April 18, 2025

# OuiRace – Algorithm Internship Test: GAP Time Correction Prototype

Written by Théo Laminie

# I - Literature Review

## 1. Strava GAP Method (Grade Adjusted Pace)

**Principle:**
Strava adjusts runners' pace based on elevation changes (uphill/downhill). The goal is to provide an "equivalent" pace to what it would have been on flat terrain.

**Model:**
Strava does not publish its exact algorithm, but analyses have shown that:

- A +1% incline slows the pace by about 12–15 seconds/km
- A -1% decline improves the pace by about 8–10 seconds/km

The correction factor is not linear:

- For small slopes ($|\theta| < 5\%$), a linear model can be used.
- For steeper slopes, a nonlinear model (quadratic or exponential) is necessary.

**Pros:** Simple method, easy to implement
**Cons:** Not transparent, not personalized (no consideration of the runner’s characteristics)

---

## 2. Daniels' VDOT Model (Running Formula)

**Principle:**
Daniels' model uses the VDOT concept to estimate the physiological effort behind a given performance, compare performances over different distances, and predict future performance on other terrains or distances (e.g., altitude, elevation changes).

**VDOT:** A theoretical functional $VO_2max$ calculated from race time and distance rather than in a lab.

**Model:**

- $VO_2$ estimation based on running speed :

  $$VO_2 = -4.60 + 0.182258 * v + 0.000104 * v^2$$

v = speed ( m/min )

$VO_2$ = estimated oxygen consumption ( ml/kg/min )

This equation estimates **the energy cost of running at a certain pace**, without yet considering the runner's abilities.

- Link between $VO_2$ of a given effort and $VO_2max$ (physiological efficiency model) :

$$\frac{VO_2}{VO_2max} = 0.8 + 0.1894393 * e^{-0.012778 * t} + 0.2989558 * e^{-0.1932605 * t}$$

t = Duration of the effort in minutes

This models the **percentage of $VO_2$max that a runner can sustain over a given duration.**

These two equations together allow for the estimation of **$VO_2$max** based on time over a known distance.

**Application to performance correction :**

1. Measure a real performance (e.g., 10k in 42 min → compute $VO_2$)
2. Use the $VO_2$ to calculate the VDOT
3. Predict another performance maintaining the same VDOT level under different conditions

We reverse the calculations: for a different duration (or another difficulty, such as an uphill), we compute the speed required to maintain the **same VDOT** level over a given time. This gives us a flat-equivalent speed corresponding to the effort over that specific duration.

The VDOT model **does not explicitly account for slope**, but it reflects **its effects through the effort duration**. An uphill slows you down → your time increases → the speed calculated via VDOT decreases → which reflects a **higher physiological cost**.

**Pros:** Robust physiological model, easy to implement, useful for comparing global performances
**Cons:** Not local (not adapted to segments), doesn't explicitly consider slope or weather, not personalized

---

## 3. Slope-Based Energy Model – Minetti (2002)

**Principle:**
Minetti measured the **energy cost of running on different slopes** using a treadmill. This enables estimation of energy expenditure based on slope percentage.

**Objective:**
Model the **energy cost per meter** (and per kg of body mass) as a function of **terrain slope**.

**Results:**
Experimental values (selected examples):

| Negative slope (%) | Energy cost Cr (J/kg/m) |
|---|---|
| -45 | 3.92 ± 0.81 |
| -40 | 3.49 ± 0.47 |
| -35 | 2.81 ± 0.54 |
| -30 | 2.43 ± 0.50 |
| -20 | 1.73 ± 0.36 |
| -10 | 1.93 ± 0.45 |

| Positive slope (%) | Energy cost Cr (J/kg/m) |
|---|---|
| 0 | 3.40 ± 0.24 |
| +10 | 5.77 ± 0.60 |
| +20 | 8.92 ± 0.84 |
| +30 | 12.52 ± 0.62 |
| +35 | 14.43 ± 1.08 |
| +40 | 16.83 ± 0.88 |
| +45 | 18.93 ± 1.74 |

**Equation (polynomial regression):**

From this table, a polynomial regression can be performed to obtain an equation that closely matches the results, linking the energy cost to the slope :

$$Cr(\theta) = -4.724 * \theta^4 - 5.676 * \theta^3 + 0.004935 * \theta^2 + 0.1764 * \theta + 3.389$$

$\theta$ = slope ( % )

**Application:**

1. Compute **local slope** for each point
2. Apply the model to estimate **instantaneous energy cost**
3. Integrate over the entire course → **total energy expenditure**
4. Deduce **equivalent flat pace**

**Corrected speed calculation** based on total energy expenditure**:**

- Theoretical energy on flat terrain

On flat terrain, the energy cost is constant, and according to Minetti's table, the energy cost on flat ground is around **3.40 J/kg/m**.

So, for a distance D in meters: $E_{flat} = C_{flat} * D$

- Energy ratio → speed correction

We assume that **time is proportional to energy expenditure**, under the assumption of constant effort (which is reasonable for estimation purposes):

$$v_{Corrected} = v_{actual} * \frac{E_{actual}}{E_{flat}}$$

**Pros:** Based on precise experimental data, locally applicable, handles slope effects well
**Cons:** Doesn't consider weather or individual specifics, requires good segmentation

---

## 4. Machine Learning Model

**Principle:** Train a model to predict adjusted pace using multiple data sources:

- Terrain data (slope, distance, elevation gain/loss)
- Environmental data (temperature, wind, humidity)
- Personal data (age, weight, $VO_2$max)

**Possible Algorithms:**

| Model | Type | Use Case |
|---|---|---|
| Linear Regression | Simple baseline | First step, low complexity |
| Random Forest/XGBoost | Decision trees | Good for tabular data |
| Neural Networks | Deep/non-deep | If large data & complex relations |
| RNN, TCN | Time series | GPX point-by-point analysis |

**How it works:**

- **Training:** Model learns from real race data, to estimate an equivalent performance *on flat terrain*, a pace correction or pace prediction.

- **Prediction:** Once trained, a new file (GPX + weather) is provided. The model returns an estimate:
  - "If this course had been flat and neutral, you would have run at XX min/km."

**Pros:** Highly adaptable, multi-factor, can improve with more data
**Cons:** Complex to train, data-intensive

---

## 5. Running Power Model

**Principle:**
Estimates mechanical power output by a runner, considering:

- Horizontal speed
- Wind resistance
- Elevation changes
- Runner's mass

Two approaches:

- Sensor-based power estimation (e.g., running watch)
- Physics-based estimation

**Model (simplified physics):**

$$P = P_{potential} + P_{Kinetic} + P_{resistance} + P_{stride}$$

- **Potential power (slope):** Based on mass, slope, and speed

  $$P_{potential} = m * g * sin(\theta) * v$$

  Data: mass *m*, slope *θ*, speed *v*, Normal acceleration due to Earth's gravity: g = **9.81 m/s²**

- **Kinetic power:** Based on speed variations

  $$P_{kinetic} = m * \frac{dv}{dt} * v$$

  Data: mass *m*, speed *v*

- **Air resistance:** Wind and air resistance

$$P_{resistance} = \frac{1}{2} * C_d * A * \rho * (v_{rel})^3$$

data : $C_d$ : Drag coefficient (≃ 1.0 for an upright runner)

A : Frontal surface area of the runner ( ≃ 0.5 | 0.6 $m^2$ )

ρ : Air density ( ≃ 1.2 kg/$m^3$ à 20°C)

$v_{rel}$ = v + $v_{vent}$ ( Headwind ) or $v_{rel}$ = v + $v_{vent}$ ( Tailwind )

This term becomes **very important at high speeds** (>12–14 km/h) or in strong winds.

- **Stride power (oscillations):** Vertical oscillations, muscular inefficiencies :

Very difficult to estimate without sensors. However, several studies approximate it as a **constant component** :

$$P_{stride} \simeq m * C_{int} * v$$

data : $C_{int}$ ≃ 1.0 J/kg/m, Representing the non-gravitational energy cost related to muscular work, rebound, stability, etc...

**After estimating power:**
Compute equivalent flat speed based on the average power.

$$v_{flat} = \frac{P_{mean}}{C_{flat} * m}$$

data : $P_{mean}$ : average power of running ( W )

$C_{flat}$ : energy cost ( ≃ 4.0 | 4.2 J/kg/m )

m : runner's mass ( kg )

**Pros:** Directly measures effort, point-by-point analysis possible
**Cons:** Complex, sensitive to data quality, requires assumptions

# II - Methodological Choice and Justification

Different methods that could lead to evaluating the adjusted speed according to elevation have been detailed previously :

- Strava GAP
- Daniels VDOT
- Minetti energy model
- Machine Learning
- Running Power

The first **GAP method from Strava** is not complete because Strava does not publish its exact model. However, it is possible to derive a simple method for calculating an adjusted pace. But this approach is imprecise and does not account for the capabilities of the evaluated runner.

The second method by **Daniels**, based on the **VDOT** evaluation, is interesting because it derives its assessment from physiological effort. The calculation of this data can be done using a maximal effort, such as a reference time for 10 km, and allows the adjusted pace to be evaluated on other performances, predicting the speed for a certain time on flat terrain or compared to different elevation changes. It could also be possible to use knowledge of the $VO_2max$ calculated by the runner's device, such as a watch, to replace the reference time. However, this method uses only times and physiological data to calculate the adjusted pace and does not explicitly take into account slope or weather data.

The third method, based on the **Minetti energy model**, uses a study directly on the energy expenditure of runners on different slopes, which gives it some reliability. The energy costs found in this study for a given slope correspond to the average measured experimentally across several runners, so the method is not personalized for the evaluated runner but rather applies to an average runner. To implement this method, it is necessary to properly segment the course to obtain the slope for the entire route.

The fourth method uses **machine learning**, offering many possibilities for implementation and improvement. Moreover, it could provide good results by taking into account all the necessary data for a reliable and realistic evaluation. However, to achieve this, a large and complete dataset is required.

The last method is based on a **power model**. This could be a good solution, as it takes into account numerous parameters to perform the evaluation. However, calculating it through data and physical formulas requires simplifications and physical assumptions combined with the use of a lot of data that must be precise, making its implementation complex. Another solution is to use power data pre-calculated, for example, by the watch worn by the runners, which simplifies the implementation and allows for an easy and potentially accurate calculation of the adjusted pace based on the power exerted by the runner.

**Method Choice:**

The Minetti method seems to be the most relevant to implement, as it is based on a study directly conducted on runners, showing a certain reliability in its use. Furthermore, the other methods either require a large amount of precise data for their implementation (AI and power) or are incomplete (GAP). The Daniels method could also have been a solution; it is easy to implement, but by basing its operation solely on speed and race time between a reference time and the race to be adjusted, it seems to not take enough parameters into account to provide a precise result.

---

# III - Results

After segmentation and application of the Minetti model through interpolation of energy cost data relative to the slope on .fit or .gpx files, the following results were obtained for the given race with the subject, as well as for two of my personal races:

- The provided 10k race
- Two personal runs: one summer trail, one training session

**10k Race:**

- Adjusted Average Pace (Minetti): 5.44 min/km
- Raw Average Pace: 5.26 min/km

**Training Run:**

- Adjusted Average Pace (Minetti): 5.12 min/km
- Raw Average Pace: 5.21 min/km

**Trail des 7 Laux:**

- Adjusted Average Pace (Minetti): 5.06 min/km
- Raw Average Pace: 6.54 min/km

For the two .fit files extracted from my races, I was able to compare the results obtained with the algorithm based on Minetti and what is shown by my application (Coros) :

- Training: Minetti 5’11 – Coros 5’04
- Trail des 7 Laux: Minetti 5’05 – Coros 5’03

These results show a good match between the Minetti-based algorithm and commercial tools like Coros, validating the model’s relevance.

---

# IV - Improvements and Future Work

The current algorithm, based on Minetti's energy model, adjusts running speed according to the slope, providing an estimate of the energy cost of the effort. While this model already delivers relevant results, several areas for improvement can be explored to increase the accuracy of the analysis and its relevance in real-world conditions.

**Improvement of slope calculation**
The current slope calculation is done on fixed 10-meter segments. However, this method can be sensitive to noise inherent in altitude data, especially in rough terrain. One improvement would be to smooth the altitude data using digital filters and calculate the slope over a longer cumulative distance. This would provide a slope more representative of the actual effort exerted by the runner.

**Personalization of the model**
The Minetti model is based on average values that do not take into account the individual characteristics of the runner (body mass, training level, biomechanical efficiency). A future evolution could involve incorporating this data to adjust the model to the specifics of the user, particularly through power sensors, heart rate monitors, or data from physiological analyses.

**Experimental validation**
Finally, it would be valuable to validate the algorithm's results using experimental data measured on the field, such as comparing energy expenditure estimates with power sensor measurements or matching speed adjustments to heart rate.
It would also be relevant to compare the results obtained with this algorithm to other calculation methods and across a large number of races to assess the quality of the results.

---

# Conclusion

This work allowed me to explore in depth various existing approaches to adjust running speed based on elevation gain, with the goal of providing a more accurate estimate of the actual effort exerted by a runner. After a comparative analysis of available models, Minetti’s energy model emerged as the most relevant to implement in a first prototype, due to its solid scientific foundation and its ability to directly integrate the impact of slope on energy cost.

The results obtained with this model show good consistency with those of a commercial tool (Coros), which partially validates the quality of the approach. However, several areas for improvement have been identified. It would be valuable to refine the slope calculation, incorporate personalized parameters related to the runner's characteristics, and validate the estimates with experimental data collected in the field.