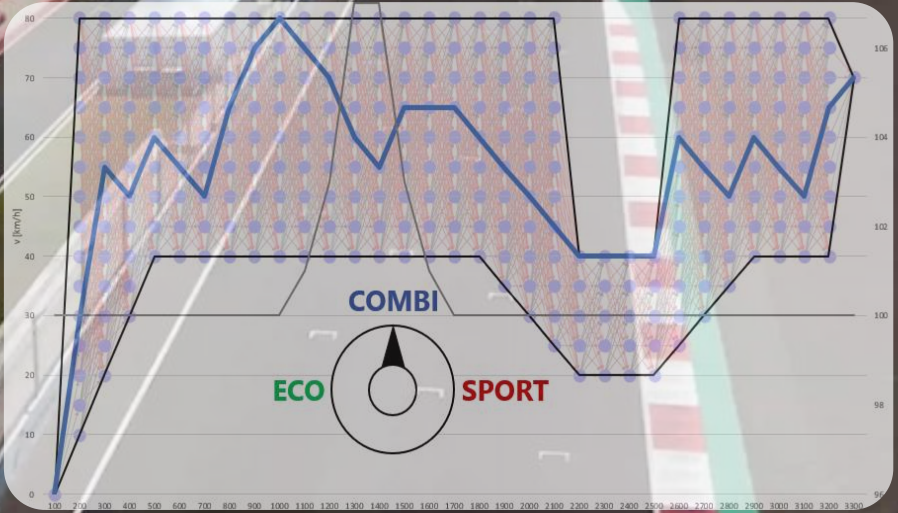
Learning Corner: Data and Telemetry Workshops at Shell Eco-marathon

Webinar 3/3 March 26th 2024

Developing Holistic Race Strategies based on Knowledge Graphs



SCHMID Elektronik

Shell Eco-marathon Partner

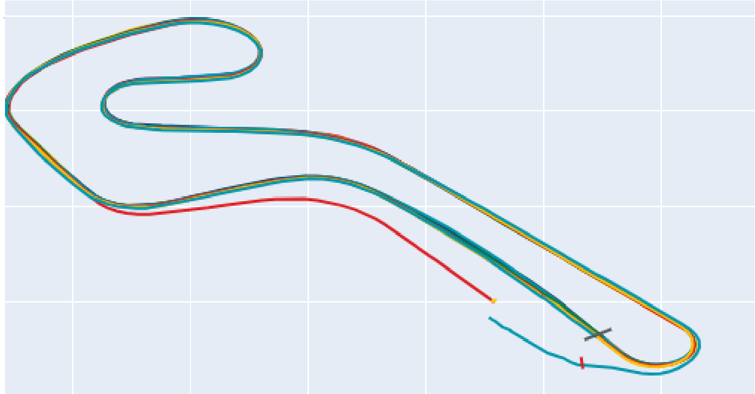
Host

Marco Schmid

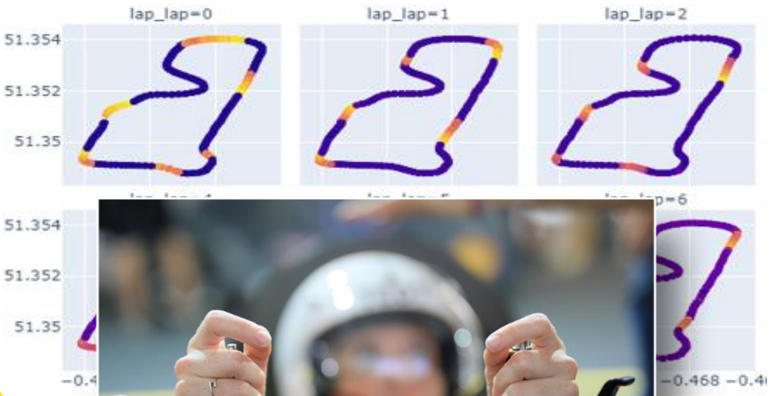
CEO Schmid Elektronik AG

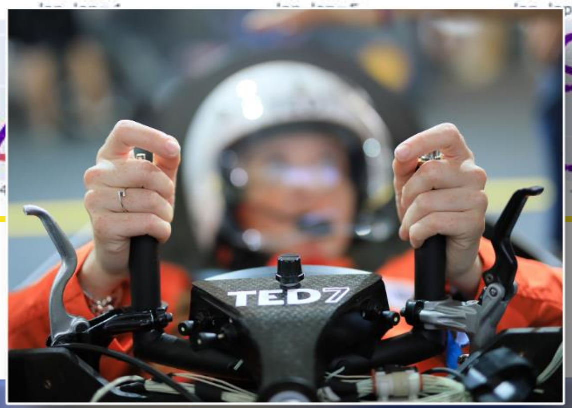
Switzerland

Webinar 1/3
Interactive Data

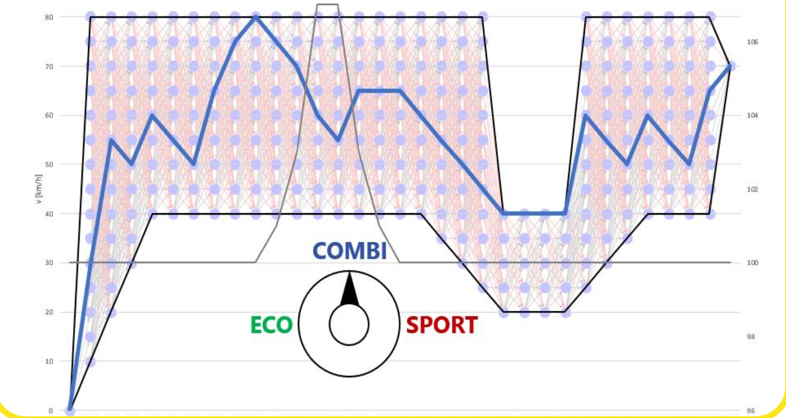


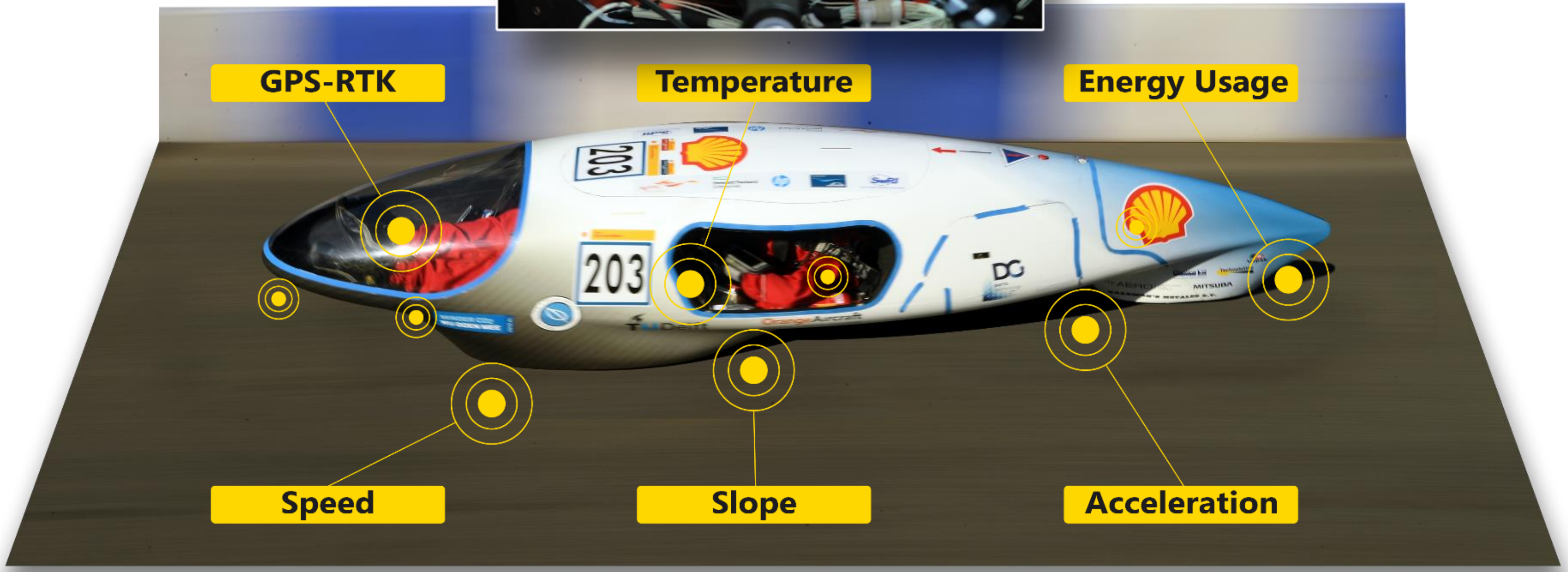
Webinar 2/3
Data Analytics





Webinar 3/3
Race Strategy





Review: From Time Series to Distance Based Data

python™

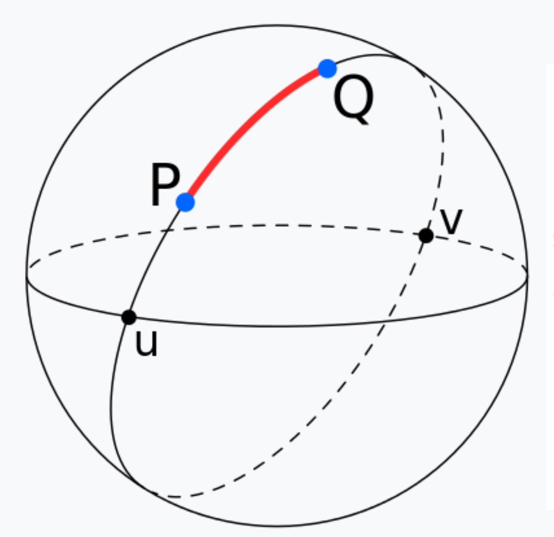
ANACONDA

jupyter

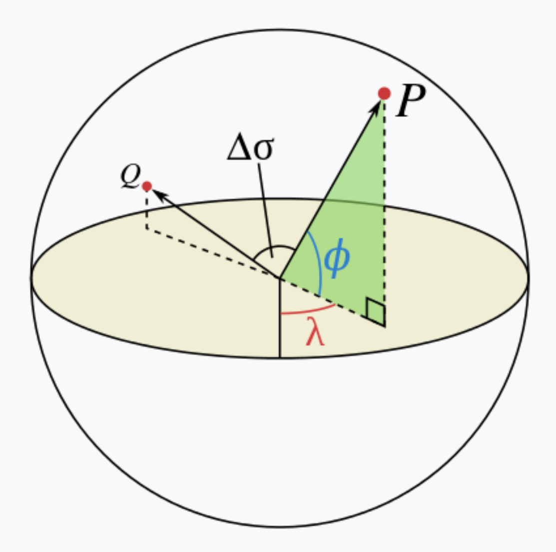
pandas

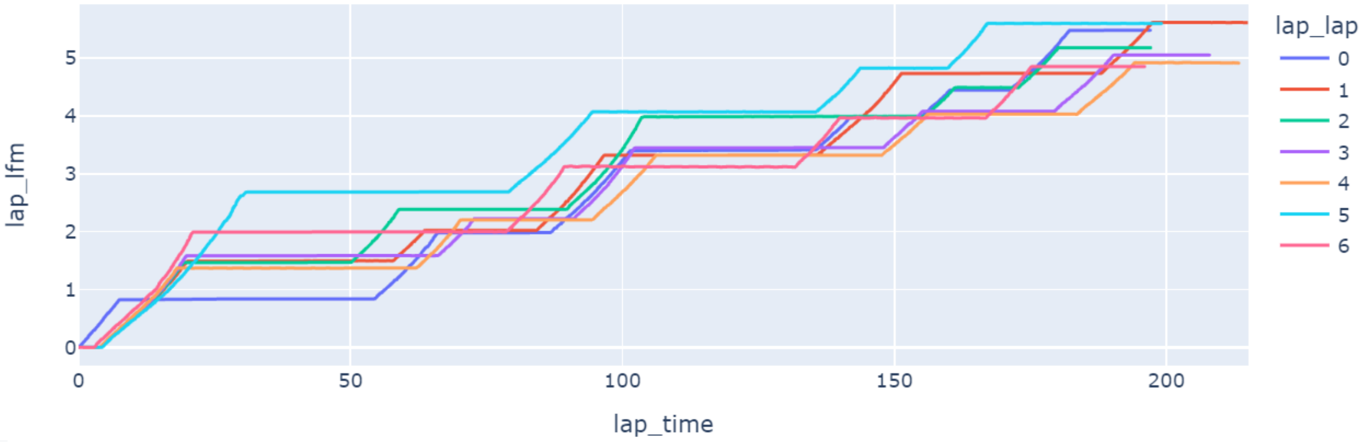
NumPy

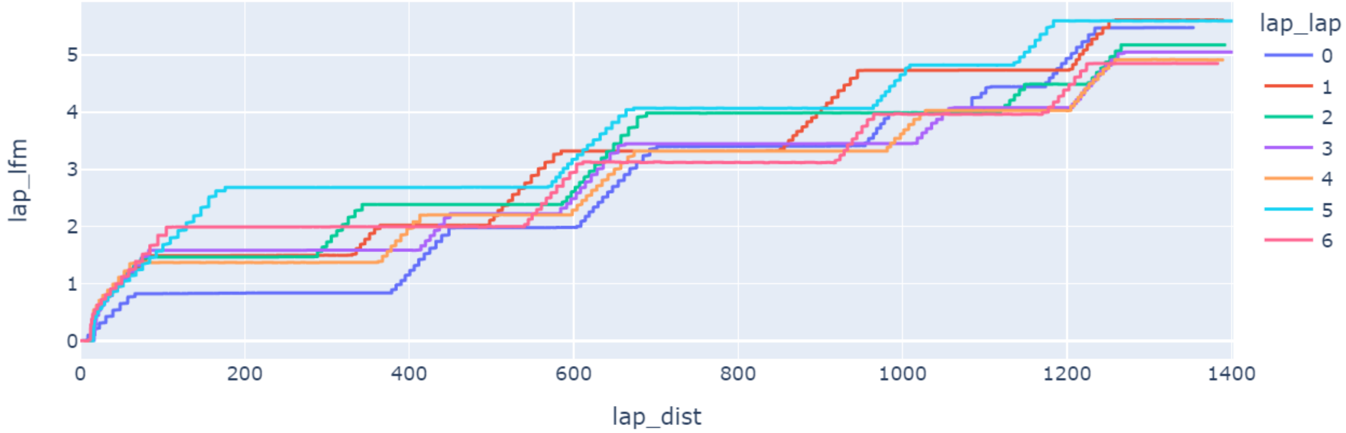
plotly



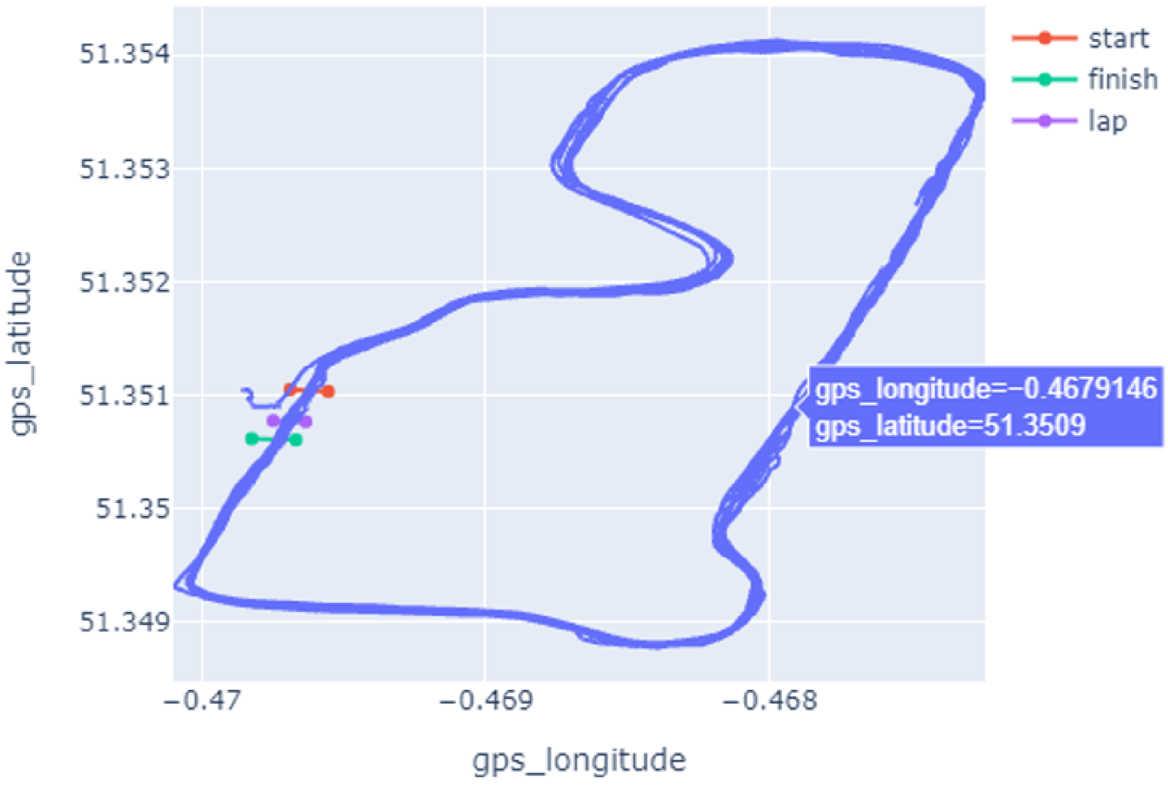

```python
df['dist'] = (
    2
    * r
    * np.arcsin(
        (
            np.sin((lat2 - lat1) / 2) **2
            + np.cos(lat1) * np.cos(lat2) * np.sin((lon2 - lon1) / 2) ** 2
        )
        ** 0.5
    )
)
df['dist'] = df.dist.cumsum()
```

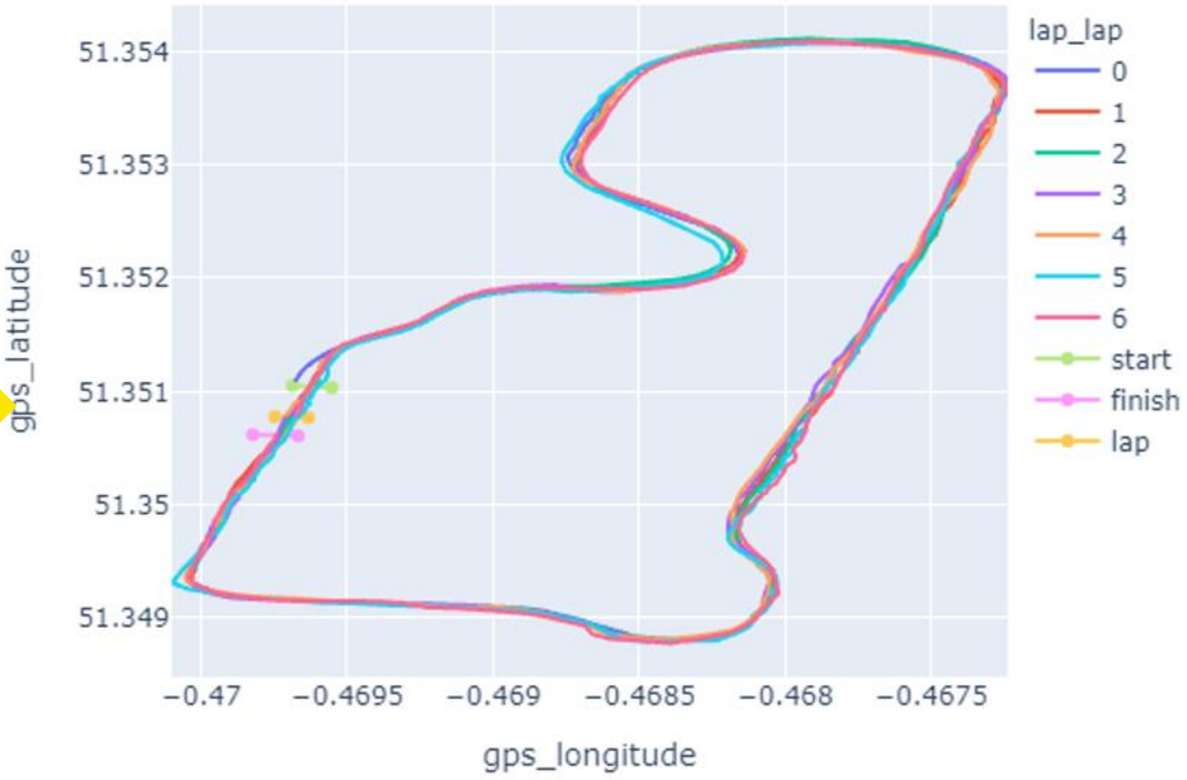




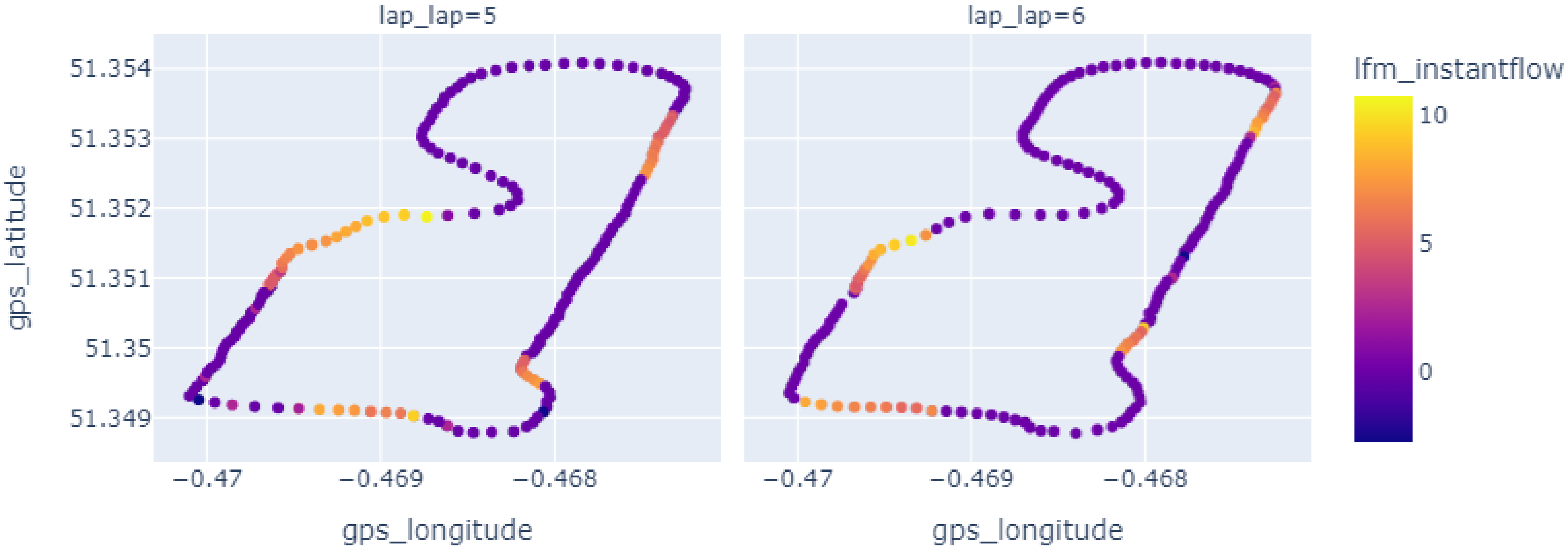


Lap-by-Lap Analytics by Detecting Line Crossings



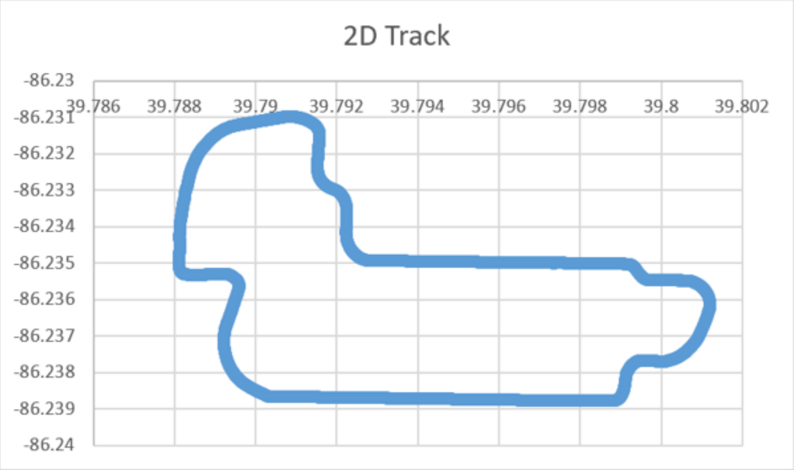


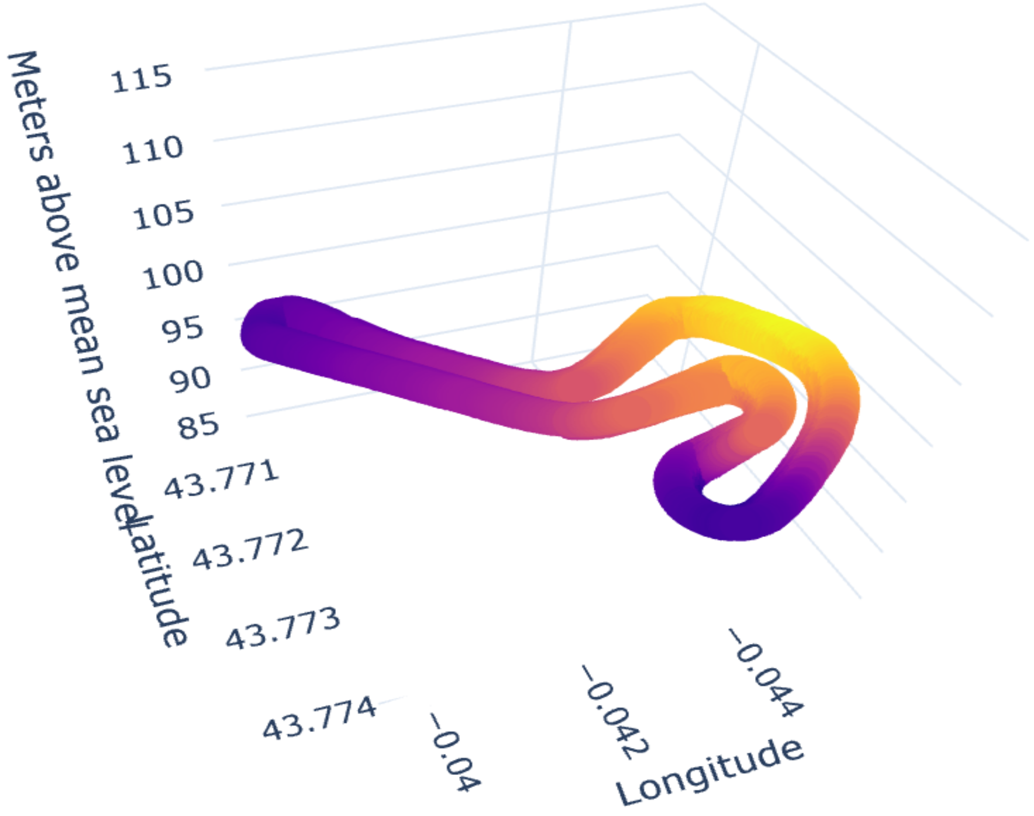
Finding the best Driving Patterns

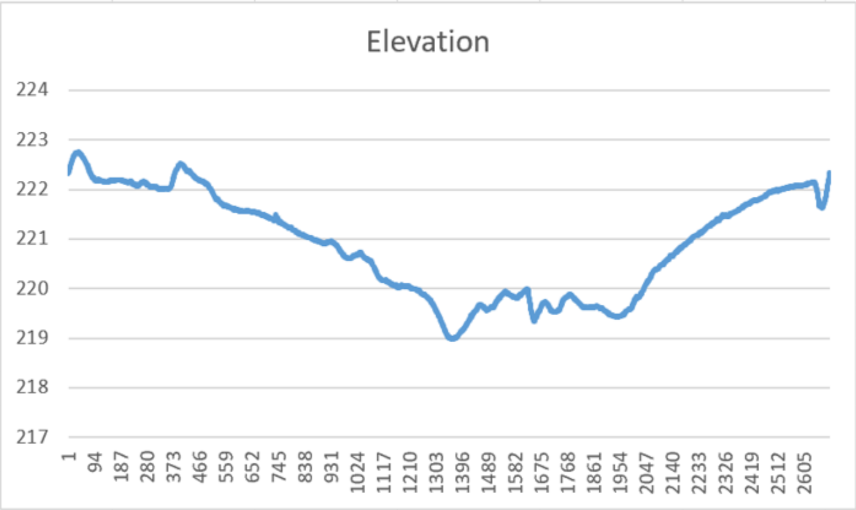


Combine Race Data with 3D-Track Data

Latitude	Longitude	Metres above sea level
39.799168	-86.2380142	222.3313
39.7991733	-86.2379987	222.3617
39.7991794	-86.2379847	222.3676
39.7991864	-86.237972	222.3918
39.7991933	-86.2379596	222.3986
39.7992001	-86.237946	222.4268
39.7992071	-86.2379319	222.4488
39.7992142	-86.237918	222.4692
39.7992211	-86.2379045	222.4826
39.7992281	-86.2378904	222.5074
39.7992364	-86.2378767	222.5145
39.7992452	-86.2378642	222.5369
39.7992539	-86.2378518	222.5572
39.7992621	-86.2378397	222.5701
39.7992708	-86.2378273	222.5817
39.7992803	-86.2378156	222.6093
39.7992904	-86.2378036	222.6299
39.7993003	-86.2377925	222.6386
39.7993102	-86.2377818	222.6497
39.7993201	-86.2377714	222.6728



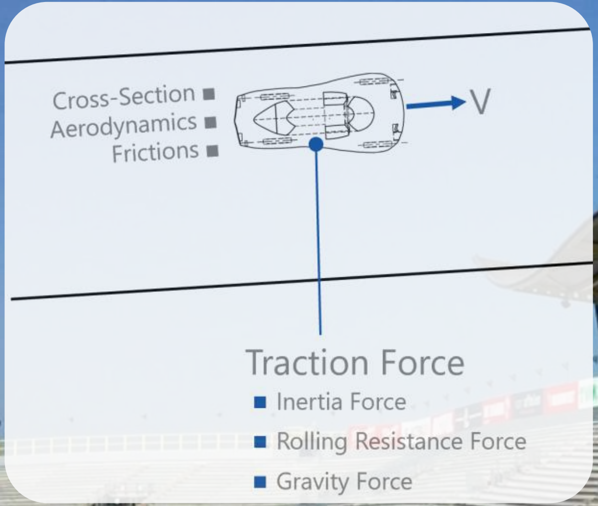




1. On-Track Use Cases
2. Tackling Opposing Goals
3. Connect the Dots!
4. Graphs with Nodes and Edges
5. Define a Race Graph
6. Graph Transformers

1. **On-Track Use Cases**
2. Tackling Opposing Goals
3. Connect the Dots!
4. Graphs with Nodes and Edges
5. Define a Race Graph
6. Graph Transformers

Use Case ❶ driving the Straigths



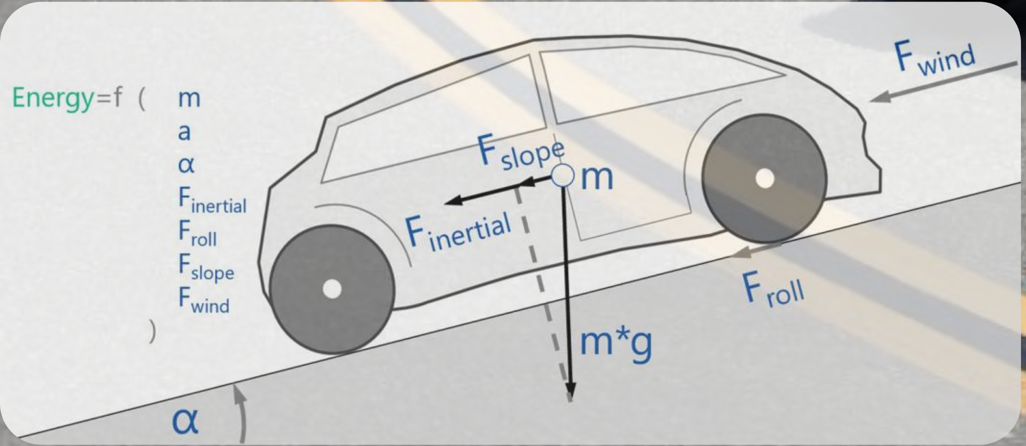
Newtons Laws of Motion

$$F = m * a$$

Use Case 2 Moving up and down **Slopes**

$$dE = \frac{1}{3600}\left[mg(f\cos\varphi + \sin\varphi) + \frac{1}{2}\left(\rho C_x A \frac{(v_{EV}+v_w)^2}{3.6}\right) + \left(m + m_f\right)\frac{dv}{dt}\right] ds$$

dE	Mechanical energy required at the wheels to drive a distance ds [kWh]
m	Total vehicle mass [kg]
m_f	Fictive mass of rolling inertia [kg]
g	Gravitational acceleration [m/s^2]
f	Vehicle coefficient of rolling resistance [-]
φ	Road gradient angle [°]
ρ	Air density [kg/m^3]
C_x	Drag coefficient of the vehicle [-]
A	Vehicle equivalent cross section [m^2]
v_{EV}	Vehicle speed between the point i and the point j [km/h]
v_w	Wind speed projected to the opposing direction of the driving direction [km/h]
ds	Distance driven from point i to point j [km]



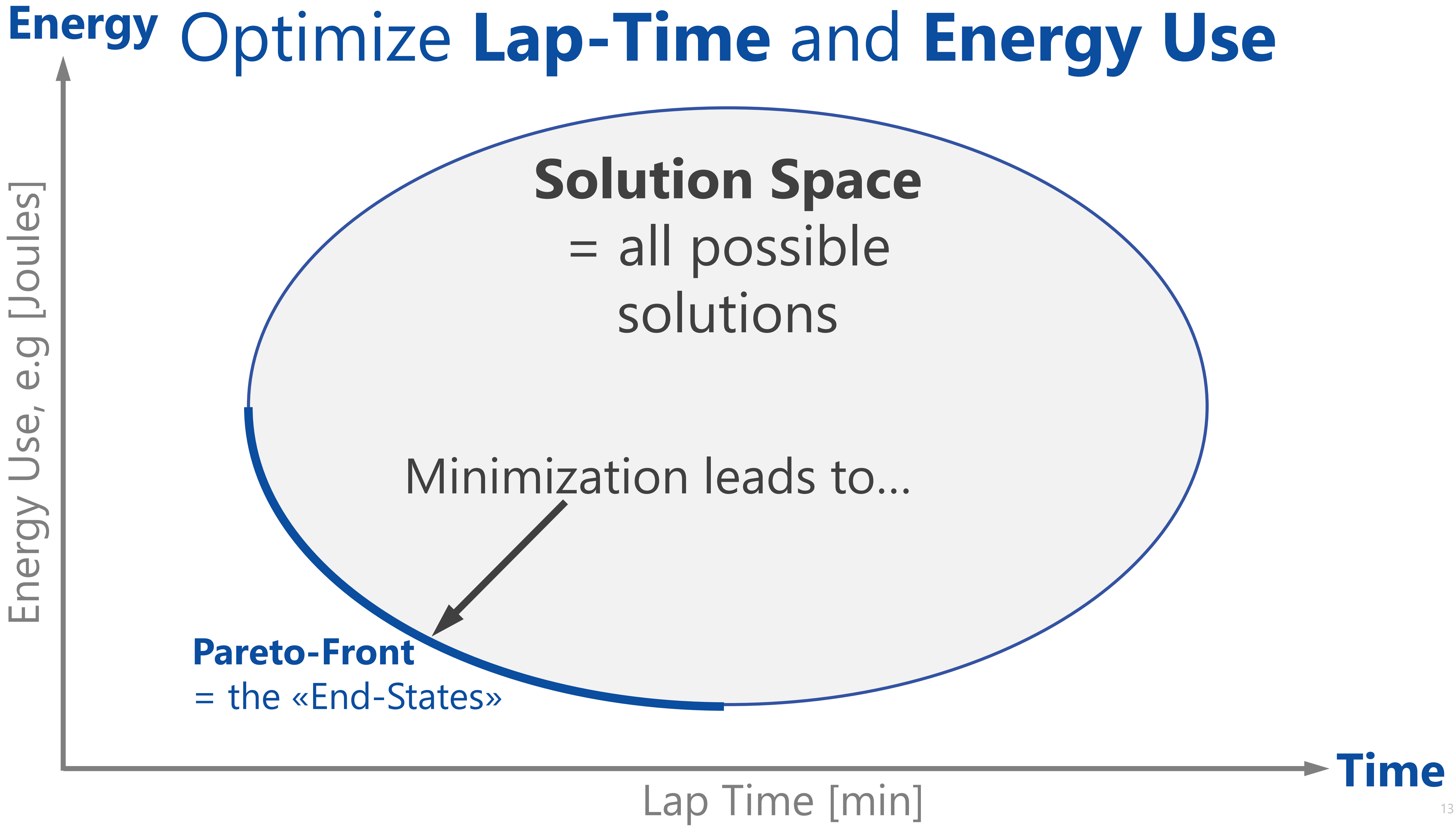
Use Case 3 Cutting the **Corners**

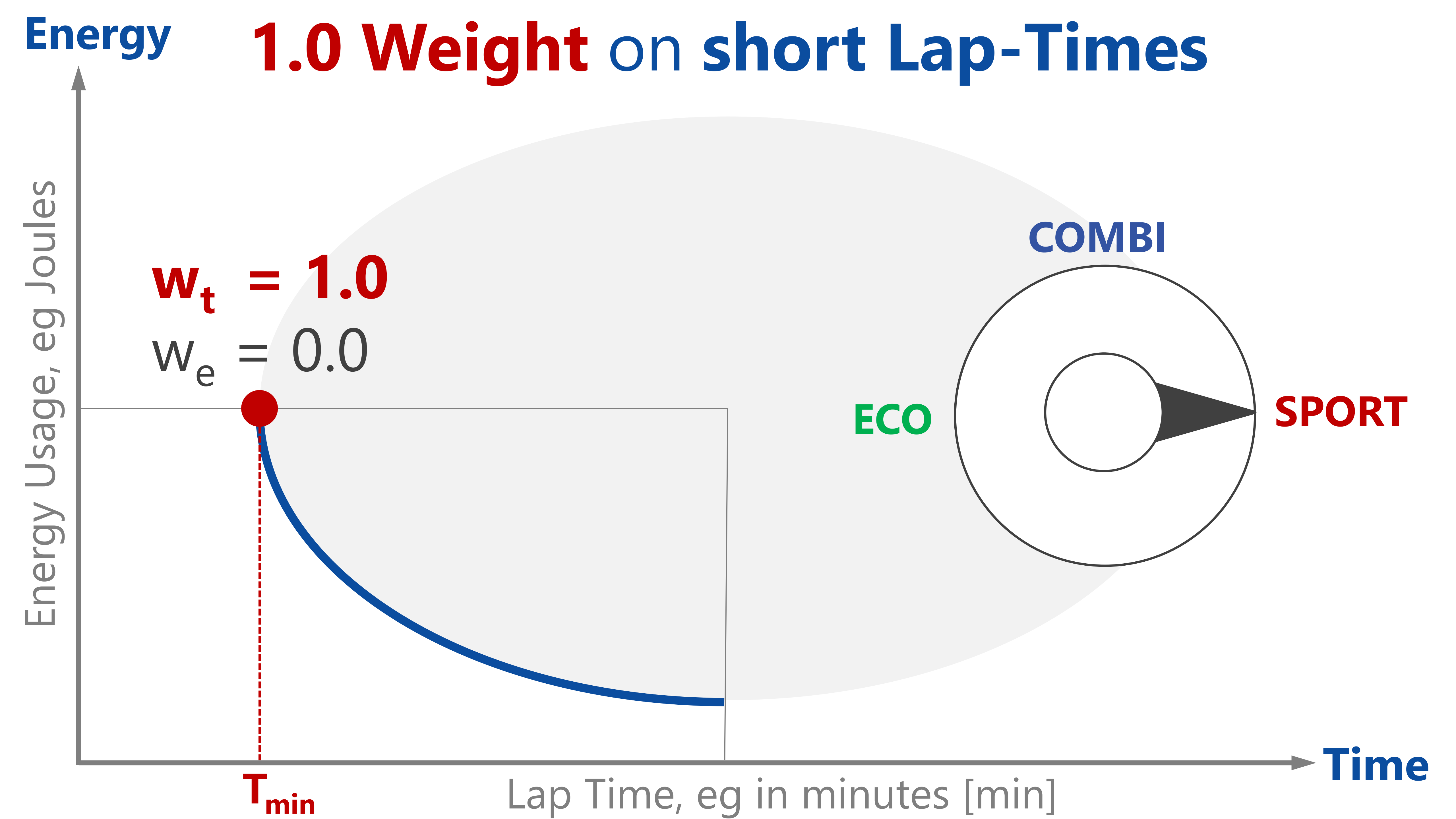
Shell Eco-marathon

SCHMID Elektronik

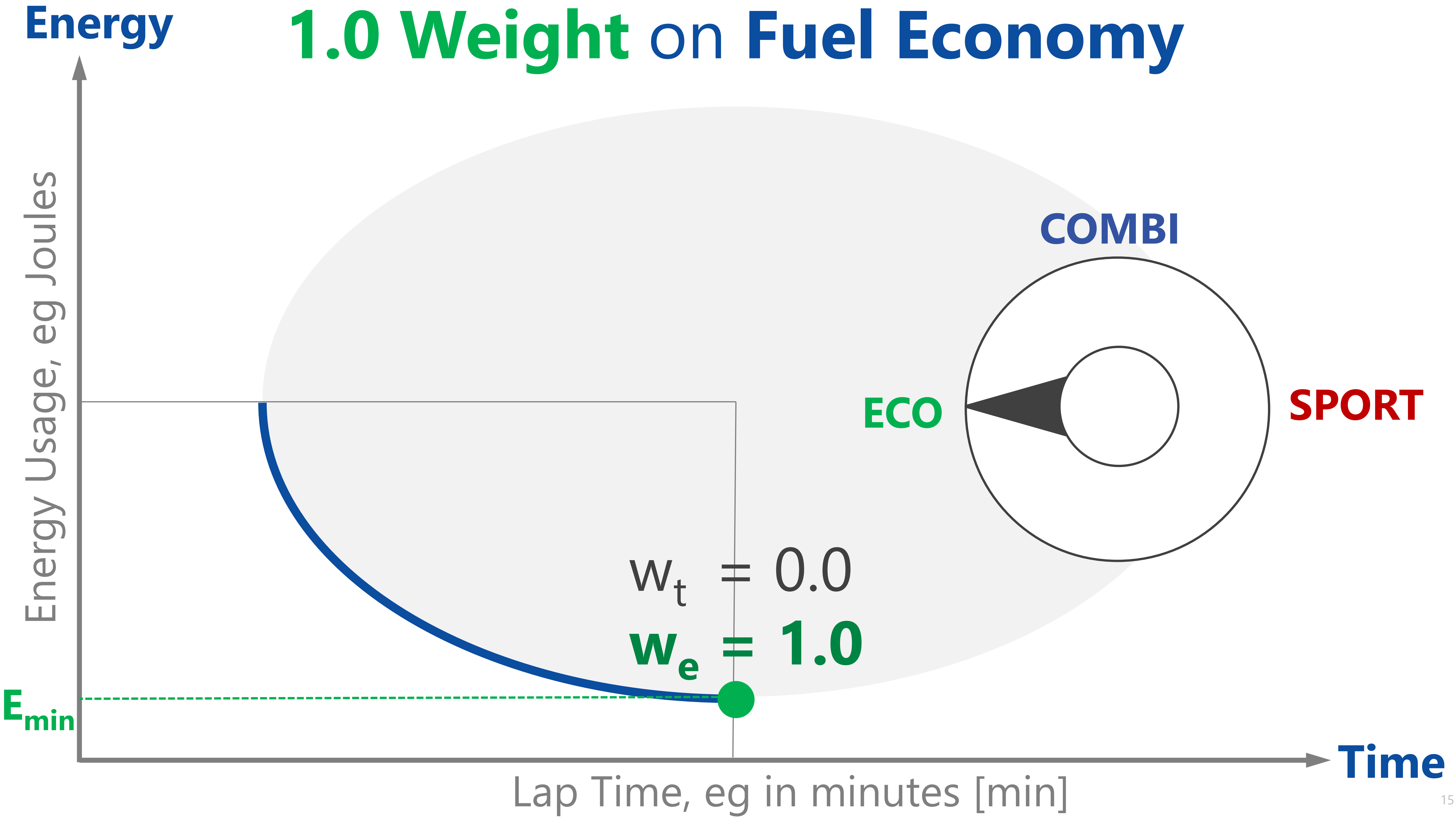
1. On-Track Use Cases
2. **Tackling Opposing Goals**
3. Connect the Dots!
4. Graphs with Nodes and Edges
5. Define a Race Graph
6. Graph Transformers

Optimize **Lap-Time** and **Energy Use**

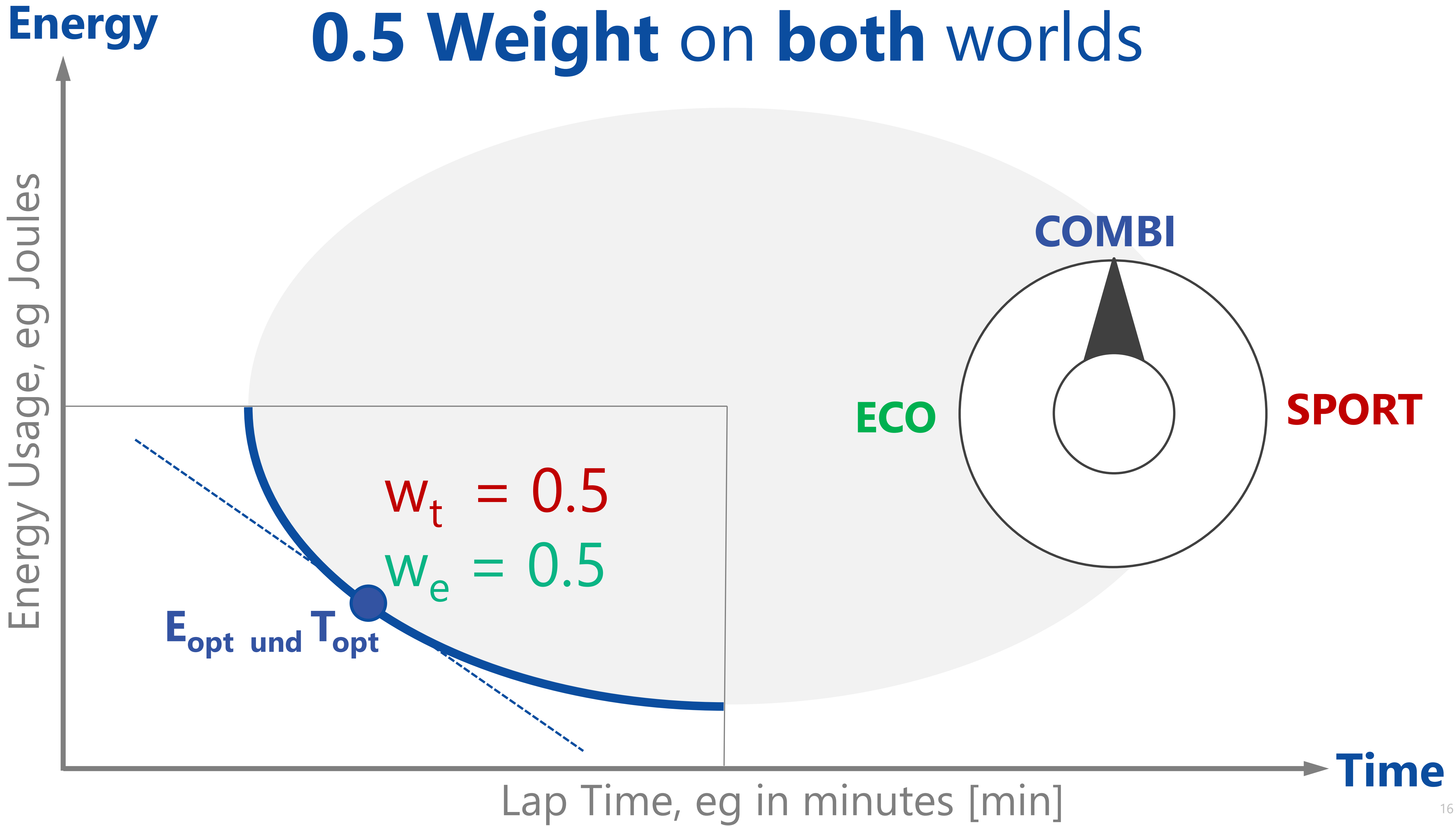



Energy
1.0 Weight on short Lap-Times
COMBI
$w_t = 1.0$
$w_e = 0.0$
ECO
SPORT
Energy Usage, eg Joules
T_{min}
Lap Time, eg in minutes [min]
Time

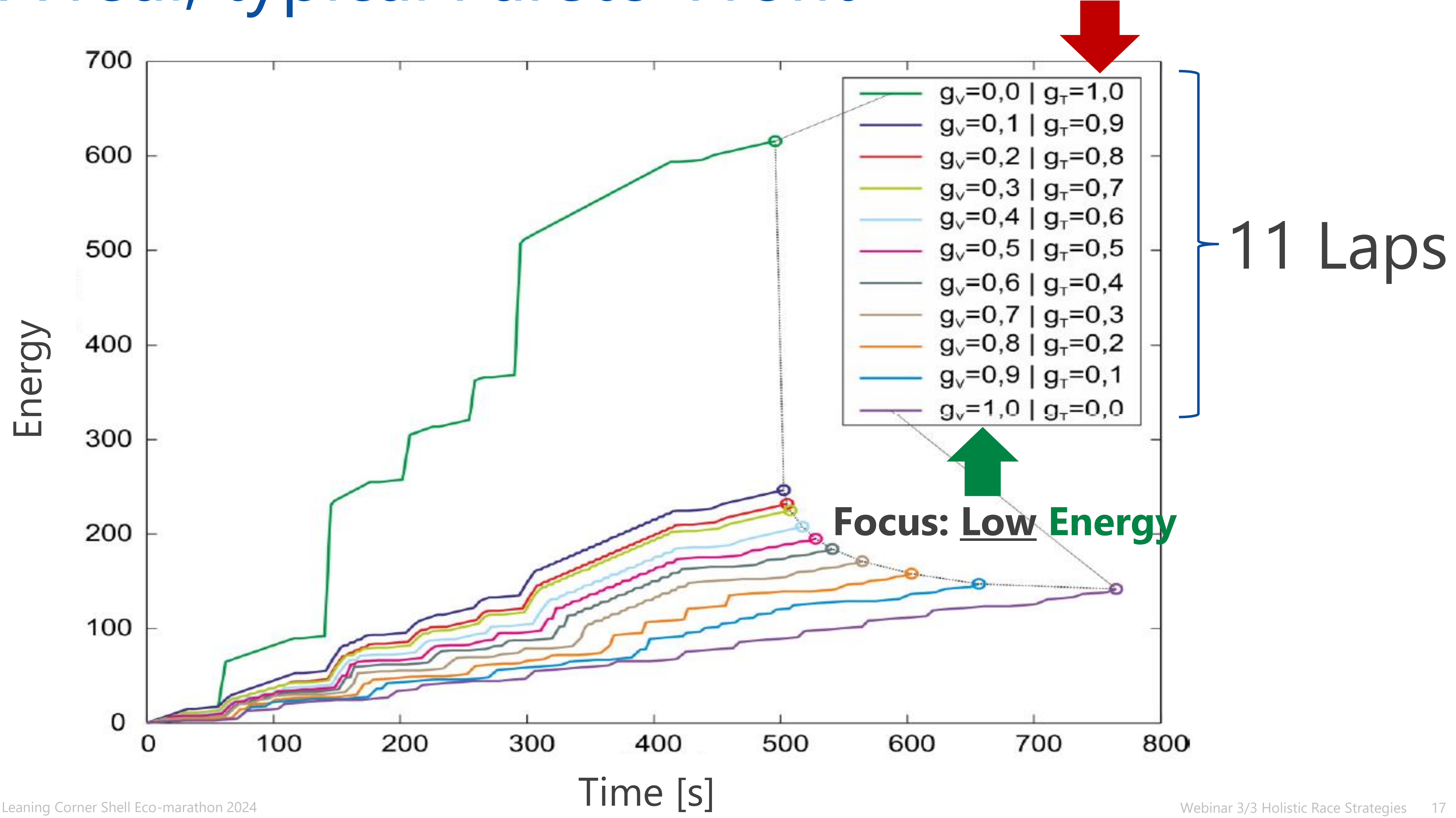
1.0 Weight on Fuel Economy



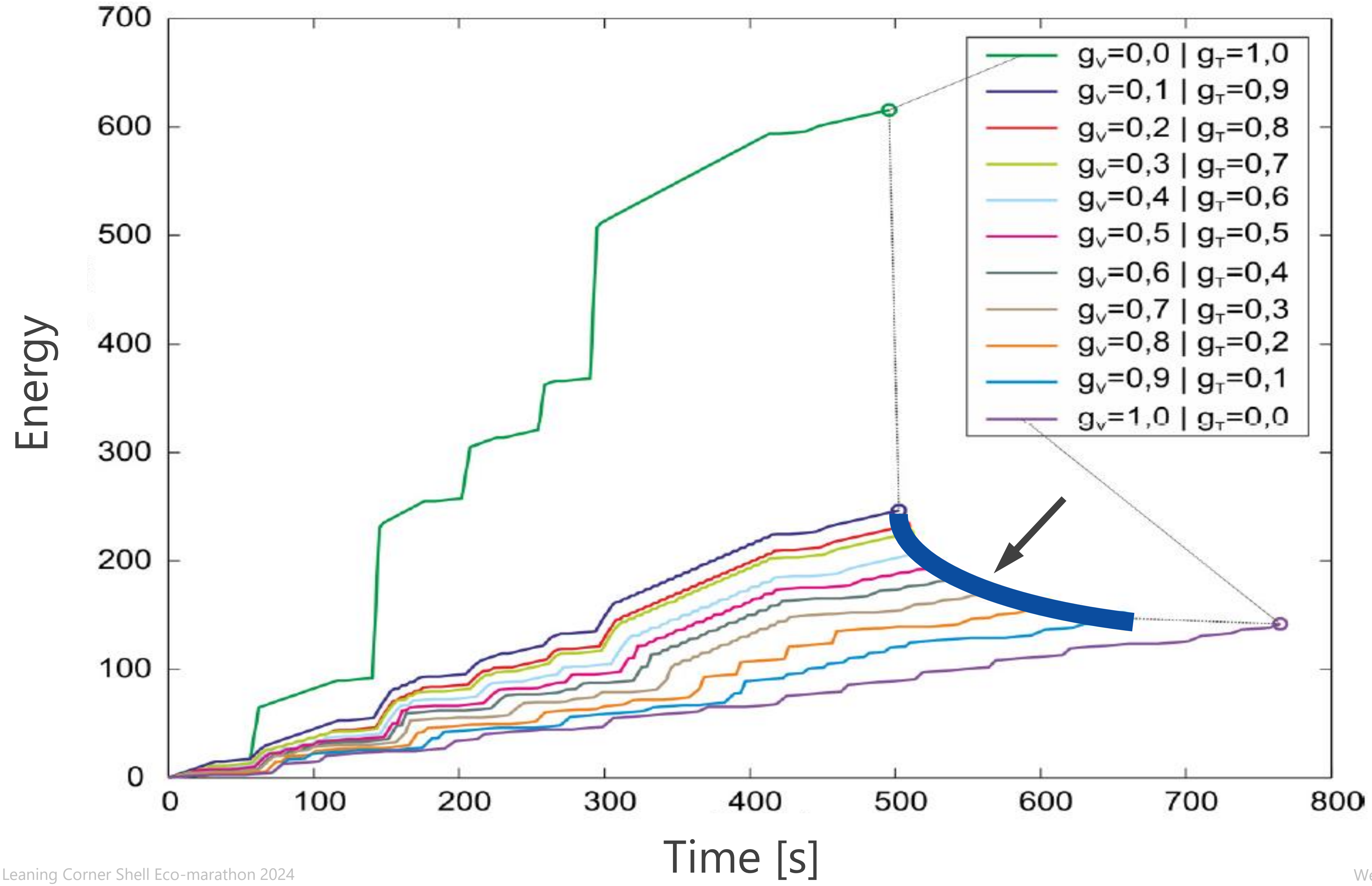
0.5 **Weight** on **both** worlds



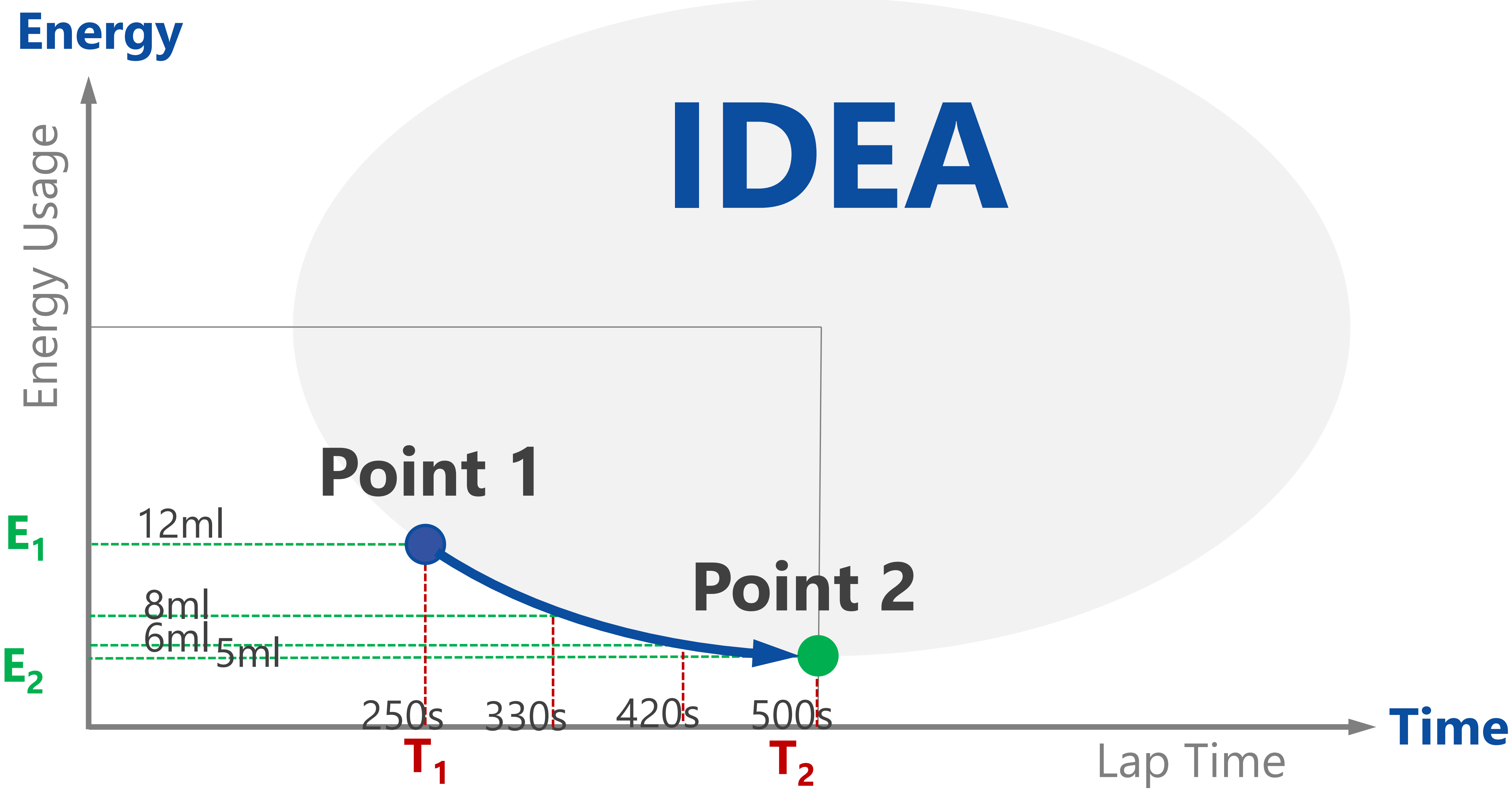
A real, typical Pareto-Front



A real, typical Pareto-Front



MVP-Philosophy: Move from Point 1 to 2



Using Minimum Viable Products (MVPs)

Check your Hypotheses

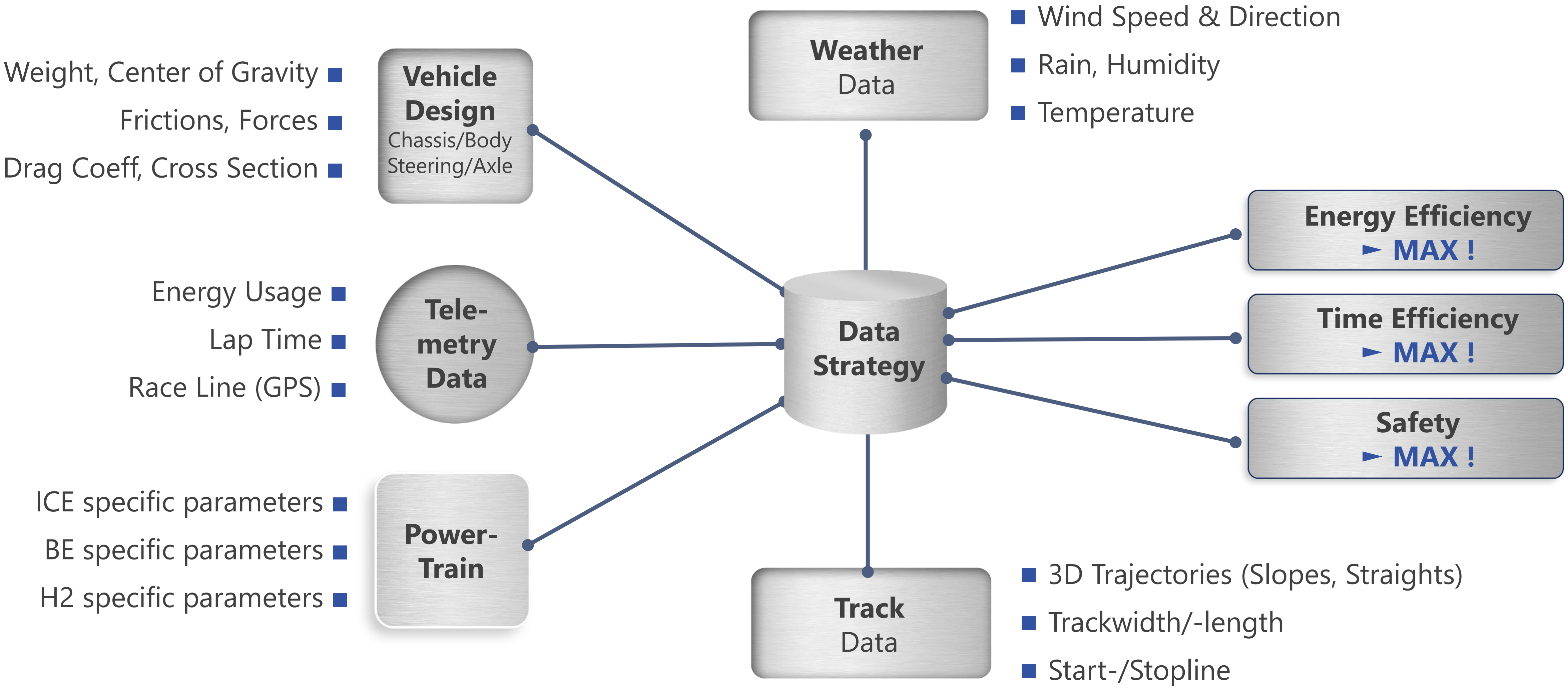
	MVP #1	MVP #2	MVP#3
	Laptime : 0% Energy : 0%	Laptime : +32% Energy : -34%	Laptime : +68% Energy : -50%
Lap Time [s]	**250s**	**330s**	**420s**
Energy Used / Lap [ml]	**12ml**	**8ml**	**6ml**
Was the lap Safe?	**OKV**	**OK**	**OK**

Shell Eco-marathon

SCHMID Elektronik

1. On-Track Use Cases
2. Tackling Opposing Goals
3. **Connect the Dots!**
4. Graphs with Nodes and Edges
5. Define a Race Graph
6. Graph Transformers

Let us start with a Data Strategy



Dilemma: bringing EVERYTHING together

1. Include **Straights**, **Slopes**, and **Corners**

2. Optimize **Energy Usage**

3. Optimize **Laptime**

4. Maintain high **Safety**

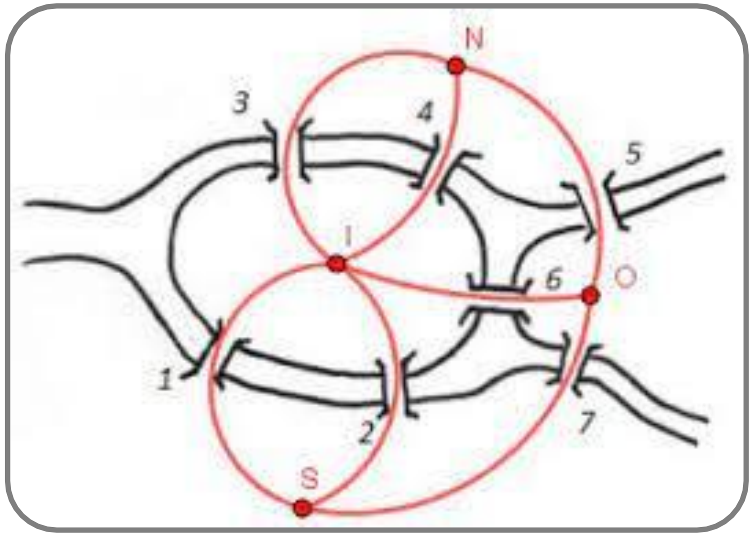
connect the **dots** and solve physical problems the data science way: **Framework**

Shell Eco-marathon

SCHMID Elektronik

1. On-Track Use Cases
2. Tackling Opposing Goals
3. Connect the Dots!
4. **Graphs with Nodes and Edges**
5. Define a Race Graph
6. Graph Transformers

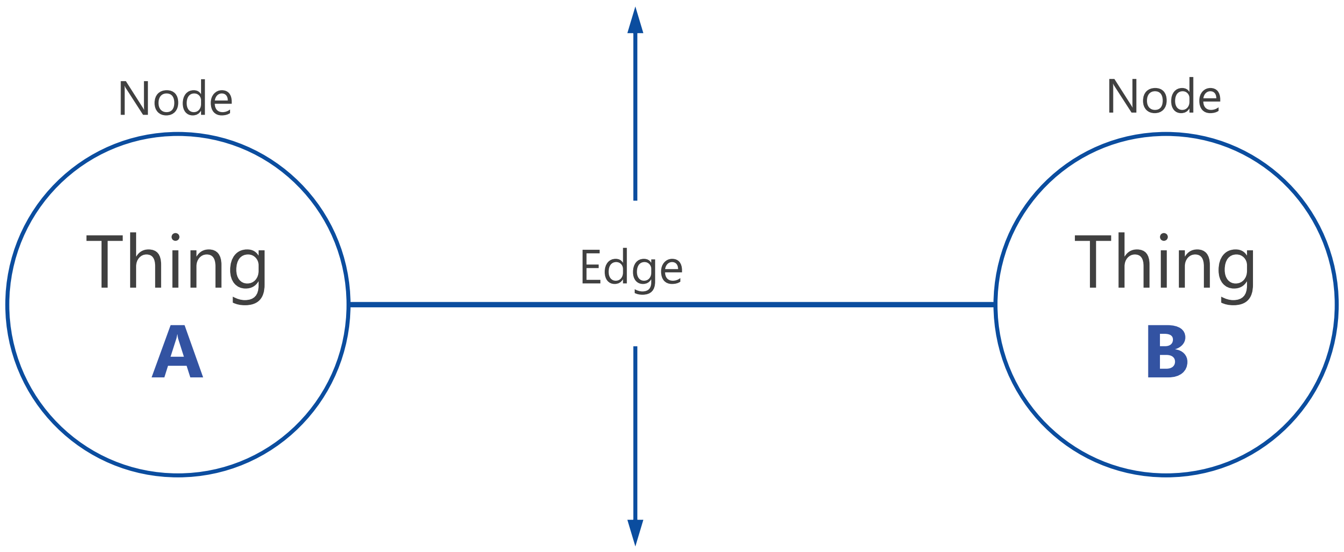
Graph Theory started around 1750



Understanding the
Nature of Things
and
Their **Relationships**
such as the
7 Bridges of Königsbert

Knowledge Graph: a language for «Things»

A Common „Currency“, e.g. Weight, Cost, Energy, etc



Well defined Semantics

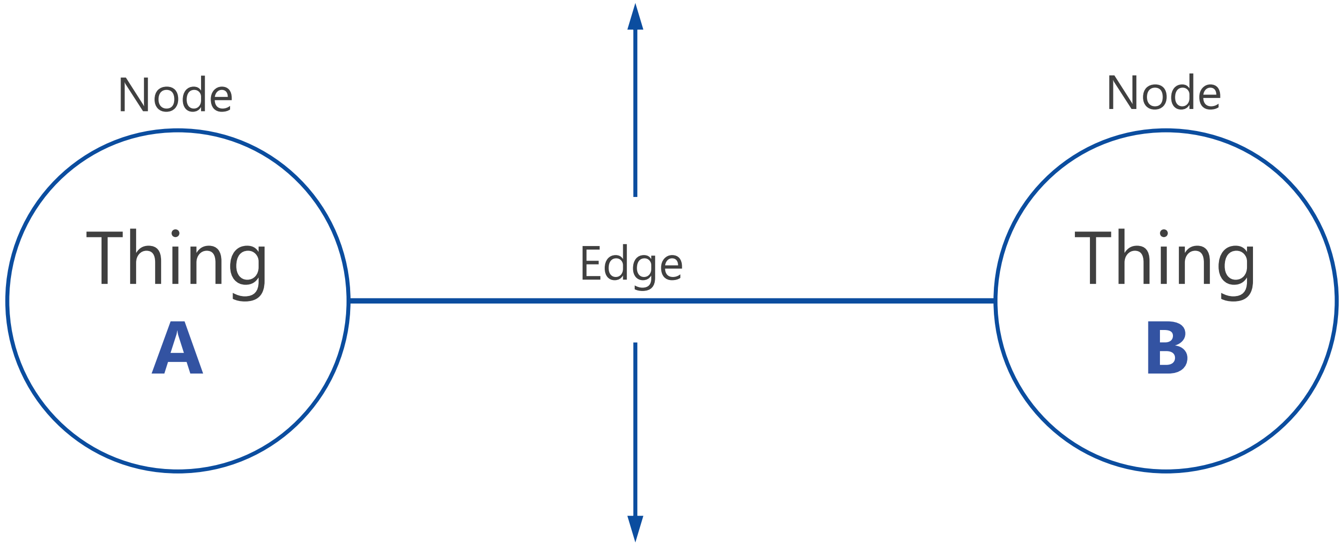
CONTAINS (<u>space</u>)
is_part_of- includes-
belongs_to
generalizes
incorporates
unites

FOLLOWS (<u>time</u>)
precedes
uses
depends_on
derived_from
happened_after
happened_before
contributes_to
caused_by

EXPRESSES (<u>information</u>)
shares_with
represents
promises
name
near
like
similar_to
approximates
connects_to
seen_together_with

Knowledge Graph: a language for «Things»

A Common „Currency", e.g. Weight, Cost, Energy, etc

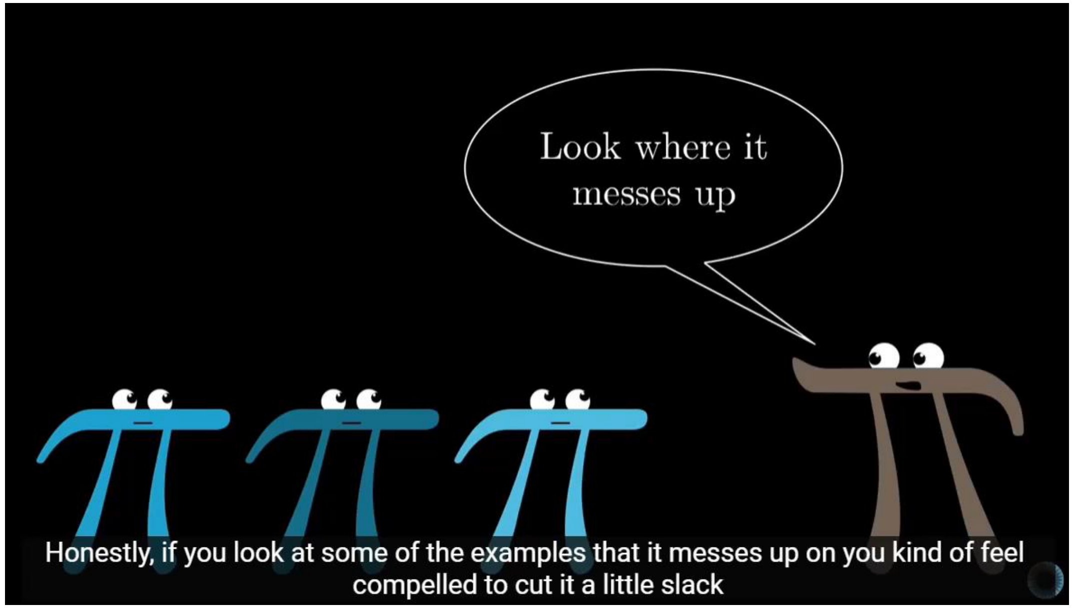


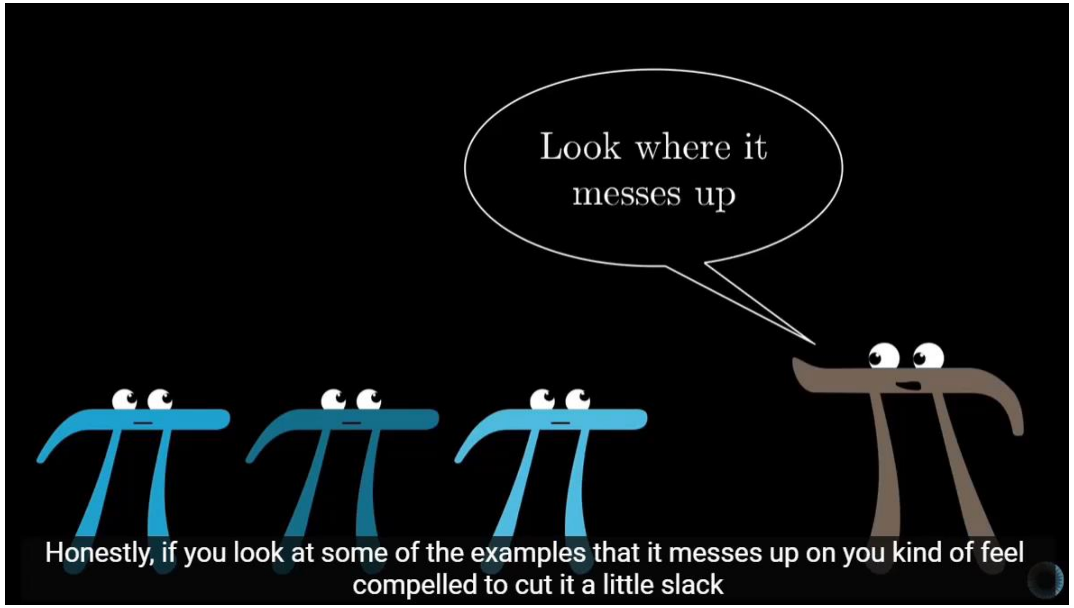
Well defined Semantics

CONTAINS (space)
is_part_of- includes-
belongs_to
generalizes
incorporates
unites

FOLLOWS (time)
precedes
uses
depends_on
derived_from
happened_after
happened_before
contributes_to
caused_by

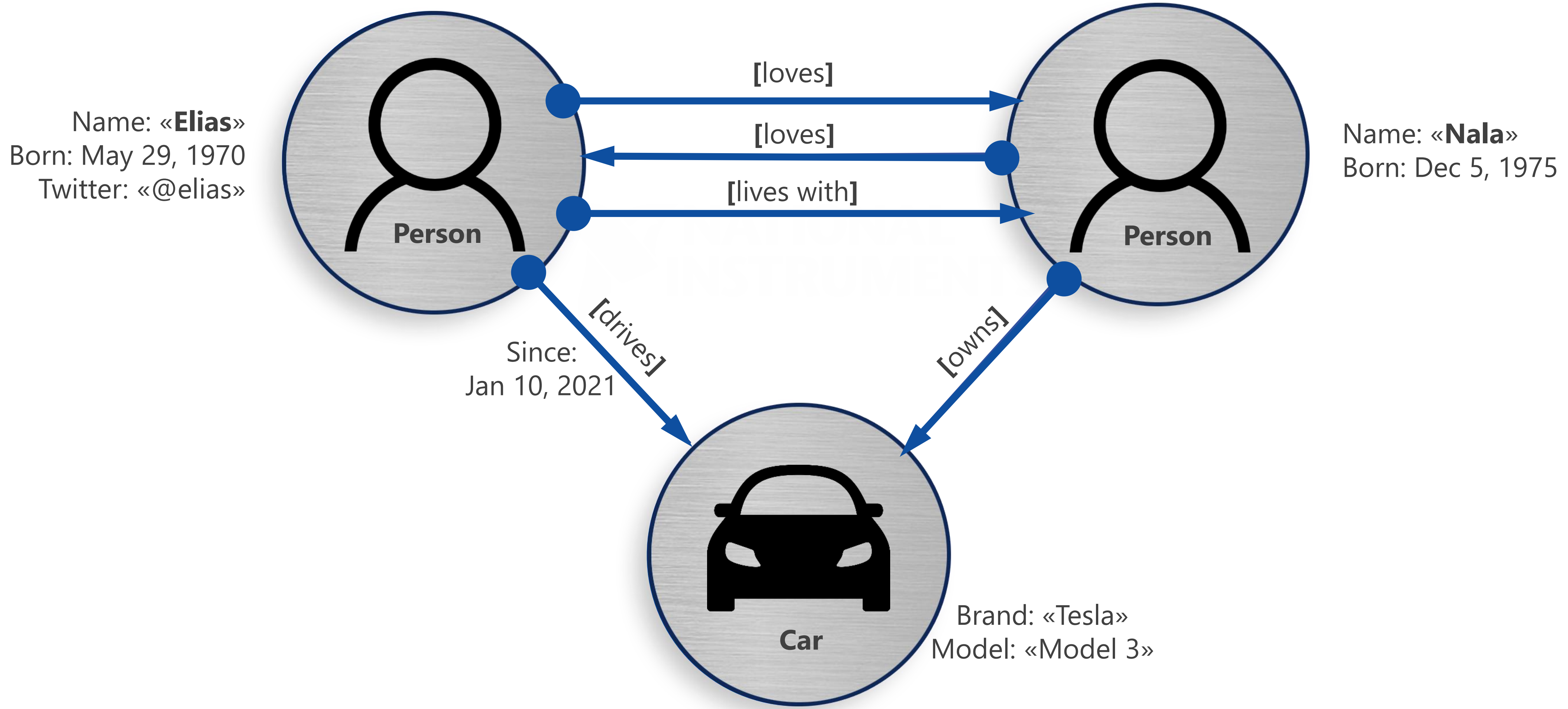
EXPRESSES (information)
shares_with
represents
promises
name
near
like
similar_to
approximates
connects_to
seen_together_with



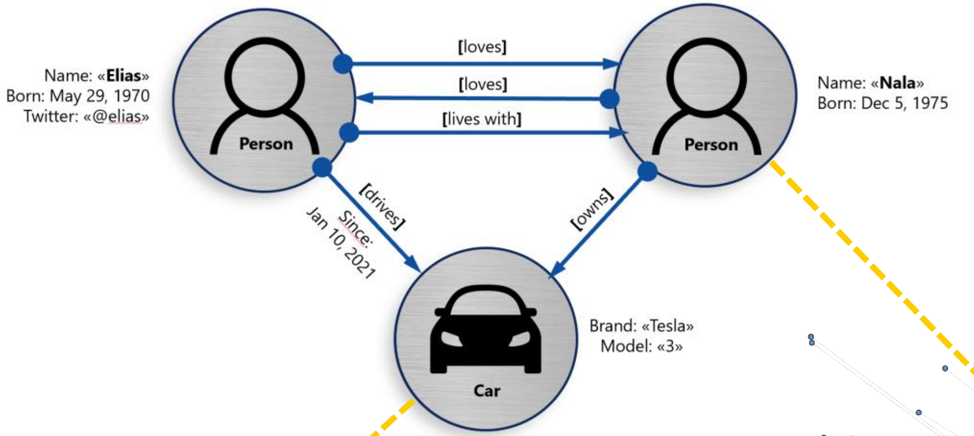


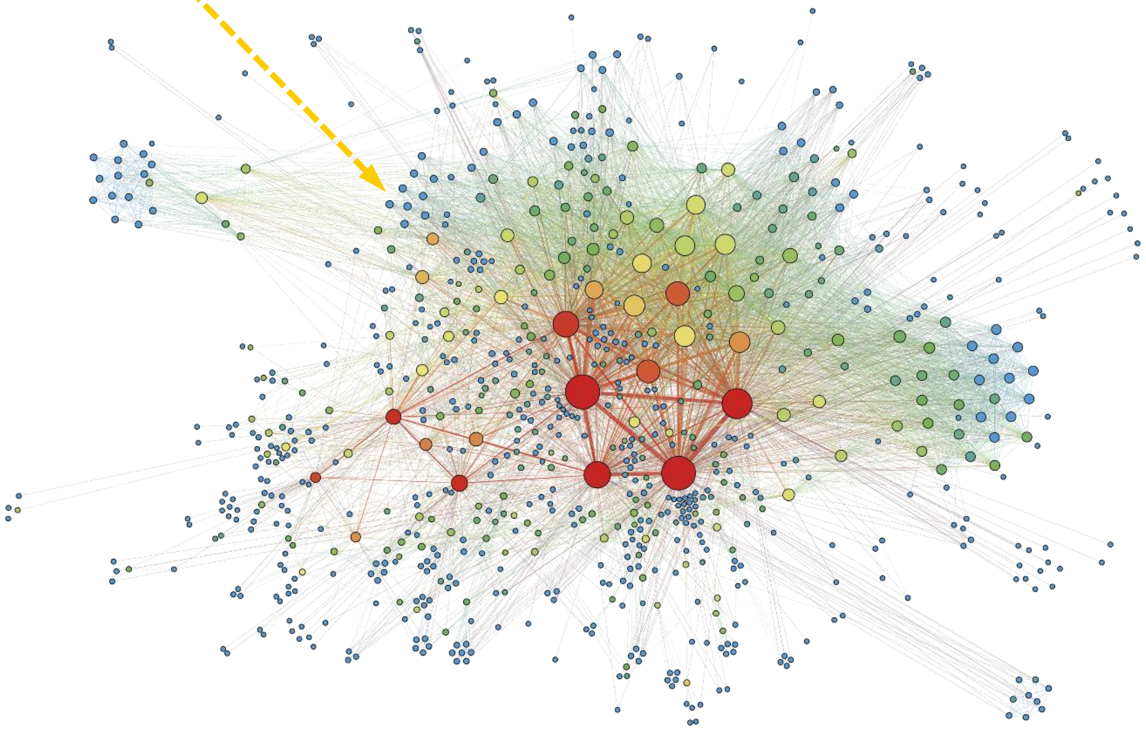
A Knowledge Graph has Nodes and Edges

Understanding the «**Essence of Things**»

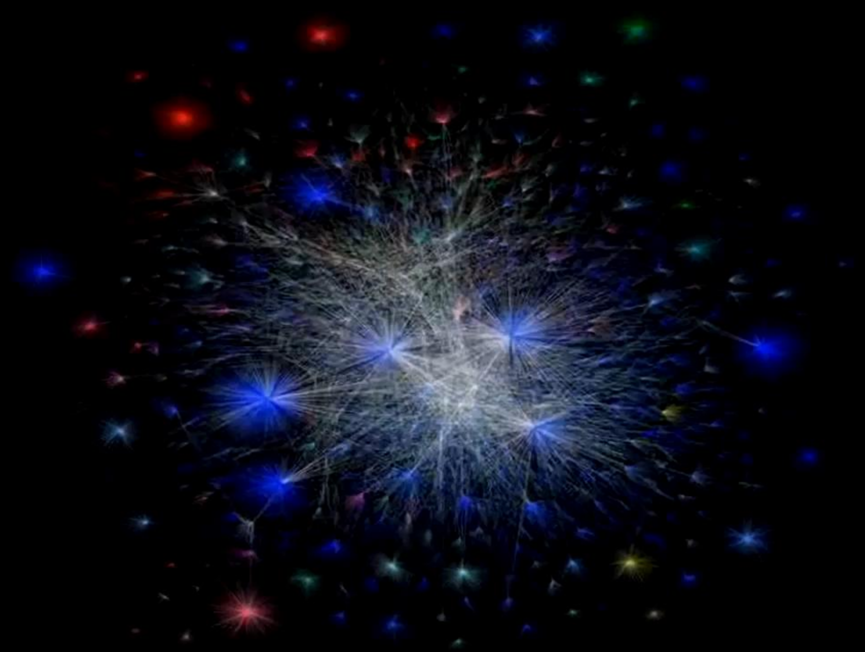


Networks and Graphs can Scale…

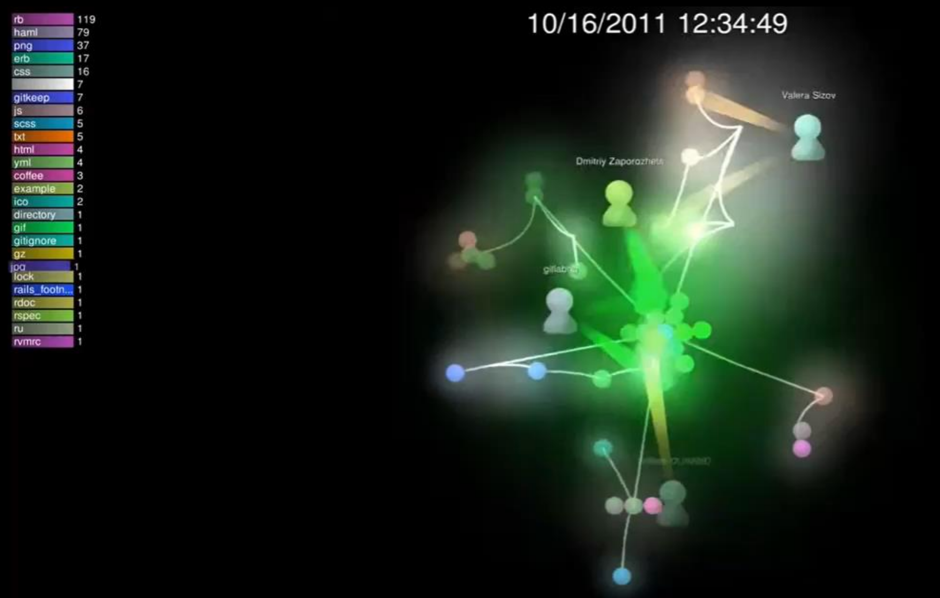




Complex, Large Scale Networks



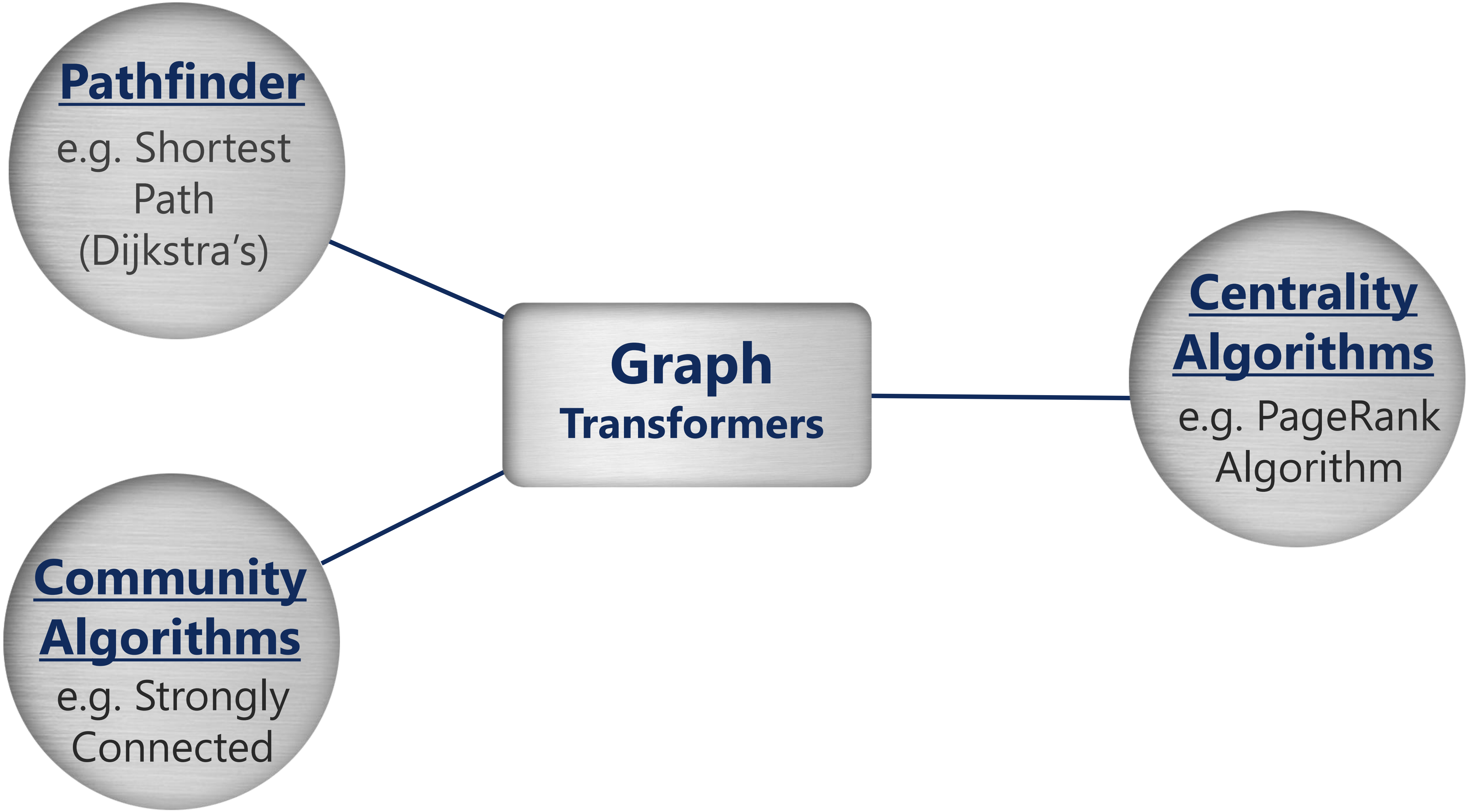
1998



Everyone Who Has Contributed

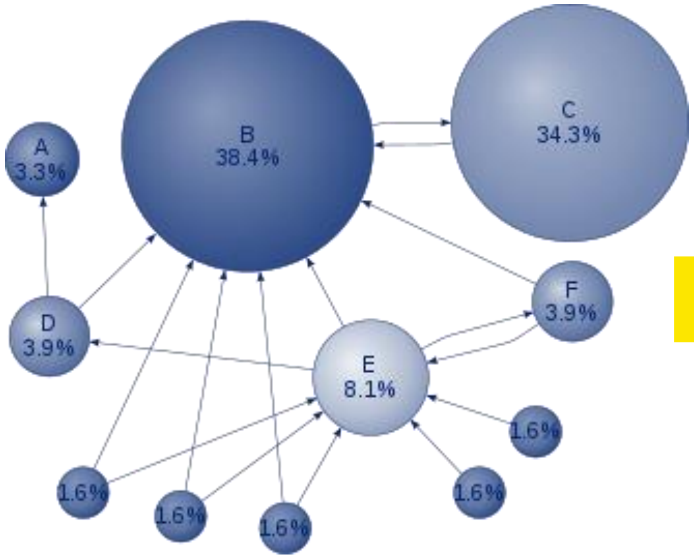
GitLab

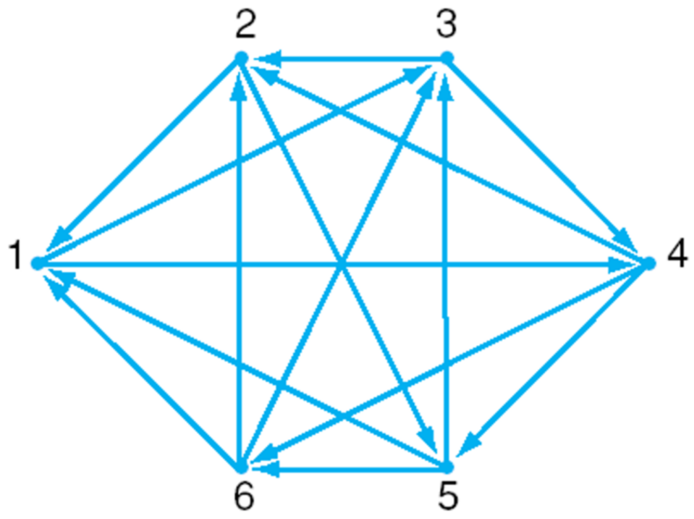
Transformers translate Graphs into Meaning



Transformers translate Graphs into Meaning

PageRank
Algorithm





$$M = \begin{bmatrix} 0 & 0 & 1/6 & 1/6 & 0 & 0 \\ 1/6 & 0 & 0 & 0 & 1/6 & 0 \\ 0 & 1/6 & 0 & 1/6 & 0 & 0 \\ 0 & 1/6 & 0 & 0 & 1/6 & 1/6 \\ 1/6 & 0 & 1/6 & 0 & 0 & 1/6 \\ 1/6 & 1/6 & 1/6 & 0 & 0 & 0 \end{bmatrix}$$

Shell Eco-marathon

SCHMID Elektronik

1 On-Track Use Cases

2 Tackling Opposing Goals

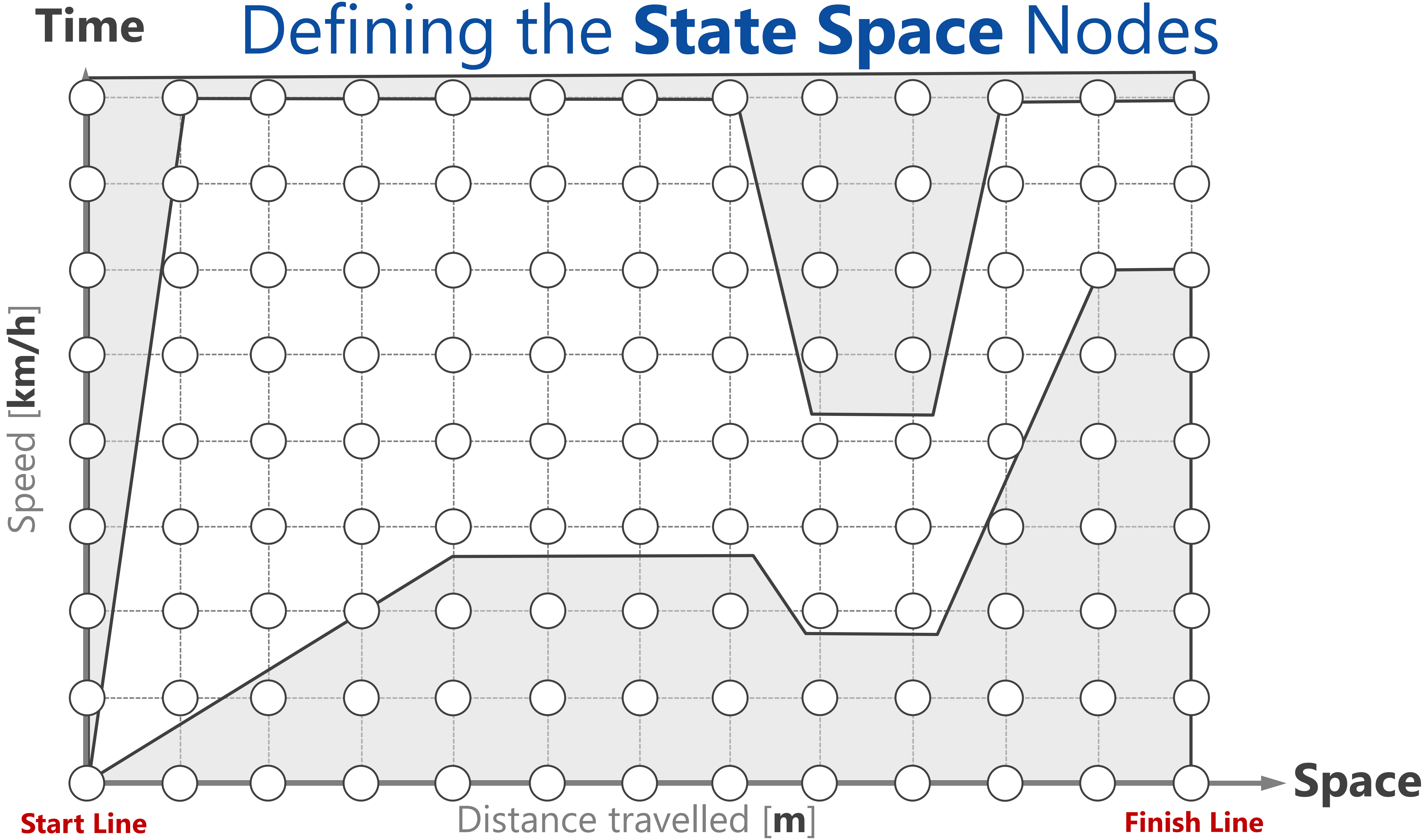
3 Connect the Dots!

4 Graphs with Nodes and Edges

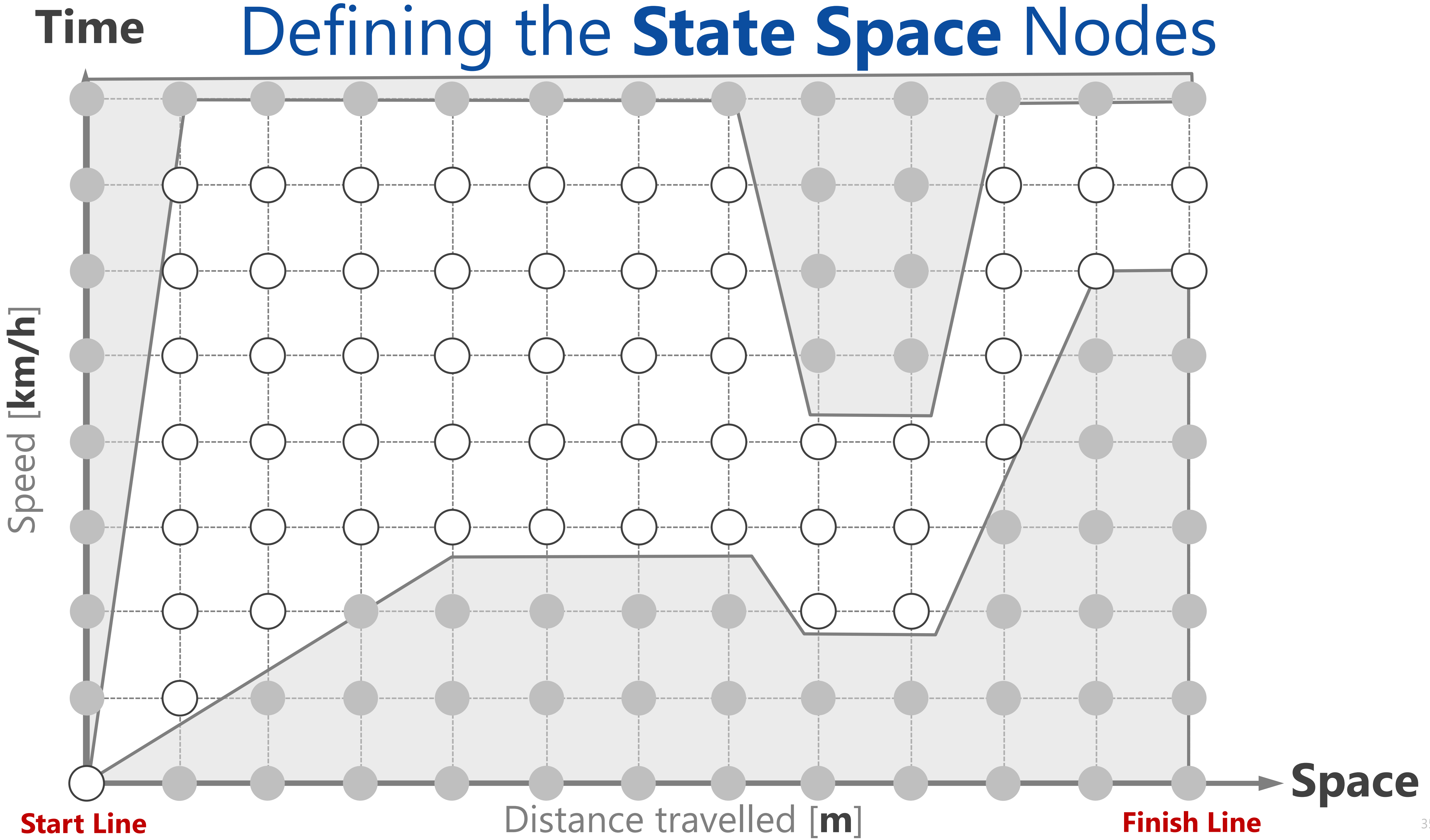
5 Define a Race Graph

6 Graph Transformers

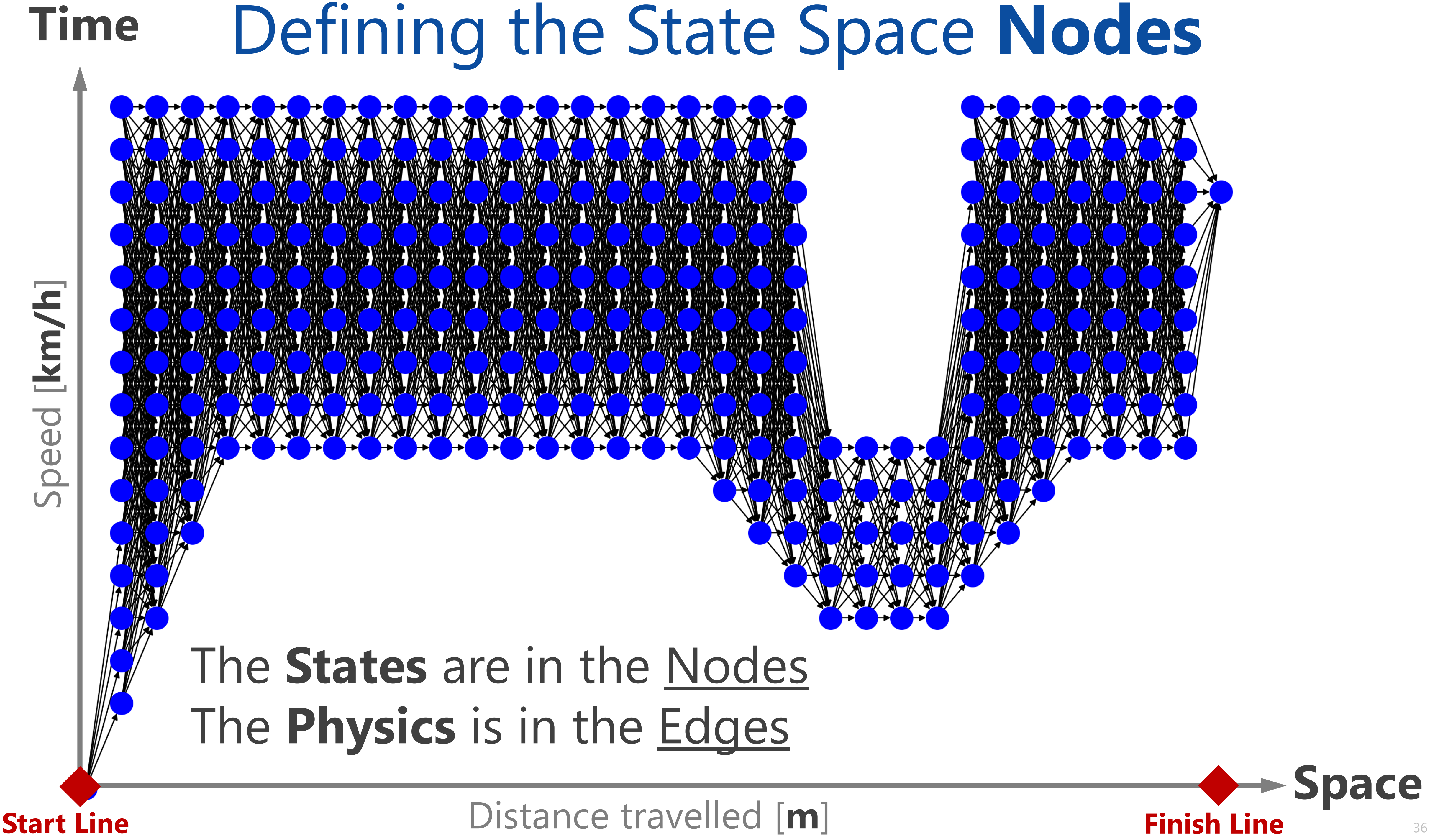
Defining the **State Space** Nodes



Defining the **State Space** Nodes

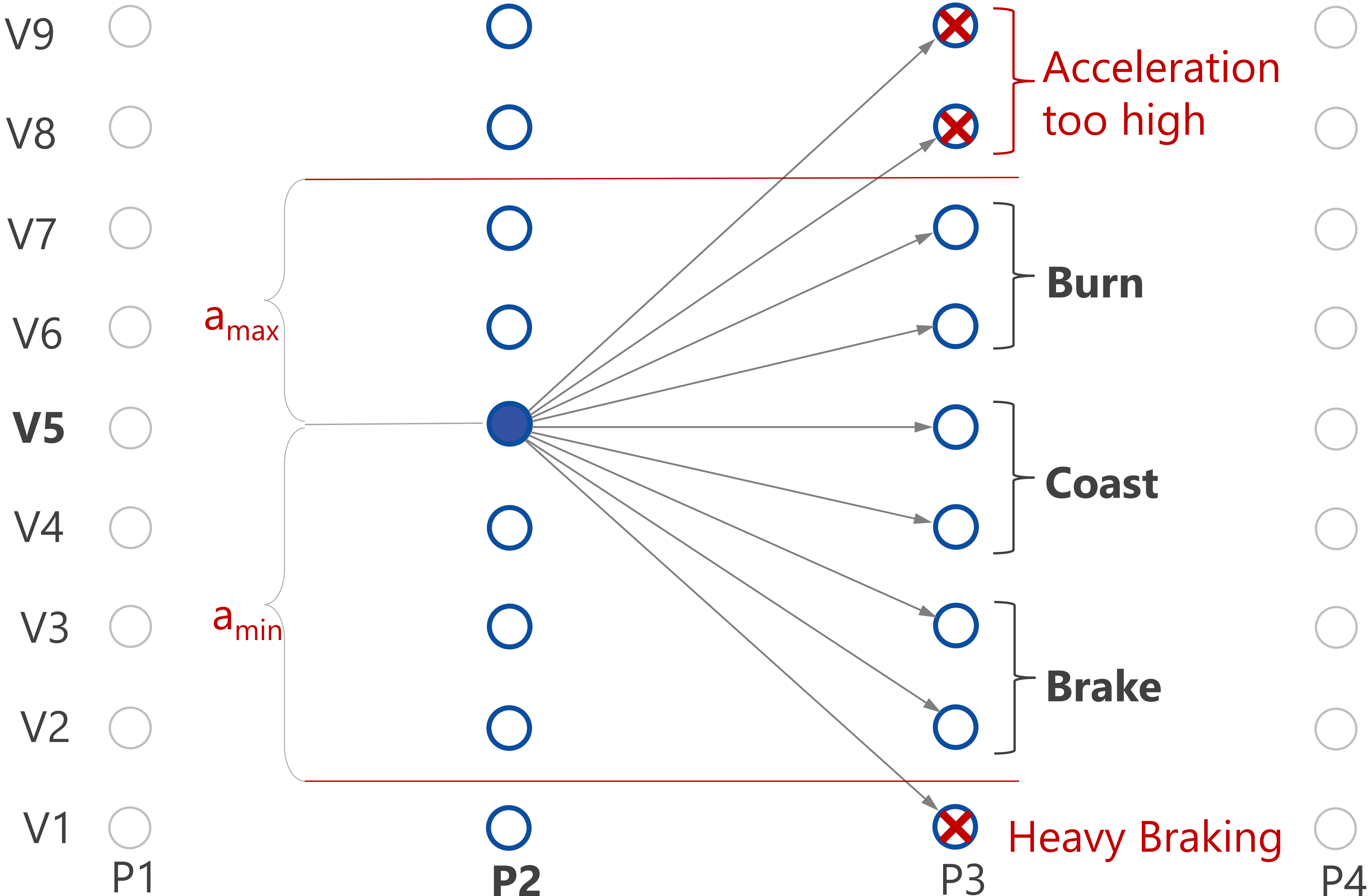


Defining the State Space **Nodes**



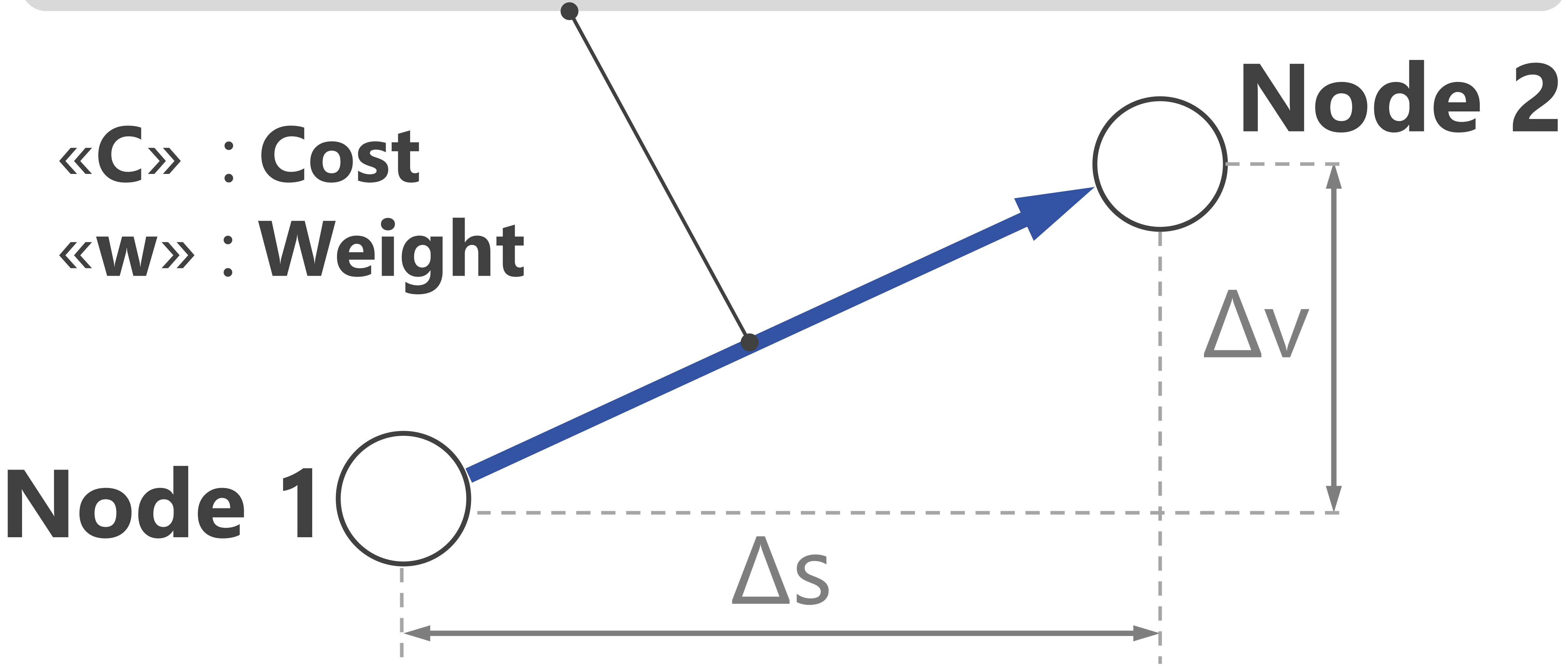
The **States** are in the <u>Nodes</u>
The **Physics** is in the <u>Edges</u>

The Graphs **Edges**



Each **State Change** is a **Decision** with a **Cost**

$$C_{edge} = C_{time} * w_{time} + C_{energy} * w_{energy} \rightarrow \text{MIN!}$$



$$C_{edge} = C_{time} * W_{time} + C_{energy} * W_{energy}$$

Cost of required for time travelled

Cost of required for energy used

$$C_{time} = f(\Delta t) = f(\Delta s / \text{Ø} v)$$

$$C_{energy} = f(\Delta e) = f(F_{total})$$

C_{time} = Cost = {0...1}

W_{time} = Weight = {0...1}

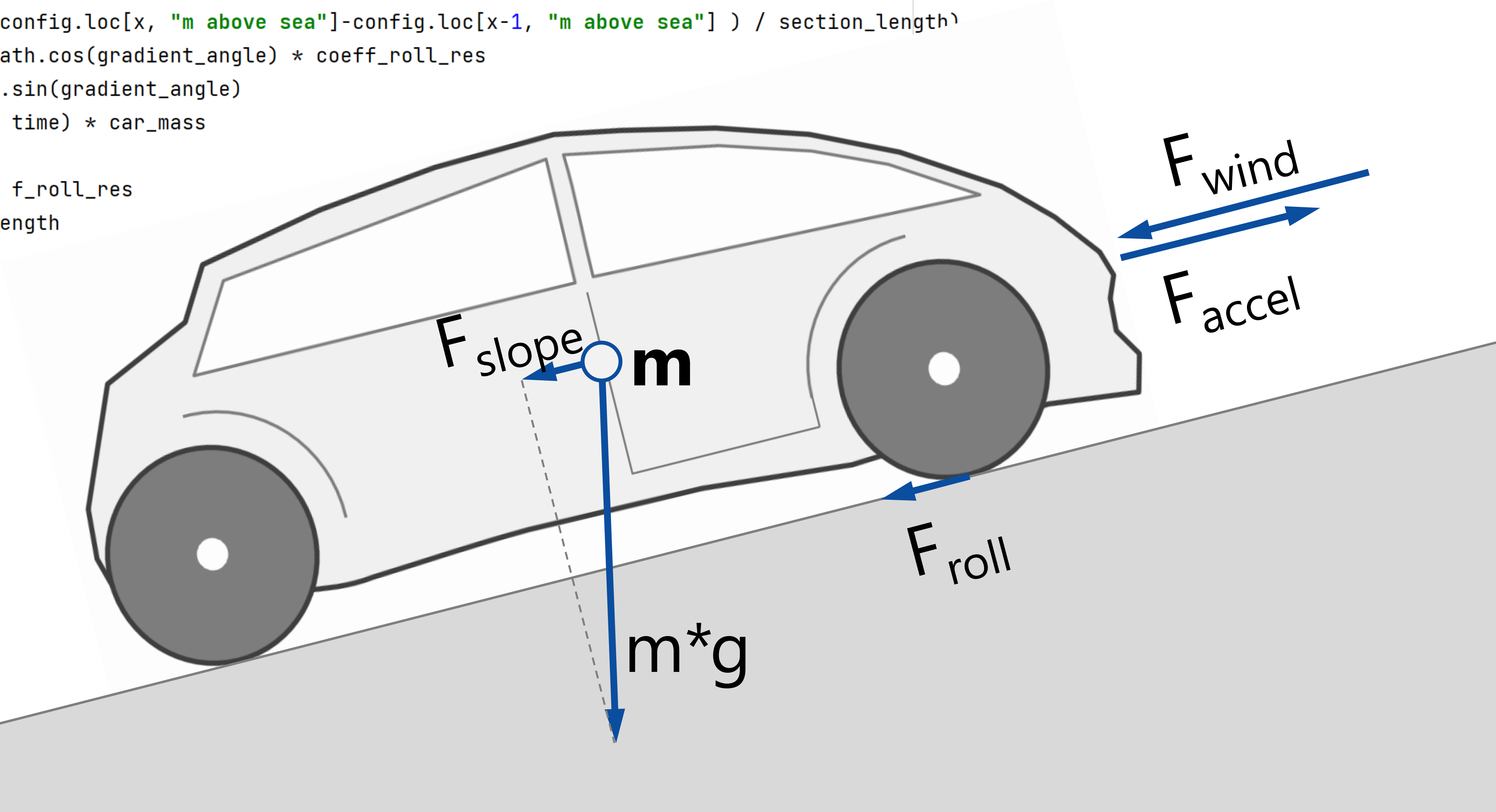
C_{energy} = {0...1}

W_{energy} = {0...1}

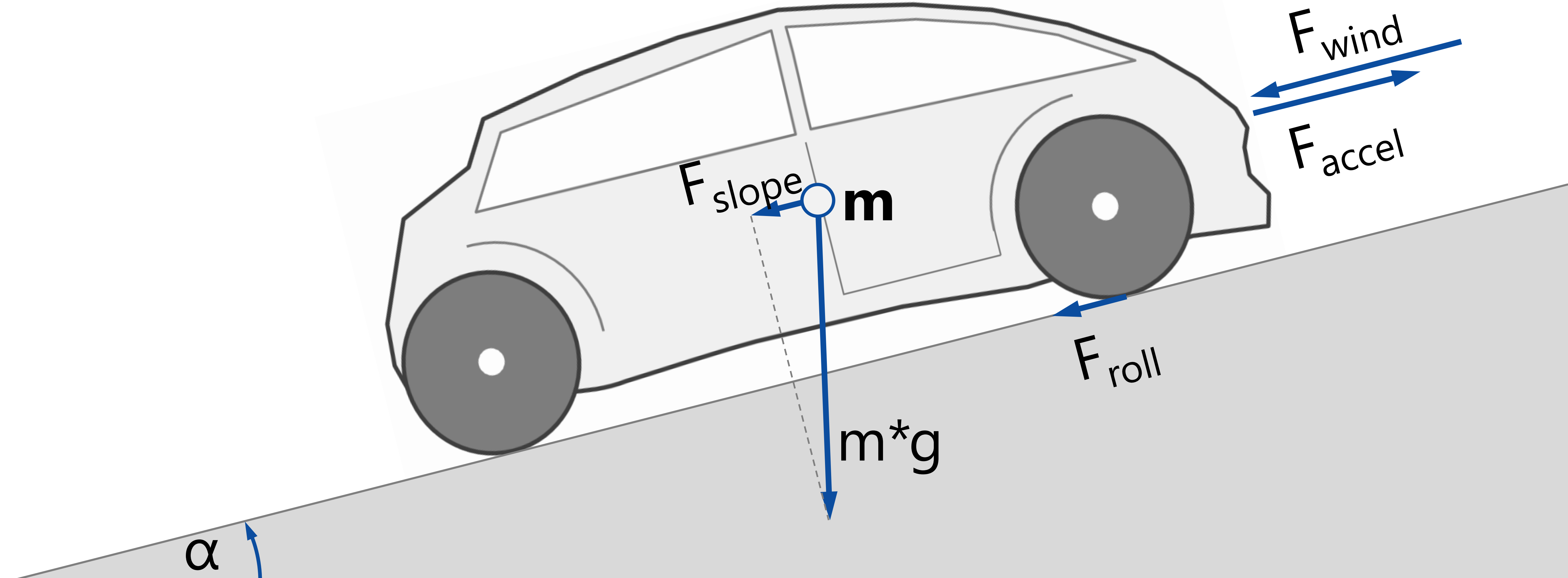
$$C_{energy} = f(\Delta e) = f(F_{total})$$

```python
vold_mps = vold/3.6
v_mps = v/3.6
time = 2 * section_length / (v_mps + vold_mps)
gradient_angle = math.tanh( (config.loc[x, "m above sea"]-config.loc[x-1, "m above sea"] ) / section_length)
f_roll_res = car_mass * g * math.cos(gradient_angle) * coeff_roll_res
f_slope = car_mass * g * math.sin(gradient_angle)
f_accel = ((v_mps-vold_mps) / time) * car_mass

f_total = f_accel + f_slope + f_roll_res
E_total = f_total * section_length
```



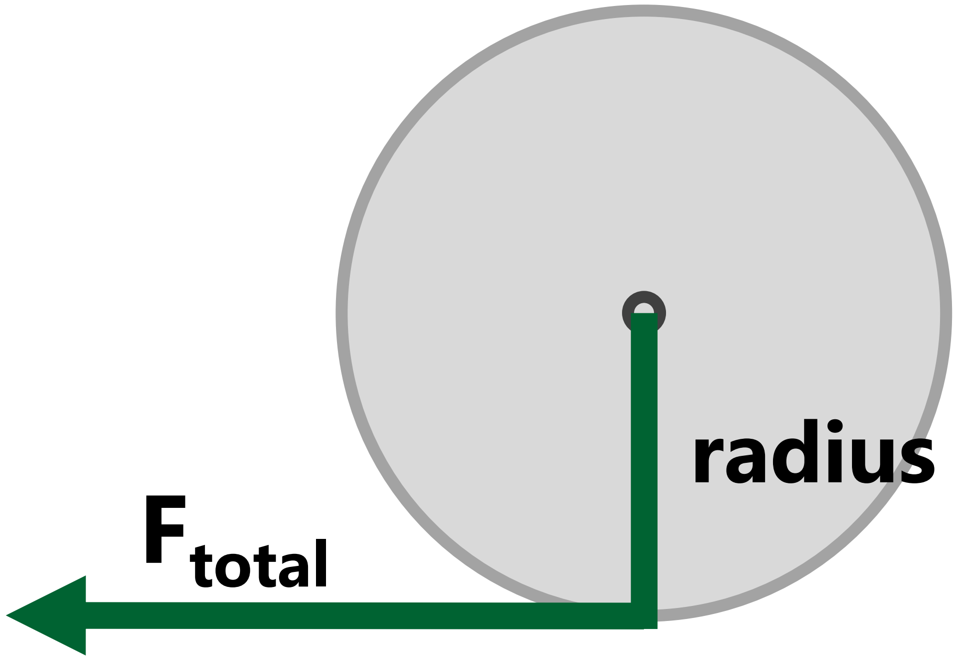
$$F_{total} = \underbrace{F_{roll} + F_{slope} + F_{accel}}_{\text{Included in the model}} + F_{wind}$$

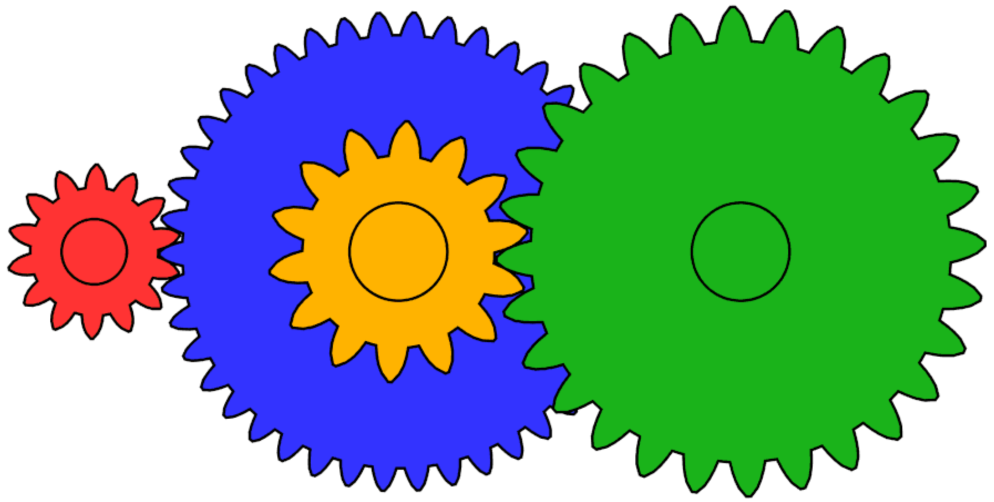


Power calculation for a series wound motor

Torque = F_{total} * radius / gear_ratio

Rotation speed = **v** / radius* gear_ratio

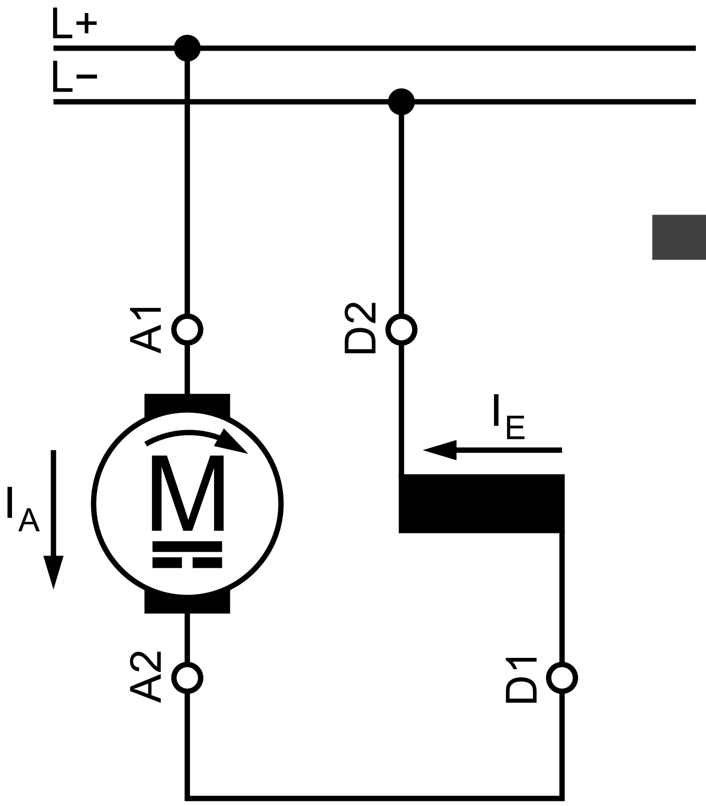




```python
motor_rotation_speed = ((v_mps+vold_mps)/2) * gear_ratio * 30 / ((diameter_tire/2) * math.pi)
motor_torque = (f_total * (diameter_tire/2) / gear_ratio)
motor_current = (motor_torque+torque_loss)*1000/torque_constant
motor_voltage = (motor_rotation_speed / velocity_constant) + (anchor_resistance * motor_current)
E_Motor = (motor_voltage*motor_current*time)
```

Power calculation for a series wound motor

Torque τ depends on the current I_A



$$\tau = I_A * K_T - \tau_{Loss}$$

$$\Rightarrow I_A = (\tau + \tau_{Loss}) / K_T$$

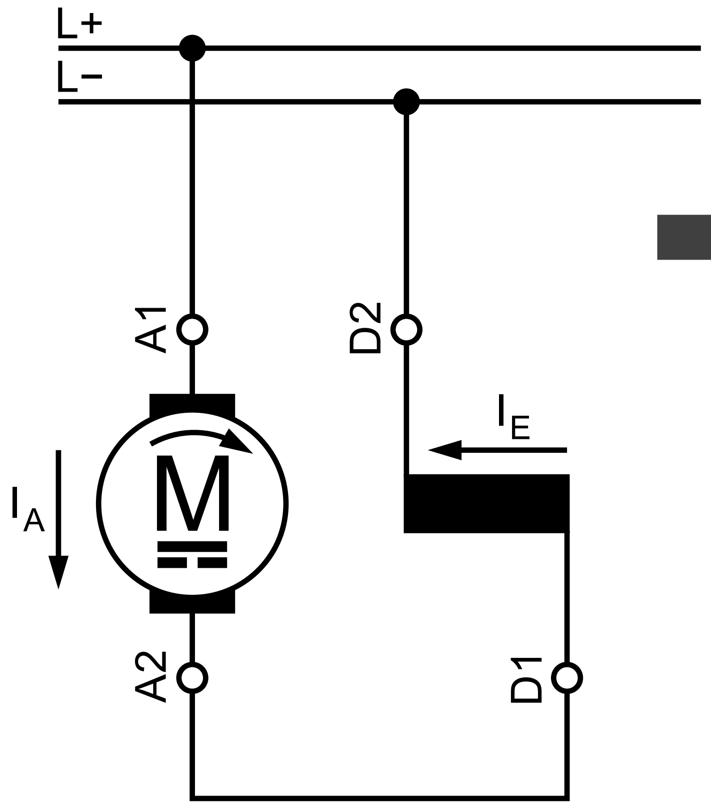
Motor specific parameters:

τ_{Loss} : Loss torque

K_T : Torque constant

Power calculation for a series wound motor

Rotation speed **n** depends on the voltage **U**



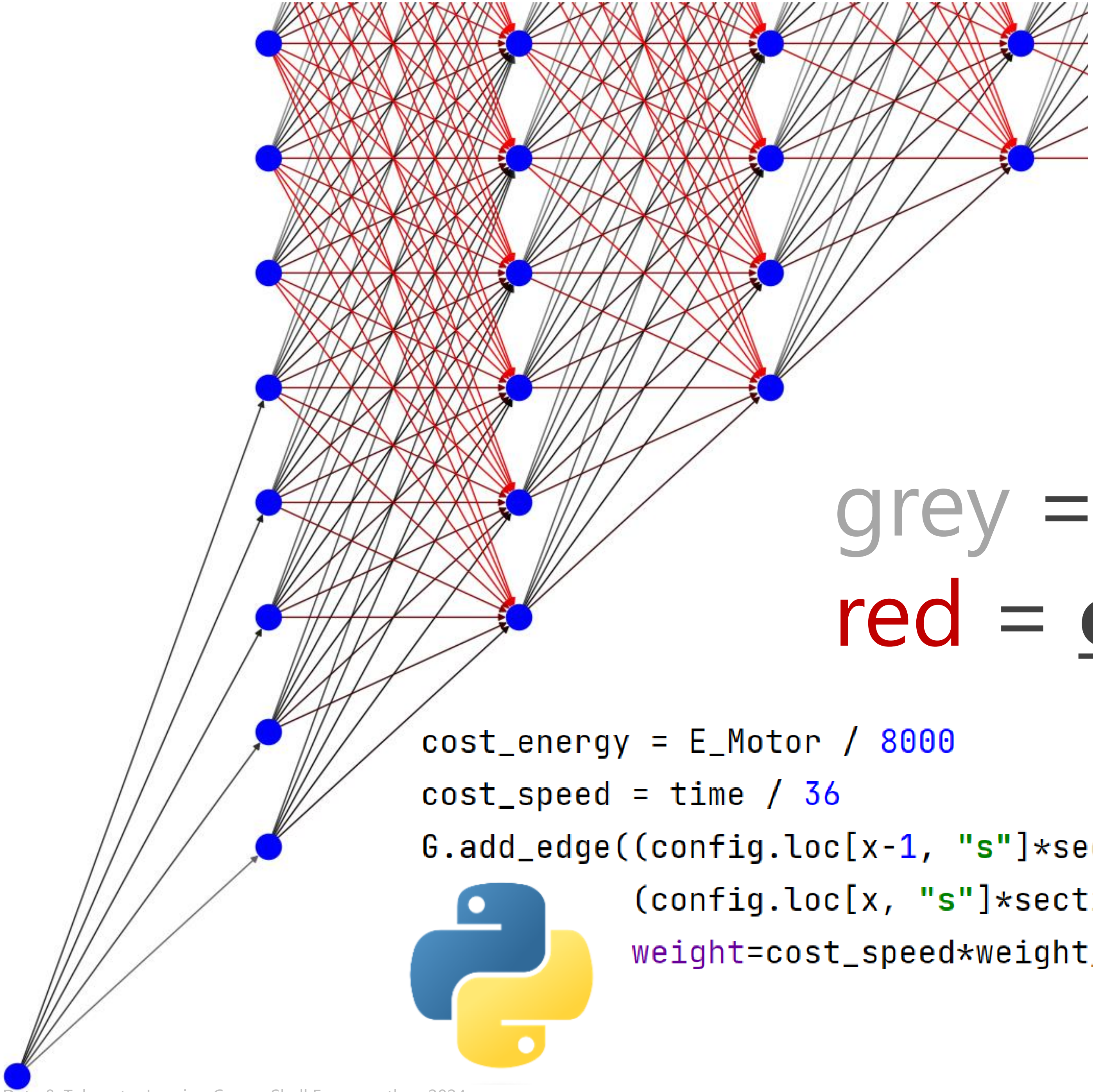
$$n = (U - R_A * I_A) * K_V$$

$$\Rightarrow U = n / K_V + R_A * I_A$$

Motor specific parameters:

R_A : Anchor resistance

K_V : Motor velocity constant

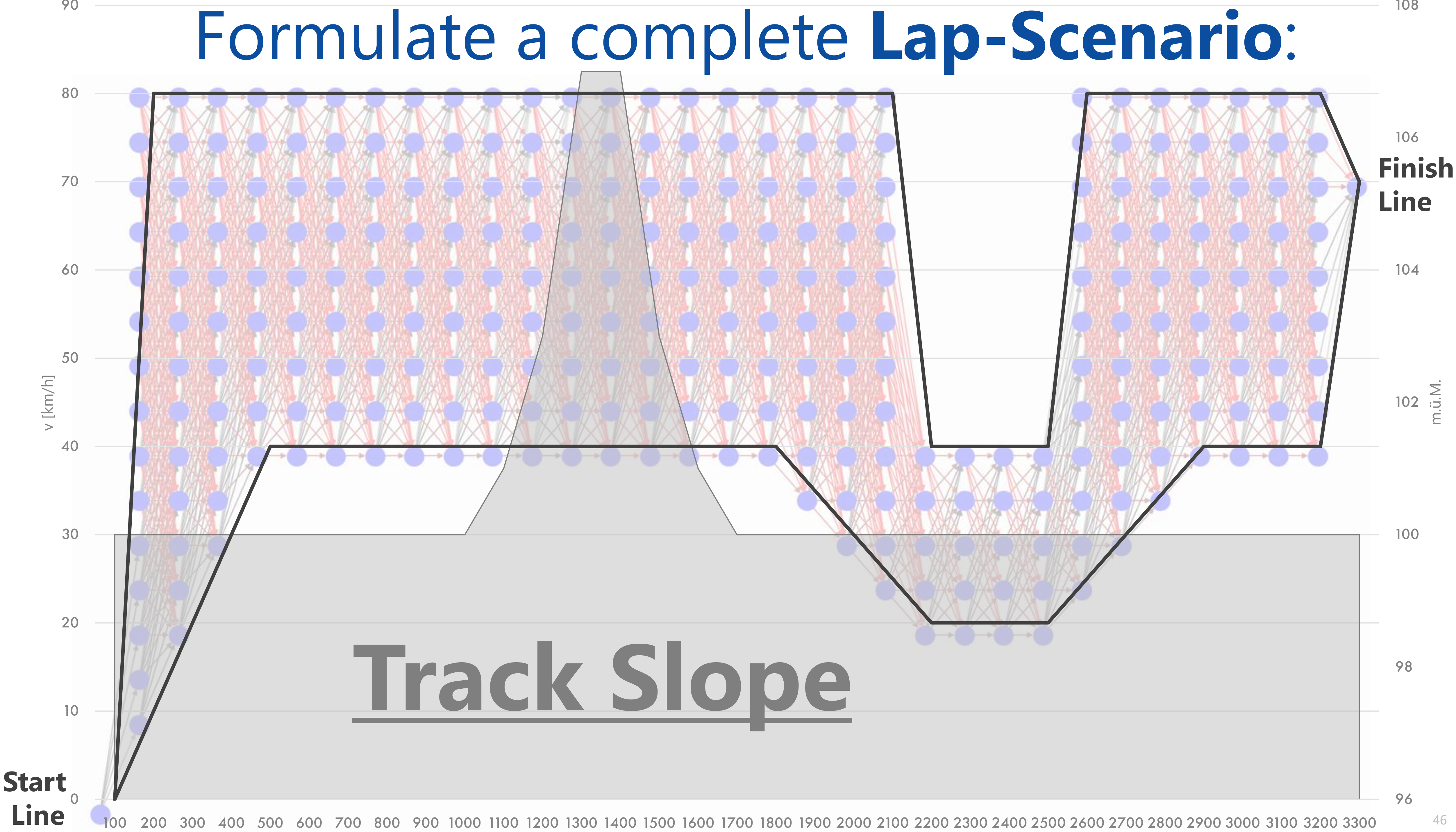


grey = **<u>expensive</u>** edges

red = **<u>cheap</u>** edges

```
cost_energy = E_Motor / 8000
cost_speed = time / 36
G.add_edge((config.loc[x-1, "s"]*section_length+vold),
           (config.loc[x, "s"]*section_length+v),
           weight=cost_speed*weight_speed+cost_energy*weight_effiency*cost_econ)
```

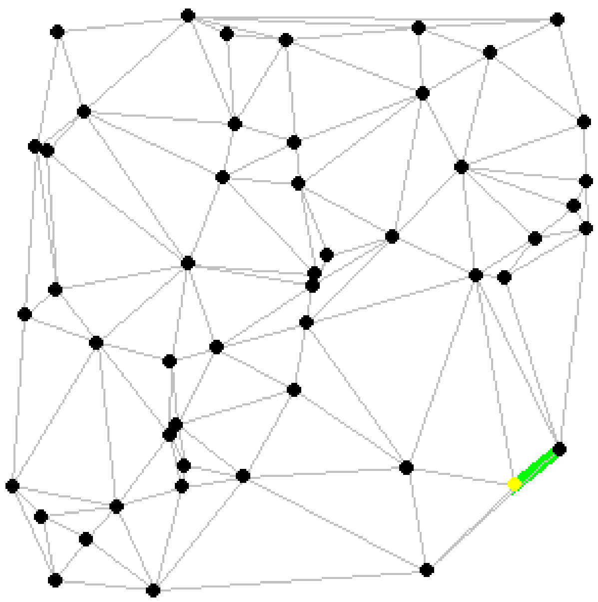
Formulate a complete **Lap-Scenario**:

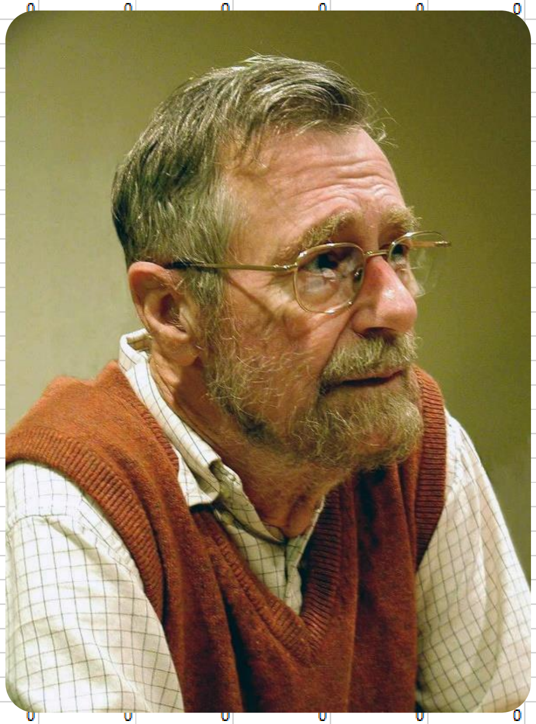


1. On-Track Use Cases
2. Tackling Opposing Goals
3. Connect the Dots!
4. Graphs with Nodes and Edges
5. Define a Race Graph
6. **Graph Transformers**

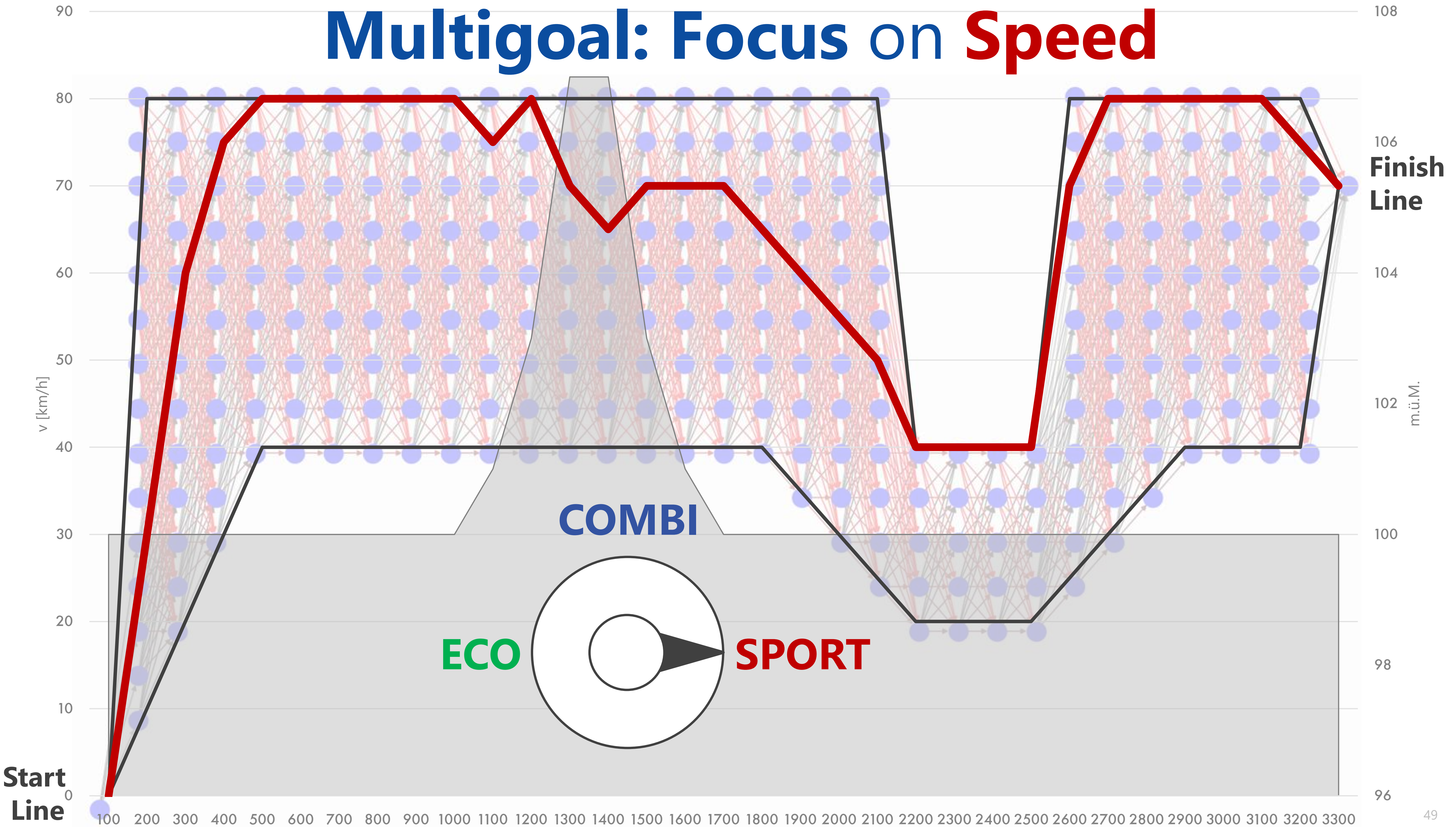
Executable by **Dijkstra** or **A*** Algorithm

```python
nodelist = nx.dijkstra_path(G, 0,config.loc[len(config.index)-1]["s"]*section_length+config.loc[len(config.index)-1]["vmin"]
nodelist = nx.astar_path(G, 0,config.loc[len(config.index)-1]["s"]*section_length+config.loc[len(config.index)-1]["vmin"])
```

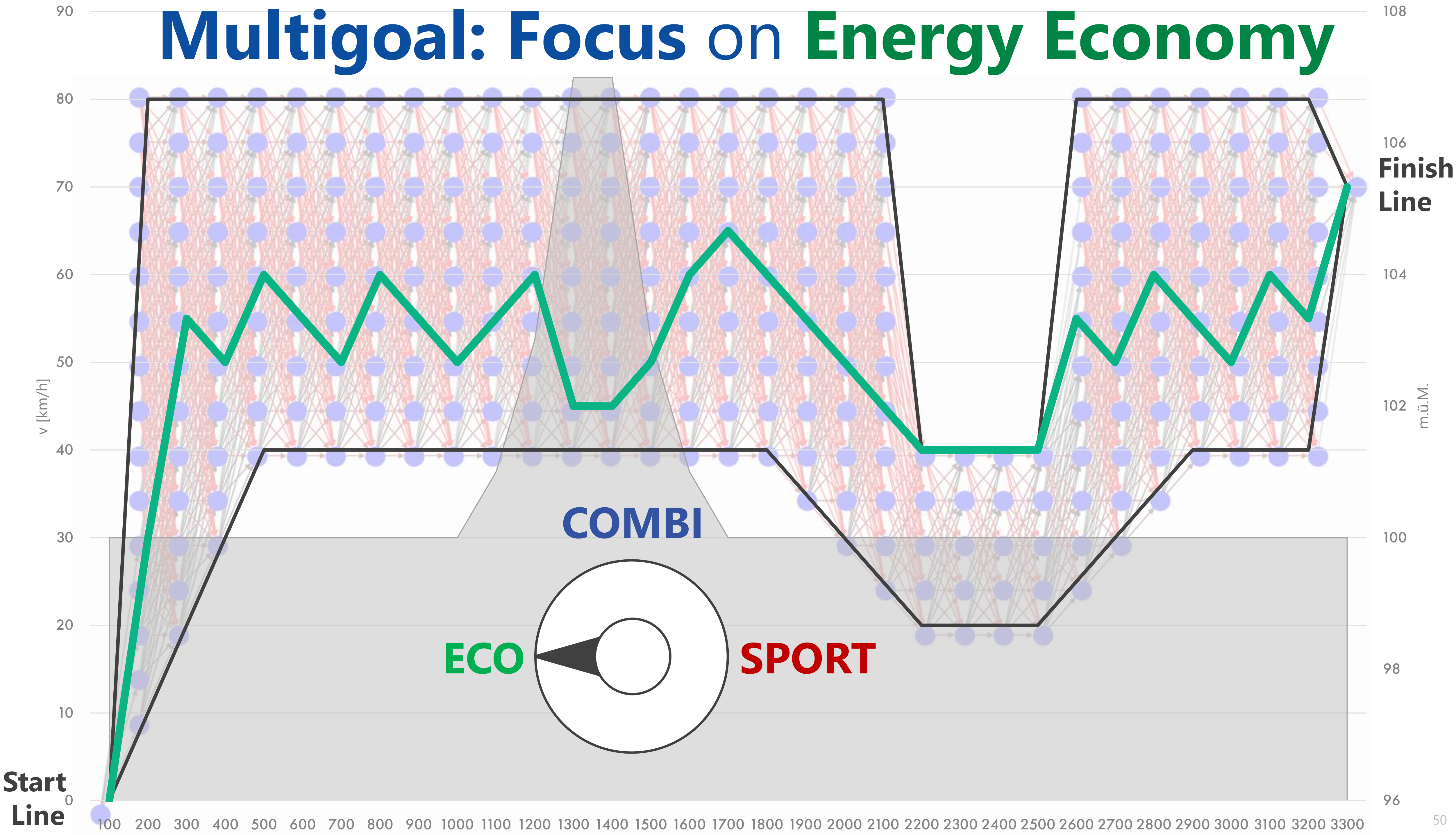




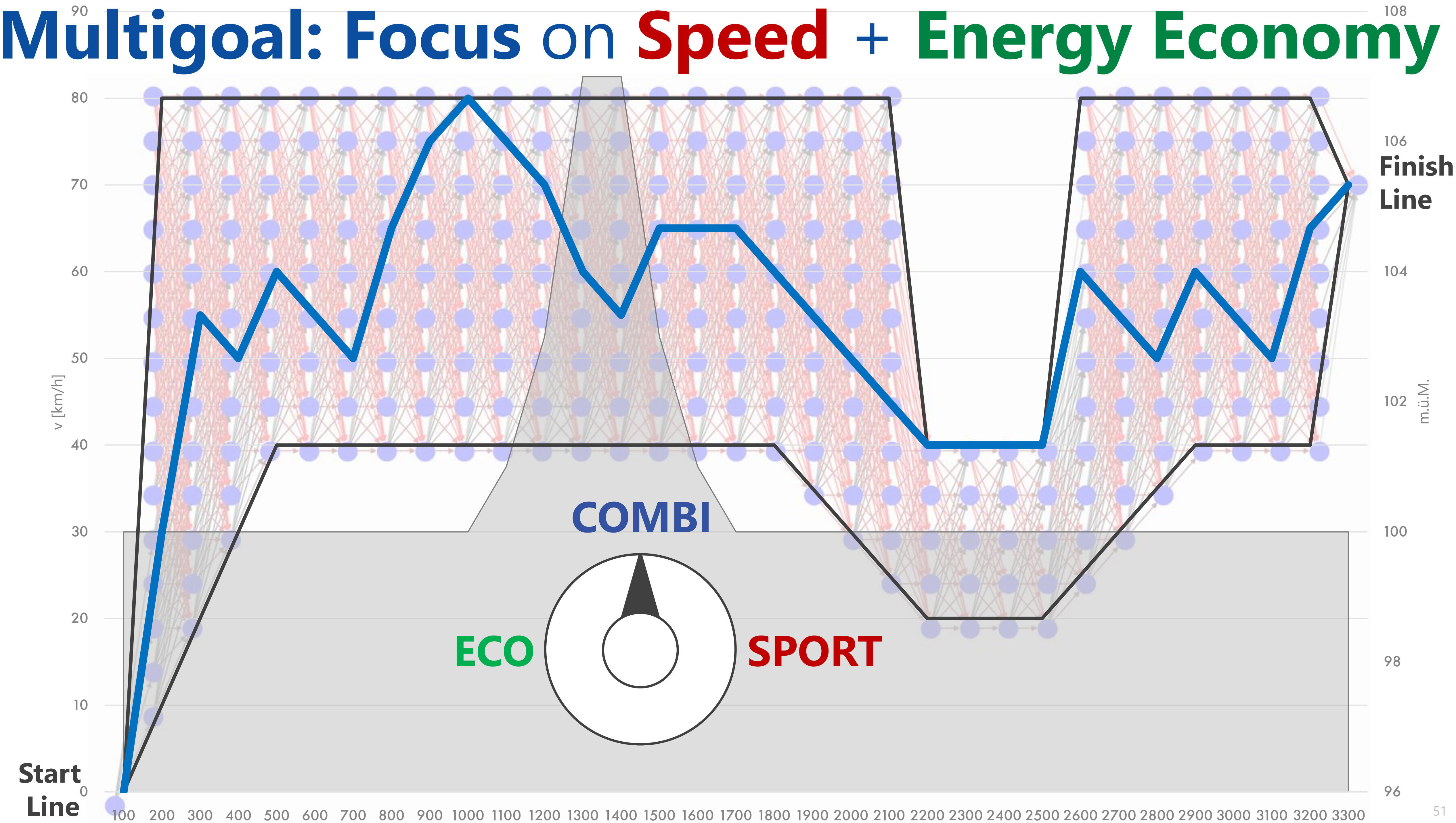
Multigoal: Focus on Speed



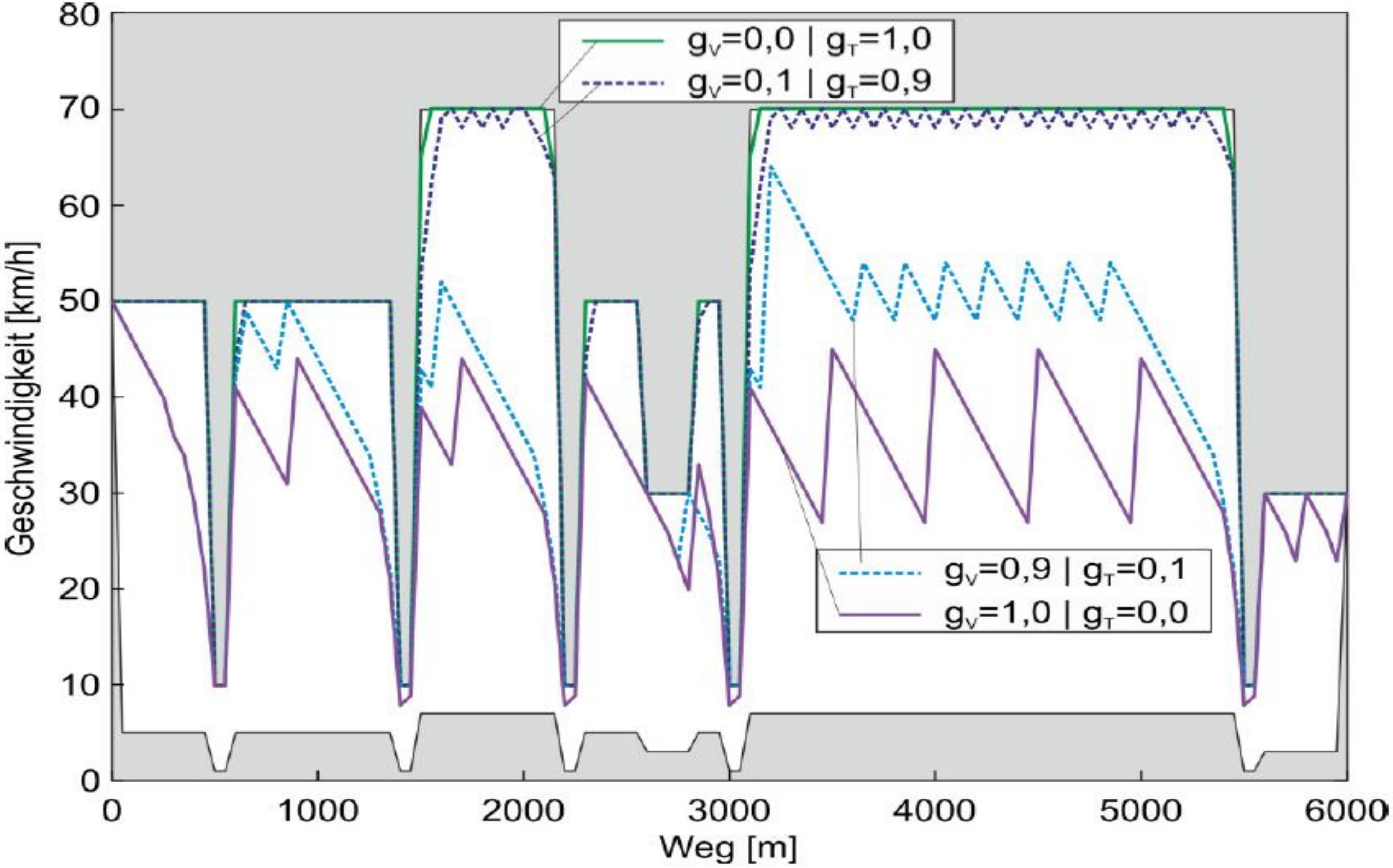
Multigoal: Focus on Energy Economy



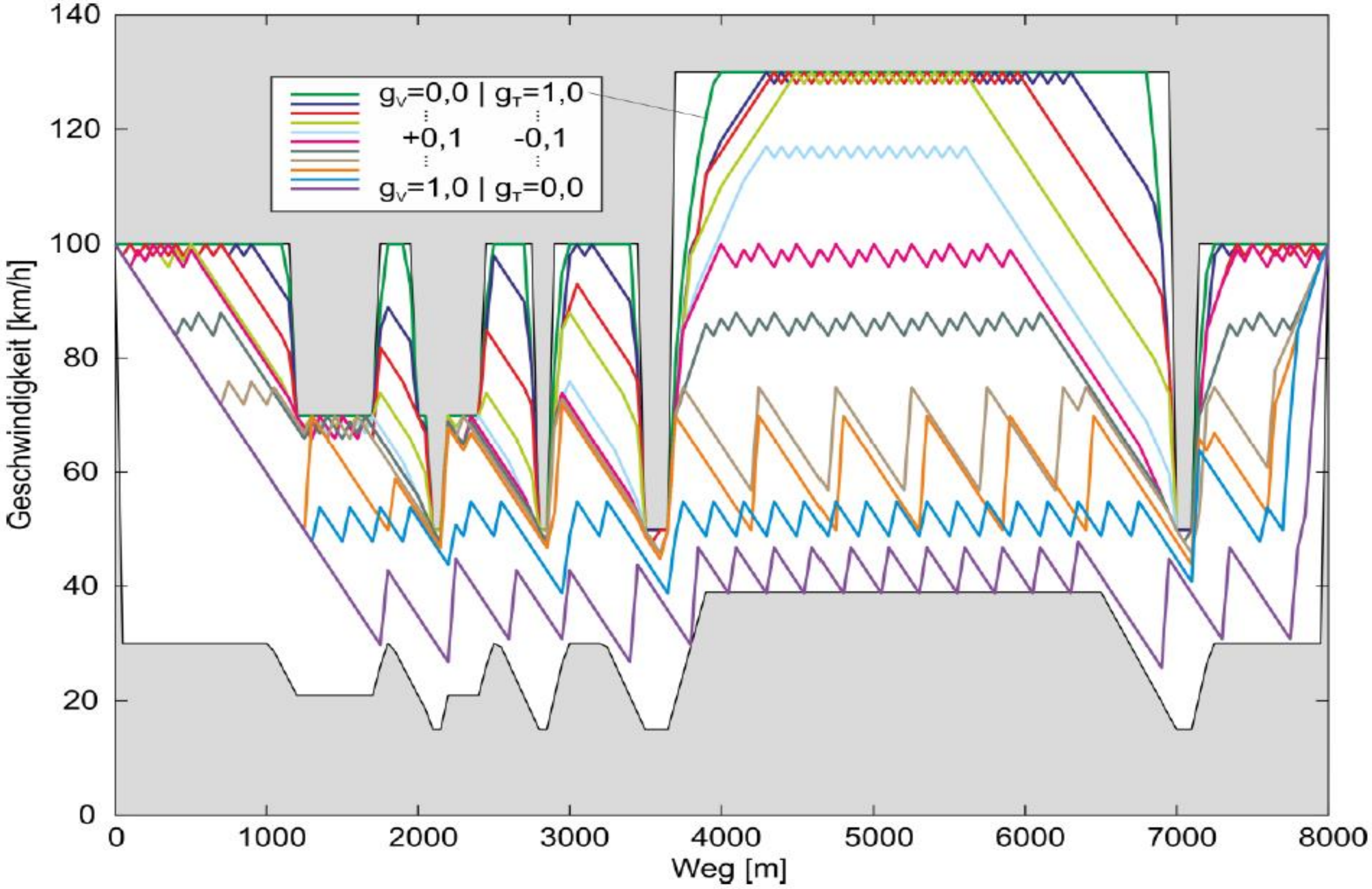
Multigoal: Focus on Speed + Energy Economy



A real Speed Profile

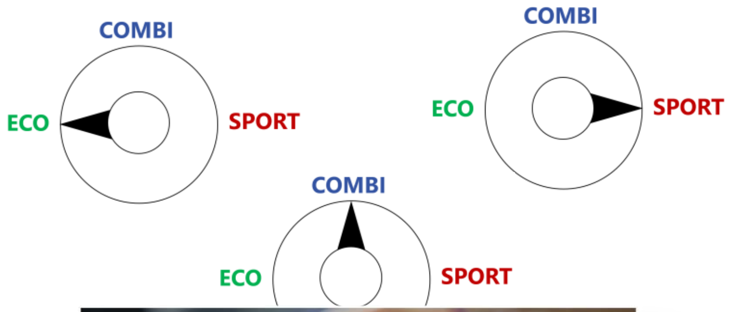


A real Speed Profile

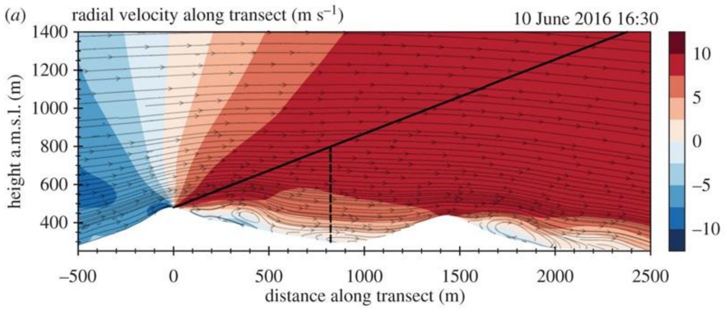


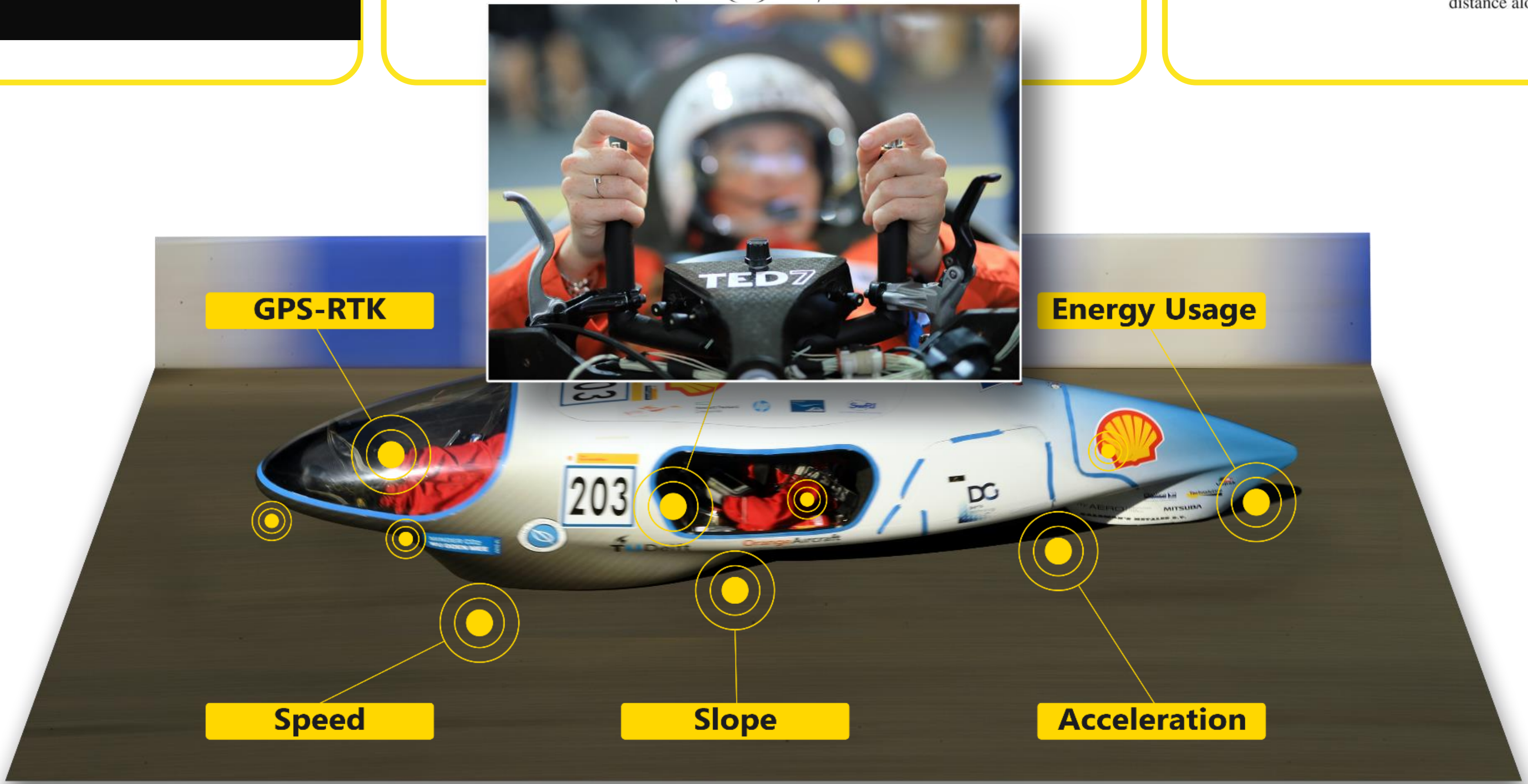
Include and adjust with **Telemetry Data**

Dynamically Changed **Driving Patterns**

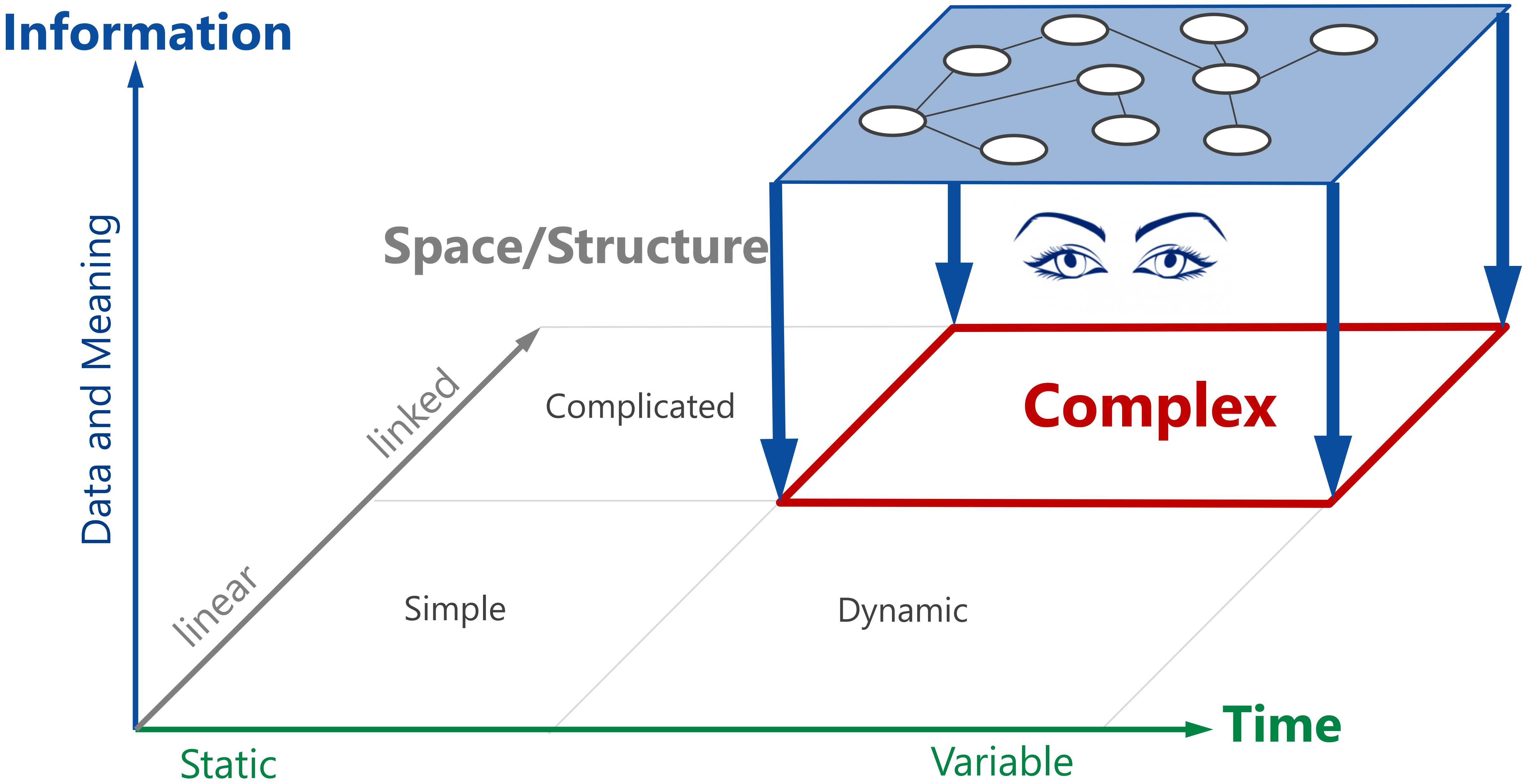


Include dynamic **Wheather Data**





Tackling Complex Problems with Information



1. On-Track Use Cases
2. Tackling Opposing Goals
3. Connect the Dots!
4. Graphs with Nodes and Edges
5. Define a Race Graph
6. Graph Transformers

Quiz

Go to

www.menti.com

Enter the code

5795 0551

Or use QR code

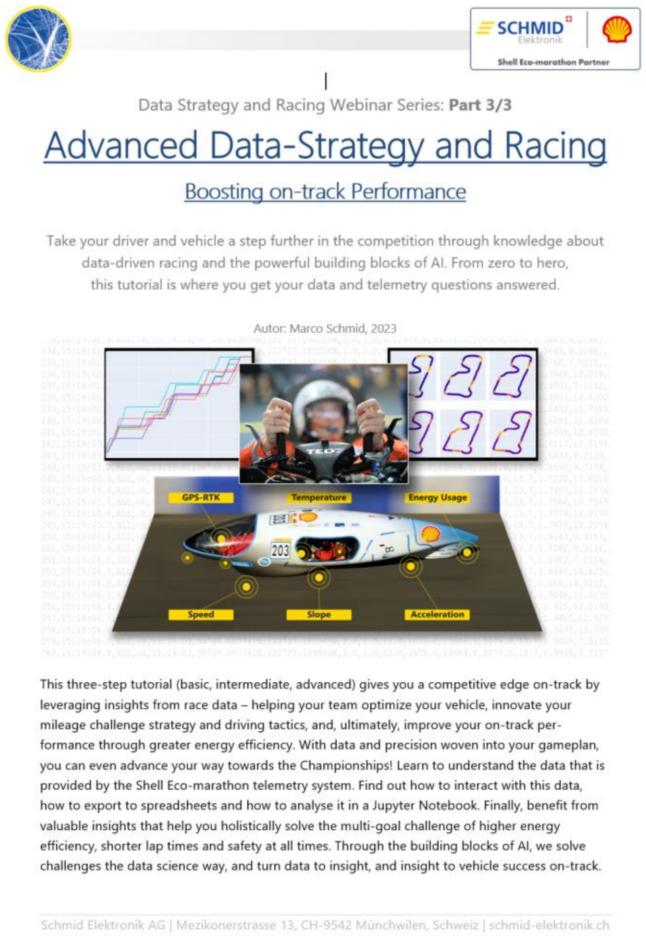
Developing Holistic Race Strategies based on Knowledge Graphs

Q&A

SCHMID Elektronik

Shell Eco-marathon Partner

Download Whitepaper and Handout



Data Strategy and Racing Webinar Series: **Part 3/3**

Advanced Data-Strategy and Racing

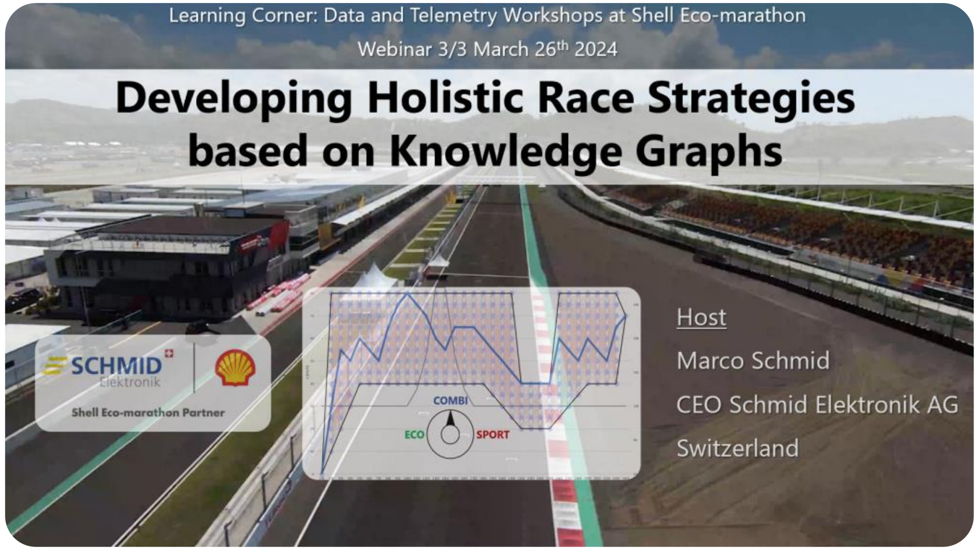
Boosting on-track Performance

Take your driver and vehicle a step further in the competition through knowledge about data-driven racing and the powerful building blocks of AI. From zero to hero, this tutorial is where you get your data and telemetry questions answered.

Autor: Marco Schmid, 2023

This three-step tutorial (basic, intermediate, advanced) gives you a competitive edge on-track by leveraging insights from race data – helping your team optimize your vehicle, innovate your mileage challenge strategy and driving tactics, and, ultimately, improve your on-track performance through greater energy efficiency. With data and precision woven into your gameplan, you can even advance your way towards the Championships! Learn to understand the data that is provided by the Shell Eco-marathon telemetry system. Find out how to interact with this data, how to export to spreadsheets and how to analyse it in a Jupyter Notebook. Finally, benefit from valuable insights that help you holistically solve the multi-goal challenge of higher energy efficiency, shorter lap times and safety at all times. Through the building blocks of AI, we solve challenges the data science way, and turn data to insight, and insight to vehicle success on-track.

Schmid Elektronik AG | Mezikonerstrasse 13, CH-9542 Münchwilen, Schweiz | schmid-elektronik.ch



Bootcamp with two Parts and Ontrack Activity

Part #1

45min: Urban Concept only

1. Recap of the most relevant race data in the Dashboard
2. Recap of the MVP Strategy
3. Plan your first attempts based on that Strategy

On Track

1. Develop several race strategies
2. Create driving patterns
3. Apply them on track, where data is created through telemetry

Part #2

45min: Urban Concept only

1. Do quantitative Analysis first
2. Then go into qualitative Analysis
3. Learn about the effects of driving patterns.
4. Iterate your Race Strategy from attempt to attempt

Thank You!

We're close the 3 webinars

Let's apply this knowledge on track

see you at the bootcamp!

About us

Schmid Elektronik

Schmid Elektronik is a family-run, Swiss technology SME for embedded systems and industrial electronics. Customers are supplied with feasibility studies, prototypes, pilot series, minimum viable products (MVP), batch size 1 as well as small and medium series. Market services include hardware and software engineering, products for LabVIEW on customer-specific hardware and production services (EMS). One of Schmid Elektronik's specialties is Internet-of-Things things.

Since 2007, Schmid Elektronik has been one of the NI Partners, specialized for LabVIEW on Embedded-Hardwae. Thus, the focus is on embedded sbRIO, customized cRIO modules, specific FlexRIO and SLSC boards.

Z-BRAIN®

The ZBrain product family from Schmid Elektronik consists of hardware platforms for embedded systems that are graphically programmable with NI LabVIEW (RT, FPGA). With this Schmid Elektronik completes the NI platform (sbRIO, cRIO, PXI, SLSC) and enables its customers to run LabVIEW on their own hardware. Be it in prototypes, MVPs or products.

Shell Eco-marathon Partner

Schmid Elektronik is an official partner of the Shell Eco-marathon and part of its Data & Technology Team. Schmid Elektronik has been providing development and production services as well as products and on-site services for the telemetry system since 2015.

Services include webinars, jury in the Data & Telemetry Off-Track-Award, Bootcamps and Technical Vehicle Inspection. The Swiss family-owned SME thus joined the ranks of global players, which are also partners in the program.

Contact

Schmid Elektronik AG
Marco Schmid
Mezikonerstrasse 13
CH-9542 Münchwilen
Switzerland

marco.schmid@schmid-elektronik.ch
Phone direct +41 (0) 71 969 35 90
Phone Info +41 (0) 71 969 35 80
info@schmid-elektronik.ch
schmid-elektronik.ch