

любой момент, вне зависимости от его расположения в дереве компонентов. Стоит упомянуть, что в Angular 2 реализован прием, позволяющий любому объекту веб-приложения получить доступ к другому, предварительно зарегистрированному объекту (provider) при необходимости.

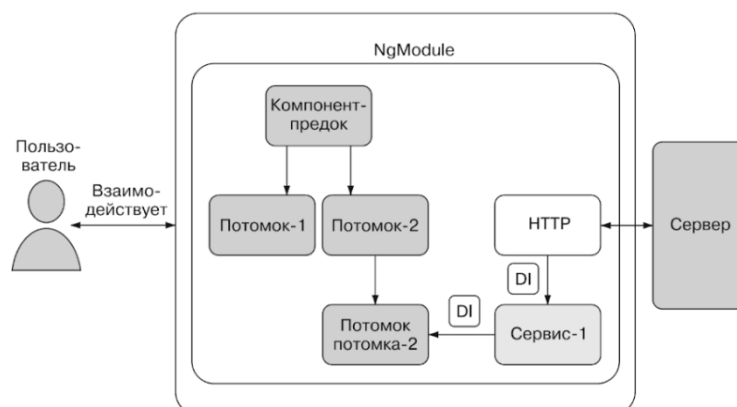


Рис. 1. Пример архитектуры Angular-приложений

Так же хотелось бы отметить, что Angular 2 разработан с нуля для мобильных приложений и оптимизирован с точки зрения эффективности памяти и меньшего количества циклов центрального процессора. Здесь объектами первого класса являются touch-события и жесты, работающие на всех устройствах. Все тесты производительности, проводимые командой Angular, находятся в открытом доступе в Github. Так что абсолютно любой может на них взглянуть и даже провести их.

Таким образом, фреймворк Angular.js является полноценным инструментом, полноценно реализующим компонентный подход, позволяя разработчикам создавать отдельные компоненты, а из компонентов создавать полноценные приложения. Созданные компоненты можно использовать в других приложениях, так как они независимы.

Список использованных источников:

1. Angular 2 [Электронный ресурс]. – Режим доступа: <https://angular.io/>.
2. Fain Y., Moiseev A. Angular 2 Development with TypeScript. – Manning Publications, 2016

## ПРИМЕНЕНИЕ СКРЫТЫХ МАРКОВСКИХ МОДЕЛЕЙ ДЛЯ СОПОСТАВЛЕНИЯ КАРТ

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Иванин Н.С.*

*Волорова Н.А. – к.т.н., доцент*

В последнее время было разработано множество подходов, позволяющих улучшить навигацию и управление транспортным средством на основе данных, собираемых его различными датчиками. Одним из таких датчиков является датчик Глобальной Системы Позиционирования (GPS). Зачастую требуется определить по какой дороге движется транспортное средство на основе данных датчика GPS – эта проблема известна как проблема сопоставления карт. В настоящей работе предлагается подход на основе использования Скрытых Марковских Моделей (СММ) для решения задачи сопоставления карт. Этот подход позволяет бороться с такими проблемами, как шумы в GPS данных и разреженность данных.

Сопоставление карт – это процесс определения дороги, по которой совершалось движение транспортного средства на основе данных, собранных специальными датчиками. Как правило, датчики собирают данные GPS, поскольку система GPS доступна повсеместно. Например, сопоставление карт используется навигационными системами. В последнее время сопоставление карт также используется для измерения скорости движения по дороге, а также построения статистических моделей пробок. В дальнейшем такие модели могут быть использованы для нахождения оптимального пути движения транспортного средства в обход пробок [1].

В работе [2] приводятся основные элементы структуры дорожной сети:

Узел. Дорога, являющаяся линейным объектом, при цифровом представлении хранится как последовательность точек (узлов). Узлами являются также точки пересечения дорог, начала и концы дорог, по которым транспортные средства могут двигаться.

Сегменты дорог. При существовании прямого пути между двумя соседними узлами такой путь будет называться сегментом дороги. Начальный и конечный узел такого пути будут называться начальными и конечными узлами сегмента.

Дуга. При существовании направленного пути между двумя соседними узлами такой путь будет называться дугой, а узлы соответственно началом и концом дуги. Дуга может принадлежать одному или нескольким сегментам.

При использовании Скрытых Марковских Моделей (СММ) для проблемы сопоставления карт движение объекта моделируется как Марковский процесс между узлами дорожной сети. Такие узлы дорожной сети не являются явно видимыми и рассматриваются как скрытые состояния. Координаты GPS в таком случае являются выходами скрытых состояний, чье распределение вероятностей зависит только от скрытых состояний.

Основными компонентами СММ являются вероятность эмиссии, вероятности изменения состояний, вероятности начальных состояний. Вероятности эмиссии определяют вероятность присвоения точки GPS (описывает позицию движущегося объекта) данной точке на сегменте дороги и определяется как  $P(z_t | point_i)$ . Вероятность эмиссии показывает вероятность наблюдения  $z_t$ , если движущийся объект находится в точке  $point_i$  в момент времени  $t$ . В работе [3] вероятность эмиссии моделируется с помощью функции Гаусса с центром в точке  $point_i$ :

$$P(z_t | point_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\|z_t - point_i\|_{gc}^2}{2\sigma^2}\right)$$

где  $\|z_t - point_i\|_{gc}$  – это расстояние между реальным положением движущегося объекта и точкой дорожного сегмента

$\sigma$  – погрешность измерений датчика GPS.

Вероятности изменения состояний определяют условную вероятность движения объекта из одного узла в другой узел дорожной сети. В работе [4] предлагается в качестве этих вероятностей принимать значения, зависящие от расстояния между узлами дорожной сети, следующим образом:

$$P(s_t | s_{t-1}) \approx \lambda \exp\{-\lambda(\|z_t - z_{t-1}\| - \|s_t - s_{t-1}\|)\},$$

где  $\lambda$ - параметр, который необходимо подбирать для каждой конкретной выборки (последовательности позиций движущегося объекта).

Вероятности начальных состояний показывают вероятность того, что начальной точкой движущегося объекта является  $point_i$ . В работе [5] начальные вероятности аппроксимируются из эмиссионных вероятностей относительно первых наблюдений  $P(z_1 | point_i)$ .

При реализации сопоставления карт было принято решение использовать готовую библиотеку [5] для языка python. Для поиска состояний СММ был использован алгоритм Витерби [7]. Алгоритм Витерби- это алгоритм динамического программирования для поиска наиболее подходящего списка состояний, называемым путем Витерби. В результате алгоритм находит последовательность наблюдаемых событий в контексте Марковских моделей. В случае сопоставления карт алгоритм для каждой GPS точки найдет наиболее подходящие узлы на карте. Сложность алгоритма составляет  $O(T*S)$ , где  $S$ -размер пространства событий, а  $T$ -число выходов.

Среди всех точек выбираются такие точки, которые имеют минимальную дистанцию до GPS-точки. Для каждой такой точки-кандидата и GPS-точки рассчитываются эмиссионные вероятности. После этого с помощью алгоритма Витерби для поиска наиболее подходящего списка состояний для текущих состояний (GPS точек). Итерации алгоритма продолжают до тех пор, пока существуют GPS точки, не сопоставленные точкам на дороге.

Список использованных источников:

1. Goh C., Dauwels J. и др., Online map-matching based on hidden Markov model for real-time traffic sensing applications // 15th International IEEE Conference. – 2012.
2. Kibal J., Hidden Markov Model based map matching.
3. Newson P., Krumm J., Hidden Markov map matching through noise and sparseness // 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. – 2009.
4. Mattheis S., Al-Zahid K. K. и др., Putting the car on the map: A scalable map matching system for the Open Source Community // INFORMATIK 2014: Workshop Automotive Software Engineering. – 2014.
5. Raymond R., Morimura T. и др., Map matching with hidden Markov model on sampled road network // 21st International Conference on Pattern Recognition (ICPR '12). – 2012.
6. HiddenMarkovpython [Электронный ресурс]. — Режим доступа: [https://pypi.org/project/hidden\\_markov/](https://pypi.org/project/hidden_markov/)
7. Forney G., The Viterbi algorithm // IEEE. – 1973.