

Virtualized traffic at metropolitan scales

David Wilkie¹, Jason Sewall², Weizi Li¹ and Ming C. Lin^{1*}

¹ Department of Computer Science, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, ² Parallel Computing Laboratory, Intel Corporation, Santa Clara, CA, USA

Few phenomena are more ubiquitous than traffic in urban scenes, and few are more significant economically, socially, or environmentally. Many virtual-reality applications and systems, including virtual globes and immersive multi-player worlds that are often set in a large-scale modern or futuristic setting, feature traffic systems. Virtual-reality models can also aid in addressing the challenges of real-world traffic – the ever-present gridlock and congestion in cities worldwide: traffic engineers and planners can diagnose system instabilities and explore control strategies in virtual worlds reconstructed from available sensor data. To create these VR systems with traffic mimicking real-world conditions, road network models need to be created and represented. Traffic needs to be realistically and efficiently simulated. To analyze real-world scenarios, the traffic conditions need to be estimated and reconstructed. To create virtual scenarios, such as simulated cities, traffic needs to be intelligently and efficiently routed. These applications all require research advances in road network capture and modeling, intelligent traffic routing and simulation, and traffic state estimation and reconstruction. New systems need to be designed that combine these components with visual and analytical infrastructure. In this paper, we present some state-of-the-art approaches for these areas as well as our vision for unified virtual-reality traffic systems that combine and integrate them to achieve virtualized traffic.

OPEN ACCESS

Edited by:

Anthony Steed,
University College London, UK

Reviewed by:

Guillaume Moreau,
Ecole Centrale de Nantes, France
Yiorgos L. Chrysanthou,
University of Cyprus, Cyprus

***Correspondence:**

Ming C. Lin,
Department of Computer Science,
University of North Carolina at
Chapel Hill, 201 S. Columbia Street,
Chapel Hill, NC 27599, USA
lin@cs.unc.edu

Specialty section:

This article was submitted to *Virtual Environments*, a section of the journal
Frontiers in Robotics and AI

Received: 07 December 2014

Accepted: 06 April 2015

Published: 07 May 2015

Citation:

Wilkie D, Sewall J, Li W and Lin MC
(2015) Virtualized traffic at
metropolitan scales.
Front. Robot. AI 2:11.
doi: 10.3389/frobt.2015.00011

Keywords: virtualized traffic, road networks, traffic simulation, traffic routing, urban scenes

1. Introduction

In both the physical world and many virtual cities, traffic is a ubiquitous phenomenon. Modeling these systems realistically and efficiently is an ongoing challenge that can lead to more immersive and more engaging virtual urban environments, as well as enable visual analysis and control to significantly improve real-world outcomes.

Typical urban environments are rife with traffic, which has significant implications both economically and socially around the world. Systematic failings are regular events; gridlock and traffic jams cost 2.9 billion gallons of wasted fuel and cost over 121 billion dollars every year in the U.S. alone, as discussed in Schrank et al. (2012). A major drive to solve these challenges entails using the road network more efficiently via *Intelligent Traffic Systems*, combining sensing, communications, and traffic controls in a managed system. Virtual-reality models have a key role to play in (1) visualizing real-world conditions, (2) visualizing potential alternative control strategies, (3) communicating conditions and proposals to the public, and (4) providing analysts a real-world context and interface.

As self-driving and other cars become more prevalent, a shared model of the road network could significantly improve their performance. Fleets of autonomous vehicles roving the road network can build a map, including up-to-date details about road blockages and closures. Such a model

can be used by the vehicles to plan more effectively than would be possible using only real-time sensor data or static road network maps. Rendered views of the model can let controllers and analysts validate the maps.

For the virtual world, research efforts have focused on creating, representing, and visualizing “digital urbanscapes” [e.g., Pausch et al. (1992), Cremer et al. (1997), and Donikian et al. (1999)]. These efforts have yielded significant results on a number of fronts, including satellite and aerial imagery melded into virtual globes, the creation of 3D models of cities, real-time rendered worlds, and new procedural modeling techniques. Google Earth and Microsoft Virtual Earth, which feature imagery of most cities and 3D models, are both products of this success. The creation of 3D city models from sensor data has been a focus area as has work on streaming imagery and geometry for real-time rendering of massive models on everything from high-end workstations to mobile devices.

Despite the impressive progress that has been made, these efforts focused only on static models of the world, ignoring dynamic elements such as traffic. The evolution of flight and driving simulators from single-user programs to vast virtual worlds magnifies the demand for a higher level of detail and for dynamic elements. The addition of traffic to these worlds, especially driven by real-time sensor data, would considerably enhance the realism and immersion of these worlds. For other applications, such as video games set in modern cities, city planning programs, and traffic simulators, the need for traffic is even more clear. Further, these traffic systems need to be realistic and need to behave intelligently, often while operating within a tight computational budget. There are existing systems for traffic simulation, [e.g., MITSIM (2011) and SUMO (2009)] and ongoing active research on new methodologies [see Delling et al. (2009) for details]. There has also been work on incorporating realistic traffic in virtual environments, such as Plumert et al. (2004), Thomson et al. (2005), and Wang et al. (2005), but these have focused on small scenarios: single roads or intersections, predominantly.

These applications demand advancements in modeling and simulating *virtualized traffic*, i.e., *the creation of virtual traffic flows directly from real-world data*. In this paper, we present a vision based on the synthesis of our recent work to construct virtualized traffic at metropolitan scales for VR applications. First, for the creation and representation of virtual road networks, we review related approaches and suggest a direction forward for virtual models suitable for the above applications, as discussed above. Our approach builds off of approaches to create simulation-enabled representations from GIS road network models, which are now publicly available in vast databases. This step involves building formal models atop of the noisy, human-authored data before creating a geometric representation. These models can also serve as a prior model to be fused with traffic sensor data. Second, to simulate virtualized traffic, we present approaches for creating hybrid traffic simulators that allow for level-of-detail simulations, enabling low-cost simulation of background traffic that can be switched to high-fidelity traffic simulation when necessary (see Section 2). Third, to estimate and communicate traffic conditions, we present a method that filters traffic sensor data, discrete in both space and time, to recreate a continuous estimate of traffic

conditions that can be used to create visualizations or predict future traffic patterns, as described in Section 3. Fourth, to enable low-cost intelligent routing for city-scale navigation, we present an approach that manages traffic flows via “*participatory traffic routing*,” in which the plans for individual vehicles are aggregated to coordinate their routes, as presented in Section 4. Using this approach, simulated, virtualized vehicles can react intelligently to the addition of roads or to anomalous conditions. Finally, in Section 5, we discuss our approaches and the directions for future work they point toward.

2. Road Networks

Road networks are an integral part of the traffic system. These networks can be models of real-world networks, captured using a variety of techniques, or can be artificial networks, generated to represent fictitious locales in virtual worlds. In either case, these networks are geometrically and topologically complex, making manual creation difficult for large-scale models.

There have been a variety of efforts directed at the problem of representing, designing, and acquiring road networks. Thomas and Donikian (2000) presented an early and influential approach to modeling human environments including streets, buildings, and free spaces. The model is used to create behavioral animations of pedestrians and vehicles in highly detailed scenarios. Willemsen et al. (2006) discuss “ribbon networks” for modeling the paths of agents in a road network, such that kinematic behavior is preserved. Wang et al. (2002) propose curves parameterized by their arc-length for motion-control and animation suitable for description of vehicle motion. Garcia-Dorado et al. (2014) take the novel approach of letting desired traffic behavior guide the creation of road networks and cityscapes. While these approaches do a good job of representing roads, they often do not address the challenges of creating large-scale models, especially from noisy data, and some road network models would not be able to support efficient simulation due to their geometric representations.

2.1. Levels of Detail

Road networks can be represented by differing levels of detail, from polygonal graphs to meshes and parameterized curves. These representations are suitable for different applications and have different costs associated for creation and use.

2.1.1. Directed Graph

A road network can be represented very simply as a directed graph for which edges, E , represent segments of roadway, and the vertices, V , represent intersections between roads and terminus points. Each edge can have attributes such as length, number of lanes, and speed limit. Representations such as this are suitable for online map display and for applications that do not require a high level of detail, such as simple routing operations.

However, a network such as this would not be suited for any visual analysis, animation, or highly detailed simulation, as the geometric details of the road are lost. Some problems would also be difficult to handle with this representation, such as determining which roads correspond to a sequence of global-positioning system (GPS) coordinates.

2.1.2. Polyline Road Networks

To address these shortcomings, road networks must be embedded in space; typically a 2D planar (or a height-field) is used. In this representation, the vertices, V^i , i.e., intersections, have coordinates indicating their positions, and each edge e , i.e., each road segment, has additional vertices, V_e^g , that approximate the geometry of the road. The edges and vertices can have additional attributes, A , as above.

This is known as the *polyline road network* representation (Safra et al., 2006), and it is the most common for road networks and is used for nearly all geographic-information system (GIS) applications, including in online mapping and routing applications, and are widely available in public data sources.

This representation also has its shortcomings. For one, lanes are not individually represented. This limits the level of detail at which simulations can be done and limits the level of analysis. Second, the polyline (or polygonal chain) geometry can be a source of significant error – real-world cars “cut” corners, following curvilinear paths derived from their kinematics (LaValle, 2006). This is exacerbated at intersections, at which simulated cars could take unrealistic turns.

This leads into the third shortcoming, that intersections are modeled as abstract, dimensionless junctions of edges. In reality, they have a geometric extent and can encompass complicated topological relationships. A consequence of this representation is that complex intersections either must be modeled using multiple vertices or their geometric and topological details must be sacrificed: modeling the intersections with multiple vertices means that vertices no longer solely represent the topological relationships between roads, but instead also represent the geometric extent and internal connections of the intersection.

2.1.3. Lanes, Geometry, and Intersections

To address these issues, additional details can be added to the representation. First, individual lanes can be represented. These lanes need geometric representations, which can be distinct or can reference a shared road segment geometry. Second, intersections also need a more complex representation: their geometric extents and topological relations need to be captured. This includes not only a bounding geometry but also *internal lanes* that connect the incoming and outgoing lanes. Third, a more detailed representation can be used for the various geometries, such as using a higher number of vertices or using a representation that can model curves, such as a spline. Finally, a model is needed for each intersection that defines the behavior of the intersection, whether the intersection uses a traffic light logic, stop sign, or some other rules of the road.

2.2. Road Network Creation

The complexity of road network geometry and topology is such that the manipulation and creation of their representations is computationally costly. Several approaches show promise for the automated creation of road network models, using GIS data, sensors, or procedural generation, depending on the desired application. We consider here only those methods for creating reconstructions or models of real-world road networks.

2.2.1. Creation from Sensing

Numerous methods for creating GIS road geometry from satellite imagery have been proposed, many of which are described in surveys including Park et al. (2002), Mena (2003), and Fortier et al. (1999). At this point, these methods are limited to creating relatively low detail networks due to the limited resolution of satellite imagery. Similar methods using aerial imagery from aircraft and UAVs have the potential to create more detailed road maps, along with 3D reconstructions of terrain and buildings.

Rich road network models can also be created by car-mounted sensors, which are becoming increasingly common as cars with auto-pilot, adaptive cruise-control, and self-driving functions enter the market. These sensors range from simple GPS devices to laser and radar sensors capable of mapping the geometry of the roads. Fleets of vehicles such as this have the potential, in the future, to use their sensors not only for real-time navigation but also to collaboratively create full 3D models of road networks. From the point clouds and reconstructed meshes created by these methods, logical road structures, as described above, need to be extracted in order to be used for traffic simulation and animation.

2.2.2. Creation from GIS Data

Digital representations of real-world road networks are commonly available in the form of GIS *polyline road networks*, but many applications, such as microscopic traffic simulation or high-quality animation, require a higher level of detail and accuracy. Some higher quality data sets already exist, but are usually limited to relatively small scale scenarios. Other techniques focus on extrapolating a detailed road network from the 2D GIS data. The SUMO traffic simulation system (Krajzewicz et al., 2012), can now generate networks from GIS input. A recent work (Wang et al., 2014) demonstrated an approach to creating 3D geometry from GIS inputs.

In Wilkie et al. (2012), we presented work on extrapolating plausible, high detail road network models suitable for microscopic simulation and animation from publicly available geographic-information system (GIS) data, i.e., polyline roads. The resulting model is composed of both (1) a graph that captures the topological relationships of the lanes that make up the road network and (2) a geometric representation that captures the shape of the roads as an optimal sequence of arcs and line segments. The model has the following features:

- It has sufficient data for high detail traffic simulation, visualization, and analysis, created automatically.
- The geometric road representation is C^1 continuous.
- Geometric operations such as calculating the distance between cars and the position along the road can be computed efficiently.

Our approach creates a *polyline road network* as described above. While each GIS data set has some notion of a road (or “way”), the usage is not necessarily consistent nor is the definition necessarily useful for further processing. We enforce a formal definition for a road on the data, easing filtering steps and enabling the creation of more complex road network features. We then create geometric representations for the roads and their intersections,



FIGURE 1 | A selection of road networks generated by our approach. (A) A large-scale highway scene. **(B)** A highway overpass created by our method for a 3D application. **(C)** A road network created by our approach and overlaid on a satellite image. **(D)** Ramps connecting to a divided highway. Wilkie et al. (2012). Image © IEEE.

including for details such as highway ramps. Some example road networks generated with this approach can be seen in **Figure 1**.

In addition, we plan to release the code for creating a 3D road network directly from GIS data as described in this section at: <http://gamma.cs.unc.edu/RoadLib>.

3. Traffic Simulation

Research on traffic simulation began in 1950s and is still an active field. Helbing (2001) serves as a good entry point to the field, although there have been many developments since. Doniec et al. (2008) apply strong behavioral models to agent-based simulation, as does Lu et al. (2014b). Shen and Jin (2012) discuss urban agent-based simulation with a focus on smooth lane changes, and recent work by Lu et al. (2014a) and focuses on preventing agent collisions in rural environments.

An essential component of traffic systems is all the vehicles – the cars, trucks, motorcycles, etc. The state of the vehicles is time-varying (except at very severe traffic jams), and a representation is needed for both the vehicles that are on the roads at a particular time and for the boundary conditions of the system.

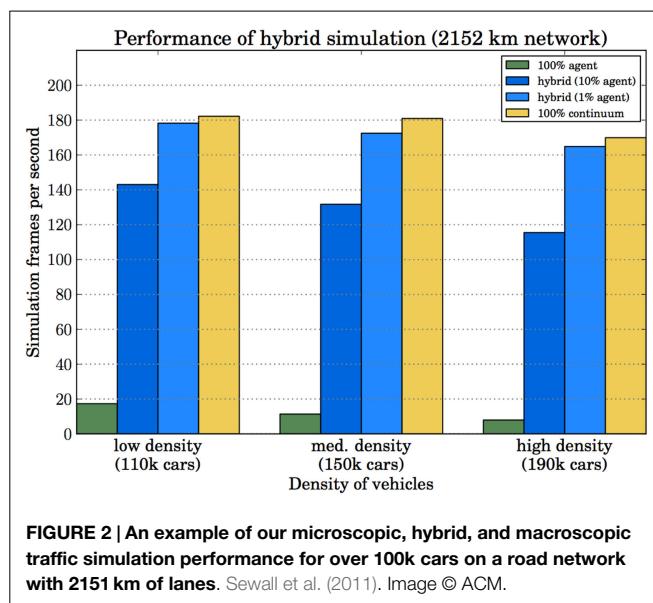
At any moment of time, we can refer to the *state* of the traffic system, which is the specification of all time-varying values, such as how many cars are on the road; how fast the cars are moving; and how many cars are entering the system. The representation of the state is dependent on how the dynamics of the traffic system are modeled. There are two primary approaches. First, every vehicle can be represented. This is the most direct approach: for every car in the real, there is a virtual car in the model with a known position along a lane, a velocity, and perhaps other parameters. This type of representation is referred to as *microscopic* in the literature or *agent-based* if the car models encapsulate some

decision making ability. Second, the vehicles can be represented as average quantities over some spatial discretization, i.e., a lane can be divided into cells, and each of these cells can contain a density value and velocity value that represent the average of these statistics in the real-world. This type of representation is called *macroscopic*. Each state representation has its associated dynamics definition, i.e., how the cars actually move. Like the state representation, there are two broad categories for these models – microscopic and macroscopic. For the former, individual vehicles are simulated, typically by calculating an acceleration from the state of the vehicle and the vehicle ahead of it along the lane. For the latter, differential equations define the evolution of the density and velocity fields over time. In either case, the model of the dynamics can be embodied in a simulation, which, if given initial and boundary conditions, can model the evolution of the traffic dynamics over time.

Real-world road networks span whole continents. Working with networks this large would be impractical, and so some region of interest must be extracted. Cars pass into and out of this region, and boundary modeling can be used to control the edge behavior of the system. “Internal boundaries,” such as large parking lots, can be similarly represented.

3.1. Animation and Immersion with Continuum Simulation

We have shown in Sewall et al. (2010) that it is possible to create realistic animations and immersive simulations using continuum (or macroscopic) simulations. This requires a continuum simulation model that is well defined for all road network features, including changes in speed limit, intersections, highway ramps, etc. In our work, we used the state-of-the-art *Aw-Rascle-Zhang* model of traffic dynamics (Aw and Rascle, 2000; Zhang, 2002) on



a fully featured road network as described above. This simulation is visualized using *carticles*, microscopic cars that are coupled with the macroscopic simulation. These cars match the traffic simulation state determined by the continuum simulator as well as determine when certain events occur within the simulation, such as lane changes. With this approach, we can easily simulate many thousands of cars at faster than real time (Sewall et al., 2010).

3.2. Level-of-Detail Simulation

Though the above work adds individual vehicles to the continuum traffic simulation, there are scenarios in which a more detailed, microscopic simulation is needed. If a user is interacting with the traffic, for example, heterogeneous, agent-like vehicles would create a more realistic and immersive experience. Given the computational cost of microscopic simulation, however, it is typically not feasible to simulate a large area using this method. To address this challenge, we proposed an approach in Sewall et al. (2011) in which subsets of the road network can be simulated using a microscopic simulator while the surrounding region can be simulated using the computationally cheaper macroscopic simulator. We are also able to *interactively* switch between simulation strategies for a particular region, allowing levels of detail based on, for example, viewing distance – all in real time. An example of our method's performance can be seen in Figure 2.

In addition, we have validated our simulation results by modifying string-distance metrics, such as LCSS and EDR, to compare agent-based, continuum, and level-of-detail simulation using NGSIM data. Our level-of-detail simulation has been able to achieve comparable accuracy while maintaining overall excellent performance, as shown in Table 1.

3.3. Other Issues

Above, we discussed our prior work on level-of-detail traffic simulation. However, further work is possible here as microscopic simulation still may not provide sufficient detail for some applications. Remember that in these simulation formulations, the space

TABLE 1 | Sequence comparison results (in percentile) along the NGSIM 101 freeway.

Metric	Agent-based	Continuum	Hybrid
Flux LCSS	0.934	0.541	0.820
Flux EDR	0.951	0.685	0.861

The rows show that our hybrid technique, in contrast to agent-based, microscopic traffic simulation, gives only a modest drop (in the mid to upper 80 percentile) in comparison scores with the real-world data.

in which the vehicles can move is restricted to a single dimension, the centerline of the lane. Real-world vehicles do not adhere to this constraint, and there are many applications that are only possible if this is relaxed: reconstructions and visualizations of anomalous traffic conditions, dangerous intersections, or accidents would require deeper models of vehicle movement. More immersive interactive traffic also requires these agents as real-world traffic does not neatly align itself to the lane. However, removing this constraint creates additional difficulties: allowing 2D movement of the cars requires providing a motion planning strategy that accounts for their non-holonomic kinematics, such as LaValle and Kuffner (2001), which is computationally costly.

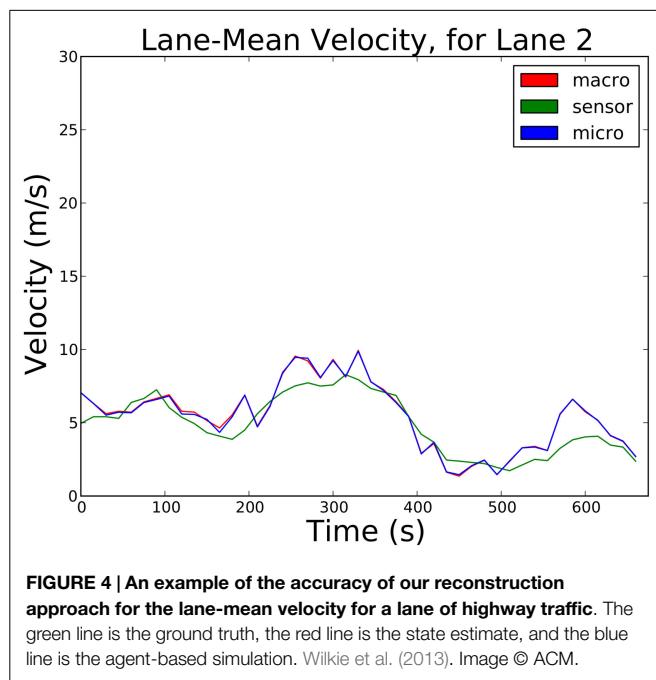
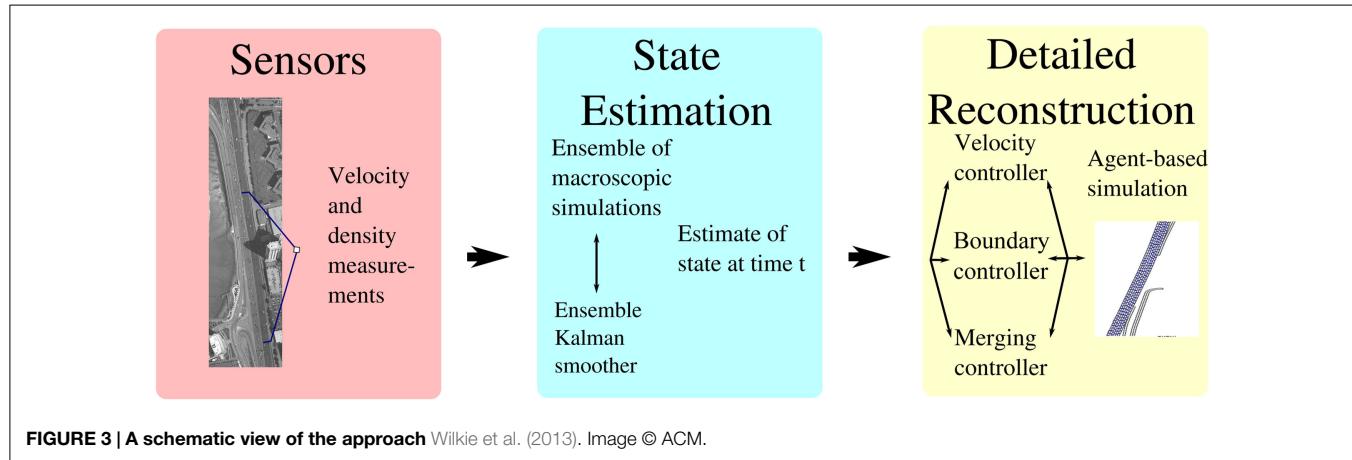
4. Traffic Reconstruction

As discussed above, one area of research has focused on the creation and representation of “digital urbanscapes,” which are digital representations of cities and, more generally, geographic areas. Commercial systems are now available that feature such models, obtained by using reconstruction techniques and aerial photography or other sensor measurements. However, these models are static: they lack dynamic elements, such as virtual people or traffic, which would increase their realism and user immersion.

Creating a digital representation of traffic that corresponds to real-world conditions is called “virtualized traffic” and was first introduced in van den Berg et al. (2009). To create virtualized traffic representations, in Wilkie et al. (2013), we proposed an efficient approach to creating animations that match observations, which could originate either from real-world traffic sensors or from a procedural system to allow the creation of controlled traffic animations. An overview of our approach is shown in Figure 3. We can see that there are two primary stages: data analysis and visualization. In the data analysis stage, the traffic state is estimated via a filter and a macroscopic traffic simulation model. In the visualization stage, a reconstruction of the traffic is created by an agent-based traffic simulator that is controlled via boundary conditions and a target velocity field. The created traffic can be displayed in 3D VR settings and is consistent with the traffic sensors and/or control points. An example of our reconstruction method's accuracy can be seen in Figure 4.

4.1. State Estimation

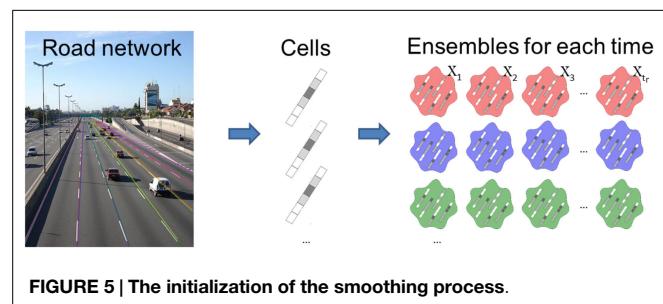
To create a traffic state estimate, the first part of our approach, we use sensor measurements (or control points), a macroscopic traffic simulator, and an Ensemble Kalman Smoother (EnKS), which was first introduced in Evensen (2003). This approach is summarized as follows. In Figure 5, we can see that each lane is discretized into cells, which hold continuum quantities. We create



multiple estimates of the traffic for each lane, which is called the ensemble. Then, we update each ensemble member to maximize the likelihood of the given observations, which again come from either sensor measurements or specified control points. As seen in **Figure 6**, this update is a combination of simulation, to evolve the ensemble member forward in time, and statistics, to align the member with the observations using a calculated Kalman Gain matrix. Finally, we use the mean of the ensemble of states for each timestep as the estimate of the traffic state.

4.2. Reconstruction Animation

Once a state has been estimated for each timestep, we create a traffic animation. To do this, we first create cars for each lane in a way that matches the target density field. This is done by separating the cars by a distance calculated from the target density. If a higher variance in the separation distances is desired, additional noise can be added to the distances or a Poisson process can be used



to place the vehicles. The vehicles are simulated using a simplified microscopic traffic simulator. Each car is first advected according to the target velocity field. Additionally, each vehicle has a velocity scaling factor, drawn from a bounded normal distribution, to model the variance in preferred driving speeds. The cars are also subject to a simplified leader-follower relationship. Other details, such as merging and boundary conditions, are also determined by the underlying state estimate. An example of reconstructed highway traffic, recreated at interactive rates, can be seen in **Figure 7**.

4.3. Other Issues

The method described above created detailed animations of a single highway. For larger scale reconstructions with tighter computational budgets, an alternative approach is needed to creating the estimate of the traffic state. One such interpolation method for offline data is the *adaptive smoothing method*, described in Treiber and Kesting (2013), which uses a superposition of two convolution kernels that account for traffic flows. It remains to be seen how well approaches such as this work in the context of visual reconstruction and whether observable details in the traffic are lost.

Another potential area for further research is fitting full microscopic simulation models to the traffic data. Our work creates vehicles using relatively simple agents that follow the estimated velocity field. Full microscopic simulations would require fitting behavioral parameters based on the data. However, a larger dataset is required to avoid the solution being underdetermined.

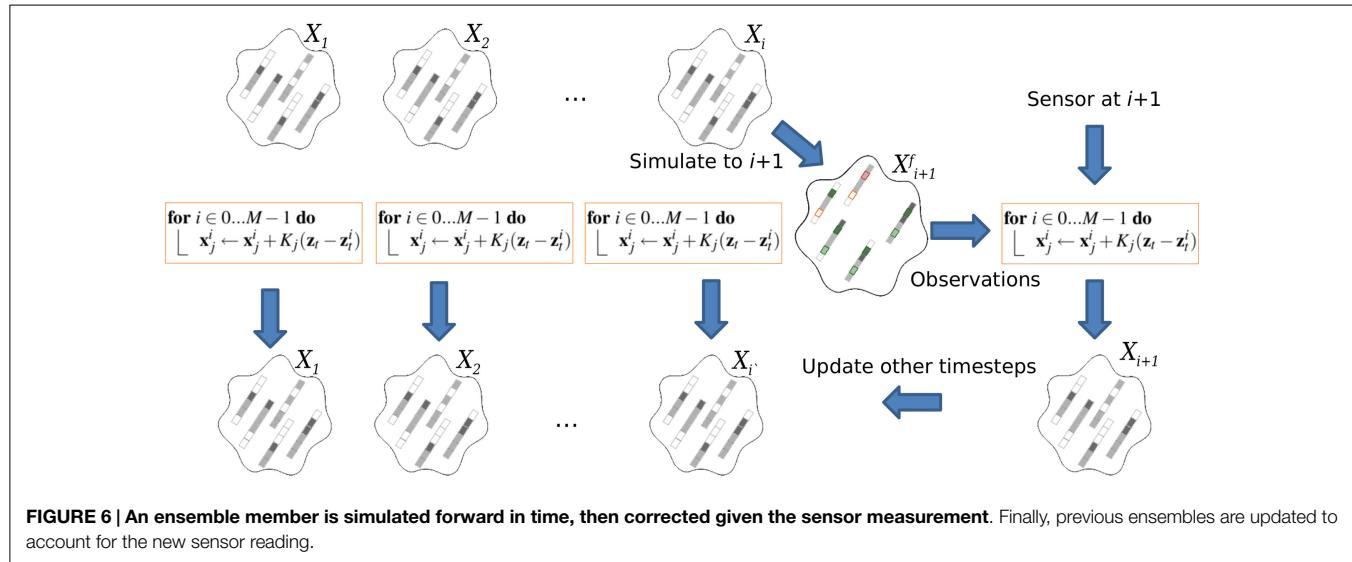


FIGURE 7 | On the bottom half of the divided highway is virtualized traffic matching real-world conditions as measured by loop-detector sensors. On the top half of the highway is virtualized traffic matching user-specified control points. The traffic is animated atop a GIS-derived road network. Wilkie et al. (2013). Image © ACM.

5. Vehicle Routing

The problem of routing a car through a road network can be abstractly described as the problem of finding a route through a graph, preferably a route that optimizes some desired criteria, which is solvable using Djikstra's algorithm or an A* algorithm. However, in their basic form, these approaches do not yield good performance for large road networks. Optimizations and alternative approaches that take advantage of the structure of the road network can yield computational speedups on the order of 10^5 and 10^6 , as described in Delling et al. (2009). These approaches build hierarchies of the road network and pre-compute distance tables and shortest path regions.

In the context of virtual reality, these optimization efforts are likely less necessary for most applications as the scale of the road networks is often smaller: few applications need continent-sized road networks. However, in virtual-reality applications, the challenge is that every vehicle must be routed, and they need to be routed in a way that does not artificially create congestion. If every car is routed naively, the "shortest" routes will be assigned more traffic than is optimal, creating congestion patterns in some

areas while leaving other parts of the road network underutilized. This can be especially frustrating in city simulation scenarios where the drivers are expected to behave intelligently to user road network designs.

5.1. Self-Aware and Participatory Route Planning

To address this challenge, we proposed in Wilkie et al. (2011) a method that can route thousands of vehicles in a manner that minimizes congestion and travel time. By utilizing a method for estimating each vehicle's effect on the traffic density field, our method allows the route planner to take other vehicles into account without any explicit cooperation.

Our method work considers the cars' own planned routes as a source of information. The method routes cars individually, and then uses the planned routes as estimates for the cars' trajectories in the near future. These trajectories, in aggregate, form an estimate of the future traffic pattern. This method is composed of (a) a *route planner* that computes paths for cars through a time-varying density field defined on the road network and (b) an *update* that modifies the density field according to the calculated route.



FIGURE 8 | On the left is the velocity field predicted by our routing system for two time intervals. On the right is the “ground truth” velocity field created by simulating the cars following their assigned routes. We can see qualitative similarities in the congestion patterns in the fields. Wilkie et al. (2014). Image © ACM.

The *route planner* makes use of a stochastic A*-search algorithm through a time-dependent density field. This field is composed of Gaussian distributions, discrete in time and space, defined over the road network graph, G . For each road explored, the cost of traversing the road is the estimated travel time

$$\tau_e(t) = \frac{\ell_e}{f_e(\rho_e(t))},$$

which is the length of the road ℓ_e divided by the estimated velocity, $f_e(\rho_e(t))$, which is a function of the estimated density. The function uses values for the maximum density and maximum velocity to determine the current velocity, which can be estimated using a number of models, as discussed in Greenshields et al. (1935) and Work et al. (2010).

Once a route has been planned, that route is considered an estimate for where that car will go in the future. For each car routed, the method adds a marginal amount of density to the road network along the planned path. For each edge of the route, the travel time estimates are used to calculate the probability that the car is on that edge during each time interval, $q_e(t)$. This car is then added to the density field for that road segment, taking into account the length of the road,

$$\rho_e(t) = \rho_e(t) + q_e(t)/\ell_e$$

The above approach uses a relatively simple model of traffic, however, and ignores certain phenomena, including congestion and delay caused by traffic lights and the spread of congestion and grid lock through a road network. In Wilkie et al. (2014), we presented an extension that models these phenomena in order to include their costs in the routing calculation. With these models,

our approach is able to predict the evolution of the traffic conditions in the future, allowing vehicles to take those conditions into account when routing and behave more intelligently. In Figure 8, for example, we can see the velocity field predicted by our router as well as the velocity field that occurred when the routes were simulated by a state-of-the-art microscopic traffic simulator.

5.2. Other Issues

Our approach can create routes for vehicles that simulate intelligent routing behavior and realistic traffic patterns. However, it is still relatively computationally costly. Years of research has gone into optimizing planning for static road networks, but it is not clear which if any of these approaches can be used in tandem with our approach.

6. Conclusion

Virtual environments, including training simulations, virtual globes, games set in cities, city simulations, traffic controller systems, and others, can all benefit from realistic, visual, efficient, and detailed traffic systems. Creating these systems, however, poses a number of ongoing research challenges. We have surveyed our approaches to creating virtual road networks and to simulating, reconstructing, and routing virtualized traffic using real-world data. We believe that these methods are among the first steps toward creating efficient virtualized traffic systems for realistic, metropolitan-scale virtual environments.

Much work remains to be done to fully achieve this vision, however. In each of the areas we discussed, there are promising future directions for research. For road network generation, current accuracy is limited to the accuracy and availability of the road network data and challenges remain in realistic procedural

road network generation. For microscopic (or agent-based) traffic simulation, more detailed simulation models can be developed, such as models that allow more realistic movement within and between lanes, and better driver behavior models for intersections and routing are needed to enable more realistic traffic simulation and avoid deadlock situations in dense scenarios. For virtualized traffic reconstruction from data, further research is needed to improve the efficiency of the approaches and extend them to handle GPS data signals and

complex road networks. For traffic routing, algorithmic improvements are needed to decrease the compute cost, especially in the context of interactive applications with a large number of vehicles.

Acknowledgments

This research is supported in part by National Science Foundation, Award #IIS-1247456.

References

- Aw, A., and Rascle, M. (2000). Resurrection of “second order” models of traffic flow. *SIAM J. Appl. Math.* 60, 916–938. doi:10.1137/S0036139997332099
- Cremer, J., Kearney, J., and Willemsen, P. (1997). Directable behavior models for virtual driving scenarios. *Trans. Soc. Comput. Simul. Int.* 14, 87–96.
- Delling, D., Sanders, P., Schultes, D., and Wagner, D. (2009). “Engineering route planning algorithms,” in *Algorithmics of Large and Complex Networks* (Berlin: Springer), 117–139.
- Doniec, A., Mandiau, R., Piechowiak, S., and Espié, S. (2008). A behavioral multi-agent model for road traffic simulation. *Eng. Appl. Artif. Intell.* 21, 1443–1454. doi:10.1016/j.engappai.2008.04.002
- Donikian, S., Moreau, G., and Thomas, G. (1999). Multimodal driving simulation in realistic urban environments. *Prog. Syst. Robot Anal. Cont. Des. (LNCIS)* 243, 321–332. doi:10.1007/BFb0110555
- Evensen, G. (2003). The ensemble kalman filter: theoretical formulation and practical implementation. *Ocean Dyn.* 53, 343–367. doi:10.1007/s10236-003-0036-9
- Fortier, A., Ziou, D., Armenakis, C., and Wang, S. (1999). *Survey of Work on Road Extraction in Aerial and Satellite Images*. Technical Report, 241: Center for Topographic Information Geomatics.
- Garcia-Dorado, I., Aliaga, D. G., and Ukkusuri, S. V. (2014). “Designing large-scale interactive traffic animations for urban modeling,” in *Computer Graphics Forum*, Vol. 33 (Wiley Online Library), 411–420.
- Greenshields, B., Bibbins, J. R., Channing, W. S., and Miller, H. H. (1935). “A study of traffic capacity,” in *Highway Research Board Proceedings* (Washington, DC: Highway Research Board), Vol. 14, 448–477.
- Helbing, D. (2001). Traffic and related self-driven many-particle systems. *Rev. Mod. Phys.* 73, 1067–1141. doi:10.1371/journal.pone.0094351
- Krajewicz, D., Erdmann, J., Behrisch, M., and Bieker, L. (2012). Recent development and applications of SUMO – Simulation of Urban MObility. *Int. J. Adv. Syst. Meas.* 5, 128–138.
- LaValle, S. M. (2006). *Planning Algorithms*. New York, NY: Cambridge University Press.
- LaValle, S. M., and Kuffner, J. J. (2001). Randomized kinodynamic planning. *Int. J. Rob. Res.* 20, 378–400. doi:10.1177/02783640122067453
- Lu, X., Chen, W., Xu, M., Wang, Z., Deng, Z., and Ye, Y. (2014a). Aa-fvdm: an accident-avoidance full velocity difference model for animating realistic street-level traffic in rural scenes. *Comput. Animat. Virtual Worlds* 25, 83–97. doi:10.1002/cav.1540
- Lu, X., Wang, Z., Xu, M., Chen, W., and Deng, Z. (2014b). A personality model for animating heterogeneous traffic behaviors. *Comput. Animat. Virtual Worlds* 25, 363–373. doi:10.1002/cav.1575
- Mena, J. (2003). State of the art on automatic road extraction for gis update: a novel classification. *Pattern Recognit. Lett.* 24, 3037–3058. doi:10.1016/S0167-8655(03)00164-8
- MITSIM. (2011). MITSIM. MIT Intelligent Transportation Systems. Available at: <http://mitsim.sourceforge.net/>
- Park, J., Saleh, R., and Yeu, Y. (2002). “Comprehensive survey of extraction techniques of linear features from remote sensing imagery for updating road spatial databases,” in *ASPRS-ACSM Annual Conference and FIG XXII Congress*.
- Pausch, R., Crea, T., and Conway, M. (1992). A literature survey for virtual environments – military flight simulator visual systems and simulator sickness. *Presence Teleoperators Virtual Environ.* 1, 344–363.
- Plumert, J. M., Kearney, J. K., and Cremer, J. F. (2004). Children’s perception of gap affordances: bicycling across traffic-filled intersections in an immersive virtual environment. *Child Dev.* 75, 1243–1253. doi:10.1111/j.1467-8624.2004.00736.x
- Safra, E., Kanza, Y., Sagiv, Y., and Doytsher, Y. (2006). “Efficient integration of road maps,” in *Proceedings of the 14th Annual ACM International Symposium on Advances in Geographic Information Systems*, GIS ’06 (New York, NY: ACM), 59–66.
- Schrank, D., Eisele, B., and Lomax, T. (2012). *Tits 2012 Urban Mobility Report*. (Texas: Texas A&M, College Station).
- Sewall, J., Wilkie, D., and Lin, M. C. (2011). “Interactive hybrid simulation of large-scale traffic,” in *ACM Transactions on Graphics (TOG)*, Vol. 30, (New York, NY: ACM), 135.
- Sewall, J., Wilkie, D., Merrell, P., and Lin, M. (2010). “Continuum traffic simulation,” in *Proceedings of Eurographics 2010, Computer Graphics Forum*, Vol. 29 (Blackwell Publishing), 439–448.
- Shen, J., and Jin, X. (2012). Detailed traffic animation for urban road networks. *Graph. Models* 74, 265–282. doi:10.1016/j.gmod.2012.04.002
- SUMO. (2009). SUMO – *Simulation of Urban MObility*. Available at: <http://sourceforge.net/projects/sumo/>
- Thomas, G., and Donikian, S. (2000). Modelling virtual cities dedicated to behavioural animation. *CGF* 19, 71–80. doi:10.1111/1467-8659.00399
- Thomson, J. A., Tolmie, A. K., Foot, H. C., Whelan, K. M., Sarvary, P., and Morrison, S. (2005). Influence of virtual reality training on the roadside crossing judgments of child pedestrians. *J. Exp. Psychol. Appl.* 11, 175. doi:10.1037/1076-898X.11.3.175
- Treiber, M., and Kesting, A. (2013). “Traffic flow dynamics,” in *Traffic Flow Dynamics: Data, Models and Simulation*, ISBN 978-3-642-32459-8, Vol. 2013 (Berlin: Springer-Verlag), 1.
- van den Berg, J., Sewall, J., Lin, M., and Manocha, D. (2009). “Virtualized traffic: reconstructing traffic flows from discrete spatio-temporal data,” in *Virtual Reality Conference, 2009. VR 2009. IEEE* (IEEE), 183–190.
- Wang, H., Kearney, J., and Atkinson, K. (2002). “Arc-length parameterized spline curves for real-time simulation,” in *Proc. 5th International Conference on Curves and Surfaces* (Brentwood, TN: CiteSeer), 387–396.
- Wang, H., Kearney, J., Cremer, J., and Willemsen, P. (2005). “Steering behaviors for autonomous vehicles in virtual environments,” in *Proc. IEEE Virtual Reality Conf*, 155–162.
- Wang, J., Lawson, G., and Shen, Y. (2014). Automatic high-fidelity 3d road network modeling based on 2d gis data. *Adv. Eng. Software* 76, 86–98. doi:10.1016/j.advengsoft.2014.06.005
- Wilkie, D., Baykal, C., and Lin, M. C. (2014). “Participatory route planning,” in *Proc. of SIGSPATIAL*, 2014. ACM.
- Wilkie, D., Sewall, J., and Lin, M. (2012). Transforming gis data into functional road models for large-scale traffic simulation. *IEEE Trans. Vis. Comput. Graph* 18, 890–901. doi:10.1109/TVCG.2011.116
- Wilkie, D., Sewall, J., and Lin, M. (2013). Flow reconstruction for data-driven traffic animation. *ACM Trans. Graph. (TOG)* 32, 89. doi:10.1145/2461912.2462021
- Wilkie, D., van den Berg, J. P., Lin, M. C., and Manocha, D. (2011). “Self-aware traffic route planning,” in *AAAI*.
- Willemsen, P., Kearney, J., and Wang, H. (2006). Ribbon networks for modeling navigable paths of autonomous agents in virtual environments. *IEEE Trans. Vis. Comput. Graph* 12, 331–342. doi:10.1109/TVCG.2006.53
- Work, D., Blandin, S., Tossavainen, O., Piccoli, B., and Bayen, A. (2010). A traffic model for velocity data assimilation. *Appl. Math. Res. Express* 2010, 1–35. doi:10.1093/amrx/abq002

Zhang, H. (2002). A non-equilibrium traffic model devoid of gas-like behavior. *Trans. Res. B Methodol.* 36, 275–290. doi:10.1016/S0191-2615(00)00050-3

Conflict of Interest Statement: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Copyright © 2015 Wilkie, Sewall, Li and Lin. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.