

Tesla, Inc. (Petitioner)
v.
iQar Inc. (Patent Owner)
Petitioner's Demonstratives

Case No. IPR2024-00545
U.S. Patent No. 8,972,161
Before Jason W. Melvin, Sean P. O'Hanlon, and
Brent M. Dougal,
Administrative Patent Judges



DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Table of Contents

• The '161 Patent and Petition Overview	3
• The claims are unpatentable over Neiss	8
• Neiss discloses or renders obvious “receiving a route”	13
• Neiss discloses “receiving a route” and “segmenting said route”	43
• The claims are unpatentable over Schramm-Clapper	51
• The Schramm-Clapper combination accounts for different per-segment speeds..	52
• The Schramm-Clapper speeds are “efficient speeds that achieve a best summation of said energy values”	61

'161 Patent and Petition Overview

'161 Patent: Independent Claim 1

1. A method of managing a power consumption of a vehicle, comprising the steps of:
 - (a) receiving a route at a computer of said vehicle;
 - (b) segmenting said route into a plurality of segment destinations;
 - (c) calculating a plurality of energy values consumed by said vehicle to travel said route based on a plurality of variations in a target speed of said vehicle between pairs of said segment destinations;
 - (d) determining a plurality of efficient speeds that achieve a best summation of said energy values; and
 - (e) notifying an operator of said vehicle about said efficient speeds to travel between each of said segment destinations.

'161 Patent Independent Claim 10

10. A method of managing a power consumption of a vehicle, comprising the steps of:

- (a) receiving a route at a computer of said vehicle;
- (b) segmenting said route into a plurality of segment destinations;
- (c) calculating a plurality of energy values consumed by said vehicle to travel said route based on a plurality of variations in a target speed of said vehicle between pairs of said segment destinations;
- (d) determining a plurality of efficient speeds that achieve a best summation of said energy values; and
- (e) applying power corresponding to said efficient speeds to an engine of said vehicle to travel between each of said segment destinations.

TESLA-1001, 26:15-27:3; Pet., 3

5

'161 Patent Overview

In this example, the route may be split into five segments of 5 km each with the following altitudes measured at the end of each segment: 0 (beginning of segment 1), 200 m (end of 1), 100 m (end of 2), 400 m (end of 3), 100 m (end of 4), 0 (end of 5). We may also make the simplifying assumption that the road grade is constant between measured points, and the road grades are calculated to be: +0.04 (seg 1), -0.02 (seg 2), +0.06 (seg 3), -0.06 (seg 4), -0.02 (seg 5). The target average speed is 100 kph.

For the sake of simplicity in this example, we can then calculate the amount of energy required to drive at a constant 100 kph speed for the entire route; as well as for 16 other combinations of the speeds 95 kph (26.39 mps), 100 kph (27.78 mps), and 105 kph (29.17 mps). In practice, this simulation may be run for hundreds or even thousands of combinations of speeds which may be tested to find the optimal speed to drive each segment. It is assumed that we enter the first segment traveling 100 kph (27.78 mps). It is also assumed that the speed listed for the segment is the final speed of the segment and the vehicle accelerates linearly throughout the segment.

The Table below shows the results of the energy calculation for all of the combinations tried. The lowest energy usage is 4275.94 Watt-hours. This was obtained by going 95 kph (seg1), 105 kph (seg2), 95 kph (seg3), 105 kph (seg4), 100 kph (seg5). The average speed over all 5 segments is still 100 kph, but the energy used is 10.2% less.

iterations	seg 1	seg 2	seg 3	seg 4	seg 5	Energy	units
1	27.78	27.78	27.78	27.78	27.78	4763.2484	Whr
2	26.39	29.17	27.78	27.78	27.78	4536.889	Whr
3	26.39	29.17	26.39	29.17	27.78	4275.938	Whr
4	26.39	29.17	26.39	27.78	29.17	4382.5026	Whr
5	26.39	29.17	27.78	26.39	29.17	4950.1438	Whr
6	26.39	29.17	27.78	29.17	26.39	4438.3467	Whr
7	26.39	26.39	29.17	29.17	27.78	4942.6325	Whr
8	26.39	26.39	29.17	27.78	29.17	5055.9124	Whr
9	26.39	26.39	27.78	29.17	29.17	4784.1865	Whr
10	29.17	26.39	27.78	27.78	27.78	5013.2565	Whr
11	29.17	26.39	26.39	29.17	27.78	4745.585	Whr
12	29.17	26.39	26.39	27.78	29.17	4852.1495	Whr
13	29.17	26.39	27.78	26.39	29.17	5137.6625	Whr
14	29.17	26.39	27.78	29.17	26.39	4914.7142	Whr
15	29.17	29.17	26.39	26.39	27.78	4602.8845	Whr
16	29.17	29.17	26.39	27.78	26.39	4493.3028	Whr
17	29.17	29.17	27.78	26.39	26.39	4765.4617	Whr

Asserted Grounds

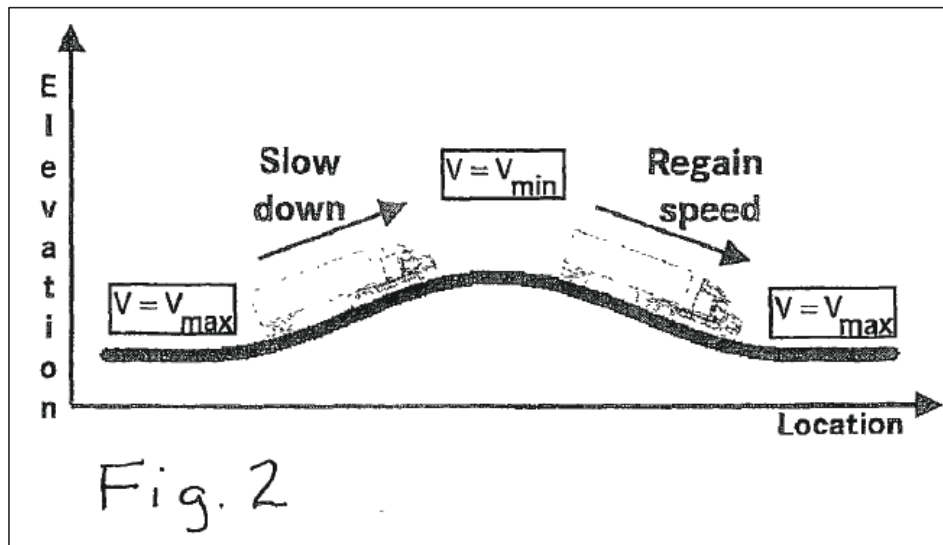
Ground	Claim(s)	35 U.S.C. §103
1A	1-3, 5-6, 8-15, 17-18	Neiss
1B	4	Neiss-Hirasago
2	1, 3, 6-10, 12-13, 15-18	Schramm-Clapper

Pet., 1.

7

The Claims are Unpatentable Over Neiss

Overview of Neiss



[0031] The basic idea of Predictive Cruise Control (PCC) is to replace or supplement the set speed of a conventional cruise control with a speed band. Forward-looking terrain information is used to determine, as a function of position, a desired speed inside the speed band (as shown in FIG. 1 for example), in order to maximize fuel savings. In the ideal case, the vehicle that is equipped with PCC slows down as it moves uphill until it reaches a minimum speed on top of the hill, and regains speed on the downhill part, thus completely converting the potential energy into kinetic energy. (See FIG. 2.) This strategy prevents unnecessary braking, or delays braking until it is unavoidable.

Neiss's Computer Performs Control Parameter Optimizations Along "the Route Ahead of the Vehicle"

[0015] In the predictive cruise control system according to the invention, a vehicle operating cost function is defined, based on a plurality of environmental parameters, vehicle parameters, vehicle operating parameters and route parameters. As the vehicle travels over a particular route for which route parameters, such as road gradient and curvature, are stored in a road map, sensors aboard the vehicle detect environmental and vehicle operating parameters, including at least vehicle speed and position relative to the road map. As the vehicle proceeds, an onboard computer iteratively calculates and stores in a memory vehicle control parameters that optimize the vehicle operating cost function for a predetermined distance (referred to as a "Prediction Horizon") along the route ahead of the vehicle. The optimal vehicle control parameters for the Prediction Horizon are then stored in a memory and continuously updated and replaced by new data as the vehicle (and hence the Prediction Horizon) moves along, thereby adjusting the "optimal" control parameters to reflect actual vehicle historical operating experience during the journey. The vehicle is then controlled by reading the optimized vehicle control parameters from the memory, corresponding to the current position of the vehicle.

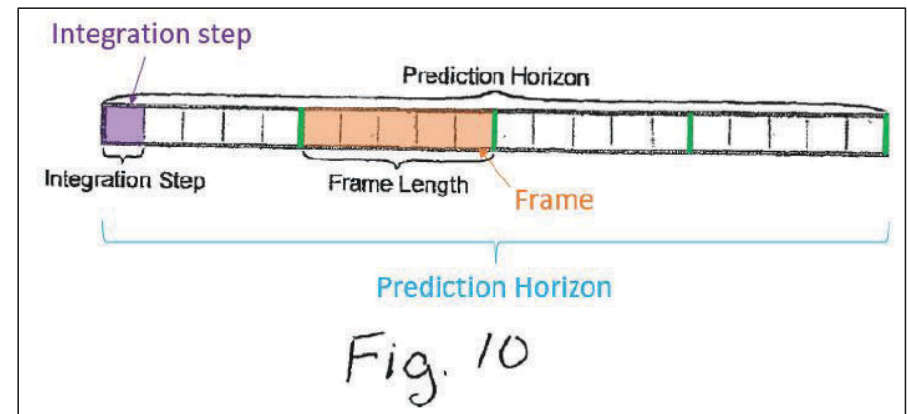
TESLA-1005, [0015]; Pet., 6-7

10

Neiss's Prediction Horizon

Papanikolopoulos Declaration

38. The “Prediction Horizon” is “a predetermined distance ... along the route ahead of the vehicle.” *Id.*, ¶[0015]. As illustrated in Figure 10, shown below, the “Prediction Horizon” along the route ahead of the vehicle “is separated into Frames.” *Id.*, ¶[0170]. Each frame corresponds to “a specified distance” to be covered by the vehicle and is composed of a number of integration steps h . *Id.*, ¶[0169]-[0170] (“the Prediction Horizon must be a multiple of the Frame Length (also measured in number of integration steps h)”).

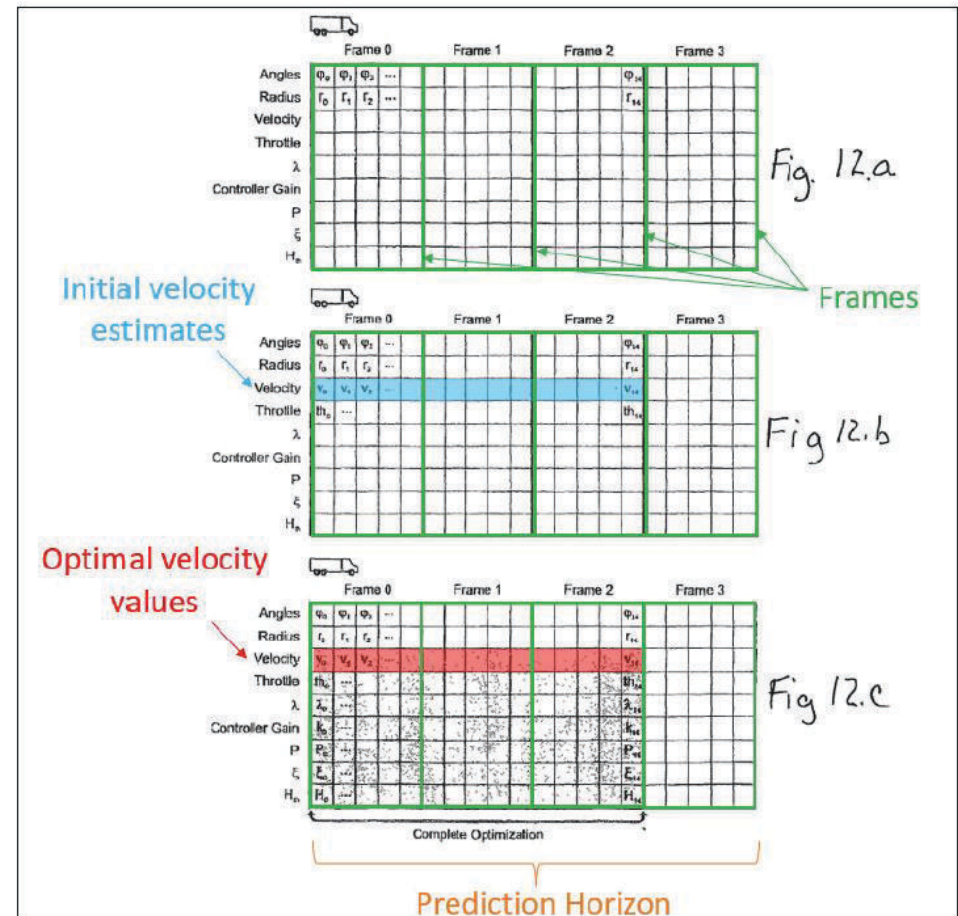


Neiss's Optimization Over the Prediction Horizon

Papanikolopoulos Declaration

39. To perform the minimization of the above-described "vehicle operating cost function," **an iterative optimization algorithm** is executed to calculate vehicle control parameters including "optimal velocity, throttle pedal positions, and controller gain series" "[f]or an area around the current vehicle position[]" ("Prediction Horizon")." *Id.*, ¶¶[0015], [0162].

41. As part of the optimization, Neiss's cruise control system uses an initial set of estimated vehicle control parameters to compute a first set of optimized vehicle control parameters, including vehicle speed, for frames along the prediction horizon. TESLA-1005, ¶¶[0171]-[0181], [0186], [0048], [0149]-[0160], FIGS. 12a-c. The initial estimates and the computed, optimized vehicle parameters are **stored in a lookup table** corresponding to the frames of the prediction horizon,



Issue 1: Neiss discloses or renders obvious “receiving a route”

Neiss discloses or renders obvious “receiving a route”

1. A method of managing a power consumption of a vehicle, comprising the steps of:

- (a) **receiving a route** at a computer of said vehicle;
- (b) segmenting said route into a plurality of segment destinations;
- (c) calculating a plurality of energy values consumed by said vehicle to travel said route based on a plurality of variations in a target speed of said vehicle between pairs of said segment destinations;

- **Issue 1A:** Neiss’s Prediction Horizon is a “route”
- **Issue 1B:** Neiss’s iterative receipt of successive Prediction Horizons corresponds to the claimed “receiving a route”
- **Issue 1C:** Neiss renders obvious receiving an entire, start-to-finish route all at once

Issue 1A: Neiss's Prediction Horizon is a "route"

Route is a Path From One Point/Place to Another

Declaration of Dr. Hoff, Patent Owner's expert

127. A POSITA would have understood the term “route” in its ordinary sense to mean a road, path, or course from one place to another. See, e.g., Ex. 2001 at 1189 (defining “route” as “[a] road, course, or way for travel from one place to another”).

Patent Owner's Sur-Reply

Next, the word “route” in the '161 patent does not refer simply to a road. As Petitioner itself admits, a “route” is a “*path* from one point/position to another point position.” Reply at 3 (emphasis added). This is how the '161 patent repeatedly uses the term “route.” Throughout the patent, a “route” always has a start, an end, and a

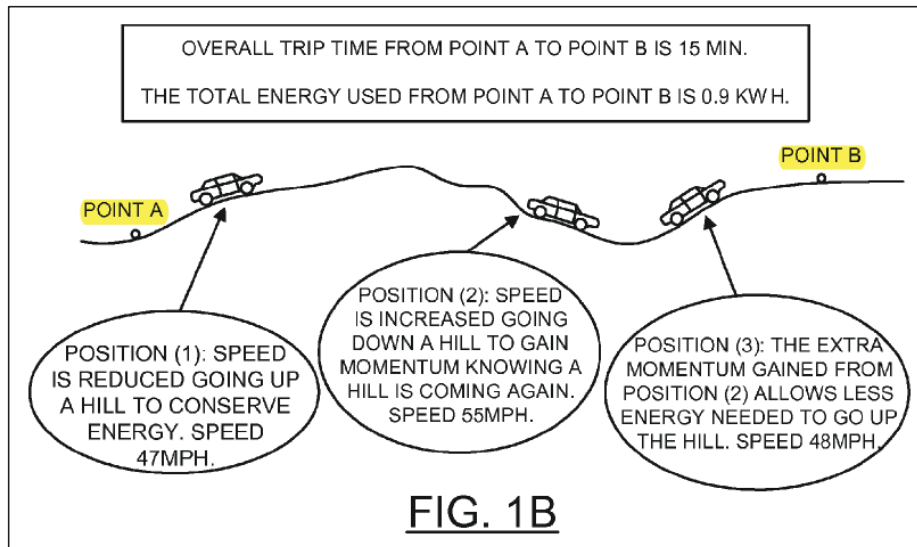
Petitioner's Reply

The independent claims require, *inter alia*, “receiving a route” and “segmenting said route.” Both parties agree that the term “route” should be afforded its plain and ordinary meaning. See Pet., 14; POR, 18. The Petition explained that this encompasses “a path traveled by the vehicle.” See Pet., 2, 14. Consistent with this meaning, iQar's expert, Dr. Hoff, cited a dictionary definition of “route” in testifying that the plain and ordinary meaning of “route” is “a road, path, or course from one place to another.” POR, 18; EX2003, ¶¶126-128 (citing EX2001). A place is simply a “particular position or point in space.” TESLA-1054, 3. Thus, the plain and ordinary meaning of “route” is a path from one point/position to another point/position.

Pet. Reply, 3; EX2003, ¶127

Route is a Path From One Point/Place to Another

'161 Patent



Deposition of Dr. Hoff

9 Q. All right. So here I think you've shown
10 **Figure 1B of the '161 patent; right?**

11 A. Yes, 1B; correct.

12 Q. Okay. And what is this figure showing?

13 A. Well, this is one figure of a pair of
14 figures from the '161 patent, trying to illustrate
15 how the '161 method varies from prior art.

16 Q. And there are two points shown on this
17 figure, point A and point B.

18 Do you see that?

19 A. I do.

20 Q. And it appears that there's a path
21 between point A to point B over which the vehicle
22 is traversing; right?

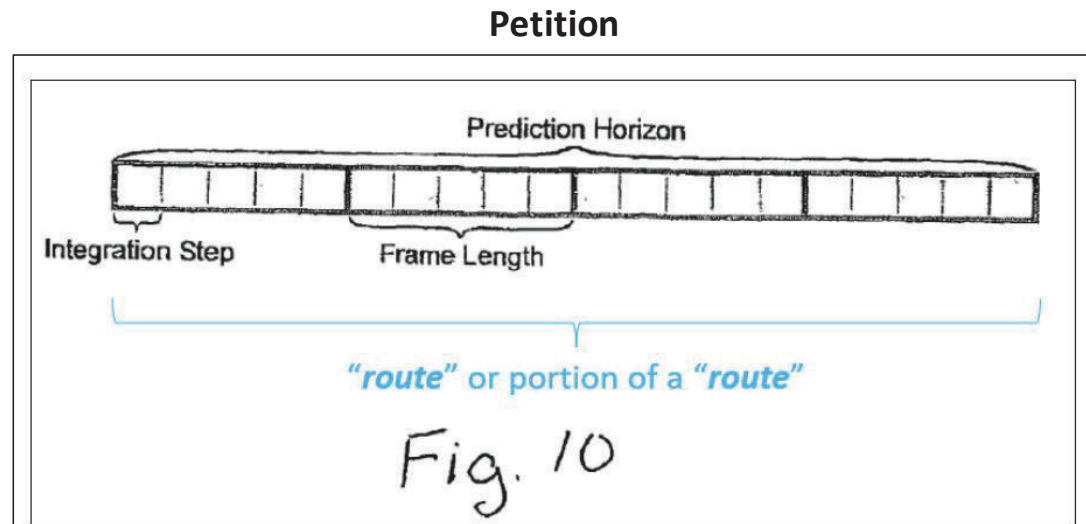
1 A. Right.

2 Q. Okay. So is this figure conveying,
3 then, **a route between two points, point A and**
4 **point B?**

5 A. **Yes,** it's showing a simple example of a
6 route between two points.

Prediction Horizon is a Path From One Point/Place to Another

[0015] In the predictive cruise control system according to the invention, a vehicle operating cost function is defined, based on a plurality of environmental parameters, vehicle parameters, vehicle operating parameters and route parameters. As the vehicle travels over a particular route for which route parameters, such as road gradient and curvature, are stored in a road map, sensors aboard the vehicle detect environmental and vehicle operating parameters, including at least vehicle speed and position relative to the road map. As the vehicle proceeds, an onboard computer iteratively calculates and stores in a memory vehicle control parameters that optimize the vehicle operating cost function for a predetermined distance (referred to as a "Prediction Horizon") along the route ahead of the vehicle. The optimal vehicle control parameters for the Prediction Horizon are then stored in a memory and continuously updated and replaced by new data as the vehicle (and hence the Prediction Horizon) moves along, thereby adjusting the "optimal" control parameters to reflect actual vehicle historical operating experience during the journey. The vehicle is then controlled by reading the optimized vehicle control parameters from the memory, corresponding to the current position of the vehicle.



TESLA-1005, FIG. 10 (annotated)

The Prediction Horizon (blue *supra*) corresponds to a route because it constitutes a path traveled by the vehicle. TESLA-1003, ¶61.

iQar Incorrectly Argues that “Route” Must Include a Destination

Patent Owner’s Response

Next, Neiss’s “Prediction Horizon” is not a route at all. A POSITA would have understood the ordinary meaning of the term “route” to be a road, path, or course from one place to another. (See, e.g., Ex. 2001 at 1189 (defining “route” as “[a] road, course, or way for travel from one space to another”).) Courts have also construed the term the same way. See, e.g., *Garmin Ltd. V. TomTom, Inc.* 468 F.Supp.2d 988, 1016-1017 (W.D. Wis. 2006) (construing “optimal route” to mean “a course of travel such that the amount of time or distance from a starting point to a destination is least or requires the fewest number of turns”). In other words, a “route” is not just the terrain immediately ahead of the vehicle. It is the path the vehicle is following to travel from its present location to its destination. (See Ex. 2003, ¶¶ 126-128.)

Patent Owner’s Sur-Reply

Next, the word “route” in the ’161 patent does not refer simply to a road. As Petitioner itself admits, a “route” is a path from one point/position to another point position.” Reply at 3 (emphasis added). This is how the ’161 patent repeatedly uses the term “route.” Throughout the patent, a “route” always has a start, an end, and a path from that start to end. Ex. 1001, 5:19-21 (a route is “for” a “destination”); 9:10-15 (a “route” has a “starting location,” “any intermediate locations,” and a “destination”); 11:13-17 (a “route” extends “from the starting position” to “an actual or estimated ending position”); 14:20-22 (a “route” extends “from the” vehicle’s “initial position to a destination (e.g., a final destination or an intermediate destination)”; see also Response at 18-20. The patent never states that there can be a “route” without a destination. The patent also never states that a route can simply be a road. That, however, is how Petitioner proposes interpreting the claims. Again, a patent’s claims must be interpreted in a way consistent with the disclosure in the specification. See *Indacon*, 824 F.3d at 1357. Petitioner would have the Board ignore the specification.

'161 Patent Broadly Describes What a Destination Is

The route used by the power management device typically includes a starting position (e.g., the current position of the vehicle, which may be indicated by GPS), an ending position, as described above, and any intermediate positions between the initial and the final positions. In some variations, the route may be broken up into segments that may be used by a power management device to optimize the power needed to travel this segment. A segment may comprise any distance to be traveled, including the entire route, or small portions of the route. Different segments in the same route may be of different lengths.

The power required by the vehicle to travel along a route, or a segment of the route, may be estimated or calculated, and this calculation may be used to determine a calculated speed for the vehicle so that the power usage is optimized or minimized. Such calculations of power requirements at different speeds typically use information inputs from the vehicle, the user, and the environment over the route from the initial position to a destination (e.g., a final destination or an intermediate destination). Any appropriate information input may be used.

'161 Patent

For instance, if a driver leaves her house to go to work, and drives toward the highway, the statistical destination logic may determine that there is a 95% chance that she is going to the highway but only a 75% chance that she is going to go South on the highway. So, if the threshold (X) is 90%, the "destination" in this case would be the highway onramp. Once she gets on the highway going south, the probability that she is going to work may have increased to 92%. Now, the "destination" would be her work. Thus, as the vehicle drives around, the probable destination determined by the statistical destination logic may constantly change to revise the destination or intermediate destination.

Prediction Horizon Includes a Destination Even Under iQar's Interpretation

Petitioner's Reply

Second, even assuming *arguendo* that a route must include a destination (which Petitioner does not concede), the '161 Patent confirms—and iQar acknowledges—that **this destination need not be a final destination**, but instead can be just an intermediate destination. *See* TESLA-1001, 14:20-22; POR, 18.² Thus, assuming *arguendo* that a route must include a destination, the route can be simply a path from a vehicle's current position to an intermediate position along the way to a final destination. *See* TESLA-1001, 12:15-20 (explaining that a destination can be the “work” location of the driver or can be merely a point on the road, such as a “highway onramp”).

Similarly, **Neiss's Prediction Horizon is a path from the vehicle's current position to an intermediate position** that is a predetermined distance ahead of the vehicle, along the path to the vehicle's final destination. *See* TESLA-1005, [0015], claim 1. As such, Neiss's Prediction Horizon is a route, even under iQar's narrow interpretation of the term.

TESLA-1001, 12:15-26; Pet. Reply, 10 21

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

iQar Argues Prediction Horizon Must Relate To Vehicle's Route

Patent Owner's Response

[in] the route while driving.” (*Id.*, ¶ [0165].) In other words, Neiss has specifically designed its system to account for the fact that a vehicle may very well *not* traverse the terrain corresponding to the Prediction Horizon and may instead take another, different path requiring its system to “adapt.” (*See id.*, ¶¶ [0163]-[0169].) When it does so, Neiss will discard its prior “Prediction Horizon” (as it is not terrain over which the vehicle will ever travel) and replace it with a new “Prediction Horizon” for the new road ahead (which, once again, the vehicle may or may not ever travel over). (*See id.*) Neiss also explains that in one example, its Prediction Horizon may be 4000 m long. (*See id.*, ¶ [0192].) A vehicle can of course follow an almost innumerable number of different paths—by making a variety of different turns—over this distance. (*See Ex. 2003*, ¶¶ 145-151; *see also Ex. 2002*, 51:10-15

• • •

Given this, a POSITA would have recognized there is no correlation between the “Prediction Horizon” and the route the vehicle is actually traveling.

Patent Owner's Sur-Reply

None of Petitioner's other arguments have any merit. According to Petitioner, even if the claimed “route” requires a destination, Neiss still teaches what the claims require because its Prediction Horizon ends at a supposed “intermediate destination.” Reply at 9-10. It does not. The Prediction Horizon is just a portion of road. Neiss, ¶¶ [0015] (the “Prediction Horizon” is a “predetermined distance” ahead of the vehicle); [0048] (noting that the “Prediction Horizon” is the “approaching road”), [0162] (defining “Prediction Horizon” as “an area around the current vehicle position[]”). It has nothing to do with where the vehicle is going. This is why Neiss continually checks to ensure that the vehicle is actually travelling along the road reflected by the Prediction Horizon: it does not know if the vehicle will actually traverse any of the Prediction Horizon let alone reach its end. *See id.*, ¶ [0188] (noting that “output values for the respective vehicle positions are read out”); Fig. 6 (noting that Neiss uses a “Position Estimator 25” to query its “Look-up

Prediction Horizon Relates to the Vehicle's "Route"

Petitioner's Reply

Third, it is also incorrect for iQar to argue that "there is no correlation between the 'Prediction Horizon' and the route the vehicle is actually traveling." POR, 22-24. Neiss's Prediction Horizon is a path from the vehicle's current position to another position ahead of the vehicle. See TESLA-1005, [0015]. And importantly, Neiss specifies that this path is along the vehicle's overall route, thus confirming that there is, in fact, a relationship between the Prediction Horizon and the vehicle's overall route.³ See TESLA-1005, claim 1 ("sensing ... parameters as the vehicle *travels a route* ...; and as the vehicle travels *said route*, iteratively calculating ... vehicle control parameters that minimize said vehicle operating cost function for a *predetermined distance along said route* ahead of said vehicle"); [0015] (defining the Prediction Horizon as "a predetermined distance *along the route* ahead of the vehicle").

Prediction Horizon Relates to the Vehicle's "Route"

Neiss

eters. As the vehicle travels over a particular route for which route parameters, such as road gradient and curvature, are stored in a road map, sensors aboard the vehicle detect environmental and vehicle operating parameters, including at least vehicle speed and position relative to the road map. As the vehicle proceeds, an onboard computer iteratively calculates and stores in a memory vehicle control parameters that optimize the vehicle operating cost function for a predetermined distance (referred to as a "Prediction Horizon") along the route ahead of the vehicle. The optimal

[0048] Optimization Algorithm Module 27: This is the main module, in which optimal velocity trajectory, the optimal throttle pedal positions, and the controller gains are calculated for the approaching road within the Prediction Horizon, based on optimization of a cost function, as described hereinafter. These series are then stored in the look-up table. The start of a new calculation is triggered every time the vehicle has covered a specified distance. It may also be triggered by other events, such as the vehicle driver's changing the set vehicle speed.

1. A method for controlling operation of a vehicle, defining an analytic vehicle operating cost function based on a plurality of environmental parameters, vehicle parameters, vehicle operating parameters and route parameters;
sensing said environmental and vehicle operating parameters as the vehicle travels a route for which route parameters are stored in a road map, said vehicle operating parameters including at least vehicle speed and vehicle position relative to said road map;
as the vehicle travels said route, iteratively calculating and storing in a memory, vehicle control parameters that minimize said vehicle operating cost function for a predetermined distance along said route ahead of said vehicle, as a function of at least vehicle speed, vehicle position and route parameters stored in said road map;
reading from said memory optimized vehicle control parameters corresponding to a current position of said vehicle; and
controlling current vehicle operation based on said optimized vehicle control parameters.

Immaterial Whether or Not Vehicle Deviates from the Route

Patent Owner's Response

[in] the route while driving.” (*Id.*, ¶ [0165].) In other words, Neiss has specifically designed its system to account for the fact that a vehicle may very well *not* traverse the terrain corresponding to the Prediction Horizon and may instead take another, different path requiring its system to “adapt.” (*See id.*, ¶¶ [0163]-[0169].) When it does so, Neiss will discard its prior “Prediction Horizon” (as it is not terrain over which the vehicle will ever travel) and replace it with a new “Prediction Horizon” for the new road ahead (which, once again, the vehicle may or may not ever travel over). (*See id.*) Neiss also explains that in one example, its Prediction Horizon may

Petitioner's Reply

Despite Neiss’s express indication of a correlation between the Prediction Horizon and the vehicle’s overall route, iQar maintains that the Prediction Horizon does not relate to the route because, according to iQar, “a vehicle may very well *not* traverse the terrain corresponding to the Prediction Horizon and may instead take another, different path.” POR, 23. To be sure, it is possible that sometimes, a vehicle might deviate from the overall route it initially set out to follow. But this does not preclude that in other instances, the vehicle would indeed continue to follow the original overall route, thus hewing to each successive Prediction Horizon as it travels along that overall route. And to the extent the vehicle may deviate from the route it originally set out to follow, the Prediction Horizons the vehicle traversed before the deviation nevertheless correspond to at least the initial portion of its original route.

Issue 1B: Neiss receives successive Prediction Horizons that correspond to “receiving a route”

Neiss's Computer Iteratively Receives a Start-To-Finish Route

Neiss

A predictive cruise control system utilizes information about the current vehicle position, as well as upcoming terrain in order to save fuel and increase driving comfort. A vehicle operating cost function is defined, based on a plurality of environmental parameters, vehicle parameters, vehicle operating parameters and route parameters. As the vehicle travels over a particular route for which route parameters, such as road gradient and curvature, are stored in a road map, sensors aboard the vehicle detect environmental and vehicle operating parameters, including at least vehicle speed and its position relative to the road map. As the vehicle proceeds, an onboard computer iteratively calculates and stores in a memory vehicle control parameters that optimize the vehicle operating cost function for a predetermined Prediction Horizon along the route ahead of the vehicle. The optimal vehicle control parameters for the Prediction Horizon are then stored in a memory and continuously updated and replaced by new data as the vehicle moves along, thereby adjusting the "optimal" control parameters to reflect actual vehicle historical operating experience during the journey. The vehicle is then controlled by reading the optimized updated vehicle control parameters from the memory, corresponding to the current position of the vehicle.

Petition

The Prediction Horizon (blue *supra*) corresponds to *a route* because it constitutes a path traveled by the vehicle. TESLA-1003, ¶61.

Alternatively/additionally, Neiss's optimization operations are performed iteratively for portions of an overall route ahead of the vehicle as the vehicle travels the overall *route* between a starting point and a final destination. TESLA-

Petitioner's Reply

The Petition also asserted that Neiss teaches or renders obvious receiving a "route" that includes a path extending from the vehicle's current position to the vehicle's final destination. The Petition offered two theories in this regard. First, the Petition explained that, as Neiss's vehicle travels from one point to another, it traverses multiple Prediction Horizons, which are iteratively received as the vehicle drives along the route. Pet., 14-16. By the time the vehicle reaches its final destination, all Prediction Horizons spanning the entire, start-to-finish journey of the vehicle have been received, thus rendering obvious receipt of a route. *Id.*

iQar's Argument Regarding Vehicle Deviating from Route

Patent Owner's Response

Decision at 14-15. This appears to be an obviousness by inherency argument. Even though Neiss does not determine the vehicle's final destination, the Board appears to have taken the position it inherently teaches such a determination as the Board assumes the end of the final Prediction Horizon necessarily corresponds with or includes the end of the vehicle's route. Not so. First, the vehicle would need to actually follow the terrain corresponding to the Prediction Horizon—instead of turning or deviating from that terrain—for the final Prediction Horizon to include the vehicle's destination. (*See id.*, ¶¶ 159-168.) As explained, this is certainly not guaranteed to occur. Neiss recognizes that the vehicle may “change ... route while driving.” (Ex. 1005, ¶ [0165].) Next, even if the vehicle did precisely follow the

Petitioner's Reply

First, iQar argues that “the vehicle would need to actually follow the terrain corresponding to the Prediction Horizon—instead of turning or deviating from that terrain—for the final Prediction Horizon to include the vehicle's destination.” POR, 25. Certainly, Neiss contemplates that that would happen in many instances, thus satisfying the “receiving a route” limitation. TESLA-1005, claim 1. As such, it is immaterial if Neiss's system also allows for the remote possibility that a vehicle might change its path partway through its travel. But even if the vehicle did change its path to deviate from an originally planned route, the vehicle would still have previously received Prediction Horizons prior to the deviation, and would continue to receive Prediction Horizons after the deviation along its new path to its ultimate destination —*i.e.*, iteratively received a “route,” under iQar's own interpretation of the term. *See* POR, 18; TESLA-1001, 5:30-31.

Neiss's Received Route Includes a Final Destination Irrespective of Any Deviations Along the Way

Petitioner's Reply

First, iQar argues that “the vehicle would need to actually follow the terrain corresponding to the Prediction Horizon—instead of turning or deviating from that terrain—for the final Prediction Horizon to include the vehicle’s destination.” POR, 25. Certainly, Neiss contemplates that **that would happen in many instances**, thus satisfying the “receiving a route” limitation. TESLA-1005, claim 1. As such, it is immaterial if Neiss’s system also allows for the remote possibility that a vehicle might change its path partway through its travel. But even if the vehicle did change its path to deviate from an originally planned route, the vehicle would still have **previously received Prediction Horizons** prior to the deviation, and would **continue to receive Prediction Horizons** after the deviation along its new path to its ultimate destination —*i.e.*, iteratively received a “route,” under iQar’s own interpretation of the term. *See* POR, 18; TESLA-1001, 5:30-31.

POR, 25; Pet. Reply, 14-15

29

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

iQar's Argument Regarding Alignment of Vehicle's Final Destination with Final Prediction Horizon

Patent Owner's Response

driving.” (Ex. 1005, ¶ [0165].) Next, even if the vehicle did precisely follow the Prediction Horizon, the overall route would need to be an exact multiple of Neiss’s frame length for the end of the Prediction Horizon to correspond with the end of the route. As the vehicle moves along, Neiss updates the Prediction Horizon one frame at a time. (See *id.*, ¶¶ [0015]; [0188]-[0191].) If the route is not a multiple of the frame length, the final Prediction Horizon will end either before or after the vehicle’s destination. By way of example, Neiss explains that one Prediction Horizon can be

Patent Owner's Sur-Reply

Second, Petitioner now proffers entirely new arguments regarding why Neiss’s system would purportedly employ a Prediction Horizon corresponding to a “route.” As noted in the Response, because the Prediction Horizon is just a fixed slice of road, Neiss’s Prediction Horizon will never correspond exactly to the route the vehicle actually travels or end at the same point as the vehicle’s destination. Instead, the Prediction Horizon will—because it is simply the road ahead and is continuously updated—cover untraveled road and extend well beyond the vehicle’s destination. See Response at 25-26. According to Petitioner, this does not matter

Final Prediction Horizon Would Terminate in the Route's Final Destination

Neiss

[0192] In the example of FIGS. 12 and 13, the parameters Prediction Horizon and Number of Frames are chosen very small in order to make the basic operation of the ring buffer clear. Realistic values for the Prediction Horizon would be 4000 assuming an Integration Step Size of 1 m, and 200 for the Number of Frames. Thus, nearly 200 optimization iteration steps are performed with regard to the values in one frame before they are actually read out. This number is more than enough for the algorithm to converge to the optimal values. Therefore, it is sufficient to compute one optimization per frame.

sensing said environmental and vehicle operating parameters as the vehicle travels a route for which route parameters are stored in a road map, said vehicle operating parameters including at least vehicle speed and vehicle position relative to said road map;

as the vehicle travels said route, iteratively calculating and storing in a memory, vehicle control parameters that minimize said vehicle operating cost function for a predetermined distance along said route ahead of said vehicle, as a function of at least vehicle speed, vehicle position and route parameters stored in said road map;

Petitioner's Reply

In this regard, iQar also leverages one illustrative example of Neiss—specifying that “one Prediction Horizon can be 4000 m (or 4 km) long and include 200 frames,” each 20 m long—to allege that “the final Prediction Horizon will end either before or after the vehicle’s destination.” POR, 25-26 (citing TESLA-1005, [0192]). But this argument ignores that some overall routes would have a length that is a multiple of the 20 m frame length, in which case the end of the last Prediction Horizon would indeed coincide with the final destination. Moreover, this example is illustrative, and Neiss provides no strict requirements on the size of either the Prediction Horizon or the frames. In practice, it would be a straightforward implementation detail to select a frame length such the length of the vehicle’s overall route is a multiple of that frame length (and/or the vehicle’s final destination coincides with the ending point of the final Prediction Horizon).

Issue 1C: Neiss renders obvious
receiving an entire, start-to-finish
route all at once

It was Obvious to Receive an Entire, Start-to-Finish Route

Petition

If it is determined that the present limitation requires receiving an *entire route* (e.g., an end-to-end route between a starting point and a final destination) *at once*, it would be obvious to modify Neiss so that its onboard computer receives the entire route at once, instead of receiving it iteratively. TESLA-1003, ¶¶68-72.

Declaration of Dr. Papanikolopoulos

68. If it is determined that the present limitation requires receiving an *entire route* (i.e., an end-to-end route with a starting point to a final destination) *at once*, a POSITA would have found that to be obvious in view of Neiss. As described above, Neiss's onboard computer iteratively receives predetermined distances of the route ahead of the vehicle, as the vehicle drives from a starting point and until it reaches its final destination. A POSITA would have found it obvious to modify Neiss so that Neiss's onboard computer receives the entire end-to-end route all at once, instead of receiving it iteratively in the manner described by Neiss.

Obvious for Neiss's Computer To Receive Start-To-Finish Route

Declaration of Dr. Papanikolopoulos

70. *First*, receiving the entire route at once would be computationally efficient because it would enable quicker computations on upcoming route segments/frames by **avoiding or reducing use of network and computational resources** that would otherwise be needed to obtain this information as the vehicle drives and thereafter perform the computations on such received information.

71. *Second*, a vehicle onboard computer **receiving an entire end-to-end route** (e.g., from external map applications) was **well-known** in the context of vehicle control systems and enabled receipt of accurate routes that, e.g., account for real-time events, such as road and/or traffic conditions. This is corroborated by, e.g., Kuroda (TESLA-1021, 1:42-43) (teaching that a vehicle can include a "route retrieving unit [that] retriev[es] a route to a destination"). Thus, modifying Neiss's onboard computer to receive the route would have been an application of known techniques (e.g., an onboard computer of a cruise control system receiving a route) to a known system (e.g., a vehicle onboard computer implementing a predictive cruise control system that processes information about the route) to yield **predictable results** (e.g., an onboard route computer for predictive cruise control that receives an entire route with respect to which optimizations and/or analytic operations can be performed).

Obvious for Neiss's Computer To Receive Start-To-Finish Route

Declaration of Dr. Papanikolopoulos

72. A POSITA would have also had a reasonable expectation of success in configuring Neiss in this manner. That is because (1) it was well-known and conventional for vehicle onboard computer to receive routes, e.g., from map applications (e.g., using user input specifying a start position and a final destination, and a map application including routing logic that identifies and provides to an onboard vehicle computer the identified route (see, e.g., Hafner (TESLA-1022, 7:52-54 (teaching that the driver “informs the system of her[] destination,” and the “system access all available prior information about this destination and various routes to the destination”))), (2) Neiss already includes a digital map that stores information about routes that the vehicle drives (e.g., TESLA-1005, ¶¶[0049], [0031], Abstract), and (3) Neiss’s optimization algorithm already obtains/receives information

about the route ahead of the vehicle for its optimization calculations (*see id.*).

Thus, it would have been straightforward and well within a POSITA’s skillset to configure Neiss’s map application (e.g., the 3D Digital Road Map) with such conventional routing logic that enables identification of the entire route and enables Neiss’s onboard computer (and its associated optimization algorithm) to receive the entire route from the map application (e.g., using the same mechanism by

iQar's Argument Regarding Performing Calculations in Advance

Patent Owner's Response

Petitioner concludes by arguing that “[i]f it is determined that the present limitation requires receiving an *entire* route (e.g., an end-to-end route between a starting point and a final destination) *at once*, it would have been obvious to modify Neiss so that its onboard computer receives the entire route at once, instead of receiving it iteratively.” Petition at 16. According to Petitioner, it was “well-known” that an “entire end-to-end route” could be received by a vehicle and performing calculations in advance for an entire route would somehow be more “computationally efficient.” *Id.* at 17. Petitioner’s expert simply repeats this

Patent Owner's Sur-Reply

Neiss, ¶¶ [00164]-[0169]. Because Neiss’s system does not store an entire route—and stores only one Prediction Horizon—it is able to account for these routine occurrences while using “[l]ess memory.” *Id.* By proposing that Neiss would for some reason store many Prediction Horizons, Petitioner ignores this. A POSITA would not consider it obvious to do what Neiss says should not be done (*i.e.*, store more than a single, short Prediction Horizon).

iQar Fails to Contend With Obviousness Theory Actually Presented

Petitioner's Reply

Third, iQar takes issue with the obviousness analysis to which the entire start-to-finish route would be received all at once. iQar characterizes this analysis in the Petition as advancing a modification to Neiss to “perform[] calculations in advance for an entire route.” POR, 26; *see* 28 (mischaracterizing the Petition’s proposed modification as having the result that “Neiss’s system would no longer perform real time calculations”); TESLA-1053, 30:12-32:9. **This is simply not the theory advanced in the Petition.**

The Petition explained that it would be **obvious to receive** the entire, end-to-end route all at once. *See* Pet., 16-17 (“it would be obvious to modify Neiss so that its onboard computer *receives* the entire route at once, instead of receiving it iteratively” (emphasis added)). **It did not contemplate computing the vehicle control parameters** for the entire, end-to-end route all at once, as iQar contends. Instead, in this modified Neiss system, the calculations would still be performed iteratively for each successive Prediction Horizon as the vehicle traverses the entirety of the route, consistent with Neiss’s disclosure; the only change would be that the route is received all at once.

iQar Argues that Prediction Horizon is a Ring Buffer Parameter

Patent Owner's Sur-Reply

First, Petitioner is wrong when it argues that the fact that Neiss continually performs new Prediction Horizon calculations as it travels along somehow satisfies the claim limitation requiring “receiving a route.” *Id.* at 13-14. Per Petitioner, “Neiss’s vehicle unequivocally traverses an overall, start-to-finish route, such that the set of Prediction Horizons that constitutes the route to the final destination will have been received (either all at once or iteratively).” *Id.* There is no such disclosure in Neiss. Neiss stores its Prediction Horizon in a “ring buffer memory.” Neiss, ¶ [0162]. And this ring buffer stores only a single Prediction Horizon at a time: earlier ring buffer entries are replaced with new entries as the vehicle travels. *See id.*, ¶¶ [0188]-[0191].

Prediction Horizon is Different From Ring Buffer and its Parameters

Petitioner's Reply

First, Neiss's Prediction Horizon is not the same as the ring buffer. As explained above, the Prediction Horizon is a path from one place to another—*i.e.*, a route. See TESLA-1005, [0015]; TESLA-1053, 22:21-23:8. The ring buffer itself is not a physical path. It is instead simply memory used to store data about this path.

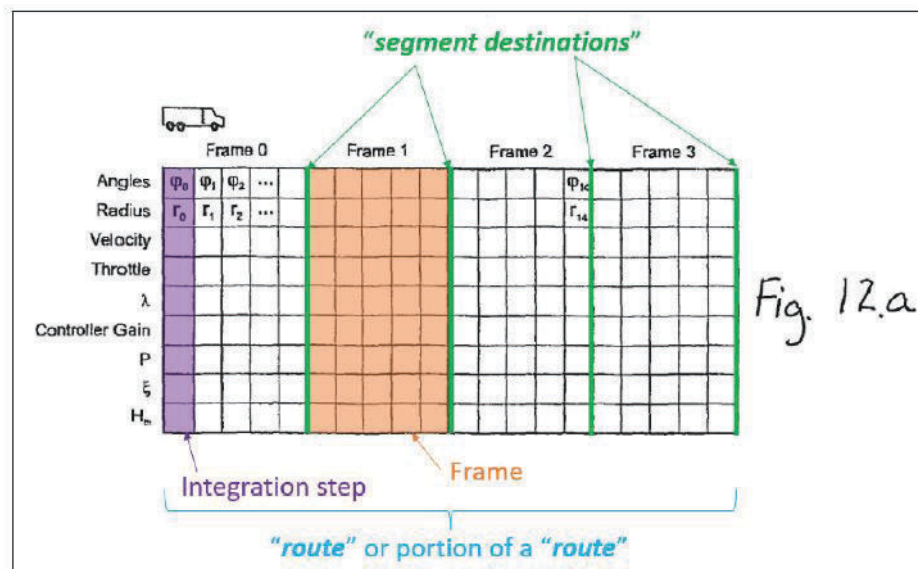
Second, road data about the Prediction Horizon is not tantamount to a route, as iQar appears to argue. A route, as discussed in §II.A, is a path from one place to another. Data about the path, such as gradient and curvature, are “parameters” of the route, not the route itself. See, *e.g.*, TESLA-1005, [0015] (“As the vehicle travels over a particular route for which *route parameters, such as road gradient and curvature*, are stored in a road map”).

Third, PO is also incorrect that “[t]he ‘Prediction Horizon’ is nothing more than a ‘parameter’ that specifies size ... of this ring buffer.” POR, 15. To be sure, the ring buffer size relates to the Prediction Horizon. See TESLA-1015, [0171]. But the ring buffer is merely a digital construct that represents the physical Prediction Horizon. See TESLA-1015, [0015]; Pet., 13. That is, *the Prediction Horizon is a distance along the road, not a buffer size parameter.* See Pet., 14 (“The Prediction Horizon (blue *supra*) corresponds to a route because it constitutes a path traveled by the vehicle”).

Prediction Horizon is Different From Ring Buffer and its Parameters

Papanikolopoulos Declaration

80. In more detail, Neiss's optimization algorithm is implemented using a look-up table "that corresponds to the length (in frames and integration steps) of the Prediction Horizon." TESLA-1005, ¶[0171]; see *id.*, Claim 18 ("said ring buffer memory is divided into a plurality of frames which collectively correspond to said prediction horizon, each frame including a predetermined number of cells corresponding to said integration interval"), FIGS. 12a(below)-c, FIGS. 13a-13d. The locations in the lookup table that correspond to the endpoints of the frames of the Prediction Horizon likewise correspond to the *segment destinations* of said route.



Prediction Horizon Is Different From Ring Buffer And Its Parameters

Petitioner's Reply

With the understanding that the Prediction Horizon is not a ring buffer parameter, but a distance of road ahead of the vehicle, iQar's argument that "no 'route' exists that can be 'received' by Neiss's vehicle computer until after Neiss populates a Prediction Horizon worth of integration steps / buffer cells with road-specific angle, radius, and other data" does not withstand scrutiny. POR, 17. The Prediction Horizon—*i.e.*, the road section ahead of the vehicle—not only exists, but is also known by the computer that manages the ring buffer, such that relevant data characterizing that road section can be populated into the ring buffer. Indeed, Neiss makes clear that the vehicle's route is known prior to the ring buffer being populated; road data for the upcoming road are "read from the road map" and used to "fill[] these ring buffer cells." TESLA-1005, [0001], [0179]; *see* claim 1. It follows that, to retrieve road data for a particular section of road—*i.e.*, the Prediction Horizon—from the road map, that section of road would have to already be known to the system. *See id.*; Pet., 15-16.

Pet. Reply, 21

41

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

iQar's Argument Regarding Neiss's Receipt of Prediction Horizons

Patent Owner's Sur-Reply

Because it stores only a single Prediction Horizon, Neiss's system never receives any "set" of Prediction Horizons. Instead, it is specifically designed not to collect or store information relating to the entirety of a route. *See id.*, ¶¶ [0007]-[0008] (discussing problems with systems that store speed control data for an entire route). This is because, according to Neiss, the vehicle may encounter a variety of "unforeseeable driving situations" that require recalculation. *Id.*, ¶¶ [0008], [0167]. Petitioner argues that these are only "remote possibilities" that have little bearing on how a POSITA would interpret Neiss. Reply at 14-15. They are not: they are routine driving events like "a slower vehicle driving ahead" or a "change in route." Neiss, ¶¶ [00164]-[0169]. Because Neiss's system does not store an entire route—and stores only one Prediction Horizon—it is able to account for these routine occurrences while using "[l]ess memory." *Id.* By proposing that Neiss would for some reason store many Prediction Horizons, Petitioner ignores this. A POSITA would not consider it obvious to do what Neiss says should not be done (*i.e.*, store more than a single, short Prediction Horizon).

PO Sur-reply, 10; Pet. Reply, 21

42

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Issue 2: Neiss discloses “receiving a route” and “segmenting said route”

Neiss Discloses Segmenting The Route

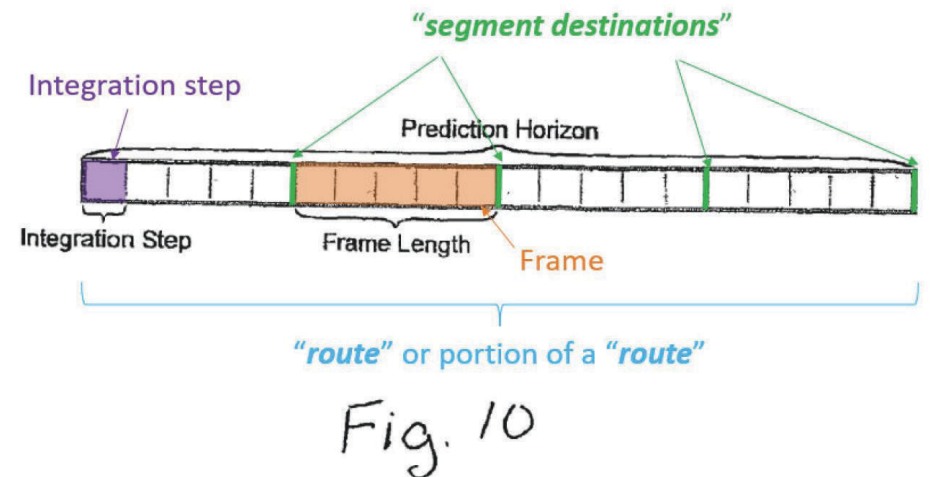
Neiss

1. A method of managing a power consumption of a vehicle, comprising the steps of:

- (a) receiving a route at a computer of said vehicle;
- (b) segmenting said route into a plurality of segment destinations;

[0170] The whole Prediction Horizon is separated into Frames, so that the Prediction Horizon must be a multiple of the Frame Length (also measured in number of integration steps b). FIG. 10 shows this separation for a Prediction Horizon of 20, the Number Of Frames is 4, and the resulting Frame Length is 5. (This is a simplified version for illustration only.) The parameter "Number of Frames" determines the rate at which the optimal trajectory is updated.

Each Prediction Horizon (the route or a subset thereof) "is separated into Frames." TESLA-1005, ¶[0170]; see ¶[0192], FIG. 10, *infra*; TESLA-1003, ¶78. Each frame (one in orange *infra*) is "a specified distance," i.e., a segment of the route. TESLA-1005, ¶[0169]. Each frame is composed of multiple "integration steps" (one in purple), each corresponding to a "small distance[]" along its frame. *Id.*, ¶[0162], [0170]; TESLA-1003, ¶78. Each frame's endpoint corresponds to a segment destination (green) because it is the end (destination) of a frame/segment. TESLA-1003, ¶78.



iQar Argues: Neiss Does Not Teach Segmenting After Receiving a Route, As Allegedly Required by the Claims

Patent Owner's Response

To begin, claims 1 and 10 require an ordered, two-step process. A “route” must first be “receiv[ed] ... at a computer of said vehicle.” Then, this already received route must be “[s]egment[ed] ... into a plurality of segment destinations.” To the extent Neiss performs these steps at all, it does so out of order.

In sum, Neiss’s process starts with an empty ring buffer that contains no data. (See, e.g., *id.*, Fig. 11.) This empty ring buffer is then iteratively filled with road data one integration step / cell at a time as the vehicle drives until a number of ring buffer cells equal to the Prediction Horizon parameter is reached. (See *id.*, ¶ [0179].) An empty ring buffer—even one whose number of cells is dictated by the Prediction Horizon parameter—is not yet a “route.” No road-specific data is present. (See Ex. 2003, ¶¶ 108-119.) It follows that even if one did consider Neiss’s Prediction Horizon to be a “route” and its frames to be “segments” as the Petition argues, no “route” exists that can be “received” by Neiss’s vehicle computer until *after* Neiss populates a Prediction Horizon worth of integration steps / buffer cells with road-specific angle, radius, and other data. (See *id.*, ¶¶ 120-122.) In other words, Neiss determines what the Petition calls “segments” first, and then accumulates these “segments” into the Petition’s “route.” (See *id.*) Once again, the claims here require

45

iQar's Argument Illustrates a Misunderstanding of Neiss

Petitioner's Reply

First, Neiss's Prediction Horizon is not the same as the ring buffer. As explained above, the Prediction Horizon is a path from one place to another—*i.e.*, a route. See TESLA-1005, [0015]; TESLA-1053, 22:21-23:8. The ring buffer itself is not a physical path. It is instead simply memory used to store data about this path.

As explained in the Petition, Neiss's Prediction Horizon is received and used to query a digital road map to retrieve road information (*e.g.*, road curvature and gradient) about each frame (*e.g.*, segment) of the Prediction Horizon. See Pet., 15-16; TESLA-1005, [0015], claim 1. This road information is stored in frames of a ring buffer along with computed vehicle control parameters for the Prediction Horizon. As iQar's expert confirmed, "data for the prediction horizon is stored in a section of memory which is the ring buffer" and "as the vehicle drives the route, past sections of memory are replaced with new calculations for when the cycle repeats." TESLA-1053, 34:2-8.

Neiss

[0162] For an area around the current vehicle positioned ("Prediction Horizon"), the optimal velocity, throttle pedal positions, and controller gain series are stored in a look-up table (which is contained in a ring buffer memory, as discussed hereinafter). The resolution of this look-up table is thereby the Integration Step Size h . (The optimal velocity is considered to be constant for small distances h .) The output values of the PCC system are then determined by reading out the values corresponding to the estimated vehicle position.

iQar's Argument Illustrates a Misunderstanding of Neiss

Petitioner's Reply

Second, road data about the Prediction Horizon is not tantamount to a route, as iQar appears to argue. A route, as discussed in §II.A, is a path from one place to another. Data about the path, such as gradient and curvature, are “parameters” of the route, not the route itself. *See, e.g.,* TESLA-1005, [0015] (“As the vehicle

Neiss

[0004] Accordingly, it would be advantageous to provide a cruise control system which utilizes information about the current position of the vehicle, as well as stored information concerning upcoming terrain along the route currently being traveled, in order to save fuel and increase driving comfort. In such a system, the current vehicle position may be determined, for example, by means of a Global Positioning System (GPS), and information about the upcoming terrain can be taken from a three-dimensional digital road map for the purpose of controlling vehicle operation.

[0015] In the predictive cruise control system according to the invention, a vehicle operating cost function is defined, based on a plurality of environmental parameters, vehicle parameters, vehicle operating parameters and route parameters. As the vehicle travels over a particular route for which route parameters, such as road gradient and curvature, are stored in a road map, sensors aboard the vehicle detect environmental and vehicle operating parameters, including at least vehicle speed and position relative to the road map. As the vehicle proceeds, an onboard computer iteratively calculates and stores in a memory vehicle control parameters that optimize the vehicle operating cost function for a predetermined distance (referred to as a “Prediction Horizon”) along the route ahead of the vehicle. The optimal vehicle control parameters for the Prediction Horizon are then stored in a memory and continuously updated and replaced by new data as the vehicle (and hence the Predic-

Neiss's System Receives A Route Before Segmenting It

Petitioner's Reply

With the understanding that the Prediction Horizon is not a ring buffer parameter, but a distance of road ahead of the vehicle, iQar's argument that "no 'route' exists that can be 'received' by Neiss's vehicle computer until after Neiss populates a Prediction Horizon worth of integration steps / buffer cells with road-specific angle, radius, and other data" does not withstand scrutiny. POR, 17. The Prediction Horizon—*i.e.*, the road section ahead of the vehicle—not only exists, but is also known by the computer that manages the ring buffer, such that relevant data characterizing that road section can be populated into the ring buffer. Indeed, Neiss makes clear that the vehicle's route is known prior to the ring buffer being populated: road data for the upcoming road are "read from the road map" and used to "fill[] these ring buffer cells." TESLA-1005, [0001], [0179]; *see* claim 1. It follows that, to retrieve road data for a particular section of road—*i.e.*, the Prediction Horizon—from the road map, that section of road would have to already be known to the system. *See id.*; Pet., 15-16.

Pet. Reply, 21

48

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Claims Do Not Require “Segmenting” Only After “Receiving”

1. A method of managing a power consumption of a vehicle, comprising the steps of:
 - (a) receiving a route at a computer of said vehicle;
 - (b) segmenting said route into a plurality of segment destinations;

Patent Owner’s Response

This is what claims 1 and 10 do here. Both require “receiving a route.” Then, “said route”—in other words, the same “route” that was previously “receiv[ed]” by the vehicle’s computer—must be “segment[ed].” A route cannot be segmented if it is not first “received” by the vehicle computer performing the segmenting. Thus, per the language of the claims and basic logic, the claimed “receiving” and “segmenting” steps must be performed in order. *See Information Techs.*, 764 F.3d

Petitioner’s Reply

Thus, because the claims do *not* specify that that same vehicle computer must perform both the receiving and the segmenting, they therefore impose no requirement that the segmenting necessarily must be performed after receipt of the route. TESLA-1001, claim 1. That is, the plain language of the claims does not require an ordering of steps, nor is there an ordering requirement implicit in the claims, and thus iQar’s reliance on *Information Techs* (POR, 12) is inapposite.

POR, 13; Pet. Reply, 6

Neiss Teaches the Claimed Steps Even Under iQar's Characterization of Neiss

1. A method of managing a power consumption of a vehicle, comprising the steps of:
 - (a) receiving a route at a computer of said vehicle;
 - (b) segmenting said route into a plurality of segment destinations;

Patent Owner's Response

To begin, claims 1 and 10 require an ordered, two-step process. A “route” must first be “receiv[ed] ... at a computer of said vehicle.” Then, this already received route must be “[s]egment[ed] ... into a plurality of segment destinations.”

To the extent Neiss performs these steps at all, it does so out of order.

Petitioner's Reply

1. *iQar's Characterization of Neiss As Segmenting a Route Prior to Receiving the Route Satisfies the Independent Claims, Which Do Not Require an Ordered Process*

iQar asserts that Neiss's system first segments the Prediction Horizon into frames and *then* receives the route when it receives the road curvature and gradient information. POR, 12. Because the claims do not require an “ordered, two-step process” in which the “receiving” step occurs before the “segmenting” step (*see* §II.B *supra*), even iQar's characterization of Neiss satisfies this claim. That said, iQar's characterization is refuted by Neiss's teachings (as explained *infra* §V.B), which confirm that the route is received and subsequently segmented (thus satisfying the claims under any interpretation of these claims).

The Claims are Obvious over Schramm-Clapper

The Schramm-Clapper Combination Accounts for Different Per-Segment Speeds

Schramm Determines An Energy-optimum Route

Schramm

[57]

ABSTRACT

A system for determining an energy-optimum route for a vehicle to take between an origin and a destination includes a computer onboard the vehicle. A global positioning system (GPS) transceiver is also onboard the vehicle for receiving vehicle position data from the satellite. The computer access a database containing both topographical map information and vehicle energy consumption data, and based thereon the computer determines and displays an energy-optimum route. The computer can also display information regarding nearby refueling stations. Also, the vehicle energy consumption data can be modified as appropriate based on the operation of the vehicle, to account for the particular vehicle's actual fuel consumption characteristics.

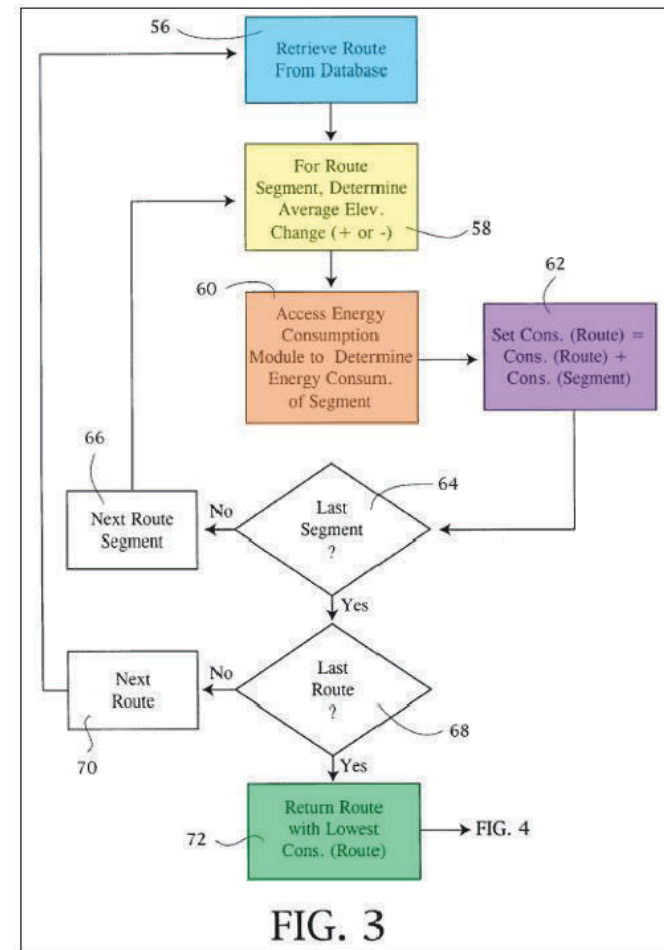


FIG. 3

TESLA-1009, Abstract, Fig. 3; Pet., 52-53.

53

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Schramm: Retrieve Routes Between An Origin and a Destination

Schramm

Referring now to FIG. 3, after receiving the inputs described above the routing module 42 retrieves from the database 36 (FIG. 1) a route between the origination point "A" and the destination point "B". It is to be understood that not all possible routes that are stored on the map of the database 36 need be retrieved. Rather, a route screening strategy can be employed to automatically screen routes that are of a greater length than a predetermined length, e.g., greater than 20% longer than the shortest available route, or greater than 30% longer than the straight line distance between the origination point "A" and the destination "B".

It is to be further understood that each route retrieved is regarded by the logic below as a series of contiguous route segments. For example, each route segment can be one mile in length. Thus, a route between points "A" and "B" that is one hundred miles in length would be composed of 100 route segments, while an alternate route of, say, one hundred five miles in length would be composed of 105 segments, and so on.

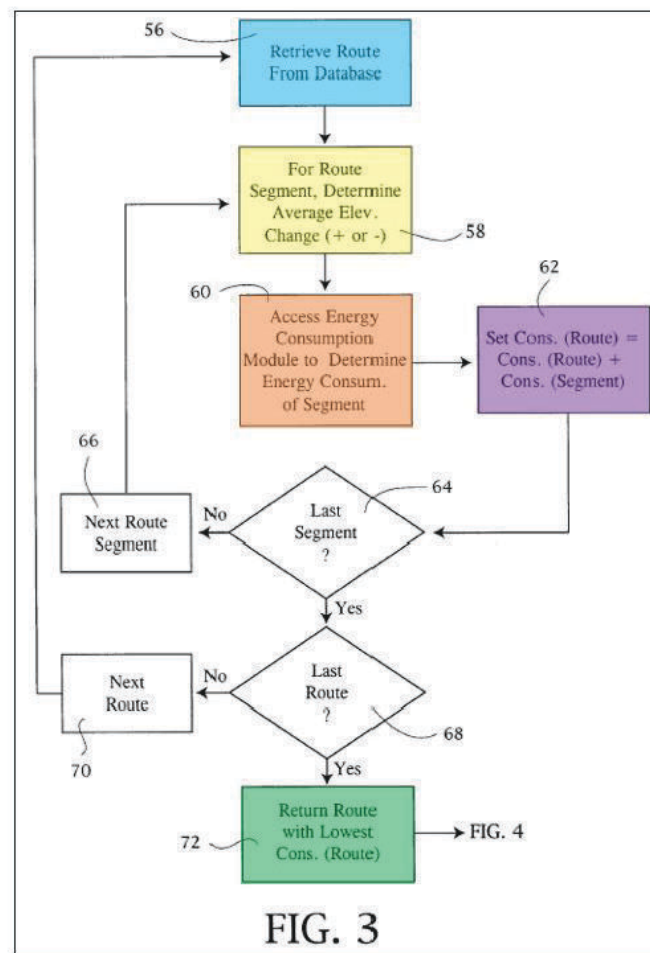


FIG. 3

TESLA-1009, 5:14-39, Fig. 3; Pet., 52-53.

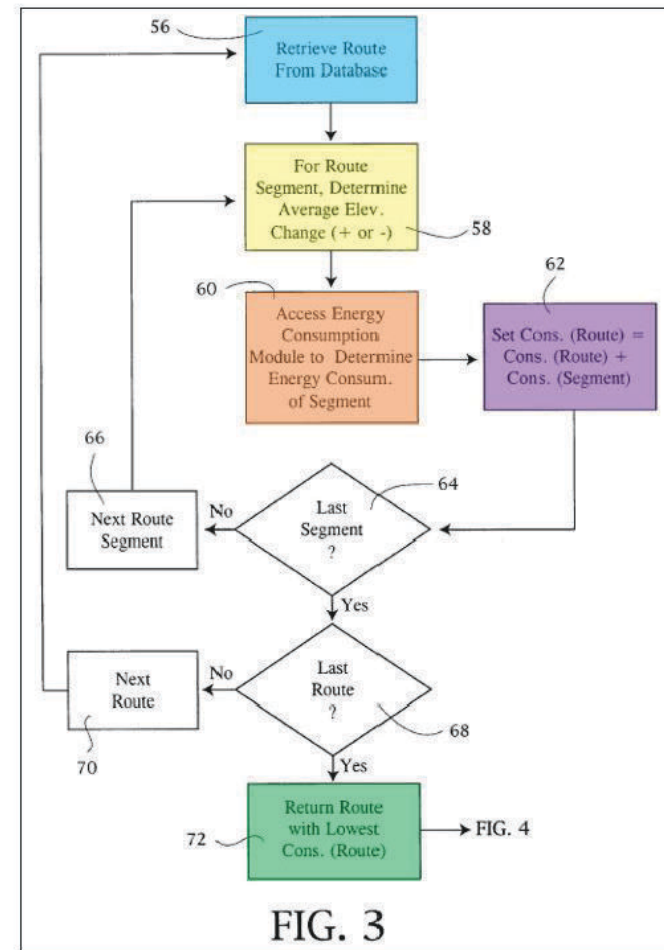
DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Schramm: Determine Energy Consumption For Each Route's Segments

Schramm

Accordingly, at block 58 the logic determines, starting with the first route segment of a first route, the average elevation change, either plus or minus ($\pm\Delta$) (or zero), as indicated by the topographical map contained in the database 36. Then, at block 60 the logic accesses the energy consumption module described below to determine the energy consumption that would be required for the vehicle 12 to traverse the route segment under test at the speed that was input at block 54 of FIG. 2. In the preferred embodiment, the energy consumption module 44 is table-driven, but it is to be understood that equivalent algorithms can be used in lieu of the tables disclosed herein.

Returning to FIG. 3, and more specifically to block 62, the logic of the routing module 42 sets a variable "consumption_{route}" (initialized at zero) to be equal to "consumption_{route}" plus the energy consumption ("consumption_{segment}") determined at block 60. Then, the routing module enters states 64–70, which indicate that the process described above is performed in a so-called "DO" loop for each segment of each possible screened route, and for each route.



Schramm: Determine the Route With Lowest Energy Consumption

Schramm

More particularly, at decision diamond 64 it is determined whether the segment under test is the last segment of the route, and if not the logic retrieves the next segment at block 66 and then loops back to block 58. Otherwise, the logic determines at decision diamond 68 whether the route under test is the last route that has been retrieved within the constraints of the screen described above, and if not the logic retrieves the next route at block 70 and then loops back to block 56 to retrieve the route from the database 36. On the other hand, when it is determined at decision diamond 68 that the last route has indeed been tested, the logic returns the route with the lowest overall energy consumption at block 72. From block 72, the logic proceeds to FIG. 4.

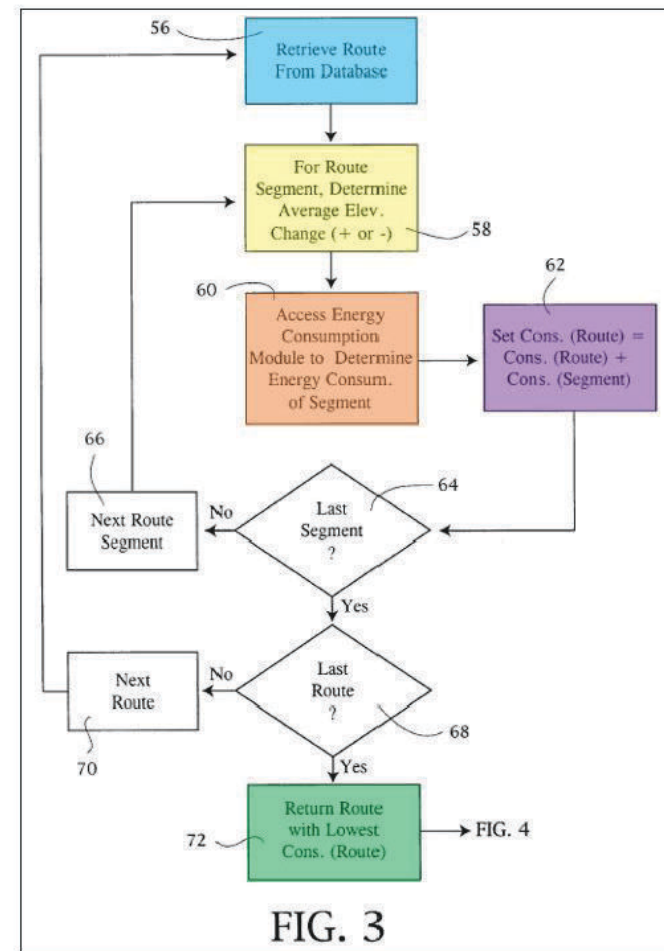


FIG. 3

Operation of Schramm's Energy Consumption Module

Schramm

Next, at block 54 of FIG. 2, the logic receives the desired average speed at which the operator intends to drive vehicle to the destination point "B". If desired, the operator may indicate a desire to select, as a default, the current speed of the vehicle 12. Accordingly, the computer 14 would access the speedometer of the vehicle 12 to set the default speed at block 54.

Accordingly, at block 58 the logic determines, starting with the first route segment of a first route, the average elevation change, either plus or minus ($\pm\Delta$) (or zero), as indicated by the topographical map contained in the database 36. Then, at block 60 the logic accesses the energy consumption module described below to determine the energy consumption that would be required for the vehicle 12 to traverse the route segment under test at the speed that was input at block 54 of FIG. 2. In the preferred embodiment, the energy consumption module 44 is table-driven, but it is to be understood that equivalent algorithms can be used in lieu of the tables disclosed herein.

Schramm, Figure 6

	Segment Δ Elev. (Aug.)	Speed values		Δ Elev. Factors, For Speech =			
		1	2	3			
Elevation values	-x	.8	.8	1	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	0	1	1.01	.	.	.	.
(base energy consumption)	1	1.01	1.02	.	.	.	.
	2	1.02	1.03	.	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	.	.	.	.	.	.	.
	+x	n	m	k	.	.	.

Schramm in View of Clapper Accounts for Different Per-Segment Speeds

Petition

Schramm's route identification process applies a single speed (e.g., "desired average speed" or current vehicle speed) to all route segments being evaluated. TESLA-1009, 5:1-16; *see* 5:41-48; TESLA-1003, ¶199. A POSITA would have recognized that a vehicle would not necessarily travel all segments at the same speed (*see* TESLA-1009, 57:23-25; TESLA-1006, 1:50-67, 3:13-27), and that speed variations impact vehicle energy consumption. TESLA-1003, ¶¶199-201. The POSITA thus would have adjusted Schramm's optimization process, per Clapper, to account for different per-segment speeds, as illustrated *infra*. *See* TESLA-1009, 5:45-50, 5:65-6:7; TESLA-1003, ¶¶200-203. Specifically, speeds for various locations, such as "[t]he driver's previous speed[s]" "at given locations" or "speed limits on existing roads and highways," are stored in a database, per Clapper. *See* TESLA-1006, 1:50-67, 2:37-44, 2:52-57, 3:13-23;

Clapper

A memory 28 coupled to a controller 22 may provide information about speeds used at given locations in the past. For example, a given driver may drive the same route to work repeatedly. Based on prior history and detected current position, the automatic cruise control 20 may set the vehicle's speed based on the speed that the driver used previously at the same location. This may reduce the amount of effort that the driver must invest in continually changing speed or resetting the cruise control.

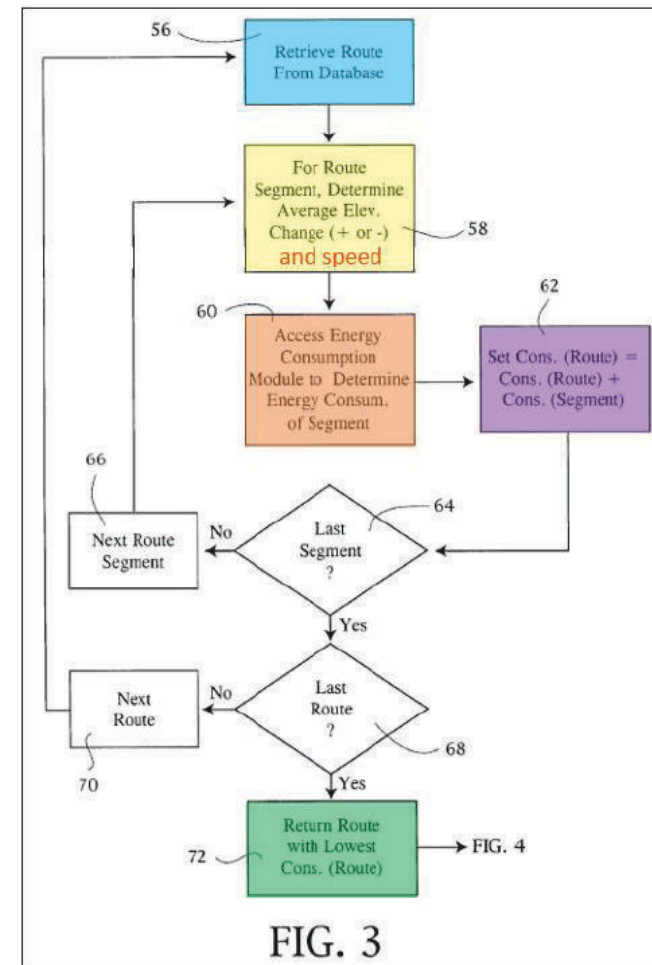
In some embodiments, a database with actual speed limit data may be provided. In one embodiment, the user may also be able to override the speed limit data and to exceed the speed limit by a given amount, or if the user prefers, to travel at a speed below the speed limit by a predetermined amount. The operator may input a customized profile that adjusts the system's suggested speeds, using the speed limit data as a reference.

Schramm in View of Clapper Accounts for Different Per-Segment Speeds

Petition

A route between origin and destination is retrieved for evaluation, per Schramm. TESLA-1009, 5:23-24 (blue *supra*). For each route segment, the segment's "average elevation change" (per Schramm) and a speed for the segment (e.g., previously-driven speed or speed limit) (per Clapper) are retrieved. TESLA-1009, 5:41-45; see TESLA-1006, 1:50-67, 2:37-57, 3:13-23; TESLA-1003, ¶204. The energy consumption value corresponding to the elevation-speed combination for each segment is retrieved, e.g., from Schramm's FIG. 6 table. TESLA-1009, 5:46-6:2; TESLA-1003, ¶¶205-206.

Per Schramm, the per-segment energy consumption values are summed to determine the total energy consumption for each route (purple *supra*) and "the route with the lowest overall energy consumption" is output/returned (green). TESLA-1009, 6:12-31; TESLA-1003, ¶207. Thus, the optimization accounts for different per-segment speeds. TESLA-1003, ¶207.



Pet., 61-62

59

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Obvious to Combine Schramm's and Clapper's Teachings

Petition

A POSITA would have been motivated to account for different per-segment speeds when identifying an energy-optimum route. TESLA-1003, ¶¶208-214.

First, a POSITA would have been motivated to modify the fixed speed constraint of Schramm's optimization process to allow for the realistic possibility that per-segment speed may vary, thereby enhancing the optimization's accuracy. TESLA-1003, ¶¶209-210.

Second, the more accurate energy-optimum route identified via Schramm-Clapper's route optimization process would provide more reliable and useful information to the driver, e.g., about whether the vehicle "has sufficient energy to travel" the route. TESLA-1009, 5:44-67; TESLA-1003, ¶211.

Third, this combination is merely the application of known techniques (maintaining per-segment speed data) to a known system (Schramm's route optimization process) that is ready for improvement (e.g., by accounting for per-segment speed variations) to yield predictable results (route optimization accounting for varying per-segment speeds). See KSR, 550 U.S. at 415-421; MPEP 2143; TESLA-1003, ¶¶212-213.

Schramm-Clapper Teaches “efficient speeds that achieve a best summation of said energy values”

The “Efficient Speeds” Limitation

‘161 Patent, claim 1

1. A method of managing a power consumption of a vehicle, comprising the steps of:
 - (a) receiving a route at a computer of said vehicle;
 - (b) segmenting said route into a plurality of segment destinations;
 - (c) calculating a plurality of energy values consumed by said vehicle to travel said route based on a plurality of variations in a target speed of said vehicle between pairs of said segment destinations;
 - (d) determining a plurality of efficient speeds that achieve a best summation of said energy values; and
 - (e) notifying an operator of said vehicle about said efficient speeds to travel between each of said segment destinations.

62

FISH.

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Energy Calculations in the '161 Patent

'161 Patent

In this example, the route may be split into five segments of 5 km each with the following altitudes measured at the end of each segment: 0 (beginning of segment 1), 200 m (end of 1), 100 m (end of 2), 400 m (end of 3), 100 m (end of 4), 0 (end of 5). We may also make the simplifying assumption that the road grade is constant between measured points, and the road grades are calculated to be: +0.04 (seg 1), -0.02 (seg 2), +0.06 (seg 3), -0.06 (seg 4), -0.02 (seg 5). The target average speed is 100 kph.

For the sake of simplicity in this example, we can then calculate the amount of energy required to drive at a constant 100 kph speed for the entire route; as well as for 16 other combinations of the speeds 95 kph (26.39 mps), 100 kph (27.78 mps), and 105 kph (29.17 mps). In practice, this simulation may be run for hundreds or even thousands of combinations of speeds which may be tested to find the optimal speed to drive each segment. It is assumed that we enter the first segment traveling 100 kph (27.78 mps). It is also assumed that the speed listed for the segment is the final speed of the segment and the vehicle accelerates linearly throughout the segment.

The Table below shows the results of the energy calculation for all of the combinations tried. The lowest energy usage is 4275.94 Watt-hours. This was obtained by going 95 kph (seg1), 105 kph (seg2), 95 kph (seg3), 105 kph (seg4), 100 kph (seg5). The average speed over all 5 segments is still 100 kph, but the energy used is 10.2% less.

iterations	seg 1	seg 2	seg 3	seg 4	seg 5	Energy	units
1	27.78	27.78	27.78	27.78	27.78	4763.2484	Whr
2	26.39	29.17	27.78	27.78	27.78	4536.889	Whr
3	26.39	29.17	26.39	29.17	27.78	4275.938	Whr
4	26.39	29.17	26.39	27.78	29.17	4382.5026	Whr
5	26.39	29.17	27.78	26.39	29.17	4950.1438	Whr
6	26.39	29.17	27.78	29.17	26.39	4438.3467	Whr
7	26.39	26.39	29.17	29.17	27.78	4942.6325	Whr
8	26.39	26.39	29.17	27.78	29.17	5055.9124	Whr
9	26.39	26.39	27.78	29.17	29.17	4784.1865	Whr
10	29.17	26.39	27.78	27.78	27.78	5013.2565	Whr
11	29.17	26.39	26.39	29.17	27.78	4745.585	Whr
12	29.17	26.39	26.39	27.78	29.17	4852.1495	Whr
13	29.17	26.39	27.78	26.39	29.17	5137.6625	Whr
14	29.17	26.39	27.78	29.17	26.39	4914.7142	Whr
15	29.17	29.17	26.39	26.39	27.78	4602.8845	Whr
16	29.17	29.17	26.39	27.78	26.39	4493.3028	Whr
17	29.17	29.17	27.78	26.39	26.39	4765.4617	Whr

TESLA-1001, 14:66-15:46; EX2003, ¶107; POR, 15

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Schramm-Clapper Teaches the “Efficient Speeds” Limitation

Papanikolopoulos Declaration

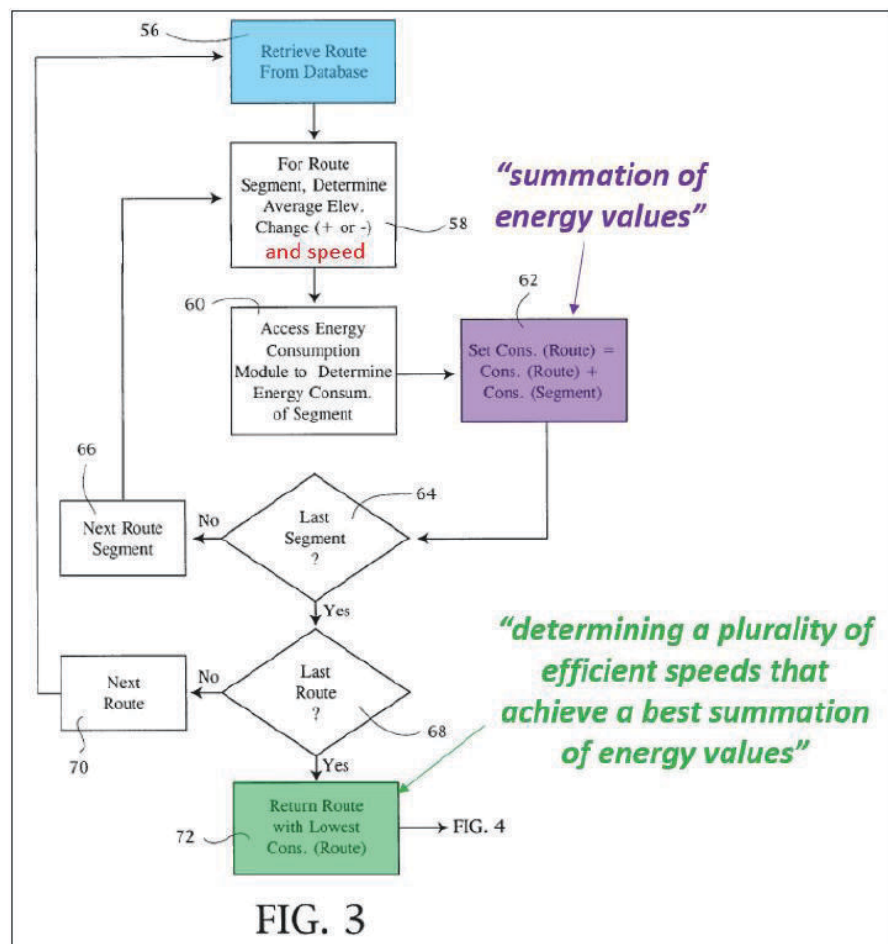
270. The identified energy-optimum route is composed of “a series of contiguous route segments.” TESLA-1009, 5:33-35; *see* my discussion of element [1b]. Based on Clapper, each route segment has a corresponding speed, *e.g.*, “the speed that the driver used previously” on that segment or the “speed limit” of that segment, as taught by Clapper. TESLA-1006, 1:53-55, 3:13-23; *see* 1:50-67, 2:37-44, 2:52-57. Indeed, these are the speeds that were used to retrieve the energy values for the segments of the route, as I discussed above. The speeds for the segments of the identified energy-optimum route correspond to *a plurality of efficient speeds* at least because they are a plurality of speeds for the plurality of segments of the identified energy-optimum (*e.g.*, efficient) route. These speeds can be considered *efficient speeds* because they correspond to energy values that, when summed, produce the “lowest overall energy consumption” (*e.g.*, *best summation* of energy values) for a route between origin and destination, as I discussed above. TESLA-1009, 6:29-31.

TESLA-1003, ¶270; Pet., 81

64

DEMONSTRATIVE EXHIBIT – NOT EVIDENCE

Schramm-Clapper Teaches the “Efficient Speeds” Limitation



Petitioner's Reply

Within the framework of Schramm's route optimization process, which aims to identify an energy optimum route given a desired average speed, the speed is explicitly allowed to vary by segment, based on Clapper's teachings of stored per-segment speeds. *See* Pet., 59-62. In this context, **determination of an energy-optimum route**—which, in the combination, takes into account variations in speed by segment—**involves identification of the segments** that make up that route, and thus also **identification of the speed** for each of those identified segments. *See id.*, 77-84. iQar's myopic focus on the purported deficiencies of Schramm fails to acknowledge the role of Clapper's speeds in this combination.

iQar Incorrectly Argues Schramm's and Clapper's Speeds are "Inefficient"

Patent Owner's Sur-Reply

must still teach everything the claims require. Again, claims 1 and 10 of the '161 patent require *"determining a plurality of efficient speeds that achieve a best summation of said energy values."* The combination of Schramm and Clapper do not teach this. Schramm takes speed as input when determining an optimal vehicle route. Clapper stores inefficient speeds. So, the combined system will simply take Clapper's inefficient speeds as inputs when performing Schramm's route determination process. The mere fact that Schramm calculates an optimal route given inefficient input speeds does not somehow transform those inefficient speeds into efficient speeds. The inefficient speeds are still inefficient speeds. This is what Petitioner misses. It has failed to identify anything in the combination of Schramm and Clapper that constitutes the required *"determining"* of *"a plurality of efficient speeds that achieve a best summation of said energy values."* This showing deficiency requires rejection of this ground.

Record Confirms that Speeds Are Varied in the Combination

Schramm

Next, at block 54 of FIG. 2, the logic receives the desired average speed at which the operator intends to drive vehicle to the destination point “B”. If desired, the operator may indicate a desire to select, as a default, the current speed of the vehicle 12. Accordingly, the computer 14 would access the speedometer of the vehicle 12 to set the default speed at block 54.

Institution Decision

57 F.3d at 1581 (emphasis added) (internal quotations omitted). Moreover, Schramm discloses that the operator inputs the desired *average* speed, at least suggesting that the speed of Schramm’s vehicle does vary. Ex. 1009, 5:14–16, 7:24–26.

Petitioner’s Reply

This ignores that the Schramm-Clapper ground relies not on the teachings of Schramm alone, but on the combined teachings of Schramm and Clapper to demonstrate obviousness of the recited determining of efficient speeds. To be sure, Schramm does take speed as an input. See Pet., 59; TESLA-1009, 5:14-20. But Schramm already recognizes that speed may vary along a route in its indication that that the speed received as an input is a “desired *average* speed at which the operator intends to drive.” TESLA-1009, 5:14-15 (emphasis added); see Pet., 59. A desired average speed encompasses the possibility that the actual speed of the vehicle may vary as it travels, as explained by iQar’s expert. TESLA-1053, 38:19-39:3 (“Q: I believe you testified that average speed could be achieved in different ways. And my question ways: Could those different ways be by driving at a constant speed throughout the route and driving at different speeds throughout the route? A. It could be achieved, yes, as you described.”).