

МАТЕМАТИЧЕСКАЯ ВИЗУАЛИЗАЦИЯ В РЕАЛЬНОМ ВРЕМЕНИ С GRAFAR

Клепов Владимир Юрьевич

Студент

Факультет компьютерных наук НИУ ВШЭ, Москва, Россия

E-mail: v.klepov@gmail.com

Визуализация — важный инструмент науки. Упрощенное графическое представление сложных объектов было одним из первых видов абстракции и стоит у истоков классической математики. Графики наглядно показывают закономерности в экспериментальных данных. У каждой науки есть особый визуальный язык, компактно подчеркивающий важные для нее детали.

Компьютеры легко визуализируют даже самые сложные объекты. Современные ученые могут обрабатывать и отображать миллионы наблюдений, исследовать связь между ними и публиковать результаты в понятной форме. Другое преимущество компьютерной визуализации — быстрое динамическое изменение изображения.

Библиотека для визуализации данных d3.js — четвертый по популярности проект на GitHub, который используют миллионы людей каждый день. Библиотека написана на языке JavaScript и позволяет любому интернет-пользователю взаимодействовать с данными и находить в них интересные закономерности. Похожий подход к математике принес бы пользу всем. Люди, не разбирающиеся в математике, смогли бы уловить практический смысл формальных результатов. Студенты — найти связь между параметрами математических объектов. Практикующим математикам, владеющим формальным аппаратом, наглядное изображение могло бы подсказать идею доказательства или перспективное направление исследования. Поэтому я разрабатываю Grafar — кроссплатформенную (работает в браузере, без проприетарных плагинов) и эффективную библиотеку для многомерной математической визуализации.

Математические объекты обычно задают в краткой символической форме. Система использует многоступенчатую формулировку задачи математической визуализации:

1. Найти алгоритмическую форму объекта;
2. Зафиксировать свободные переменные и вычислить координаты близкого к исходному конечного объекта;
3. Визуализировать полученный набор данных.

Каждый из этих шагов хорошо исследован и может быть эффективно реализован. На третьем шаге можно использовать любую библиотеку для визуализации данных. Проблема в том, что для математических объектов статистическая обработка часто не имеет смысла, и приходится отображать десятки тысяч движущихся точек. Существующие пакеты не рассчитаны на такую работу. На каждом шаге Grafar использует методы, специально разработанные для математических объектов.

Вычисления в реальном времени реализованы в Grafar через парадигму реактивного программирования. При обновлении нескольких переменных обычно нет необходимости повторять все вычисления. Зависимости между данными представляются в виде графа потока данных (Data Flow Graph) вершины соответствуют переменным, ребра — зависимостям. Простая система уравнений

$$x = 1 \wedge c = 2 \wedge y = cx \wedge d = c + x$$

даёт граф, изображенный на рис. 1. При обновлении переменной нужно заново вычислить только значения потомков соответствующей вершины. Кроме того, граф потока данных позволяет находить «мертвый» код и автоматически распараллеливать выполнение. Как декларативная техника, реактивное программирование похоже на математическую нотацию. Эту парадигму используют редакторы таблиц (MS Excel) и программы для процедурного моделирования (Grasshopper).

Иллюстрации

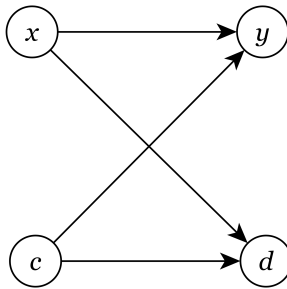


Рис. 1. Граф потока данных

Я предлагаю несколько обобщений реактивного программирования. Так, у программы с циклическим графом потока данных нет корректного порядка выполнения. Численные методы часто могут разрешить циклические зависимости. С таким расширением система аппроксимирует и визуализирует объекты, заданные неявными функциями и даже дифференциальными уравнениями. Кроме того, вычисления на списках вместо атомарных переменных эффективно реализуются через реляционные базы данных.

Полученные данные (реляционные таблицы) отображаются с помощью WebGL. Эта технология дает браузеру доступ к видеокarte, и изображение обновляется без задержек. Чтобы показать направления оптимизации, я рассматриваю некоторые принципы работы GL и их связь с реактивными вычислениями на таблицах. Матрицу смежности точек, наглядно показывающую непрерывность объекта, можно анализировать с точки зрения алгебраической теории графов.

Визуальный аппарат математики отличается от принятого в информационной графике. Ученые часто уделяют мало внимания «эстетике» представления, хотя эта часть может сделать график понятным. Grafar поддерживает несколько методов визуализации многомерных объектов: параллельные координаты и цветовое кодирование. Оптимальные цветовые схемы для разных видов данных легко генерировать, а результат делает изображения понятнее и привлекательнее.

Grafar подходит не только для визуализации непрерывных математических объектов. Передавая системе объекты в алгоритмическом виде можно иллюстрировать цифровую обработку сигналов или теорию графов. Система отображает большие объемы данных в реальном времени даже на мобильных устройствах.

Литература

1. Demetrescu C., Finocchi I., Ribichini A. Reactive Imperative Programming with Dataflow Constraints. In Proceedings of the 26th ACM International Conference on Object-Oriented Programming, Systems, Languages and Applications, Portland, Oregon, USA, 2011, P. 407–426.
2. Vickers P., Faith J., Rossiter N. Understanding Visualization: A Formal Approach Using Category Theory and Semiotics. IEEE Transactions on Visualization and Computer Graphics. 2013. Vol 19, № 6. P. 1048–1061.
3. Theus M. Interactive Data Visualization using Mondrian. Journal of Statistical Software. 2002. Vol 7, № 11. P. 1–9.