться нетривиальной задачей, в частности, когда изучаемое отображение является необратимым (к «счастью» базовые преобразования пространства таковыми не являются). Потребуется привлечь дополнительный арсенал дидактических средств, например, воспользоваться специальными трехмерными поверхностями, подвижными зонами и т. д. На рис. 9 изображена модель функции $f(z) = \sin z$, демонстрирующая множественные прообразы комплексного числа, находящего в точке, на которую указывает мышка. Для пояснения цвета точки под мышкой и причин потери колорита исходной области в целом привлечены дополнительные окна с поверхностями, на которых располагаются синхронизированные прообразы.

разуются в прямые. Вопрос относительно произвольно расположеных прямых остается открытым. Введем две свободные точки P и R и соединим их прямой. Определим точку T как свободно перемещающуюся вдоль этой прямой. Это можно сделать мышкой, воспользовавшись интерфейсом программы, или напрямую, введя выражение «pointOnSegment(P, R, 0.15)», где 0.15 – это коэффициент линейной комбинации, определяющей положение точки на прямой PR (точка лежит на отрезке PR, когда значения коэффициента находятся в интервале [0,1]). В любом случае точка будет определена именно им. Особенность точек, тем или иным способом определенных выражениями семейства pointOn... (pointOnSegment, pointOnSurface, pointOnConic и т. д.), состоит в том, что, будучи зависимыми от других геометрических объектов, при своем перемещении с помощью мышки они изменяют положение посредством подходящего автоматического изменения своих числовых параметров или, если это невозможно, не перемещаются вовсе (ср. с принципами организации интерфейса, иллюстрированными на рис. 6). Подвигаем наши три точки. Обращаем внимание учащихся на сохранение тестируемых отношений $T \in PR$ и $T' \in P'R'$. В продвинутой аудитории можно задать положение точки Q явной формулой $d \cdot P + (1 - d) \cdot R$, где d – свободный параметр, и, меняя его значения, убедиться, что наблюдаемое свойство выполняется и в этом случае. Обратим внимание учащихся, что образы точек (их имена программа метит апострофами) и прямой построены автоматически, просто потому что они принадлежат трансформируемой плоскости (ср. с порядком построения точки P_1 на рис. 6). Скептик может усомниться в истинности результатов, полученных исключительно визуально. В любом случае чрезвычайно важно обсудить такую вероятность, выяснить и ввести в компьтер аналитические критерии, подтверждающие принадлежность точек отрезкам (первые четыре строки в окне условий в нижнем правом углу рис. 10). Повторив манипуляции с точками и параметром d, убеждаемся в том,

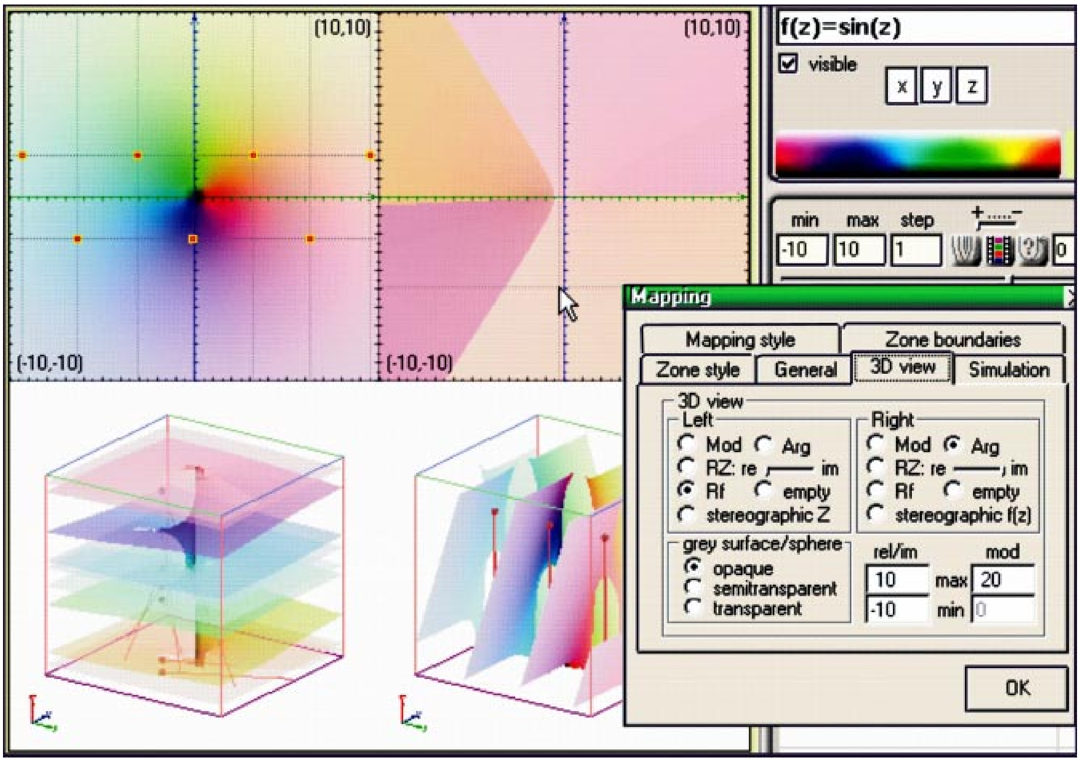

Рис. 9

что до тех пор пока точки T и Q остаются внутри отрезка, правый столбец этого окна целиком заполнен значениями True.

Свойство 2) не столь «наглядно». Для его проверки введем дополнительную точку S и дополнительную строку в окне условий (нижняя строка на рис. 10). Меняя мышкой положения трех точек P, R и S, обнаруживаем неизменность значения True этого условия.

Меняя значения параметров a, b и c, изменим само преобразование. Наблюдая соответствующие изменения изображений в правом и нижнем окне, обратим внимание на неизменность значений в столбике *окна условий*. Дополнительное варьирование положений точек сцены также оставляет это окно неизменным. Похоже, что наши условия справедливы всегда, а вместе с ними и свойства 1) и 2).

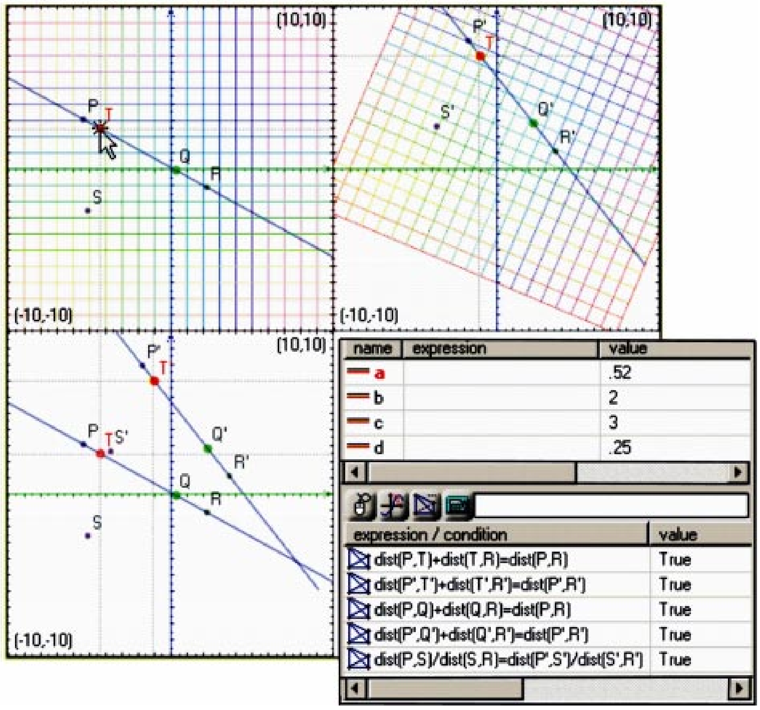

Рис. 10

Окончание в следующем номере.

Литература

1. *Болтянский В.Г.* Формула наглядности – Изоморфизм плюс простота // Советская педагогика, 1970, № 5.

2. *Антоновский М.Я.* Простота восприятия – важнейшая часть понятия наглядности // Математика в школе, 1971, № 4.

3. *Антоновский М.Я., Болтянский В.Г., Волович М.Б. и др.* Комплексы учебного оборудования по математике. Педагогика, 1971.

4. *Нодельман В.С.* Создание обучающих программ для ЭВМ. Минск, 1988.

5. *Горелик Л.В.* Электронный учебник с динамическими моделями // Компьютерные инструменты в школе, 2008, № 1.

Dr. Vladimir Nodelman,
Holon Institute of Technology,
Faculty of Sciences,
Dept. of Computer Scienes, Israel.