Automation of Ultra-Light Vehicles

Tyler C. Folsom
Computing and Software Systems
University of Washington
Bothell, USA
tfolsom@uw.edu

Rob Cotter
Organic Transit
Durham, NC, USA
elfcars@gmail.com

Abstract—Most urban trips can be handled by vehicles weighing less than the riders, using 30 times less energy than a car. Such vehicles are commercially available, but not widespread. Automation of bicycle-class vehicles could form the backbone of an urban transportation system. Such a system is competitive with commuter trains in terms of capacity, speed and cost, but uses much less energy since it is not moving the same vehicle weight. This paper presents an open-source system to automate ultra-light vehicles.

Keywords—Intelligent Transportation, Driverless, Vehicle Automation, Bicycles, ELF, Open-source, Autonomy.

I. Introduction

Present urban transportation systems are a poor match to energy sustainability. Most urban trips can be well served by a vehicle weighing less than the riders. There are only a few places in the world (such as Amsterdam) that use bicycles as a major part of the transportation system. Technology may extend the acceptability of vehicles weighing less than the riders. If vehicle automation becomes widespread, traffic accident rates are likely to plummet, making a motorcycle almost as safe as an SUV.

There are several factors that inhibit wider use of bicycles. These include adverse weather, avoiding perspiration in business clothes, disarranging a hairdo, safety, the need to transport items, and distances [1].

A vehicle that overcomes these barriers is Organic Transit's three-wheeled ELF, a vehicle bigger than a bicycle, but smaller than a car (Figure 1). It is a hybrid electric / human / solar powered vehicle, weighing 73 kg (160 pounds) with dimensions of 1.2 m (4 ft.) wide, 2.6 m (8.4 ft.) long, and 1.5 m (5 ft.) high [2]. The vehicle has a shell providing shelter from rain, protection in low-speed crashes, and improved aerodynamics. The vehicle can be operated by pedaling or driven by an electric motor. A solar panel on the roof can recharge a depleted battery in six hours. The ELF is legally a bicycle when the electric speed is limited to 30 km/h (20 mph), but it is easy to configure a scooter version traveling at higher speeds.

This work was funded in part by a grant from Amazon.

Figure 1. Organic Transit ELF

The ELF appeals to a niche market. However, a similar automated ultra-light vehicle, incorporating the driverless technology that is poised to reshape the automotive industry, could handle the majority of city trips. Such a vehicle would reduce the typical single occupancy vehicle (SOV) energy consumption from 30 kW per person to less than 1 kW at 50 km/h (30 mph), as shown in Figure 2 [3]. The ultra-light vehicle is far more efficient than the 6 to 7 kW per person achievable with transit, and has the advantage of going directly from origin to destination without schedules.

One may object that 50 km/h (30 mph) is way too slow, since it is assumed that cars and light rail move people at 80-110 km/h (50-70 mph). Such high speeds are unusual in the city, where stop-and go traffic is the norm. U.S. EPA city mileage ratings are based on a vehicle with average speed of 30 km/h (19 mph) that spends 43% of the time stopped or decelerating. Average speed in New York City is 13.7 km/h (8.5 mph) [4]. With 2000 new for-hire vehicles registering every month, the speed drops another 9% every four years. Scheduled average subway speeds range from 17 mph in New York and Tokyo to 24 to 29 mph in Seattle or Washington DC. A system that can operate vehicles at a constant 50 km/h with no stop-and-go is faster than what we have today, and it minimizes energy consumption. Energy sustainable transportation depends not just on vehicles, but on system design.

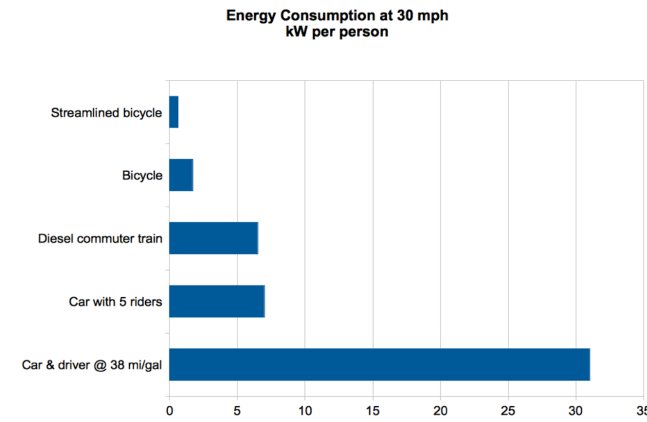

Figure 2. Vehicle Energy Consumption

A key reason for making ultra-light vehicles the transportation backbone is battery size. Most U.S. urban trips are less than 20 km (12 miles). A light vehicle with a 50 km range can be powered by a 10 kg lithium battery [5]. Batteries this light can be robotically swapped at refueling stations. The vehicle can operate as a shared driverless taxi, and swap out the battery between rides, eliminating range anxiety. A bank of depleted batteries enables harnessing renewable energy whenever the sun shines or the wind blows.

Another benefit is the vehicle's footprint. Three ELFs fit into a conventional parking space and take up an equivalent minimal space on the road.

The small size of an ELF also serves as an advantage when doing local deliveries. Rather than double parked large delivery vans, the micro sized ELF can even pull up onto sidewalks. Since the ELF devotes so little energy to propelling itself, it can carry many times its own weight, unlike most delivery vehicles. This allows for an extremely efficient delivery method. (Figure 3)

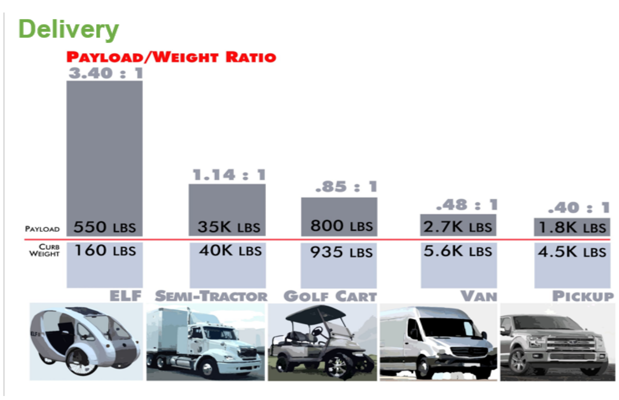

Figure 3. Payload / Weight Ratios

II. Bicycle Automation

A. Scope

The Elcano Project seeks to develop open-source hardware and software for vehicle automation at a price point that includes bicycle-class vehicles. We use a three-wheeled vehicle for several reasons:

- In the U.S., three-wheeled vehicles with pedals and a limited electric motor are legally bicycles; four-wheelers are not.
- Three-wheeled vehicles are stable at rest; two-wheelers are not.
- The format easily supports a shell.

B. Simplify the Environment

Machines have not yet mastered the difficult challenge of driving in any condition. Simplifying the environment is the key to deploying automated vehicles without needing breakthroughs in technology.

- An elevator is an autonomous vehicle operating in a highly simplified environment. Automated elevators were introduced in the first half of the 20th century, and there are few manually driven elevators left. [6]
- Rail vehicles have been automated since 1975 [7]. Today there are more than 130 driverless rail systems in operation.
- Small driverless transit vehicles on restricted paved paths have been operating since 1999 [8].

Much of the emphasis in driverless cars has been on a vehicle that can drive on any roadway at any time. Even if this vision were to be achieved, it would rob us of an important benefit of vehicle automation. Unconnected autonomous vehicles do nothing for congestion. When automated vehicles are connected, and controlled by an external system, road capacity increases by a factor of two to three [9]. It would be necessary to exclude manually operated vehicles from automated lanes, since they would snarl the flow. If the ultimate objective of automated vehicles is to operate in a reserved lane, why not design them that way from the start? It greatly simplifies the technology and decreases the cost of sensors and other equipment.

C. Software Design

The Elcano Project is designed for a vehicle that operates on a restricted lane in a controlled environment. For security purposes, the core software and hardware components are kept small, and do not depend on an Operating System (OS), Machine Learning, Cloud computing, or connection to the Internet. Since the system is capable of complete autonomy, vulnerability to unanticipated exploits is minimized. It is practical to mathematically prove that software below a certain size correctly implements its specifications [10]. The open-source version of the software is designed for experimental vehicles, making autonomy accessible to academics, students, and hobbyists. A deployed version would best be constructed from scratch using formal methods.

To keep software modules small and OS independent, the Elcano System is split among six or more microprocessors. It would be practical to port the entire system to tasks running under a Real-Time Operating System on a single processor, such as an ARM system used in smart phones.

The Elcano architecture runs on a stack of Arduino Atmel AVR microcontrollers. The processors are loosely coupled by passing serial data. The key processors are:

- Localization: Find current position from GPS and other instruments.
- Route planning: Map best route from current position to destination.
- Obstacle avoidance: Read sonars to avoid collisions.
- Pilot: Produce speed and turn profiles.
- Low-level vehicle control: Implement the desired speed and turn angle.

In addition to the core processors, data intensive sensors (such as vision) may have their own larger bandwidth processors. A Raspberry Pi is used to visually locate key objects and the edge of the lane. Traffic flow is optimized when ultra-light vehicles are choreographed by an external controller. This would be accomplished by a communications processor that receives desired speeds and relays the information to the Pilot. All processors other than the vehicle-specific low-level controller are designated as “high-level”. This divide means that any vehicle could be automated by building an appropriate low-level controller.

D. Feasibility

Can bicycle class vehicles replace mass transit? Here is a rough calculation of feasibility. People with better numbers are invited to improve this estimate.

Sound Transit operates Link Light Rail, connecting downtown Seattle to the SeaTac Airport. That section of the route is 22.4 km and was completed in 2009. Trains run every 6, 10 or 15 minutes except that there is no service between 1:00 AM and 5:00 AM. A trip from downtown Westlake Station to the Airport Station takes 37 minutes, making an average speed of 36.3 km/h. When that number is adjusted for an average wait time of 3 minutes at peak or 7.5 minutes off-peak, the average travel speed becomes 33.6 to 30.2 km/h (20.8 to 18.7 mph) A $131M contract buys 30 Siemens S70 light rail vehicles, which comes to $4.3M each [11]. The four-car trains carry an average of 203 passengers per trip and a weekday daily ridership on all lines is about 70,000 [12]; the new cars will seat 74 people with potential to pack up to 230 people per car. The vehicle dimensions are 29.4m x 2.65m x 3.87m (96.4 ft. x 8.7 ft. x 12.7 ft.) and empty weight is 44,000 kg (97,200 lb.) [13]. The key statistic on energy consumption per person is missing. The estimate from Wilson [3] is for a diesel train and does not apply to the electric S70. However, that number does appear to be approximately correct since Siemens states “each trip on a LRV instead of in a car reduces greenhouse gas emissions by more than 70 percent per mile” [14] which is less favorable than Wilson's 4.8:1 car to train energy ratio.

Maximum capacity for the Link Light Rail system would be 230 people per car times four cars times 10 trips per hour or 9200 people per hour; typical ridership is about a quarter of that figure. Could an automated ELF come anywhere close to this figure?

Personal Rapid Transit (PRT) is operational today. It is based on small automated vehicles which never stop at intermediate stations. PRT can operate on vehicle headways of 2 seconds, which would allow 1800 light vehicles per hour, with everyone seated [15]. Capacity of the train with everyone seated is 74 x 4 x 10 = 2960. Since the ELF is less than half the width of a train, and automation enables narrow lanes, two light vehicle lanes could replace one track. Since the ELF is only 1.5m (5 ft.) high, lanes could be stacked, and four lanes could fit into the space that a train requires, moving 7,200 people per hour on a light SOV system. Replace the single seat light vehicles with a dual seat model, and the capacity becomes 14,400 people per hour.

One can argue that trains can run on two-minute headways, and the peak capacity of the train is thus 27,600 people per hour. Such calculations are theoretical, but it is more instructive to look at how many people a system actually moves. The light SOV system always runs the right number of vehicles; it does not waste energy moving near-empty large vehicles.

Assuming that a separate vehicle makes each trip in an hour, light rail is making ten trips per hour with four vehicles, for 40 vehicles at a vehicle cost of $172M. The ELF list price is $8500, and the automation equipment under development is expect to add $2000. Thus 7200 ELFs would cost $75.6M. However, the maintenance costs of the light vehicle fleet may be higher, and the vehicle lifetimes shorter.

In an hour, the trains travel an average of 36.3 km. The total length of the trains is 40 * 29.4m = 1.12 km. Thus 3% of the track is occupied by vehicles. Light vehicles travel 50 km in an hour and one lane handles 1800 vehicles of 2.6m, increasing lane occupancy to 9.4%. Empty weight of vehicles per hour is 1760 metric tons for the trains and 131 for ELFs.

With enough batteries and a suitable motor, a light electric vehicle can be designed to operate at any speed. Suppose that the light vehicles operate on a dedicated lane, and never need to stop or slow before the destination, which maximizes energy efficiency. It is reasonable to operate them at the 50 km/h of Figure 2, resulting in a system that

- Has comparable capacity to light rail.
- Is 50% faster.
- Costs half as much for vehicles.
- Uses 15% as much energy.
- Is quiet.
- Is available 24/7 without schedules.

The main operating costs for transit are driver salary and fuel consumption, giving the driverless light vehicle system a significant advantage.

PRT has often been touted as the solution to covering the first and last km to a line-haul system. Instead, it makes sense for an automated light vehicle system to **be** the line-haul system. In classical PRT, the vehicles are captive to the guideway. It is feasible to allow bicycle-class vehicles to leave the paved guideway and operate under manual control. Such vehicles could be publically owned driverless taxis or privately owned.

III. Results

A. Software and circuits

Two automated recumbent tadpole tricycles have been built, with one shown in Figure 4. A Catrike chassis was converted to drive-by-wire with removal of chain, gears, handlebars and brake levers. Vehicle power is supplied by a hub motor, and activated by an electric bike controller that accepts a 5V analog throttle. Steering and brakes are operated by PWM signals to linear actuators. Battery packs supply 36V to the hub motor, 12V to the actuators, and 9.6V to the microprocessors, which regulate it to 5V.

Vehicle control has been demonstrated in four ways:

1. From on-board joystick
2. Via Bluetooth from a hand-held device
3. From a hobbyist Radio Control (RC) device
4. From the commands sent by the high-level processors.

Figure 4. Catrike modified for automation

The vehicle is designed for drive-by-wire, with no user mechanical control. This permits setting all vehicle speeds from an external traffic control computer that optimizes traffic flow. Drive-by-wire eliminates the danger that the rider will panic from close following distances, slam on the brakes, and cause a multi-vehicle crash. The rider can always notify the controlling computer of an emergency, or request to leave the platoon. The system will comply as soon as it can do so safely.

Work in progress centers on improving the performance of the high-level systems. Software and hardware are available as open source [16]. Documentation is dynamically updated [17].

Arduino processors for Localization and Pilot are housed on the High-Level printed circuit board (PCB). The task of localization is based on fusing information from GPS with dead-reckoning from speedometer and magnetometer. The task can conceptually be done with a Kalman filter, but the undergraduate students designing the system opted for a fuzzy estimator that finds the most likely position based on the instrument readings and history. This method provides an estimated position with an error on the order of a meter.

The estimated current position is then passed to a mapping unit that constructs the best path to the destination using the A* search algorithm [18]. This algorithm is widely used in computer games, but it can easily consume the bulk of a game system's processing power. This problem is avoided in two ways

1. The system operates on a directed graph of routes between intersections, which is far more efficient than allowing travel through any mesh point.
2. Route searching is done on a dedicated processor, which is loosely coupled to the other processors. Route updates do not have to happen more often than every few seconds. Meanwhile, the immediate route section is sent to the Pilot as needed.

Obstacle detection could be done by a variety of sensors, such camera, LIDAR, radar or sonar. Visual obstacle detection is a promising technology, but requires a processor with sophisticated algorithms and sufficient power. The Elcano Project will make use of a low-cost vision or LIDAR system when one becomes available. In the meantime. it uses an array of Maxbotic sonar detectors. These are housed on a Sonar Array PCB. The board allows up to eight sonars; one points backwards and the others are arranged in a semi-circle with each finding the closest object within a 30-degree arc. All sonars cannot be fired at the same time, since the sonar pulse echo from a neighbor could produce incorrect information. Firing each sonar in turn would result in an unacceptably long response time. Thus, the software and hardware fires three sonars at a time, grouped to avoid interference. The output from the sonar array to the pilot is a list of the distance to the closest object within each sector.

The Pilot processor takes the next path section from the Mapper and combines it with obstacle information. Obstacles will cause the Pilot to stop or find a route around. The Pilot then formulates a speed and turn profile. This information is passed to the low-level processor for implementation.

The Low-Level PCB provides robust connections between the High-Level system and the vehicle's actuators. The low-level system also needs to decide whether it is taking commands from High-Level, on-board joystick or an RC receiver.

There is another PCB that handles the proper power-on sequence required by the Kelly e-bike controller. This circuit board also cleans up the speed signal from the hub motor, producing 19 to 23 pulses per wheel revolution. This system allows finer speed measurement than was provided by the original cyclometer, which produced only one pulse per revolution. The Power-On PCB also slams on the brakes in the case of an emergency stop. The prior design for e-stop removed 36V power, but was dependent on proper execution of the low-level system. The Power-On board acts as a watchdog, and applies an e-stop if the low-level board fails to check in. The Power-On board is powered separately from the Low-Level board. The Power-On board will have power if the brakes do.

All interprocessor communication is serial. Most of it uses UART serial at 74,800 baud, with complete messages exchanged at 10 Hz. Messages are passed as a ring from

Localization to Mapper to Pilot to Low Level to Localization. There is a separate bidirectional serial link between the Localization Processor and the Vision Processor. The system uses a serial library to encode the messages exchanged. It also incorporates a CRC checksum with retransmit requests as needed. The communication between the Pilot and Sonar Array board uses Serial Peripheral Interface (SPI).

The link to Vision is low bandwidth. Inspiration is taken from the human brain primary visual cortex, which receives more inputs from higher cognitive centers than from the eyes. The vision system is told which features to look for and their expected position in world coordinates. The Raspberry Pi translates this information to expected pixel positions, which defines a region of interest (ROI) in the video frame from the camera [19]. Vision searches for a feature near this point, and returns an updated feature position along with a probability measure ranging for 1.0 for definite match to 0.0 if it does not see the feature.

B. System integration and test

The drive-by-wire system has been extensively validated. It shows that sending an analog signal to the throttle will make the trike go, sending a PWM signal will produce enough pull on the brake cables to stop the vehicle, and sending a PWM signal to the steering actuator will turn the vehicle. However, these signals are qualitatively different than the signals from the high level. The high-level signals need to achieve a specific speed, a particular rate of acceleration, and a particular turn angle.

Controlled operation was achieved by developing PID controllers [20]. These pointed to the need for better sensors. A commercial bicycle cyclometer measures zero for speeds less than 3 km/h, and may take several seconds to decide that the bike has stopped. This is an inevitable effect of getting a signal once per wheel revolution, where the wheel circumference is two to three meters. The speed signal from the hub motor allows for finer control. Steering is also problematic, and several turn angle sensors were tried. A magnetic shaft sensor proved to be too delicate. A sensor mechanically coupled to the steering column gave better results. However, the analog signal picked up significant noise. The turn sensor was improved in several ways

- The single-ended analog measurement was made differential.
- The resolution was increased by using an analog part responsive to 60° instead of 360°.
- The analog signal was digitized to further reduce noise.

The first tests of autonomous operations were

- Set the wheels for a circle and verify that the trike will move in an autonomous circle at a commanded speed.
- Set the wheels straight and verify that the trike will move a fixed distance.
- Move the trike straight, then turn the wheels to make a right-angle turn at a given speed.
- Make the trike autonomously drive in a rectangle.
- Show that the trike can drive between two points.

IV. Future Work

As of June 2017, the above autonomous operations have been done, but work to improve their reliability is continuing. Projects scheduled for summer include

- Improvements to speed and turn sensors, enabling finer control.
- Improved emergency-stop, effective on loss of computer power.
- Demonstrate route-following ability.
- Show ability to react to obstacles by stopping or going around.
- Improved sonar or laser obstacle detection.
- Visual landmark recognition and localization.
- Visual obstacle detection.
- Testing of integrated systems.
- Improved documentation.

In the fall, we expect to begin automating an Organic Transit ELF, and develop Advanced Driver Assistance Systems (ADAS) for it.

Others are encouraged to replicate and improve on this open-source hardware and software project. When satisfactory performance is achieved, the prototype electronics will be made available for purchase. We plan to develop an environment that enables academics and entrepreneurs to experiment with vehicle automation. With multiple vehicles available, a next stage would be to develop V2X communication, and choreograph efficient traffic flow.

References

[1] Heinen, E., K Maat, and B van Wee (2011), The day-to-day choice (not) to commute by bicycle, Transportation Research Board, Washington, D.C.

[2] https://organictransit.com

[3] D. G. Wilson, Bicycling Science, 3rd ed, Cambridge, MA, 2004

[4] http://www.wnyc.org/story/traffic-speeds-slow-nyc-wants-curb-car-service-growth/

[5] W. C. Morchin and H. Oman, Electric Bicycles, IEEE Press, 2006.

[6] http://www.npr.org/2015/07/31/427990392/remembering-when-driverless-elevators-drew-skepticism

[7] http://transportation.wvu.edu/prt

[8] http://www.2getthere.eu/projects/rivium-grt/

[9] S. E. Shladover, "Reasons for operating AHS vehicles in platoons," in Automated Highway Systems, P. A. Ioannou, Ed. New York, NY, Springer, 1996.

[10] Ivo Van Horebeek, Johan Lew, Algebraic Specifications in Software Engineering: An Introduction, Springer, 2012.

[11] https://m.soundtransit.org/About-Sound-Transit/News-and-events/News-releases/sound-transit-order-30-additional-light-rail

[12] https://www.soundtransit.org/sites/default/files/march-2017-service-performance-report.pdf

[13] https://w3.usa.siemens.com/mobility/us/en/interurban-mobility/rail-solutions/high-speed-and-intercity-trains/Documents/Houston1_DataSheet_2014_LR.pdf

[14] https://w3.usa.siemens.com/mobility/us/en/urban-mobility/rail-solutions/trams-and-light-rail/pages/metro-transit-light-rail-case-study.aspx

[15] M. Lowson and J. Hammersley, High Capacity PRT Station Design, https://www.princeton.edu/~alaink/Orf467F10/PRT@LHR10_Conf/PRT_StationDesignHammersley_paper.pdf

[16] https://github.com/elcano/elcano.

[17] http://www.elcanoproject.org/wiki.php

[18] J. Matthews, "Basic A* Pathfinding Made Simple" in AI Game Programming Wisdom, S. Rabin Ed., Charles River Media, Hingham, MA, 2002

[19] J. F. Hughes et al., Computer Graphics: Principles and Practice, 3rd Ed., Pearson Education, 2014

[20] E. Forrester, "Intelligent Steering using PID Controllers" in AI Game Programming Wisdom 2, S. Rabin Ed., Charles River Media, Hingham, MA, 2004