Modeling attractive and repulsive forces in semantic properties

D. Gueorguiev 2/8/2022

Table of Contents

Modeling attractive and repulsive forces in semantic properties ***1***

Notation **1**

The Concept of Semantic Aspect and modeling interaction between Semantic Aspects **2**

Modeling interaction between semantic properties **7**

On the Semantic Mass of a Property **7**

On the Semantic Energy of a Semantic Property **8**

Bound state of an ensemble of semantic properties **10**

Energy-weighted centroid of an ensemble of semantic properties 11

Determining the ending position of the travel path for each property 15

Forming an Ensemble of semantic properties 20

Appendix ***20***

Constructing the property tree: constraints and inequalities based on binding force **20**

Construction of semantic properties **22**

Notation

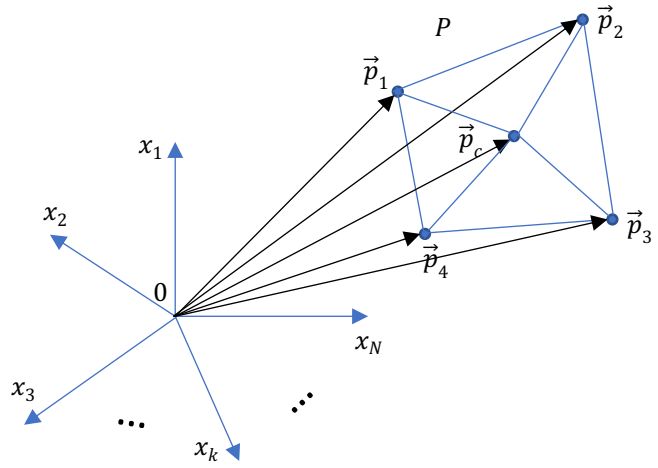
L : the number of semantic dimensions
K : the number of semantic dimensions in a property represented as K-polytope
N : number of semantic aspects in a property
$\mathcal{P}$: set of points forming the K-polytope of a semantic property
A_i : denotes the i-th semantic aspect of a semantic property
P_i : denotes semantic property
V_i : denotes primitive semantic particle
$\vec{p}_c$: in the context of a property: the center of mass of the property
In the context of an ensemble of properties: the center of mass of the ensemble
$\vec{p}_i$: In the context of a property: semantic position of the aspect A_i
In the context of an ensemble of properties: the center of mass of the property
l_i : the type of the aspect A_i
θ_j : angle between the current aspect and semantic axis x_j
$\boldsymbol{\theta}$: a vector with all angular coordinates of the current aspect to the semantic axes
Φ : Aspect Type Matching function
Θ : Aspect Value Matching function
Ψ : Semantic Energy Density function
Λ : Energy Dissipation Density function
$f_{12}(A_1, A_2)$: attractive/repulsive force between the aspects A_1, A_2 of two different properties

$f(P_1, P_2)$: attractive/repulsive force between the properties P_1, P_2
M_P : semantic mass of the property P
IC_P : information content of the property P
VL_P : semantic valence of the property P
$E(P)$: semantic energy of a property P
$E(\vec{p}_i + \Delta\vec{p}, l_i)$: semantic energy of an aspect A_i
$\tilde{E}(s + \Delta s)$: harmonic semantic energy of a property at point $s + \Delta s$
$\tilde{E}_t(s + \Delta s)$: harmonic net semantic energy of a property at point $s + \Delta s$
$x_i(s_i^0 + \Delta s)$: distance from property P_i to the energy-weighted center of mass of an ensemble of properties at the position $s_i^0 + \Delta s$ of P_i's travel path
$\vec{p}_E^{(0)}$: the energy weighted center of mass for an ensemble of properties on the 0-th iteration. Equal to the center of mass of the ensemble $\vec{p}_c$
$\vec{p}_E^{(k)}$: the energy weighted center of mass for an ensemble of properties on the k-th iteration

The Concept of Semantic Aspect and modeling interaction between Semantic Aspects

The internal structure of a semantic property P is represented by a set $\mathcal{P}$ of points $\vec{p}_i, i = 1..|\mathcal{P}|$ in semantic space forming a K-polytope which occupies a subset K of the L semantic dimensions i.e. $K \leq L$. On the picture below it is depicted an L-dimensional 4-polytope. With $\vec{p}_i, i = 1..4$ are denoted the vertices of the polytope. With $\vec{p}_c$ we denote the center of mass of the polytope a.k.a *centroid* given with:

$$\vec{p}_c = \frac{\sum_{i=1}^{|\mathcal{P}|} \vec{p}_i}{|\mathcal{P}|}$$



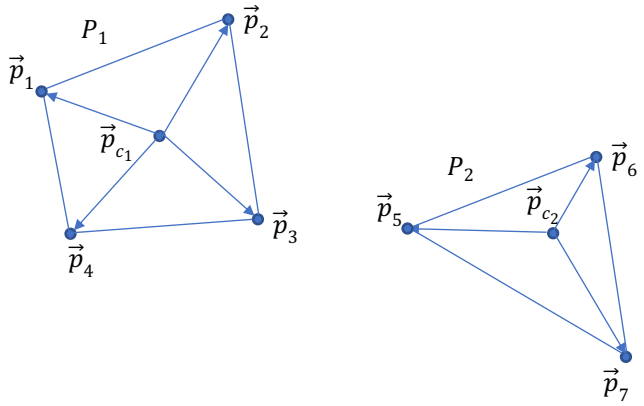
From now on we will denote the polytope associated with property P as P-polytope or *property polytope*. Each vertex of the property polytope models specific semantic aspect of the property. The distance of each vertex $\vec{p}_i$ from the centroid $\vec{p}_c$ for the property identifies the type of the semantic aspect this vertex accounts for. The specific position of that vertex $\vec{p}_i$ in semantic space relative to the centroid $\vec{p}_c$ encodes the current value of the semantic aspect in that property.

In other words, the $K-1$-tuple $(\theta_1, \theta_2, \ldots, \theta_{K-1})$, where each $\theta_j, j = 1..K-1$ denotes the angle between the aspect tip $\vec{p}_i$ and the semantic axis x_j, uniquely identifies a specific value for the semantic aspect $\vec{p}_i$. For instance, if the property describes the gender of an animal or a human, one particular semantic aspect of this property is if it refers to a subject (a human) or a verb (an action). In both cases the distance of the vertex from the center of mass will be the same but the vertex orientation would be different.

Now imagine the following scenario – we have a primitive semantic particle V_1 which denotes the personal pronoun "she". We have a second primitive semantic particle V_2 which denotes the verb "to be" in third person, singular "is". Clearly, the expectation is to be able to combine the two particles as:

$$\begin{matrix} V_2 \\ / \\ V_1 \end{matrix}$$

The question is how to design the properties those two particles are made of so they will "choose" each other. One way to achieve this is to model some sort of attractive force between the two particles. Each property has a set of semantic aspects which we differentiate by type/kind and by current value. We are going to define an attractive/repulsive force between semantic aspects of the same type.
Let us consider the following example: we have two semantic properties P_1 and P_2 where the first one has 4 vertices while the second one has 3 as shown on the Figure below.



Let us assume that vertex $\vec{p}_3$ and vertex $\vec{p}_5$ describe semantic aspects of the same type/kind. This fact is modeled by their relative distances to the corresponding centroid where the following relationship holds:

$$\left|\vec{p}_3 - \vec{p}_{c_1}\right| \sim \left|\vec{p}_5 - \vec{p}_{c_2}\right|$$

Here we will make an important relaxation the need of which will become clear later in the discussion.

We will allow the type of the semantic aspects to be continuous. That is: the types of the semantic aspects are represented by an uncountable set which is modeled with some segment of the real axis $[R_{min}, R_{max}]$. Here the value of R_{min} corresponds to the minimum semantic aspect type and R_{max} corresponds to the maximum semantic aspect type.

Definition: *semantic function* – a function which accepts a set of parameters each of which describes one or more of the following:

a) a specific semantic aspect, property, particle or structure
b) the relative or absolute position of its constituents in semantic space,
c) how the above-mentioned constituents exert influence relative to each other.

Definition: *Local semantic function* – a semantic function which does not have explicit dependence on the absolute position in semantic space of its constituents.

Definition: *Aspect Type Matching function* Φ
We would like semantic aspects of the same type to attract each other or repel each depending on their values. But how to discriminate between different types of semantic aspects? One way to do that is by introduction of a *local semantic* function Φ:

$$\Phi = \Phi(l_1, l_2) \qquad (1)$$

which we will name *aspect type matching function*. Here l_1 and l_2 represent the types of the semantic aspects for which we want to estimate attracting / repulsive force. The aspect type matching function gives an estimate how likely is the two semantic aspects given with their centroid distances l_1 and l_2 to influence each other through attractive or repulsive force. The aspect type matching function has a range $[0,1]$.
In some of our future investigations we will use the following aspect type matching function:

$$\Phi(l_1, l_2) = e^{-c|l_1 - l_2|^2} \qquad (2)$$

where c is a constant.

Definition: *Aspect Value Matching function* Θ
Let us have two semantic aspects $\vec{p}_1$ and $\vec{p}_2$, their centroids $\vec{p}_{c_1}$ and $\vec{p}_{c_2}$, and their types $l_1 = |\vec{p}_1 - \vec{p}_{c_1}|$, $l_2 = |\vec{p}_2 - \vec{p}_{c_2}|$. Let is denote with $\boldsymbol{\theta}^{(1)} = \left(\theta_1^{(1)}, \theta_2^{(1)}, \dots, \theta_{K-1}^{(1)}\right)$ the coordinates of first centered aspect vector $\vec{p}_1$ and with $\boldsymbol{\theta}^{(2)} = \left(\theta_1^{(2)}, \theta_2^{(2)}, \dots, \theta_{K-1}^{(2)}\right)$ the coordinates of the second centered aspect vector $\vec{p}_2$. Here each of the pairs $\left(l_1, \boldsymbol{\theta}^{(1)}\right)$ and $\left(l_2, \boldsymbol{\theta}^{(2)}\right)$ uniquely identifies the positions of the tip of each of the two aspect values with respect to their corresponding centroids. Let us assume that $\Phi(l_1, l_2)$ is 1 so their types are matching. The question is under what conditions the two aspect values encoded in their corresponding coordinate positions $\boldsymbol{\theta}^{(1)}$ and $\boldsymbol{\theta}^{(2)}$ will attract, repel each other and won't influence each other. The answer to this question is given by the aspect value matching function Θ defined as:

$$\Theta = \Theta\left(\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}\right) \qquad (3)$$

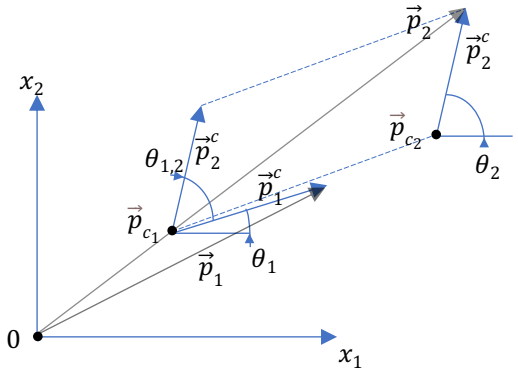
The aspect value matching function $\Theta\left(\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}\right)$ has a range of $[-1,1]$.

Example of an aspect value matching function is:

$$\Theta = -\sin\theta_{1,2} \qquad (4)$$

where $\theta_{1,2}$ is the angle between the two centered aspect vectors $\vec{p}_1 - \vec{p}_{c_1}$ and $\vec{p}_2 - \vec{p}_{c_2}$.

Here are some special cases for which it is particularly easy to express $\theta_{1,2}$ in terms of $\boldsymbol{\theta}^{(1)}$ and $\boldsymbol{\theta}^{(2)}$. When $K = 2$ we have $\boldsymbol{\theta}^{(1)} = \theta_1$ and $\boldsymbol{\theta}^{(2)} = \theta_2$. Then $\theta_{1,2} = \theta_1 - \theta_2$ and we have $\Theta = -\sin(\theta_1 - \theta_2)$.



When $K = 3$ the picture is a bit more involved but it is not difficult to find a closed form expression for $\theta_{1,2}$ in terms of $\theta_1^{(1)}, \theta_2^{(1)}, \theta_1^{(2)}$ and $\theta_2^{(2)}$. Here the triplet $\left(l_1, \theta_1^{(1)}, \theta_2^{(1)}\right)$ uniquely identifies the position of the centered semantic aspect vector $\vec{p}_1 - \vec{p}_{c_1}$ and the triplet $\left(l_2, \theta_1^{(2)}, \theta_2^{(2)}\right)$ uniquely identifies the position of the centered semantic aspect vector $\vec{p}_2 - \vec{p}_{c_2}$.

Definition: *Semantic energy density function* Ψ : this is a non-local function giving the density of the semantic energy field at every point $\mathbf{r} = (x_1, x_2, \dots, x_N)$ in semantic space and for every aspect type l:

$$\Psi = \Psi(\mathbf{r}, l) \qquad (5)$$

Everywhere in the future discussion we will assume that:

1) the energy density $\Psi(\mathbf{r}, l)$ is continuous function with respect to the semantic position $\mathbf{r}$ and with respect to the aspect type l .
2) The energy density $\Psi(\mathbf{r}, l)$ will have continuous first derivative with respect to the semantic position $\mathbf{r}$ and with respect to the aspect type l as well.

Examples of semantic energy density functions:
Constant semantic energy density for every aspect type:

$$\Psi = \Psi(l) = const \qquad (6)$$

The energy density Ψ assumes constant value throughout the semantic space for a given aspect type l and is continuous function with respect to l.

Gaussian semantic energy density:

$$\Psi(\mathbf{r}, l) = \sum_i A_i \times e^{-|\mathbf{r} - \mathbf{r}_i(l)|^2} \qquad (7)$$

Here $\mathbf{r}_i(l)$ represent the positions of a set of points in semantic space which are energy density peaks. The positions of the peaks obviously depend on the type l of the semantic aspect. $A_i = A_i(C)$ are a set of scalars which values depend on the current context C.

Potential Well semantic energy density:

$\Psi(\mathbf{r}, l) = \sum_i A_i - B_i \times e^{-|\mathbf{r}-\mathbf{r}_i(l)|^2};\ A_i \geq B_i$ (8)

Here $\mathbf{r}_i(l)$ represent the positions of a set of points in semantic space which are energy density wells. The positions of the wells obviously depend on the type l of the semantic aspect. $A_i(C)$ and $B_i(C) \leq A_i(C)$ represent a set of scalars which values depend on the current context C.

Definition: *closely related semantic aspect types* – two semantic aspect types l_1 and l_2 are closely related if there exist a pair of aspect values having the types l_1 and l_2 for which the product of the *aspect type matching function* and the *aspect value matching function* is close enough to 1 by absolute value.

Let us denote with $\vec{p}_1$ and $\vec{p}_2$ two aspect vectors and with $\vec{p}_{c_1}$ and $\vec{p}_{c_2}$ the associated centroids, such that $\left|\vec{p}_1 - \vec{p}_{c_1}\right| = l_1$ and $\left|\vec{p}_2 - \vec{p}_{c_2}\right| = l_2$. Let us denote with $\boldsymbol{\theta}^{(1)} = \left(\theta_1^{(1)}, \theta_2^{(1)}, \dots, \theta_{K-1}^{(1)}\right)$ the coordinates of first centered aspect vector $\vec{p}_1 - \vec{p}_{c_1}$ and with $\boldsymbol{\theta}^{(2)} = \left(\theta_1^{(2)}, \theta_2^{(2)}, \dots, \theta_{K-1}^{(2)}\right)$ the coordinates of the second centered aspect vector $\vec{p}_2 - \vec{p}_{c_2}$.

Strictly, two semantic types are closely related *iff* for each of the aspect types l_1 and l_2 there exist two coordinate points $\boldsymbol{\theta}^{(1)}$ and $\boldsymbol{\theta}^{(2)}$ and aspect values $\vec{p}_1\left(\boldsymbol{\theta}^{(1)}\right)$ and $\vec{p}_2\left(\boldsymbol{\theta}^{(2)}\right)$ such that:

$$1.0 - \Phi(l_1, l_2) \times \left|\Theta\left(\vec{p}_1\left(\boldsymbol{\theta}^{(1)}\right), \vec{p}_2\left(\boldsymbol{\theta}^{(2)}\right)\right)\right| < \varepsilon \quad (9)$$

for some small enough $\varepsilon > 0$.

In other words, the types of the semantic aspects need to be matching and there must be a pair of matching aspect values having those types.

In the future we will explore the case where only closely related aspect types can exert attractive or repelling force to each other.

For example, with the example aspect type matching function in (2) a necessary condition for two types l_1 and l_2 to be closely related is that l_1 and l_2 are close enough. That is,
there exists a monotonously increasing function $\delta = \delta(\varepsilon)$ such that

$$|l_1 - l_2| < \delta \quad (10)$$

Note that with other chosen aspect type matching functions this condition requiring proximity of the aspect types is no longer necessary. In the future discussion we will explore only such aspect type matching functions Φ which require condition (10) for close relatedness of aspect types.

Definition: *Attractive/Repulsive force between semantic aspects*

Let us have the two semantic aspects A_1 and A_2 given with their coordinate tuples $\left(\vec{p}_1, \vec{p}_{c_1}\right)$ and $\left(\vec{p}_2, \vec{p}_{c_2}\right)$. Here $\vec{p}_1$ and $\vec{p}_2$ represent the position of the tip of each of the aspects in semantic space. The vectors $\vec{p}_{c_1}$ and $\vec{p}_{c_2}$ represent the position of each of the centroids. The types of A_1 and A_2 are given with $l_1 = \left|\vec{p}_1 - \vec{p}_{c_1}\right|$ and $l_2 = \left|\vec{p}_2 - \vec{p}_{c_2}\right|$. The positions in semantic space of each of the two aspect types l_1 and l_2 in generalized spherical coordinates with respect to their centroids are denoted with $\boldsymbol{\theta}^{(1)}$ and $\boldsymbol{\theta}^{(2)}$.

Then the attractive/repulsive force between the two aspects is given with:

$f_{12}(A_1, A_2) = \mathcal{C}\frac{\Theta(\boldsymbol{\theta}^{(1)},\boldsymbol{\theta}^{(2)})\Phi(l_1,l_2)}{|\vec{p}_1-\vec{p}_2|^2}$ (11)

Here $\mathcal{C} > 0$ is some proportionality constant which will give the dimensions of semantic force on the RHS.

Notice that the sign of f_{12} is given with sgn $\Theta\left(\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}\right)$.

Modeling interaction between semantic properties

Let us have two properties P_1 and P_2 given with their centroids $\vec{p}_{c_1}$, $\vec{p}_{c_2}$ and aspect sets $\mathcal{P}_1 = \left\{\vec{p}_1^{(1)}, \vec{p}_2^{(1)}, \dots, \vec{p}_a^{(1)}\right\}$ and $\mathcal{P}_2 = \left\{\vec{p}_1^{(2)}, \vec{p}_2^{(2)}, \dots, \vec{p}_b^{(2)}\right\}$. We would like to model the attractive / repelling force between the two properties. An assumption comes to mind which makes the modeling simple - let us assume that the force between P_1 and P_2 is a linear superposition of the forces acting on every pair of aspects $A_i^{(1)} \in \mathcal{P}_1$ and $A_j^{(2)} \in \mathcal{P}_2$. Then we can write the following expression for the total force between P_1 and P_2:

$f(P_1, P_2) = \sum_{i,j} f_{12}\left(A_i^{(1)}, A_j^{(2)}\right)$ (12)

Definition: *relevant aspect pair* $\left(A_i^{(1)}, A_j^{(2)}\right)$ is such pair which has absolute binding force value not in the first ℓ -quantile for some $\ell > 0$. In other words, all region pairs which are *in* the first ℓ-quantile are *irrelevant*

The RHS of the expression above can be split into two sets of terms

$f(P_1, P_2) = f^+(P_1, P_2) + f^-(P_1, P_2)$ (13)

where

$f^+(P_1, P_2) = \sum_{i,j} f_{12}^+\left(A_i^{(1)}, A_j^{(2)}\right) > 0$. Here $A_i^{(1)} \in \mathcal{P}_1$ and $A_j^{(2)} \in \mathcal{P}_2$ represent all *relevant* aspect pairs which generate attractive force.

Similarly

$f^-(P_1, P_2) = \sum_{i,j} f_{12}^-\left(A_i^{(1)}, A_j^{(2)}\right) < 0$ represent all *relevant* aspect pairs which generate repelling force.

On the Semantic Mass of a Property

Each property has specific semantic mass. The path from the root in the particle property tree for a given property is determined based on its semantic mass, attractive or repelling force to other properties in the tree as well as the semantic energy stored in the property.

The semantic mass of a property can be represented as a product of two terms. The first term is determined based on the *semantic information* a property conveys. The second term is determined based on the property *valence*.

$Property\ Mass \sim Information\ Content \times\ Property\ Valence$

or in symbol notation:

$\mathfrak{m}_P \sim IC_P \times VL_P$ (14)

where IC_P is the *information content* and VL_P is the *valence* of the property P. The *information content* of a property depends on the number of different semantic aspect types and aspect points from which the property is constructed. It also depends on their relative positions and orientation in semantic space.
Certain properties have the affinity to bind to multiple child properties which reveal additional details for the semantic information provided by the parent. The more child properties a parent property can bind to - the higher will be its property valence. Property valence is related to the number of semantic aspect types and their specific orientation in semantic space.

The carrier of semantic mass in a property is the semantic aspect. We will assume that each semantic aspect in a property carries a unit of mass. Obviously, the more semantic aspect points a property is composed of - the higher information content and property valence will be attributed to that property.

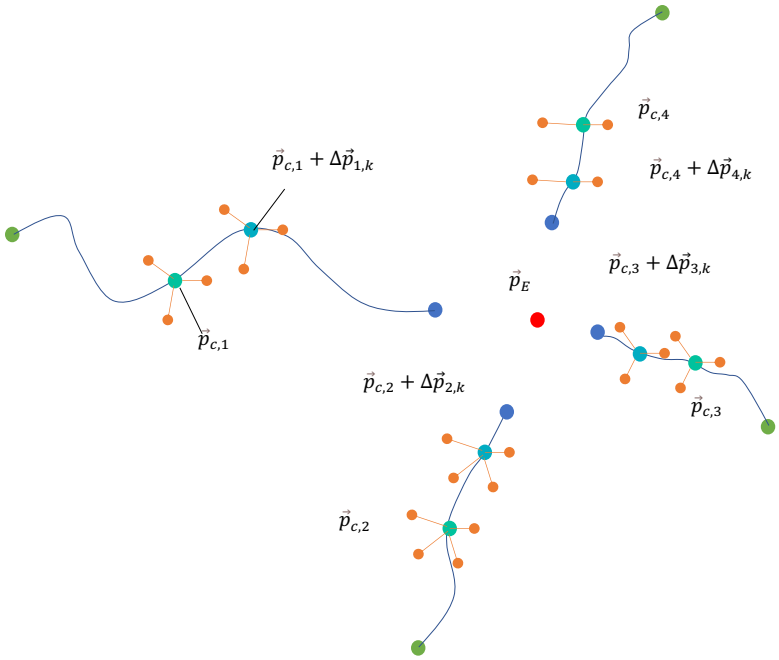
On the Semantic Energy of a Semantic Property

Another characteristic of the Semantic Property is the Semantic Energy $E(P)$ stored in it.

Definiton: *Semantic Energy of a Property*

Note: The *Semantic Energy* of a Property exists as an independent semantic entity only until the property travels toward its bound state in the semantic particle ensemble. As soon as the property is bound to a semantic particle it no longer can be considered independent semantic entity with energy and other semantic attributes. Rather it adds to the semantic quantities (such as semantic energy) of the semantic particle it is part of.

The *Semantic Energy* of a property P is determined by the path the property has travelled in semantic space from its original unbound *in-situ* position until its bound position inside its semantic particle assembly. Every property starts its travel to the centroid $\vec{p}_E(V)$ of the semantic particle V with zero semantic energy.
In the Figure below it is depicted an ensemble of 4 properties travelling toward their bound states. With green are depicted the in-situ positions of each of the properties. With blue are depicted the bound state positions of each of the properties. The red point in the center is the energy weighted center of mass of the ensemble $\vec{p}_E$ when each of the properties are in their bound states. The mass center of each property in some position between in-situ and bound state, denoted with $\vec{p}_{c,i}$, is depicted as a magenta point. The orange points connected to each magenta point represent the the semantic aspects of each property. The blue lines between the in-situ and bound state positions are the trajectories (travel paths) of the mass centers of each property. Notice that the trajectories are not straight lines because the position of the energy weighted center of mass of the ensemble $\vec{p}_E$ moves while the properties are travelling toward their bound states. How exactly $\vec{p}_E$ moves while the properties are travelling toward their bound states will be found in a later discussion in this article.



The semantic energy of P is gradually accumulated along its travel path $S(P)$ and can be computed by the relationship below:

$$E(P) = \sum_k \sum_{i=1}^{N} \Psi(\vec{p}_i + \Delta\vec{p}_k, l_i)\Delta s_k \qquad (15)$$

Here $\Delta\vec{p}_k$ represents the new position of each aspect after advancing the property with step Δs_k along the path toward its bound state. The step $\Delta s_k = |\Delta\vec{p}_k|$ multiplied by the energy density at each of the aspects gives us the total energy accumulated in the property at the end of this step. Note that in (15) we are making the following approximation

$$\frac{\Psi(\vec{p}_i+\Delta\vec{p}_k,l_i)+\Psi(\vec{p}_i,l_i)}{2} \sim \Psi(\vec{p}_i + \Delta\vec{p}_k, l_i) \qquad (16)$$

which is a reasonable one if Ψ is slowly varying in the vicinity of $\vec{p}_i$ with the chosen step Δs_k.

Statement: The energy state and semantic position of a semantic property P on every point s along the travel path $S(P)$ toward bound state is represented uniquely by the 4-tuple $\vec{p}_i(s), \vec{p}_E(s), l_i(s), E(s)$.

Additional parameter which we will introduce in our semantic construct will allow for reducing the accumulated energy in a semantic property.

Definition: *Energy Dissipation Rate*
When a semantic property P travels along some path $S(P)$ toward bound state not only will accumulate semantic energy but also will dissipate certain amount of energy on every step. The density of the dissipated energy is given with:

$\Lambda = \Lambda(\vec{r}, l)$ (17)

Note that on the density of the energy dissipation will be imposed the following constraints which will simplify the task of modeling semantic structures:

1) $\Lambda(\vec{r}, l)$ is continuous function in semantic space $\vec{r}$ as well as with respect to the aspect type l.
2) $\Lambda(\vec{r}, l)$ has continuous first derivative in semantic space $\vec{r}$ as well as with respect to the aspect type l.
3) For a given $\vec{r}$ and l the density of semantic energy dissipation cannot exceed the density of the semantic energy: $\Psi(\vec{r}, l) \geq \Lambda(\vec{r}, l)$

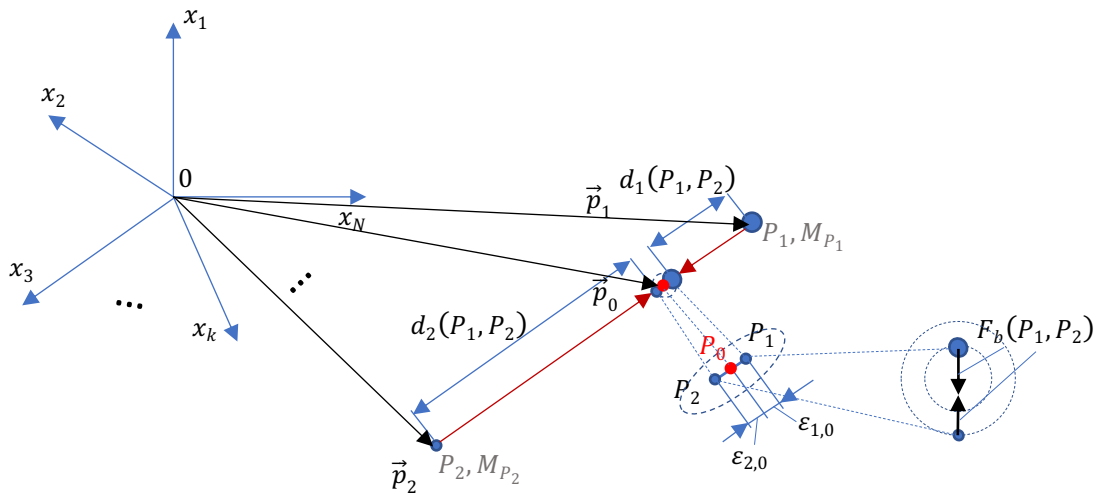
Imposing those constraints on the energy dissipation density together with the continuity requirement for the energy density and its first derivative assure that the net semantic energy accumulated along the travel path of a semantic structure (property/particle/particle compound) will be continuous and monotonously increasing function.

Definition: *Net Semantic Energy of a property*

The Net semantic energy of a property is the total net accumulated energy in the property along its travel path toward bound state. It is given with an expression like the one below:

$E_t(P) = \sum_k \sum_{i=1}^{N} \Psi(\vec{p}_i + \Delta\vec{p}_k, l_i)\Delta s_k - \Lambda(\vec{p}_i + \Delta\vec{p}_k, l_i)\Delta s_k$ (18)

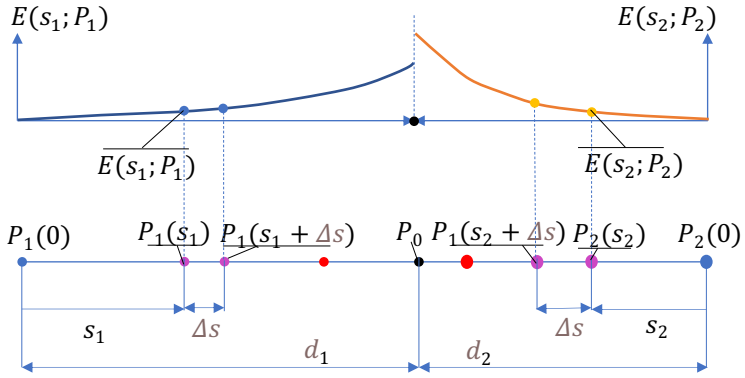
Bound state of an ensemble of semantic properties



In the Figure above there are shown two properties P_1 and P_2 given with the initial (*in-situ*) positions of their centroids $\vec{p}_1$ and $\vec{p}_2$. Each of P_1 and P_2 travels certain distance d_1 and d_2 toward the center of mass of the ensemble (P_1, P_2). Each property stops at some distance $\varepsilon_{1,0}/\varepsilon_{2,0}$ from the common center of mass $\vec{p}_0$.

Energy-weighted centroid of an ensemble of semantic properties

Problem: *Energy-weighted center of mass (centroid) of an ensemble of semantic properties*
Let us consider a set of semantic properties $P_i, i = 1..M$ which are in a general position i.e. not necessarily in their original (*in-situ*) positions on their travel paths toward bound state. Let us denote with $s_i > 0$ the current location of P_i along its travel toward bound state i.e. toward the energy-weighted center of the ensemble. On the Figure below it is depicted an ensemble with two properties $\{P_1, P_2\}$ on distances s_1 and s_2 accordingly from their in-situ positions.



Thus, each property would have acquired some non-zero amount of semantic energy $E_i, i = 1..M$. Note that for simplicity of the visualization we have not depicted the individual structure of the properties. We have simply represented the properties by their centroid $\vec{p}_{c,i}$ to which we have assigned the natural coordinate s_i. However, the semantic aspects in each of the properties obviously exist and will be used to calculate the new energy chunk consumed by each of the properties. The set of semantic aspects of P_i will be denoted with $A_{i,j} = (\vec{p}_{i,j}, l_{i,j}), j = 1..N_i$.

So, we would like to find the position of the energy-weighted centroid of the ensemble of properties at every step during the travel path of each property toward the centroid.

Let us move each property P_i with a small enough incremental step Δs along its path $S(P_i)$ toward its bound position in the ensemble. The bound position of each property is depicted by red dot in the Figure above.
The new position of each aspect then becomes $\vec{p}_{i,j} + \Delta\vec{p}_i, j = 1..N_i$. The new vector increment for any aspect $A_{i,j}$ of P_i is denoted by $\Delta\vec{p}_i$. The energy of the aspect $A_{i,j}$ accumulated over its travel from initial (in-situ) position to some point $s_{i,j} + \Delta s$ is given with:

$$E(\vec{p}_{i,j} + \Delta\vec{p}_i, l_{i,j}) = E(\vec{p}_{i,j}, l_{i,j}) + \Psi(\vec{p}_{i,j} + \Delta\vec{p}_i, l_{i,j})\Delta s \quad (19)$$

Then the energy of P_i in its new position will be given with:

$$E(\vec{p}_{c,i} + \Delta\vec{p}_i) = E(\vec{p}_{c,i}) + \sum_{j=1}^{N_i} \Psi(\vec{p}_{i,j} + \Delta\vec{p}_i, l_{i,j})\Delta s \quad (20)$$

Here we assume that step Δs is small enough so that $\Psi\left(\vec{p}_{i,j}+\Delta\vec{p}_i, l_{i,j}\right) \sim \Psi\left(\vec{p}_{i,j}, l_{i,j}\right)$.

Then we can compute a first order approximation of the energy-weighted center of mass of the ensemble of properties $\{P_1, \ldots, P_M\}$ as:

$$\vec{p}_E^{(1)} = \sum_{i=1}^{M} \frac{\tilde{E}^{(1)}(s+\Delta s)\,\mathfrak{m}_i}{E\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)}\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right) \qquad (21)$$

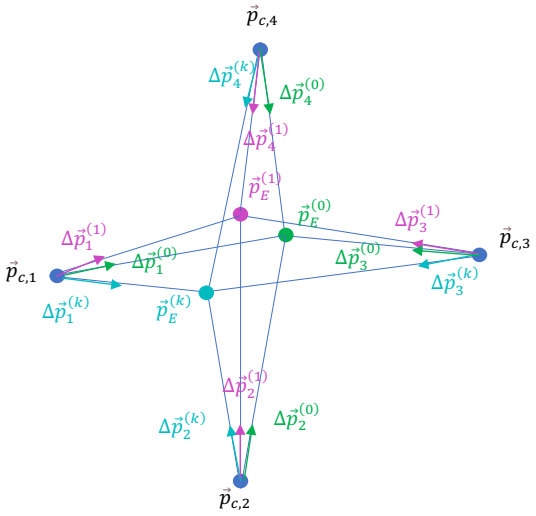
where the first order approximation of the harmonic mass weighted aggregate of the energy of the ensemble $\tilde{E}^{(1)}(s+\Delta s)$ is given with:

$$\tilde{E}^{(1)}(s+\Delta s) = \frac{1}{\sum_{i=1}^{M} \frac{\mathfrak{m}_i}{E\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)}} \qquad (22)$$

Here the semantic energy $E\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)$ of the property P_i is given with (20).

In (21) $\mathfrak{m}_i$ denotes the semantic mass of property P_i, $\vec{p}_{c,i}$ is the current position of the centroid of P_i. In the one-dimensional case of an ensemble of two properties (1-polytope) depicted in the Figure above we simply can mark the position of $\vec{p}_{c,i}$ with the natural coordinate s_i.

(21) gives us the position of the new weighted center of mass of the ensemble $\vec{p}_E^{(1)}$. In general, this necessitates the recalculation of $\Delta\vec{p}_i$ such that the incremental step will point in the new direction of $\vec{p}_E^{(1)}$. Let us denote the recalculated incremental step toward $\vec{p}_E^{(1)}$ by $\Delta\vec{p}_i^{(1)}$. On the Figure below it is shown an ensemble of four properties represented by their centroids $\vec{p}_{c,i}$, $i = 1..4$. The consecutively recalculated vector increments $\Delta\vec{p}_i^{(0)}$, $\Delta\vec{p}_i^{(1)}$, … , $\Delta\vec{p}_i^{(k)}$ and energy-weighted mass center of the ensemble $\vec{p}_E^{(0)}, \vec{p}_E^{(1)}, \ldots, \vec{p}_E^{(k)}$ are shown on the Figure as well.



Obviously, we can write:

$$\Delta\vec{p}_i^{(1)} = \frac{\vec{p}_E^{(1)} - \vec{p}_{c,i}}{\left|\vec{p}_E^{(1)} - \vec{p}_{c,i}\right|}\Delta s \qquad (23a)$$

With $\Delta\vec{p}_i = \Delta\vec{p}_i^{(1)}$ (21) and (22) give us the second order approximation of the energy-weighted center of mass:

$$\vec{p}_E^{(2)} = \sum_{i=1}^{M} \frac{\tilde{E}^{(2)}(s+\Delta s)\, \mathfrak{m}_i}{E\left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(1)}\right)} \left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(1)}\right) \qquad (23)$$

where the second order approximation of the harmonic mass-weighted aggregate of the energy of the ensemble is given with:

$$\tilde{E}^{(2)}(s+\Delta s) = \frac{1}{\sum_{i=1}^{M} \frac{\mathfrak{m}_i}{E\left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(1)}\right)}} \qquad (24)$$

The solution is to continue calculating $\vec{p}_E^{(1)}, \vec{p}_E^{(2)}, \ldots$ iteratively until for some k both $\left|\vec{p}_E^{(k)} - \vec{p}_E^{(k+1)}\right|$ and $\left|\tilde{E}^{(k)} - \tilde{E}^{(k+1)}\right|$ get smaller than some predefined ε.

On the $k+1$-th iteration we compute the semantic energy and energy-weighted centroid:

$$E\left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(k)}\right) = E\left(\vec{p}_{c,i}\right) + \sum_{j=1}^{N_i} \Psi\left(\vec{p}_{i,j} + \Delta\vec{p}_i^{(k)}, l_{i,j}\right)\Delta s \qquad (25)$$

$$\tilde{E}^{(k+1)}(s+\Delta s) = \frac{1}{\sum_{i=1}^{M} \frac{\mathfrak{m}_i}{E\left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(k)}\right)}} \qquad (26)$$

$$\vec{p}_E^{(k+1)} = \sum_{i=1}^{M} \frac{\tilde{E}^{(k+1)}(s+\Delta s)\, \mathfrak{m}_{P_i}}{E\left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(k)}\right)} \left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(k)}\right) \qquad (27)$$

$$\Delta\vec{p}_i^{(k+1)} = \frac{\vec{p}_E^{(k+1)} - \vec{p}_{c,i}}{\left|\vec{p}_E^{(k+1)} - \vec{p}_{c,i}\right|}\Delta s \qquad (28a)$$

We stop the iterations when both are satisfied:

$$\left|\vec{p}_E^{(k)} - \vec{p}_E^{(k+1)}\right| < \varepsilon \text{ and } \left|\tilde{E}^{(k)} - \tilde{E}^{(k+1)}\right| < \varepsilon \qquad (28)$$

Then we denote by $\vec{p}_E = \vec{p}_E^{(k+1)}$ the true energy-weighted centroid of the ensemble $\{P_1, P_2, \ldots, P_M\}$ on its current position $\left(\vec{p}_{c,1} + \Delta\vec{p}_1^{(k)}, \vec{p}_{c,2} + \Delta\vec{p}_2^{(k)}, \ldots, \vec{p}_M + \Delta\vec{p}_M^{(k)}\right)$
Note that if $\Psi(\vec{p}, l) = const$ for each of the semantic aspects in the ensemble in some neighborhood around the current position $\left(\vec{p}_{c,1} + \Delta\vec{p}_1^{(k)}, \vec{p}_{c,2} + \Delta\vec{p}_2^{(k)}, \ldots, \vec{p}_M + \Delta\vec{p}_M^{(k)}\right)$ then

$$\vec{p}_E^{(0)} = \vec{p}_E^{(1)} = \cdots = \vec{p}_E^{(k+1)} = \vec{p}_c \qquad (29)$$

However, if $\Psi(\vec{p}, l) = f(l)$ for each of the semantic aspects in the ensemble then we will end up with energy weighted centroid $\vec{p}_E$ which in general will be different than the mass centroid $\vec{p}_c$ of the ensemble:

$$\vec{p}_E^{(0)} = \vec{p}_E^{(1)} = \cdots = \vec{p}_E^{(k+1)} \neq \vec{p}_c \quad (30)$$

Taking into account the dissipation energy density given with (17) we can augment the process of calculating the true energy-weighted centroid of the property described in (15) - (28) as:

$$E_t\left(\vec{p}_{c,i} + \Delta\vec{p}_i\right) = E\left(\vec{p}_{c,i}\right) + \sum_{j=1}^{N_i} \Psi\left(\vec{p}_{i,j} + \Delta\vec{p}_i, l_{i,j}\right)\Delta s - \Lambda\left(\vec{p}_{i,j} + \Delta\vec{p}_i, l_{i,j}\right)\Delta s \quad (31)$$

Let us denote with $E_t\left(\vec{p}_{c,i} + \Delta\vec{p}_i\right) = E\left(\vec{p}_{c,i} + \Delta\vec{p}_i\right) - D\left(\vec{p}_{c,i} + \Delta\vec{p}_i\right)$ the net energy accumulated by the property at the point $s + \Delta s$.
Note that $E_t\left(\vec{p}_{c,i} + \Delta\vec{p}_i\right) > 0 \; \forall \; i = 1..M$ otherwise there will be a property P_i which does not belong to the ensemble $\{P_1, \ldots, P_M\}$ as it will not be able to arrive at its bound state (see (46)).

Then for the first order approximation of the net energy weighted centroid we have:
$\vec{p}_E^{(1)} = \sum_{i=1}^{M} \frac{\tilde{E}_t^{(1)}(s+\Delta s)\, \mathrm{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)} \vec{p}_i \quad (32)$

Here $\tilde{E}_t^{(1)}(s + \Delta s)$ denotes the first order approximation of the harmonic mass weighted aggregate of the net energy accumulated by all properties at the point $s + \Delta s$. It is given with:

$$\tilde{E}_t^{(1)}(s + \Delta s) = \frac{1}{\sum_{i=1}^{M} \frac{\mathrm{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)}} \quad (33)$$

Note that $\tilde{E}_t^{(1)}(s + \Delta s) > 0 \; \forall \; s \geq s_0$ because otherwise there would be present a property in the ensemble which would have zero net energy at $s + \Delta s$.

$$E_t\left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(k)}\right) = E\left(\vec{p}_{c,i}\right) + \sum_{j=1}^{N_i} \Psi\left(\vec{p}_{i,j} + \Delta\vec{p}_i^{(k)}, l_{i,j}\right)\Delta s - \Lambda\left(\vec{p}_{i,j} + \Delta\vec{p}_i^{(k)}, l_{i,j}\right)\Delta s \quad (34)$$

$$\tilde{E}_t^{(k+1)}(s + \Delta s) = \frac{1}{\sum_{i=1}^{M} \frac{\mathrm{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i^{(k)}\right)}} \quad (35)$$

$$\vec{p}_E^{(k+1)} = \sum_{i=1}^{M} \frac{\tilde{E}_t^{(k+1)}(s+\Delta s)\, \mathrm{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i^{(k)}\right)} \left(\vec{p}_{c,i} + \Delta\vec{p}_i^{(k)}\right) \quad (36)$$

$$\Delta\vec{p}_i^{(k+1)} = \frac{\vec{p}_E^{(k+1)} - \vec{p}_{c,i}}{\left|\vec{p}_E^{(k+1)} - \vec{p}_{c,i}\right|} \Delta s \quad (37)$$

We stop the iterations when both conditions below are satisfied:

$$\left|\vec{p}_E^{(k)} - \vec{p}_E^{(k+1)}\right| < \varepsilon \text{ and } \left|\tilde{E}_t^{(k)} - \tilde{E}_t^{(k+1)}\right| < \varepsilon \quad (38)$$

Finally, using (34)-(37), we can formulate the problem of finding the energy weighted centroid $\vec{p}_E$, the net energy for each property E_t at the current incremental step, harmonic mass weighted aggregate of the net energy of the ensemble $\tilde{E}_t$, the vector increment $\Delta\vec{p}_i$ for each property and the net energy for each property as a solution of a coupled system of non-linear equations:

$$\begin{cases} E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)=E\left(\vec{p}_{c,i}\right)+\sum_{j=1}^{N_i}\Psi\left(\vec{p}_{i,j}+\Delta\vec{p}_i,l_{i,j}\right)\Delta s-\Lambda\left(\vec{p}_{i,j}+\Delta\vec{p}_i,l_{i,j}\right)\Delta s \\ \tilde{E}_t=\frac{1}{\sum_{i=1}^{M}\frac{\mathfrak{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)}} \\ \vec{p}_E=\sum_{i=1}^{M}\frac{\tilde{E}_t\,\mathfrak{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)}\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right) \\ \Delta\vec{p}_i=\frac{\vec{p}_E-\vec{p}_{c,i}}{\left|\vec{p}_E-\vec{p}_{c,i}\right|}\Delta s \end{cases} \quad (39)$$

which we obviously have solved by successive iteration relying on a fixed point assumption.

Determining the bound state for each property in the ensemble

The bound state of each property is represented by two things – the ending position of the travel path and the net energy of the property at that position.
The question is how we determine how far from the common center of mass each property will stop. We have some approximate idea what we want to happen – a property with the highest mass should land closes to the center. But that is not all – we want to account for the accumulated energy along the travel path. The properties which have accumulated more energy will stay further from the mass center of the ensemble. The properties with less energy will come closer to the center. The closest to the center will be the property with the largest mass and least energy. Let us be more precise –
Let us define the following quantity for each property part of the ensemble:

$$\upsilon=\sqrt{\frac{E_t}{\mathfrak{m}}} \quad (40)$$

where E_t denotes the total net energy accumulated during the travel of all properties in the ensemble and $\mathfrak{m}$ denotes the semantic mass of the current property P. The quantity υ represents semantic velocity of the particle P when subjected to the total energy E_t of the ensemble.
Using (15) we can write

$$E_t=\sum_{P\in\{P_1,P_2,..\}}\sum_{k:\,\Delta\vec{p}_k\in T(P)}\sum_{j=1}^{N(P)}\Psi\left(\vec{p}_j+\Delta\vec{p}_k,l_j\right)\Delta s-\Lambda\left(\vec{p}_j+\Delta\vec{p}_k,l_j\right)\Delta s \quad (33)$$

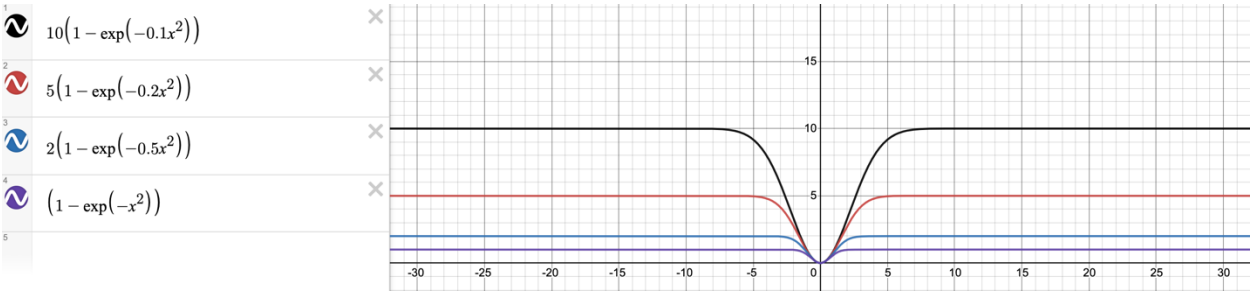
Let us define the following function $E(x)$ which we will denote as *Gaussian Inverse Semantic Energy Well*

$$E(x)=\mathfrak{m}_e\upsilon^2\left(1-e^{-\frac{f^2x^2}{\upsilon^2}}\right) \quad (34)$$

We will use this function to determine the end position of each property in its bound state.
The coordinate x represents the distance from the center of mass of the ensemble in bound state of a property P which has energy $E=E(x)$ and velocity υ.
Here $\mathfrak{m}_e$ is some normalization coefficient which has the dimension of semantic mass, given in *semantic mass units* (**smu**). f is another normalization coefficient which converts the term $\frac{fx}{\upsilon}$ into a dimensionless quantity. f has the dimension of semantic frequency ($\mathbf{stu}^{-1}$). The Semantic Energy has dimension of $\mathbf{smu}\times\frac{\mathbf{sme}^2}{\mathbf{stu}^2}$.

Here **smu** denotes *semantic mass unit*, **sme** denotes *semantic metric unit* and **stu** denotes *semantic time unit*. The system of measurement for semantic quantities will be discussed in detail in a separate article.

Below are shown example plots for $\mathfrak{m}_e = 1\ \mathbf{smu}$ and $f = 1\ \mathbf{stu}^{-1}$. The **black line** corresponds to $\upsilon = \sqrt{10}\ \mathbf{sme/stu}$. The **red line** corresponds to $\upsilon = \sqrt{5}\ \mathbf{sme/stu}$. The **blue line** corresponds to $\upsilon = \sqrt{2}\ \mathbf{sme/stu}$. The **purple line** corresponds to $\upsilon = 1\ \mathbf{sme/stu}$.



Note the expression (34) is a solution of the following ODE

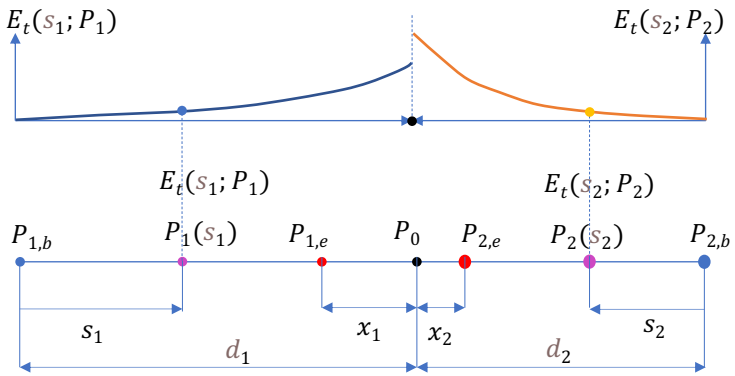
$$\frac{1}{2(1-2\xi^2)}\frac{d^2y}{d\xi^2} + y(\xi) = \mathfrak{m}_e\upsilon^2 \quad \text{where } \xi = \frac{f}{\upsilon}x \qquad (35)$$

subject to the following boundary conditions

$$\lim_{\xi\to\infty} y(\xi) = \mathfrak{m}_e\upsilon^2 \qquad (36)$$

$$\lim_{\xi\to 0} \frac{dy}{d\xi} = 0 \qquad (37)$$

With this information in mind now we can discuss a process which will allow us to determine the end position of each property in its bound state. Let us assume that the energy of each of the properties of the ensemble will grow smoothly and monotonously on the way to bound state of each property. This is depicted on the picture below of an ensemble composed of two properties:



Let us denote with s_1 the current position of P_1 on its way towards the mass center of the ensemble $\{P_1, P_2\}$. With d_1 we denote the distance between the starting (*in-situ*) position of P_1 and the center of mass of the ensemble. With $\mathfrak{m}_1$ we will denote the semantic mass of the property P_1. Let us denote with $E_t(s_1; P_1)$ the total net energy accumulated by property P_1 so far. Then with the value of $E_t(s_1; P_1)$ as $E(x)$ we solve for x_1 in (34):

$$x_1 = \sqrt{-\frac{v^2}{f^2}\ln\left(1 - \frac{E_t(P_1)}{\mathfrak{m}_e v^2}\right)} \qquad (38)$$

Obviously in order x_1 to be real number we require

$$\frac{E_t(P_1)}{\mathfrak{m}_e v^2} < 1 \qquad (39)$$

The last inequality can be rewritten as:

$$\frac{E_t(P_1)}{E_t(\{P_1,P_2\})} < \frac{\mathfrak{m}_e}{\mathfrak{m}_1} \quad (40)$$

Obviously if we set the normalization constant $\mathfrak{m}_e$ to be equal to $\mathfrak{m}_1$ we will guarantee that x_1 will be real number. Setting the other normalization constant f to 1 will give us:

$$x_1 = \sqrt{-\frac{E_t(\{P_1,P_2\})}{\mathfrak{m}_1}\ln\left(1 - \frac{E_t(P_1)}{E_t(\{P_1,P_2\})}\right)} \qquad (41)$$

With λ_1 we define the energy ratio between the net semantic energy stored in P_1 over the net semantic energy stored in the whole ensemble at the current moment:

$$\lambda_1 = \frac{E_t(P_1)}{E_t(\{P_1,P_2\})} \qquad (42)$$

Obviously, the energy ratio is a function of the current position s_i of each of the properties P_i constituting the ensemble.
Thus, we can write general formula for the bound state distances to the center of mass of the ensemble:

$$x_i = \sqrt{-\frac{E_t(\{P_1,\ldots,P_n\})}{\mathfrak{m}_i}\ln(1 - \lambda_i)};\ i = 1..n \qquad (43)$$

where

$$\lambda_i = \frac{E_t(P_i)}{E_t(\{P_1,\ldots,P_n\})} \qquad (44)$$

Since $\lambda_i < 1$ we can expand

$$\ln(1 - \lambda_i) = -\lambda_i - \frac{{\lambda_i}^2}{2} - \frac{{\lambda_i}^3}{3} - \ldots - \frac{{\lambda_i}^n}{n} - \mathcal{O}\left({\lambda_i}^{n+1}\right) \quad (44b)$$

Thus, we can rewrite (43) as:

$$x_i = \sqrt{\frac{E_t(P_i)}{\mathfrak{m}_i} K_i} \qquad (44c)$$

where the energy coefficient K_i of the particle P_i is given with

$$K_i = 1 + \frac{\lambda_i}{2} + \frac{{\lambda_i}^2}{3} + \mathcal{O}\left({\lambda_i}^3\right) \qquad (44d)$$

The distance x_i depends on how two quantities in (43) will change with the travelled path s toward bound state. The energy of the particle $E_t(P_i)$ and the energy of the ensemble $E_t(\{P_1, \ldots, P_n\})$ will keep growing from 0 to their final values accumulated in bound state. Thus $x_i = x_i(\mathrm{s})$ will be continuous function which will grow slowly if λ_i is constant and will be monotonously increasing $if\ \lambda_i = \lambda_i(\mathrm{s})$ is monotonously increasing function. Additionally, if $\lambda_i \ll 1$ then $x_i \sim \sqrt{\frac{E_t(P_i)}{\mathfrak{m}_i}}$. The higher the mass of the particle (property) and the lower its energy the closer to the semantic center the particle will end in its bound state. The higher the accumulated energy by the particle $E_t(P_i)$ compared to the total accumulated energy of the ensemble the farther from the center the particle P_i will end in its bound state.
Let us denote the initial *in-situ* positions with s_i^0 and the bound state positions with s_i^b for $P_i, i = 1..n$.

Problem Statement:
We want to find a set of points $s_i, i = 1..n$ on the travel paths of each property P_i toward the energy weighted center of mass $\vec{p}_E$ of the ensemble of properties $\{P_1, \ldots, P_n\}$ such that

$$d_i - x_i(s_i) = s_i \ \forall\ i = 1..n \qquad (45)$$

such that the total accumulated energy $E_t(\{P_1, \ldots, P_n\})$ and the individual accumulated energies $E_t(P_i)$ acquired while travelling from s_i^0 to $s_i\ \forall\ i = 1..n$ will satisfy (43). This set of points $s_i, i = 1..n$ as well as the final position of the weighted center of mass $\vec{p}_E$ of the ensemble will represent the bound state of the ensemble and we will denote it as a *primitive semantic particle*. We will use capital V letter with or without subscript to denote primitive semantic particles. When required we will include the ensemble of properties as an argument to the primitive semantic particle:

$$V_1 = V_1(\{P_1, \ldots, P_n\})$$

Precondition for a property to advance from *in-situ* state along its travel path:
The property P must satisfy the following inequality

$$\Psi(\vec{p}_i + \Delta\vec{p}_i, l_i)\Delta s > 0\ \forall\ A_i \in P \qquad (46)$$

where $\vec{p}_i$ is the *in-situ* semantic position of each aspect A_i with type l_i

Algorithm for finding the bound state positions for ensemble of properties

We start by advancing all properties along their travel paths in small enough incremental step Δs in round robin fashion. First, we advance one step Δs P_1, then P_2, ..., finally P_n. Assuming that (46) is satisfied for all properties we will end up with positive net energy for each property after this first round

of incremental steps. We compute the corresponding bound state distances to the center of mass $x_i\left(s_i^0 + \Delta s\right)$. Since (49) is satisfied we will have:

$x_i\left(s_i^0 + \Delta s\right) > 0$ for $i = 1..n$. (47)

With small enough step Δs we can assume that $d_i - x_i\left(s_i^0 + \Delta s\right) > s_i$ for $i = 1..n$; in other words, we are not close to satisfying (45).

After the first iteration has completed calculating $x_i\left(s_i^0 + \Delta s\right), i = 1..n$ in round-robin fashion the next step is to recalculate the energy-weighted center of mass for the ensemble using expression similar to (32):

$\vec{p}_E^{(1)} = \sum_{i=1}^{n} \frac{\tilde{E}_t\left(s^0+\Delta s\right) \mathrm{m}_i}{E_t\left(\vec{p}_{c,i}+\Delta\vec{p}_i\right)} \vec{p}_i$ (48)

We will recompute the energy-weighted center of mass of the ensemble after each iteration k using expression similar to (48) and we will recompute the new semantic positions $x_i\left(s_i^0 + k\Delta s\right)$ of each property P_i related to the new position of the energy-weighted center of mass $\vec{p}_E^{(k)}$.

While advancing all properties from their initial *in-situ* positions along their paths we will see that all computed distances $x_i(s)$ will be increasing and possibly fluctuating around particular set of values if $\lambda_i = \lambda_i(s)$ is not constant or monotonously increasing function.

Eventually, we will reach an iteration k and position s_1, where

$d_1 - x_1(s_1) \leq s_1$ (49)

In such case we freeze temporarily the movement of P_1 at this position. We continue moving the rest of the properties with the same incremental step Δs. If after an incremental step Δs applied to another property P_2, the inequality (49) is no longer satisfied (due to decreased energy ratio (44) for $i = 1$) then we "unfreeze" P_1 and advance it again forward with step Δs in the next iteration. Then we continue round robin with the rest of the properties $P_3, \ldots, P_n$. We proceed with the next iteration in a similar fashion. Eventually we will encounter an iteration m where we will have two or more properties $P_{i_1}, \ldots, P_{i_k}, k \geq 2$ satisfying the inequality (49). If $k = n$ we stop here and we are done – the current values of $s_i^0 + m_i\Delta s$ will be on a distance less than Δs from $x_1\left(s_i^0 + m_i\Delta s\right)$ which represents good approximation of (45) if Δs is chosen small enough. If $k < n$ we will have at least one property P_m for which the inequality (49) will not be satisfied. We will advance this property by the same step Δs unfreezing already "frozen" properties as needed. Eventually we will reach iteration where for all properties inequality (49) would be satisfied. This will be our final approximation of (45) which will be a good approximation if Δs is chosen small enough. It is guaranteed that we will reach such iteration before we encounter the mass center of the ensemble because of (46) and (47).

Here is a step-by-step implementation of the algorithm discussed above:

Step 0. Pick a small step Δs

Step 1. Compute the center of mass of the ensemble $\mathcal{P} = \{P_1, \ldots, P_n\}$ using $\vec{p}_c = \frac{\sum_{i=1}^{|\mathcal{P}|} \vec{p}_i}{|\mathcal{P}|}$

Step 2. Advance each property P_i represented by $\vec{p}_i$ toward the center of mass $\vec{p}_c$ of the ensemble. For the purpose compute the incremental change $\Delta\vec{p}_i$ such that the new position of the property will be $\vec{p}_{c,i} + \Delta\vec{p}_i$.
Step 3. After advancing each property toward the center of mass by $\Delta\vec{p}_i$ we compute $E_t(\vec{p}_i + \Delta\vec{p}_i)$ as $E_t(\vec{p}_{c,i} + \Delta\vec{p}_i) = \sum_{j=1}^{N(P_i)} \Psi(\vec{p}_{i,j} + \Delta\vec{p}_{i,j}, l_j)\Delta s - \Lambda(\vec{p}_{i,j} + \Delta\vec{p}_{i,j}, l_j)\Delta s$. Here we have included dissipation term in the net energy expression.
Step 4. Compute the total net energy of the ensemble $E_t(s^0 + \Delta s)$ after applying the step Δs to all properties.
Step 5. Compute the energy-weighted center of mass $\vec{p}_E^{(1)}$ of the ensemble

Step 6. Compute the bound state distance $x_i(s_i^0 + \Delta s)$ to the ensemble center of mass for each property P_i
Step 7. **If** $d_i - x_i(s_i) \leq s_i$ is obeyed **for all** properties in the ensemble **then stop** the iterations as we have reached the desired set of points s_i.
Step 8. **If** $d_i - x_i(s_i) \leq s_i$ is obeyed for the property P_i **and** P_i has not already been frozen **then** freeze that property that is do not advance it in the next iteration
Step 9. **If** $d_i - x_i(s_i) > s_i$ is obeyed for the property P_i **and** P_i has already been frozen **then** unfreeze that property that is advance it in the next iteration
Step 10. Repeat Steps 2 - 6 with $\vec{p}_c := \vec{p}_E^{(1)}$ and $s^0 := s^0 + \Delta s$

Forming an Ensemble of semantic properties

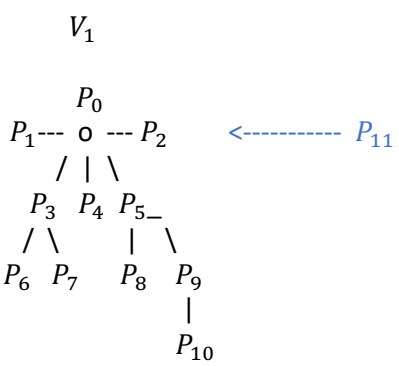
An Ensemble of semantic properties will be constructed from a set of semantic properties $P_1, P_2, \ldots, P_n$ placed in their original *in-situ* positions in Semantic Space.

//TODO: finish this

Appendix

Constructing the property tree: constraints and inequalities based on binding force

Let us consider the property tree of a V-particle:



Let us imagine we want to add new P-particle to the property tree. The following steps toward forming a new ensemble take place:

Step 1. All P-particles which are about to participate in the new ensemble become disassociated / disentangled.
Step 2. Bound state characteristics for all ensemble particles are determined:

a) the energy weighted centroid at bound state for the new ensemble is determined,
b) the bound state distances to the ensemble centroid for all properties are determined,
c) the accumulated net energy at bound state in each of the properties is determined

Step 3. The particle P_i with the smallest product of the mass-to-net-energy ratio $\frac{E_t(P_i)}{\mathfrak{m}_i}$ and energy coefficient K_i will be closest to the ensemble centroid and will be root of the property tree.

Special Case:
there are two P-particles with the same semantic mass-to-energy ratio which happens to be the largest ratio in the particle tree. Then the particle with the lower semantic energy will be closer to the semantic center than the particle with the higher energy. ~~(not sure about the energy condition)~~. ~~An ensemble in which the heaviest two particles are having the same masses and same energies is ill-formed and one of the two properties has to have a region discarded so it will end up with lower semantic mass.~~

Step 4. Let us have two particles P-particles P_j and P_k such that $M_{P_j} \geq M_{P_k}$. Let us denote with P_i the particle with the closest but larger semantic mass than that of P_j and P_k. Thus $M_{P_i} > M_{P_j} \geq M_{P_k}$. Then each one of the following configurations are possible:

a)	*b)*	*c)*	*d)*	*e)*	*f)*	*g)*
P_i	P_i	P_i	P_i	P_i	P_i	P_i
/ \	/ \	|	|	|	|	x x
P_j P_k	P_k P_j	P_j	P_k	P_j	P_k	P_j P_k
		|	|	x	x	
		P_k	P_j	P_k	P_j	

Case a) will occur when there is non-zero binding force $f_{i,j}^{+} = f^{+}(P_i, P_j) > 0$ between P_i and P_j and also between P_i and P_k - $f_{i,k}^{+} = f^{+}(P_i, P_k) > 0$. In this case either $M_{P_j} > M_{P_k}$ or $M_{P_j} = M_{P_k}$ and $f_{i,j}^{+} > f_{i,k}^{+}$.

Case b) will occur when there is non-zero binding force $f^{+}(P_i, P_j) > 0$ between P_i and P_j and also between P_i and P_k - $f^{+}(P_i, P_k) > 0$. In this case $M_{P_j} = M_{P_k}$ and $f_{i,k}^{+} > f_{i,j}^{+}$.

Case c) will occur when $f_{i,j}^{+} > 0$, $f_{i,k}^{+} = 0$, and $f_{j,k}^{+} > 0$ when either $M_{P_j} > M_{P_k}$ or $M_{P_j} = M_{P_k}$.

Case d) will occur when $f_{i,k}^{+} > 0$, $f_{i,j}^{+} = 0$, and $f_{j,k}^{+} > 0$ when $M_{P_j} = M_{P_k}$.

Case e) will occur when $f_{i,j}^{+} > 0$ and $f_{i,k}^{+} = 0$

Case f) will occur when $f_{i,j}^{+} = 0$ and $f_{i,k}^{+} > 0$

Case g) will occur when $f_{i,j}^{+} = 0$ and $f_{i,k}^{+} = 0$

//TODO: finish this

Construction of semantic properties

There are set of optimization problems which are related to the construction of new properties. Those are posed below:

Let us have a given set of semantic aspect types $l_1, l_2, \dots, l_k$. We would like to construct a new property P which can be bound to a given set of primitive semantic particles $V_1, V_2, \dots, V_m$ in an enclosing semantic structure S. When adding a new property to a subset of primitive particles of S then obviously this will cause a displacement of the centroid of S. Let us denote the displacement of the centroid of S as a result of the introduction of the new property P with $\Delta\vec{r}_c(S, P)$. Let us denote by $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(k)}$ the positions of the k semantic aspect values which correspond to the new aspect types $l_1, l_2, \dots, l_k$. We would like to obtain the values $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(k)}$ of the k semantic aspects based on the minimization of certain cost function.

Problem 1: Determine the values $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(k)}$ of the semantic aspects given by their types $l_1, l_2, \dots, l_k$ in the new property such that the centroid of the enclosing semantic structure S is moved by the least amount from its original position before the introduction of the new property. Thus, we have:

$$\min_{\boldsymbol{\theta}^{(1)},\dots,\boldsymbol{\theta}^{(k)}} |\Delta\vec{r}_c| \qquad \text{(A.1)}$$

Problem 2: Determine the values $\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, \dots, \boldsymbol{\theta}^{(k)}$ of the semantic aspects given by their types $l_1, l_2, \dots, l_k$ in the new property such that the centroid of the enclosing semantic structure S is moved in a direction which is as close as possible to some given direction $\vec{D}_0$ in semantic space. Then, we want to minimize the angle between $\Delta\vec{r}_c$ and $\vec{D}_0$ which is equivalent to maximizing $\frac{\vec{D}_0\,\Delta\vec{r}_c}{|\vec{D}_0||\Delta\vec{r}_c|}$:

$$\max_{\boldsymbol{\theta}^{(1)},\dots,\boldsymbol{\theta}^{(k)}} \frac{\vec{D}_0\,\Delta\vec{r}_c}{|\vec{D}_0||\Delta\vec{r}_c|} \qquad \text{(A.2)}$$

Related to the second optimization problem is maximization of the displacement of the centroid of the enclosing semantic structure S in the given direction $\vec{D}_0$. In this case we want to maximize:

$$\max_{\boldsymbol{\theta}^{(1)},\dots,\boldsymbol{\theta}^{(k)}} \frac{\vec{D}_0\,\Delta\vec{r}_c}{|\vec{D}_0|} \qquad \text{(A.3)}$$

A more general weighted objective function, linear combination of (A.2) and (A.3) would be

$$\max_{\boldsymbol{\theta}^{(1)},\dots,\boldsymbol{\theta}^{(k)}} W_1 \frac{\vec{D}_0\,\Delta\vec{r}_c}{|\vec{D}_0||\Delta\vec{r}_c|} + W_2 \frac{\vec{D}_0\,\Delta\vec{r}_c}{|\vec{D}_0|} \qquad \text{(A.4)}$$

where $W_1 = W_1(C)$ and $W_2 = W_2(C)$ depend on the enclosing the semantic structure S context C.

And here there is a variation of the previous problems:

In all of the following cases we have a given set of aspect types $l_1, l_2, \dots, l_k$ and we want to find an extra set of types $l_{k+1}, l_{k+2}, \dots, l_m$ which

Problem 3:

//TODO: finish this