алгоритма нечёткого вывода и т.п. задаются априори и не подвергаются изменению в процессе обучения сети, что нейронечеткие сети могут оказаться малоэффективными. В связи с этим предлагается алгоритм самоорганизации нейронечетких сетей, позволяющей настраивать не только параметры, но и структуру сети в процессе управления во время эксплуатации объекта.

Литература

1. Круглов В.В. Искусственные нейронные сети. Теория и практика / В.В.Круглов, Н.Н.Борисов. – М.:Горячая линия – Телеком, 2001. 382 с.
2. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы, пер.с польск. И.Д. Рудинского. –М.: Горячая линия – Телеком, 2006. -452 с.
3. Плетнев Г.П. Автоматизированное управление объектами тепловых электростанций /Г.П.Плетнев. –М.: Энергоиздат, 1986. -368 с.
4. Леоненков А.Ю. Нечёткое моделирование в среде Matlab и fuzzyTech / А.Ю.Леоненков. –С.-Птр.: БХВ, 2003. -720 с.
5. Юсупбеков Н.Р., Алиев Р.А., Алиев Р.Р., Юсупбеков А.Н. Интеллектуальные системы управления и принятия решений. Ташкент «Узбекистон миллий энциклопедияси», 2014. -490 с.

Ю.Н. Косников, А.И. Афанасьев

ФОРМООБРАЗОВАНИЕ НЕАНАЛИТИЧЕСКИХ ПОВЕРХНОСТЕЙ В РЕЖИМЕ РЕАЛЬНОГО ВРЕМЕНИ С ПРИМЕНЕНИЕМ СЕЧЕНИЙ

(Пензенский государственный университет)

Во многих предметных областях требуется визуальное представление пространственных объектов, поверхность которых задана набором характерных (опорных) точек, то есть скалярным полем. Типичным примером таких объектов является рельеф земной поверхности. Другой пример – показанное в виде поверхности распределение физической величины (температуры, количества осадков, уровня загрязнения воздуха) по площади региона. В обоих случаях поверхности не имеют точного аналитического описания, и для из визуализации требуется применить интерполяционные методы. Во всех случаях обязательным этапом процесса визуализации является представление поверхности в виде полигональной сетки, необходимое для графической системы компьютера. При этом следует учесть, что к визуализации подобных объектов предъявляется требование динамики в реальном времени (РВ).

У наблюдателя может возникнуть потребность изменить ракурс обзора или масштаб объекта, что предполагает выполнение операций сдвига, поворота (вращения) и масштабирования объектов. Из них особенно ресурсоемким является поворот объектов, так как для его выполнения необходимо перемножение координат всех точек объекта на тригонометрические функции углов поворота.

В то же время объекты, зачастую, имеют высокую детальность представления, чтобы при увеличении изображения не терялась его информативность. Количество опорных точек может исчисляться тысячами и десятками тысяч. Наконец, следует иметь в виду, что на визуализацию не могут быть направлены все вычислительные ресурсы компьютера, большинство ресурсов идет на решение основной прикладной задачи – на сбор данных, вычисления, моделирование и т.п.

Анализ перечисленных факторов позволяет утверждать, что визуализация неаналитических поверхностей высокой детальности с динамикой РВ – комплексный процесс, требующий применения эффективных по производительности алгоритмических решений, в том числе, в части формообразования.

Можно предложить следующую последовательность действий.

1) Предварительно (не в режиме РВ) нужно перейти от исходных неупорядоченных опорных точек к новым, также принадлежащим поверхности объекта, но лежащим в узлах ортогональной параметрической сетки. Это можно сделать с помощью любого известного метода интерполяции, работающего на неупорядоченном множестве точек и имеющего точность, удовлетворяющую разработчика. Например, хорошим инструментом интерполяции являются радиальные базисные функции (РБФ), причем для рассматриваемых поверхностей интерполянт, как правило, может быть представлен однозначной функцией $z=f(x,y)$ [1,2,3]. Обход области определения аргументов с некоторым шагом и вычисление значений интерполянта дает упорядоченное множество новых опорных точек. Номера шагов образуют отсчеты параметрических координат u,v. Для того, чтобы впоследствии строить поверхность по сечениям, шаги по одному аргументу (например, y) должны быть меньше шагов по второму аргументу (x).

2) Геометрические преобразования сдвига, поворота, масштабирования, необходимые для динамики объекта, следует проводить над его опорными точками, а полигональную сетку формировать в режиме РВ с помощью интерполяции. Исходные данные для интерполяции образует полученный набор новых опорных точек. В результате геометрических преобразований координаты опорных точек в пространстве сцены изменяются, но не изменяются их положения в параметрической системе координат. Тогда координаты промежуточных точек в процессе интерполяции РВ находятся по трем интерполяционным выражениям вида

$$x = f_1(u,v), \qquad y = f_2(u,v), \qquad z = f_3(u,v).$$

3) Вершины полигональной сетки получаются в результате интерполяции не отсеков поверхности, а их сечений. Принципиальная возможность такого приема упоминается, например, в монографии [4]. Применение сечений позволяет значительно ускорить вычисление промежуточных точек поверхности. Например, математическое описание для отсека сплайновой линии содержит 4 вычисляемых слагаемых, а для сегмента сплайновой поверхности – 16 слагаемых.

Упорядоченность опорных точек дает возможность применить интерполянты различного вида. Функции радиального базиса в данном случае не очень подходят, так как содержат дроби, радикалы, тригонометрические и экспоненциальные функции, затратные в вычислительном отношении. Хорошими интерполянтами на сетке являются сплайны, однако не все они подходят для задач визуализации в РВ. Краткий анализ интерполяционных возможностей сплайновых кривых приведен в [5]. В рассматриваемом случае следует отдать предпочтение сплайнам Кэтмулла-Рома. Отсек этого сплайна точно проходит через опорные точки, а его изгибы в концевых точках определяются только опорными точками этого отсека [6].

Отсек кривой Кэтмулла-Рома задается четырьмя опорными точками $P_i(x_i,y_i,z_i)$, где i=0,..,3 – номер точки. Математическое описание отсека имеет вид

$$x = 0.5(-t(1-t)^2x_0 + (2-5t^2+3t^3)x_1 + t(1+4t-3t^2)x_2 - t^2(1-t)x_3),$$

$$y = 0.5(-t(1-t)^2y_0 + (2-5t^2+3t^3)y_1 + t(1+4t-3t^2)y_2 - t^2(1-t)y_3),$$

$$z = 0.5(-t(1-t)^2z_0 + (2-5t^2+3t^3)z_1 + t(1+4t-3t^2)z_2 - t^2(1-t)z_3),$$

где t – параметр.

При задании набора опорных точек (узлов интерполяции) алгоритм интерполяции последовательно переходит от предыдущего отсека к следующему, добавляя одну очередную («новую») опорную точку в качестве P_0 и отбрасывая одну «старую» (P_3).

Новые опорные точки упорядочены в ряды, которые и становятся основой для получения сечений. В пределах одного сечения изменяется только одна параметрическая координата, например, u. Тогда в матричной форме интерполянт сечения имеет вид

$$\boldsymbol{C} = 0.5 \cdot \boldsymbol{U} \cdot \boldsymbol{M} \cdot \boldsymbol{CP},$$

где $\boldsymbol{C}$ – матрица текущих координат точек сечения;

$\boldsymbol{U}$ – матрица степеней аргумента (параметра) u;

$\boldsymbol{M}$ – базисная матрица сплайна Кэтмулла-Рома;

$\boldsymbol{CP}$ – матрица координат опорных точек отсека кривой:

$$\boldsymbol{C} = |x \quad y \quad z|, \qquad \boldsymbol{U} = |u^3 \quad u^2 \quad u \quad 1|, \qquad \boldsymbol{M} = \begin{vmatrix} 0 & 2 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 2 & -5 & 4 & -1 \\ -1 & 3 & -3 & 1 \end{vmatrix},$$

$$\boldsymbol{CP} = \begin{vmatrix} x_0 & y_0 & z_0 \\ x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \end{vmatrix}.$$

На каждом сегменте интерполяции в матрице $\boldsymbol{CP}$ происходит сдвиг содержимого на одну строку вверх. При этом верхняя строка пропадает, а на место сдвинувшейся четвертой строки встает новая строка, содержащая координаты очередной опорной точки сечения. Дойдя до конца очередного сечения, алгоритм интерполяции переходит на соседнее, и процесс повторяется.

В пределах одного сегмента находится несколько промежуточных точек сечения. Их количество определяется желаемой детальностью будущей полигональной сетки. Для этого параметр изменяется в интервале [0,1] с шагом (1/*N*), где *N* – желаемое число полигонов в пределах отсека.

4) Промежуточные точки соседних сечений принимаются за вершины полигональной сетки. Поскольку опорные точки лежат в узлах регулярной сетки и количество шагов в пределах каждого отсека одинаково (*N*=const), выбор вершин для каждого полигона очевиден. Множество вершин полигональной сетки передается в графическую плату компьютера, где выполняются остальные операции, обеспечивающие желаемый внешний вид объекта.

В итоге, высокая производительность формообразования поверхности объектов с применением сечений достигается, благодаря:

- предварительной регуляризации опорных точек;
- выполнению геометрических преобразований над ограниченным числом опорных точек, а не над множеством вершин полигональной сетки;
- полигонизации поверхности в режиме РВ путем интерполяции по сечениям поверхности;
- применению для интерполяции математического аппарата сплайнов Кэтмулла-Рома, не требующего никаких дополнительных действий с касательными, вспомогательными опорными точками и пр.

Кроме того, для вычисления координат промежуточных точек могут быть применены быстрые алгоритмы. Смешивающие функции сплайна Кэтмулла-Рома являются кубическими полиномами, причем каждая смешивающая функция всегда имеет один и тот же вид. Это означает, что значения смешивающих функций могут быть предварительно занесены с нужной точностью в память компьютера. Тогда очередное значение смешивающей функции не вычисляется, а считывается из памяти. В качестве адреса памяти используется значение параметра *u*, приведенное к выделенному диапазону адресов. Кроме того, нахождение промежуточных точек на сечениях идет последовательно, что дает возможность применить вычисления по приращениям. Например, метод конечных разностей позволяет получить значение кубической функции за три операции суммирования [7].

Литература

1. Buhmann M.D. Radial Basis Functions: Theory and Implementations. – Cambridge: Univ.Press, 2008. – 259 p.
2. Anjyo K., Lewis J.P., Pighin F. Scattered Data Interpolation for Computer Graphics // ACM SIGGRAPH. – 2014 – pp.1–69. – Электронный ресурс. – Доступно из URL: https://doi.org/10.1145/2614028.2615425
3. Косников Ю.Н. Методика и технология компьютерного моделирования поверхностей свободных форм с применением радиальных базисных функций // XXI век: итоги прошлого и проблемы настоящего плюс. Научное периодическое издание. Серия: Технические науки. Информационные технологии. – Пенза: Изд-во ПГТУ, 2014. – №03(19). – С. 176 – 183.

4. Фокс А., Пратт М. Вычислительная геометрия. Применение в проектировании и на производстве. – М.: Мир, 1982. – 304 с.

5. Косников Ю.Н., Афанасьев А.И. Расширение изобразительных возможностей средств визуализации с помощью управления гладкостью геометрических форм // Перспективные информационные технологии (ПИТ 2021): труды Международной научно-технической конференции /под ред. С.А. Прохорова. – Самара: Издательство Самарского научного центра РАН, 2021. – С.39 – 42. – Электронный ресурс. – Доступно из URL: https://ssau.ru/events/1137-mezhdunarodnaya-nauchno-tekhnicheskaya-konferentsiya-perspektivnye-informatsionnye-tekhnologii-pit-2021

6. Catmull, E., Rom, R. A Class of Local Interpolating Splines // Computer Aided Geometric Design / R.E. Barnhill and R.F. Reisenfeld, Editors, Academic Press. – NY, 1974. – pp. 317–326.

7. Косников, Ю.Н. Применение бикубических сплайнов в графических системах реального времени // Вестник Саратовского государственного технического университета. – 2005 – №4(9). – С.30 – 36.

Н.П. Кривулин

МЕТОДЫ ПОСТРОЕНИЯ ИЗМЕРИТЕЛЬНЫХ СИСТЕМ ВОССТАНОВЛЕНИЯ ВХОДНЫХ СИГНАЛОВ

(Пензенский государственный университет)

Проблема восстановления входных сигналов измерительных систем занимает одну из ключевых позиций во многих областях техники. Она возникает при динамических измерениях, в радиолакции, теории разпознавания изображениях, теории связи и т. д. Обзоры методов решения данной проблемы можно найти в работах [1−4].

Как отмечается в монографии [1] существует два подхода к решению данной проблемы. Первый, априорный, который заключается в том, что путем совершенствованием конструкции измерительных систем (ИС) добиваются минимума искажений в системе приема сигнала (один из таких подходов рассмотерен в работе [5]), и второй, апостериорный, в основу которого полжены алгоритмы восстановления входного сигнала по выходному сигналу. Построение таких алгоритмов осуществляет редукцию к идеальному прибору.

Предлагаются методы, основанные на втором подходе, восстановления входных сигналов нестационарных систем путем построения ИС. ИС состоит из первичного измерительного преобразователя ПИП и последовательного соединного виртуального измерительного преобразователя (ВИП). ВИП обрабатывает выходной сигнал от ПИП, таким образом, что на его выходе наблюдается измеряемый входной сигнал.