Marine Range Estimation Calculator for EV Systems

STUDENTS: Ean Barnawell, Elliot Liu, Ryan Ostrander, Theo Reid

PHOTON

XX

Background

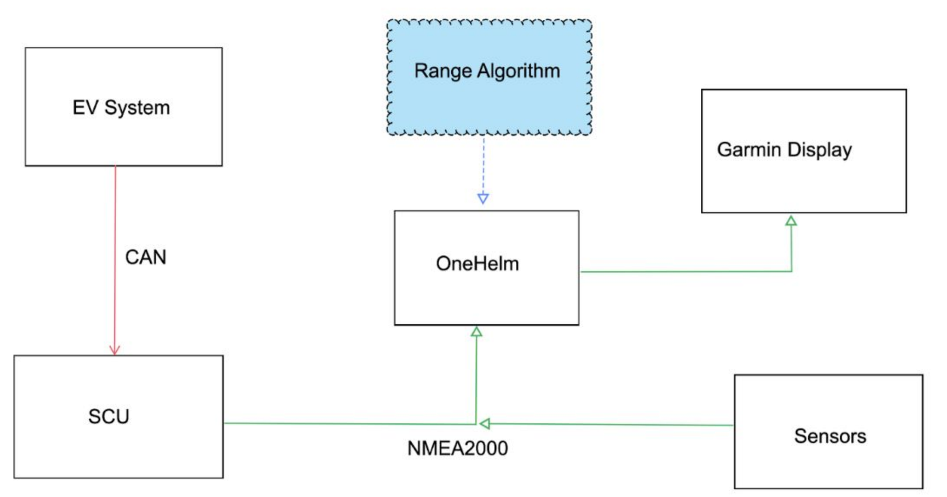
- Photon Marine is a company that specializes in designing and manufacturing electric propulsion engines for marine usage. This is either done by retrofit or by complete top down installation on newly built vessels.
- Currently they are working towards scaling production of their propulsion engines and are actively doing sea trials on their 21ft center console vessel built by a third party manufacturer (Silverback Marine). These sea trials are actively collecting useful data to improve and perfect their software and hardware.
- Our initial project scope when pitched to us was a marketing tool that a potential customer could use on Photon's website, in which a potential customer could compare their current combustion engine to Photon's fully electric engine.
- After extensive research into ICE data and the data Photon currently has as well as communication between are team and Photon, we came to the conclusion that it would be best if we changed the scope of our project.

Objective

- Develop a tool that estimates the range remaining in nautical miles using data available from the onboard battery management system and other ancillary inputs through the NMEA bus.
  - This involves predicting an estimate of distance remaining in nautical miles before recharging is required, incorporating a battery reserve of around 10%.
- Fully integrate range estimation tool onto existing hardware, onboard a 21-foot center console vessel powered by Photon Marine's EV propulsion system.
- Avoid major fluctuations in the estimated range by use of a cached average energy consumption based on performance from previous test runs and rolling average range algorithm.
- Take into account external parameters, such as adverse environmental conditions, that can alter the range of the vessel on a subsequent return trip.

OneHelm System and Implementation

- OneHelm is the name of a piece of Garmin Software that has the ability to be fully customized. It is therefore the system that Photon uses for their onboard UX.
- CAN data and NMEA2000 data from the BMS and other sensors on the vessel are received on an RPi located near the onboard display.
- Within the RPi is where our algorithm is implemented as it's own class that takes in the adjusted data and outputs range and time remaining to the OneHelm server.
- This data is then displayed on the onboard vessel.



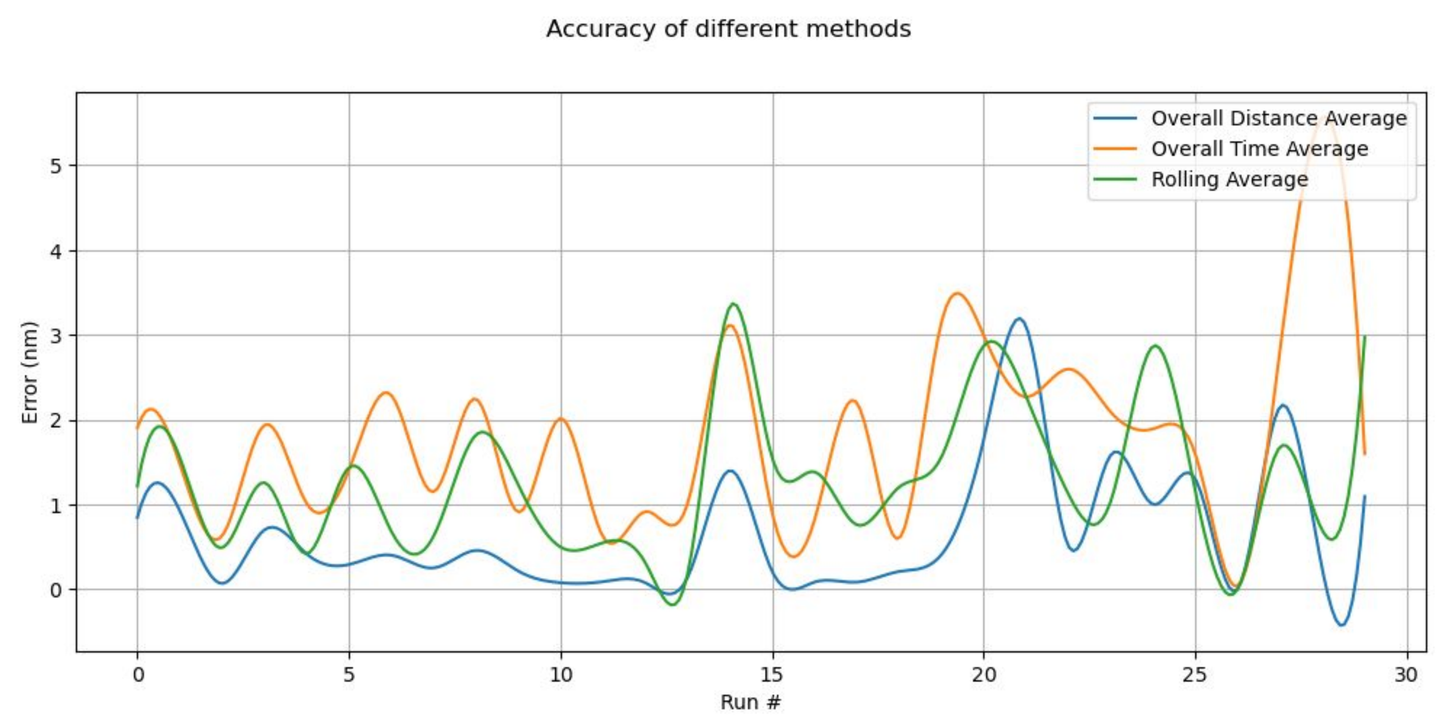
Algorithm Design

- **Overall Average:** A classic method which utilizes the historical energy consumption data to project future range performance based on energy remaining in battery. Two versions - one distance-based, one time-based multiplied by avg speed. **PROS:** Simple and accurate. **CONS:** Does not take user's immediate actions into account.

$$operating\ efficiency\ [kWh/NM] = \frac{energy\ used\ [kWh]}{distance\ traveled\ [NM]}$$

- **Rolling Average:** Builds on the overall average but incorporates more immediate performance based on the past n_minutes of data. A weighting parameter controls how much of the algorithm depends on the overall efficiency vs. the rolling efficiency. **PROS:** Incorporates a more immediate consumption rate based on recent history of the current trip. **CONS:** Complicated to implement and stores more global variables than the overall average. Proved to be less accurate than expected.

$$Rolling\ Average = (efficiency_{rolling} * \alpha) + (efficiency_{overall} * (1 - \alpha))$$



- **Average error of different methods:**
  Overall Distance Average: 0.67 nm | Overall Time Average: 1.79 nm | Rolling Average: 1.29 nm

Algorithm Testing

Data Stream Simulation

- Since our team was off-site we needed to develop a way to test the algorithm remotely on .csv files of historical data. To do this we put our function within a for-loop cycling through each row of the data. This simulated a data stream similar to what the NMEA2000 would be providing on the boat.
- To ensure our algorithm referenced variables in the same manner the oneHelm server would, we further created a function to transform the .csv data row values into class variables matching the names of the oneHelm server class. In doing this, we could be confident that our range algorithm could be implemented directly into the boat's script without any major changes in code.

Accuracy

- Comparing actual range with predicted range proved to be more challenging than expected with the available data. In the ideal case, one would gather data from precise test runs where the battery is charged and discharged to the same level each time, providing benchmarks for comparison. In our case, each dataset was different.
- We chose two arbitrary points in the dataset that were n_minutes apart and compared the distance traveled in that time with the predicted range n_minutes ago minus the current predicted range. This method gives a rough indicator of the error, but since the bottom baseline is still a prediction, it is not an extremely accurate value.

	tripDistance	Dist Prediction (nm)
200	0.0	19.488
201	0.000059	19.488
202	0.000059	19.488
203	0.00014	19.488
204	0.000238	19.488
...	...	...
2495	8.907774	10.904
2496	8.909433	10.904
2497	8.911101	10.904
2498	8.912742	10.904
2499	8.914341	10.904

Future Work and Limitations

Future Goals

- Establish a warning system with remaining range to alert the pilot at the halfway point in the trip.
- Work to set up a way to show optimum speed to maximize range for given heading and conditions.
- After optimum speed display, set up a display output of remaining runtime at median speed that would be calculated over the run history.
- Final stretch goal: Work to integrate with a weather app so that all the time/range remaining could be used to help plot the best possible courses for pilots in all weather conditions, taking safety into the primary consideration.

Limitations

- Calculations are conducted with speed over ground (SOG) and does not factor in speed through water (STW).
- Range calculation does not take into consideration adverse weather conditions which can cause changes in water current and wind speed.
- Since Photon is in prototype stage, eventual hardware/system changes might affect how algorithm will function.

Conclusion

- We implemented two methods for estimating the range remaining of an EV-powered vessel.
- Overall average, which utilizes historical energy consumption data to predict future range performance.
- Rolling average, takes into account more immediate performance metrics to calculate the average energy consumption on a rolling interval basis.
- Given more time, we would have liked to add more functionality for the onboard display – predicting time remaining, active warning system, and taking into account live weather conditions.

Special thanks to industry sponsors Brian and Marcelino, faculty advisor John, software engineer Matt and everyone else at Photon, for making this project possible! We wouldn't have been able to do this without your help.

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY of WASHINGTON

ADVISORS: John Reese

SPONSORS: ELECTRICAL & COMPUTER ENGINEERING DEPARTMENT, UNIVERSITY OF WASHINGTON