Incremental semantic scales by strings

Tim Fernando
Computer Science Department
Trinity College Dublin
Dublin, Ireland
Tim.Fernando@tcd.ie

Abstract

Scales for natural language semantics are analyzed as moving targets, perpetually under construction and subject to adjustment. Projections, factorizations and constraints are described on strings of bounded but refinable granularities, shaping types by the processes that put semantics in flux.

1 Introduction

An important impetus for recent investigations into type theory for natural language semantics is the view of "semantics in flux," correcting "the impression" from, for example, Montague 1973 "of natural languages as being regimented with meanings determined once and for all" (Cooper 2012, page 271). The present work concerns scales for temporal expressions and gradable predicates. Two questions that loom large from the perspective of semantics in flux are: how to construct scales and align them against one another (e.g. Klein and Rovatsos 2011). The formal study carried out below keeps scales as simple as possible, whilst allowing for necessary refinements and adjustments. The basic picture is that a scale is a moving target finitely approximable as a string over an alphabet which we can expand to refine granularity. Reducing a scale to a string comes, however, at a price; indivisible points must give way to refinable intervals (embodying underspecification).

Arguments for a semantic reorientation around intervals (away from points) are hardly new. Best known within linguistic semantics perhaps are those in tense and aspect from Bennett and Partee 1972, which seem to have met less resistance than arguments in the degree literature from Kennedy 2001 and Schwarzschild and Wilkinson 2002 (see Solt 2013). At the center of the present argument for intervals is a notion of finite approximability, plausibly related to cognition. What objection might there be to it? The fact that no finite linear order is dense raises the issue of compatibility between finite approximability and density — no small worry, given the popularity of dense linear orders for time (e.g. Kamp and Reyle 1993, Pratt-Hartmann 2005, Klein 2009) as well as measurement (e.g. Fox and Hackl 2006).

Fortunately, finite linear orders can be organized into a system of approximations converging at the limit to a dense linear order. The present work details ways to form such systems and limits, with density reanalyzed as refinability of arbitrary finite approximations. A familiar example provides some orientation.

Example A (calendar) We can represent a calendar year as the string

$$s_{mo} := \boxed{\text{Jan}}\boxed{\text{Feb}}\boxed{\text{Mar}}\cdots\boxed{\text{Dec}}$$

of length 12, or, were we interested also in days d1,d2. . .,d31, the string

$$s_{mo,dy} := \boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}}\cdots\boxed{\text{Jan,d31}}$$
$$\boxed{\text{Feb,d1}}\cdots\boxed{\text{Dec,d31}}$$

of length 365 for a non-leap year (Fernando 2011).[1] In contrast to the points in the real line $\mathbb{R}$, a box can split, as $\boxed{\text{Jan}}$ in s_{mo} does (30 times) to

$$\boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}}\cdots\boxed{\text{Jan,d31}}$$

in $s_{mo,dy}$, on introducing days d1, d2,. . ., d31 into the picture. Reversing direction and generalizing from

$$mo := \{\text{Jan,Feb,}\ldots\text{Dec}\}$$

[1] We draw boxes (instead of the usual curly braces { and }) around sets-as-symbols, stringing together "snapshots" much like a cartoon/film strip.

Proceedings of the EACL 2014 Workshop on Type Theory and Natural Language Semantics (TTNLS), pages 63–71,
Gothenburg, Sweden, April 26-30 2014.

to any set A, we define the function ρ_A on strings (of sets) to componentwise intersect with A

$$\rho_A(\alpha_1 \cdots \alpha_n) \ := \ (\alpha_1 \cap A) \cdots (\alpha_n \cap A)$$

(throwing out non-A's from each box) so that

$$\rho_{mo}(s_{mo,dy}) \ = \ \boxed{\text{Jan}}^{31} \boxed{\text{Feb}}^{28} \cdots \boxed{\text{Dec}}^{31}.$$

Next, the *block compression* $bc(s)$ *of* a string s compresses all repeating blocks α^n (for $n \geq 1$) of a box α in a string s to α for

$$bc(s) \ := \ \begin{cases} bc(\alpha s') & \text{if } s = \alpha\alpha s' \\ \alpha \, bc(\beta s') & \text{if } s = \alpha\beta s' \text{ with } \\ & \qquad \alpha \neq \beta \\ s & \text{otherwise} \end{cases}$$

so that if $bc(s) = \alpha_1 \cdots \alpha_n$ then $\alpha_i \neq \alpha_{i+1}$ for i from 1 to $n-1$. In particular,

$$bc(\boxed{\text{Jan}}^{31} \boxed{\text{Feb}}^{28} \cdots \boxed{\text{Dec}}^{31}) \ = \ s_{mo}.$$

Writing bc_A for the function mapping s to $bc(\rho_A(s))$, we have

$$bc_{mo}(s_{mo,dy}) \ = \ s_{mo}.$$

In general, we can refine a string s_A of granularity A to one $s_{A'}$ of granularity $A' \supseteq A$ with $bc_A(s_{A'}) = s_A$. Iterating over a chain

$$A \subseteq A' \subseteq A'' \subseteq \cdots ,$$

we can glue together strings $s_A, s_{A'}, s_{A''}, \ldots$ such that

$$bc_X(s_{X'}) \ = \ s_X \ \text{ for } X \in \{A, A', A'', \ldots\}.$$

Details in section 2.

We shall refer to the expressions we can put in a box as *fluents* (short for temporal propositions), and assume they are the elements of a set Φ. While the set Φ of fluents might be infinite, we restrict the boxes that we string together to finite sets of fluents. Writing $Fin(\Phi)$ for the set of finite subsets of Φ and 2^X for the powerset of X (i.e. the set of X's subsets), we will organize the strings over the infinite alphabet $Fin(\Phi)$ around finite alphabets 2^A, for $A \in Fin(\Phi)$

$$Fin(\Phi)^* \ = \bigcup_{A \in Fin(\Phi)} (2^A)^*.$$

In addition to projecting $Fin(\Phi)$ down to 2^A for some $A \in Fin(\Phi)$, we can build up, forming the componentwise unions of strings $\alpha_1 \cdots \alpha_n$ and $\beta_1 \cdots \beta_n$ of the same number n of sets for their *superposition*

$$\alpha_1 \cdots \alpha_n \,\&\, \beta_1 \cdots \beta_n \ := \ (\alpha_1 \cup \beta_1) \cdots (\alpha_n \cup \beta_n)$$

and superposing languages L and L' over $Fin(\Phi)$ by superposing strings in L and L' of the same length

$$L \,\&\, L' \ := \ \{s\&s' \mid s \in L, \ s' \in L' \text{ and } \\ \text{length}(s) = \text{length}(s')\}$$

(Fernando 2004). For example,

$$s_{mo,dy} \ = \ \rho_{mo}(s_{mo,dy}) \,\&\, \rho_{dy}(s_{mo,dy})$$

where $dy := \{\mathsf{d1}, \mathsf{d2} \ldots, \mathsf{d31}\}$. More generally, writing L_A for the image of L under ρ_A

$$L_A \ := \ \{\rho_A(s) \mid s \in L\},$$

observe that for $L \subseteq (2^B)^*$ and $A \subseteq B$, L is included in the superposition of L_A and L_{B-A}

$$L \ \subseteq \ L_A \,\&\, L_{B-A}.$$

The next step is to identify a language L' such that

$$L \ = \ (L_A \,\&\, L_{B-A}) \cap L' \qquad (1)$$

other than $L' = L$. For a decomposition (1) of L into (generic) contextual constraints L' separate from the (specific) components L_A and L_{B-A}, it will be useful to sharpen L_A, L_{B-A} and &, factoring in bc and variants of bc (not to mention $\cap$). Measurements ranging from crude comparisons (of order) to quantitative judgments (multiplying unit magnitudes with real numbers) can be expressed through fluents in Φ. We interpret the fluents relative to suitable strings in $Fin(\Phi)^*$, presented below in category-theoretic terms connected to type theory (e.g. Mac Lane and Moerdijk 1992). Central to this presentation is the notion of a *presheaf on* $Fin(\Phi)$ — a functor from the opposite category $Fin(\Phi)^{op}$ (a morphism in which is a pair (B, A) of finite subsets of Φ such that $A \subseteq B$) to the category **Set** of sets and functions. The $Fin(\Phi)$-indexed family of functions bc_A (for $A \in Fin(\Phi)$) provides an important example that we generalize in section 2.

An example of linguistic semantic interest to which block compression bc applies is

Example B (continuous change) The pair (a), (b) below superposes two events, soup cooling and an hour passing, in different ways (Dowty 1979).

(a) The soup cooled in an hour.

(b) The soup cooled for an hour.

A common intuition is that *in an hour* requires an event that culminates, while *for an hour* requires a homogeneous event. In the case of (a), the culmination may be that some threshold temperature (supplied by context) was reached, while in (b), the homogeneity may be the steady drop in temperature over that hour. We might track soup cooling by a descending sequence of degrees, $d_1 > d_2 > \cdots > d_n$, with d_1 at the beginning of the hour, and d_n at the end. But *no* sample of finite size n can be complete. To overcome this limitation, it is helpful to construe the ith box in a string as a description of an interval I_i over the real line $\mathbb{R}$. We call a sequence $I_1 \cdots I_n$ of intervals a *segmentation* if $\bigcup_{i=1}^{n} I_i$ is an interval and for $1 \leq i < n$, $I_i < I_{i+1}$, where $<$ is *full precedence*

$$I < I' \quad \text{iff} \quad (\forall r \in I)(\forall r' \in I')\ r < r'.$$

Now, assuming an assignment of degrees $sDg(r)$ to real numbers r representing temporal instants, the idea is to define satisfaction $\models$ between intervals I and fluents $sDg < d$ according to

$$I \models sDg < d \quad \text{iff} \quad (\forall r \in I)\ sDg(r) < d$$

and similarly for $d \leq sDg$. We then lift $\models$ to segmentations $I_1 \cdots I_n$ and strings $\alpha_1 \cdots \alpha_n \in Fin(\Phi)^n$ of the same length n such that

$$I_1 \cdots I_n \models \alpha_1 \cdots \alpha_n \quad \text{iff} \quad \text{whenever } 1 \leq i \leq n \text{ and } \varphi \in I_i,\ I_i \models \varphi_i$$

and analyze (a) above as (c) below, where d is the contextually given threshold required by *in an hour*, and x is the start of that hour, the end of which is marked by hour(x).

(c) $\boxed{x, d \leq sDg}\boxed{d \leq sDg}\boxed{\text{hour}(x), sDg < d}$

All fluents φ in (c) have the stative property

(†) for all intervals I and I' whose union $I \cup I'$ is an interval,

$$I \cup I' \models \varphi \quad \text{iff} \quad I \models \varphi \text{ and } I' \models \varphi$$

(Dowty 1979). (†) holds also for the fluents in the string (d) below for (b), where the subinterval relation $\sqsubseteq$ is inclusion restricted to intervals,

$$I \models [\sqsupseteq]\varphi \quad \text{iff} \quad (\forall I' \sqsubseteq I)\ I' \models \varphi$$

and $sDg_{\downarrow}$ is the fluent

$$\exists x\ (sDg < x \land \text{Prev}(x \leq sDg))$$

saying the degree drops (with $I \models \text{Prev}(\varphi)$ iff $I'I \models \boxed{\varphi}\boxed{}$ for some $I' < I$ such that $I \cup I'$ is an interval).

(d) $\boxed{x}\boxed{[\sqsupseteq]sDg_{\downarrow}}\boxed{\text{hour}(x), [\sqsupseteq]sDg_{\downarrow}}$

(†) is intimately related to block compression bc (Fernando 2013b), supporting derivations of (c) and (d) by a modification $\&_{bc}$ of $\&$ defined in §2.3 below.

Our third example directly concerns computational processes, which we take up in section 3.

Example C (finite automata) Given a finite alphabet A, a (non-deterministic) *finite automaton* $\mathcal{A}$ *over* A is a quadruple (Q, δ, F, q_0) consisting of a finite set Q of states, a transition relation $\delta \subseteq Q \times A \times Q$, a subset F of Q consisting of final (accepting) states, and an initial state $q_0 \in Q$. $\mathcal{A}$ *accepts* a string $a_1 \cdots a_n \in A^*$ precisely if there is a string $q_1 \cdots q_n \in Q^n$ such that

$$q_n \in F \text{ and } \delta(q_{i-1}, a_i, q_i) \text{ for } 1 \leq i \leq n \qquad (2)$$

(where q_0 is $\mathcal{A}$'s designated initial state). The *accepting runs of* $\mathcal{A}$ are strings of the form

$$\boxed{a_1, q_1} \cdots \boxed{a_n, q_n} \ \in\ (2^{A \cup Q})^*$$

satisfying (2). While we can formulate such runs as strings over the alphabet $A \times Q$, we opt for the alphabet $2^{A \cup Q}$ (formed from $A \cup Q \in Fin(\Phi)$) to link up smoothly with examples where more than one automata may be running, not all necessarily known nor in perfect harmony with others. Such examples are arguably of linguistic interest, the so-called *Imperfective Paradox* (Dowty 1979) being a case in point (Fernando 2008). That said, the attention below is largely on certain category-theoretic preliminaries for type theory.[2]

We adopt the following notational conventions. Given a function f and a set X, we write

[2]Only the most rudimentary category-theoretic notions are employed; explanations can be found in any number of introductions to category theory available online (and in print).

- $f \upharpoonright X$ for f restricted to $X \cap \mathit{domain}(f)$
- $\mathit{image}(f)$ for $\{f(x) \mid x \in \mathit{domain}(f)\}$
- fX for $\mathit{image}(f \upharpoonright X)$
- $f^{-1}X$ for $\{x \in \mathit{domain}(f) \mid f(x) \in X\}$

and if g is a function for which $\mathit{image}(f) \subseteq \mathit{domain}(g)$,

- $f; g$ for f composed (left to right) with g

  $$(f; g)(x) \ := \ g(f(x))$$

  for all $x \in \mathit{domain}(f)$.

We say f is a function *on* X if

$$\mathit{domain}(f) = X \supseteq \mathit{image}(f)$$

— i.e., $f : X \to X$. The *kernel of* f, $\mathit{ker}(f)$, is the equivalence relation on $\mathit{domain}(f)$ that holds between s, s' such that $f(s) = f(s')$. Clearly,

$$\mathit{ker}(f) \ \subseteq \ \mathit{ker}(f; g)$$

when $f; g$ is defined.

2 Some presheaves on $Fin(\Phi)$

Given a function f on $Fin(\Phi)^*$ and $A \in Fin(\Phi)$, let us write f_A for the function $\rho_A; f$ on $Fin(\Phi)^*$

$$f_A(s) \ := \ f(\rho_A(s))$$

(recalling $\rho_A(\alpha_1 \cdots \alpha_n) := (\alpha_1 \cap A) \cdots (\alpha_n \cap A)$ and generalizing bc_A from Example A). To extract a presheaf on $Fin(\Phi)$ from the $Fin(\Phi)$-indexed family of functions f_A, certain requirements on f are helpful. Toward that end, let us agree that

- f *preserves* a function g with domain $Fin(\Phi)^*$ if $g = f; g$
- f is *idempotent* if f preserves itself (i.e., $f = f; f$)
- the *vocabulary* $\mathit{voc}(s)$ *of* $s \in Fin(\Phi)^*$ is the set of fluents that occur in s

  $$\mathit{voc}(\alpha_1 \cdots \alpha_n) \ := \ \bigcup_{i=1}^{n} \alpha_i$$

  whence $s \in \mathit{voc}(s)^*$.

Note that for idempotent f, $\mathit{image}(f)$ consists of canonical representatives $f(s)$ of $\mathit{ker}(f)$'s equivalence classes $\{s' \in Fin(\Phi)^* \mid f(s') = f(s)\}$.

2.1 Φ-preserving functions

A function f on $Fin(\Phi)^*$ is Φ*-preserving* if f preserves voc and f_A, for all $A \in Fin(\Phi)$. Note that bc is Φ-preserving, as is the identity function id on $Fin(\Phi)^*$.

Proposition 1. *If f is Φ-preserving then f is idempotent and*

$$f_B; f_A \ = \ f_{A \cap B}$$

for all $A, B \in Fin(\Phi)$.

Let $\mathbf{P}_f$ be the function with domain

$$Fin(\Phi) \cup \{(B, A) \in Fin(\Phi) \times Fin(\Phi) \mid A \subseteq B\}$$

mapping $A \in Fin(\Phi)$ to $f(2^A)^*$

$$\mathbf{P}_f(A) \ := \ \{f(s) \mid s \in (2^A)^*\}$$

and a $Fin(\Phi)^{op}$-morphism (B, A) to the restriction of f_A to $\mathbf{P}_f(B)$

$$\mathbf{P}_f(B, A) \ := \ f_A \upharpoonright \mathbf{P}_f(B).$$

Corollary 2. *If f is Φ-preserving then $\mathbf{P}_f$ is a presheaf on $Fin(\Phi)$.*

Apart from bc, we get a Φ-preserving function by stripping off any initial or final empty boxes $\square$

$$\mathit{unpad}(s) \ := \ \begin{cases} \mathit{unpad}(s') & \text{if } s = \square s' \text{ or} \\ & \quad \text{else } s = s'\square \\ s & \text{otherwise} \end{cases}$$

so that $\mathit{unpad}(s)$ neither begins nor ends with $\square$. Notice that $\mathit{bc}; \mathit{unpad} = \mathit{unpad}; \mathit{bc}$.

Proposition 3. *If f and g are Φ-preserving and $f; g = g; f$, then $f; g$ is Φ-preserving.*

2.2 The Grothendieck construction

Given a presheaf F on $Fin(\Phi)$, the *category* $\int F$ *of elements of* F (also known as the *Grothendieck construction* for F) has

- objects $(A, s) \in Fin(\Phi) \times F(A)$ (making $\sum_{X \in Fin(\Phi)} F(X)$ the set of objects in $\int F$)
- morphisms (B, s', A, s) from objects (B, s') to (A, s) when $A \subseteq B$ and $F(B, A)(s') = s$

(e.g. Mac Lane and Moerdijk 1992). Let π_f be the left projection

$$\pi_f(A, s) \ = \ A$$

from $\int \mathbf{P}_f$ back to $Fin(\Phi)$. The *inverse limit of* $\mathbf{P}_f$, $\varprojlim \mathbf{P}_f$, is the set of ($\int \mathbf{P}_f$)-valued presheaves p on $Fin(\Phi)$ (i.e. functors $p : Fin(\Phi)^{op} \to \int \mathbf{P}_f$) that are inverted by π_f

$$\pi_f(p(A)) \ = \ A \qquad \text{for all } A \in Fin(\Phi).$$

That is, $p(A) = (A, s_A)$ for some $s_A \in f(2^A)^*$ such that

($\ddagger$) $s_A = f_A(s_B)$ whenever $A \subseteq B \in Fin(\Phi)$.

($\ddagger$) is the essential restriction that $\varprojlim \mathbf{P}_f$ adds to objects $\{s_X\}_{X \in Fin(\Phi)}$ of the dependent type $\prod_{X \in Fin(\Phi)} \mathbf{P}_f(X)$.

2.3 Superposition and non-determinism

Taking the presheaf $\mathbf{P}_{id}$ induced by the identity function id on $Fin(\Phi)^*$, observe that in $\int \mathbf{P}_{id}$, there is a product of

$$(\emptyset, \boxed{}\boxed{}) \text{ and } (\{\varphi\}, \boxed{\varphi}\boxed{})$$

but not of

$$(\{\varphi\}, \boxed{}\boxed{}) \text{ and } (\{\varphi\}, \boxed{\varphi}\boxed{}).$$

The tag A in (A, s) differentiating $(\emptyset, \boxed{}\boxed{})$ from $(\{\varphi\}, \boxed{}\boxed{})$ cannot be ignored when forming products in $\int \mathbf{P}_{id}$. A necessary and sufficient condition for (A, s) and (B, s') to have a product is

$$\rho_B(s) \ = \ \rho_A(s')$$

presupposed by the pullback of

$$(A, s) \ \to \ (A \cap B, \rho_B(s)) \ \leftarrow \ (B, s').$$

By comparison, the superposition $s \& s'$ exists (as a string) if and only if

$$\rho_\emptyset(s) \ = \ \rho_\emptyset(s')$$

for

$$(voc(s), s) \ \to \ (\emptyset, \rho_\emptyset(s)) \ \leftarrow \ (voc(s'), s')$$

(or length(s) = length(s') as $\rho_\emptyset(s) = \Box^{length(s)}$). Products in $\int \mathbf{P}_{id}$ are superpositions, but superpositions need not be products.

Next, we step from id to other Φ-preserving functions f such as bc and $bc; unpad$. A pair (A, s) and (B, s') of $\int \mathbf{P}_f$-objects may fail to have a product *not* because there is *no* $\int \mathbf{P}_f$-object $(A \cup B, s'')$ such that

$$(A, s) \ \leftarrow \ (A \cup B, s'') \ \to \ (B, s')$$

but too many non-isomorphic choices for such s''. Consider the case of bc; *unpad*, with $(\emptyset, \epsilon)$ terminal in $\int \mathbf{P}_{bc;unpad}$ (where ϵ is the null string of length 0). For distinct fluents a and $b \in \Phi$, there are 13 strings $s \in \mathbf{P}_{bc;unpad}(\{a, b\})$ such that

$$(\{a\}, \boxed{a}) \leftarrow (\{a, b\}, s) \rightarrow (\{b\}, \boxed{b})$$

corresponding to the 13 interval relations in Allen 1983 (Fernando 2007).

The explosion of solutions $s'' \in \mathbf{P}_f(A \cup B)$ to the equations

$$f_A(s'') = s \ \text{ and } \ f_B(s'') = s'$$

given

$$(A, s) \ \to \ (A \cap B, f_B(s)) \ \leftarrow \ (B, s')$$

(i.e., $f_B(s) = f_A(s')$) is paralleled by the transformation, under f, of a language L to

$$L_f \ := \ f^{-1} f L$$

used to turn the superposition $L \& L'$ of languages L and L' into

$$L \, \&_f \, L' \ := \ f(L_f \, \& \, L'_f).$$

For $f := bc; unpad$, the set $\boxed{a} \, \&_f \, \boxed{b}$ consists of the 13 strings mentioned above. (We follow the usual practice of conflating a string s with the singleton language $\{s\}$ whenever convenient.)

Stepping from strings to languages, we lift the presheaf $\mathbf{P}_f$ to the presheaf $\mathbf{Q}_f$ mapping $A \in Fin(\Phi)$ to

$$\mathbf{Q}_f(A) \ := \ \{fL \mid L \subseteq (2^A)^*\}$$

and a $Fin(\Phi)^{op}$-morphism (B, A) to the function

$$\mathbf{Q}_f(B, A) \ := \ (\lambda L \in \mathbf{Q}_f(B)) \ f_A L$$

sending $L \in \mathbf{Q}_f(B)$ to $f_A L \in \mathbf{Q}_f(A)$. Then, for non-identity morphisms between $\int \mathbf{Q}_f$-objects (A, L) and (A, L') where $L \subseteq L'$, we add inclusions from (A, L) to (A, L') to the $\int \mathbf{Q}_f$-morphisms for the category $\mathfrak{C}(\Phi, f)$ with

- objects the same as those in $\int \mathbf{Q}_f$, and
- morphisms (B, L', A, L) from objects (B, L') to (A, L) whenever $A \subseteq B$ and $f_A L' \subseteq L$.

As is the case with $\int \mathbf{Q}_f$-morphisms, the sources (domains) of $\mathfrak{C}(\Phi, f)$-morphisms entail their targets (codomains). To make these entailments precise, we can identify the space of possible worlds with the inverse limit of $\mathbf{P}_f$, and reduce (A, L) to

$$\begin{aligned} [\![A, L]\!]_f \ := \{p \in \varprojlim \mathbf{P}_f \mid \\ (\exists s \in L)\ p(A) = (A, s)\}. \end{aligned}$$

The inclusion

$$[\![B, L']\!]_f \ \subseteq \ [\![A, L]\!]_f$$

can then be pronounced: (B, L') *f-entails* (A, L).

Proposition 4. *Let f be a Φ-preserving function and (A, L) and (B, L') be $\int \mathbf{Q}_f$-objects such that $A \subseteq B$. (B, L') f-entails (A, L) iff there is a $\mathfrak{C}(\Phi, f)$-morphism from (B, L') to (A, L).*

Relaxing the assumption $A \subseteq B$, one can also check that for $f \in \{\mathit{bc}, \mathit{unpad}, (\mathit{bc}; \mathit{unpad})\}$, pullbacks of

$$(A, L) \to (A \cap B, (f_\emptyset L) \cap f_\emptyset L') \leftarrow (B, L')$$

in $\mathfrak{C}(\Phi, f)$ are given by

$$(A, L) \leftarrow (A \cup B, L \&_f L') \to (B, L') \qquad (3)$$

although (3) need not hold for $L \&_f L'$ to be well-defined.

3 Constraints and finite automata

We now bring finite automata into the picture, recalling from section 1 Example C's superpositions

$$\boxed{a_1}\cdots\boxed{a_n} \ \& \ \boxed{q_1}\cdots\boxed{q_n} \qquad (4)$$

where $a_1 \cdots a_n$ is accepted by a finite automaton $\mathcal{A}$ going through the sequence $q_1 \cdots q_n$ of (internal) states. We can assume the tape alphabet $A \supseteq \{a_1, \ldots, a_n\}$ and the state set $Q \supseteq \{q_1, \ldots, q_n\}$ are two disjoint subsets of the set Φ of fluents; fluents in A are "observable" (on a tape), while fluents in Q are "hidden" (inside a black box). Disjoint though they may be, A and Q are tightly coupled by $\mathcal{A}$'s transition table $\delta \subseteq Q \times A \times Q$ (not to mention the other components of $\mathcal{A}$, its initial and final states). That coupling can hardly be recreated by superposition & (or some simple modification $\&_f$) without the help of some machinery encoding δ. But first, there is the small matter of formulating the map $a_1 \cdots a_n \mapsto \boxed{a_1}\cdots\boxed{a_n}$ implicit in (4) above as a natural transformation.

3.1 Bottom $\perp$ naturally

If the function η_A such that for $a_1 \cdots a_n \in A^*$,

$$\eta_A(a_1 \cdots a_n) \ = \ \boxed{a_1}\cdots\boxed{a_n}$$

is to be the A-th component of a natural transformation $\eta : \mathbf{S} \Rightarrow \mathbf{P}_{id}$, we need to specify the presheaf $\mathbf{S}$ on $\mathit{Fin}(\Phi)$. To form a function $\mathbf{S}(B, A) : \mathbf{S}(B) \to \mathbf{S}(A)$ for $A \subseteq B \in \mathit{Fin}(\Phi)$ with $B^* \subseteq \mathbf{S}(B)$ and $A^* \subseteq \mathbf{S}(A)$, it is handy to introduce a bottom $\perp$ for $B - A$, adjoining $\perp$ to a finite subset X of Φ for $X_\perp := X + \{\perp\}$ before forming the strings in $\mathbf{S}(X) := {X_\perp}^*$. We then set $\mathbf{S}(B, A) : {B_\perp}^* \to {A_\perp}^*$

$$\mathbf{S}(B, A)(\epsilon) \ := \ \epsilon$$

$$\mathbf{S}(B, A)(\beta \mathsf{s}) \ := \ \begin{cases} \beta\, \mathbf{S}(B, A)(\mathsf{s}) & \text{if } \beta \in A_\perp \\ \perp \mathbf{S}(B, A)(\mathsf{s}) & \text{otherwise} \end{cases}$$

(e.g. $\mathbf{S}(\{a, b\}, \{a\})(ba\perp) = \perp a \perp$) and let $\eta_A : {A_\perp}^* \to (2^A)^*$ map ϵ to itself, and

$$\eta_A(\alpha \mathsf{s}) \ := \ \begin{cases} \boxed{}\, \eta_A(\mathsf{s}) & \text{if } \alpha = \perp \\ \boxed{\alpha}\, \eta_A(\mathsf{s}) & \text{otherwise} \end{cases}$$

(e.g. $\eta_{\{a\}}(\perp a \perp) = \boxed{}\boxed{a}\boxed{}$).

Proposition 5. *η is a natural transformation from $\mathbf{S}$ to $\mathbf{P}_{id}$.*

3.2 Another presheaf and category

Turning now to finite automata, we recall a fundamental result about languages that are regular (i.e., accepted by finite automata),[3] the Büchi-Elgot-Trakhtenbrot theorem (e.g. Thomas 1997)

> *for every finite alphabet $A \neq \emptyset$, a language $L \subseteq A^+$ is regular iff there is a sentence φ of MSO$_A$ such that*
>
> $$L \ = \ \{\mathsf{s} \in A^+ \mid \mathsf{s} \models_A \varphi\} \,.$$

MSO$_A$ is Monadic Second Order logic with a unary relation symbol U_a for each $a \in A$, plus a binary relation symbol S for successors. The predicate $\models_A$ treats a string $a_1 a_2 \cdots a_n$ over A as an MSO$_A$-model with universe $\{1, 2, \ldots, n\}$, U_a as its subset $\{i \mid a_i = a\}$, and S as

$$\{(1, 2), (2, 3), \ldots, (n-1, n)\}$$

[3] Whether or not this sense of *regular* has an interesting connection with regular categories (which are, among other things, finitely complete), I do not know.

so that, for instance,

$$a_1 \cdots a_n \models_A \exists x \exists y\, \mathsf{S}(x,y) \;\text{ iff }\; n \geq 2 \tag{5}$$

for all finite $A \neq \emptyset$. Notice that no $a \in A$ is required to interpret $\exists x \exists y\, \mathsf{S}(x,y)$, which after all is an MSO$_\emptyset$-sentence suited to strings $\perp^n \in \mathbf{S}(\emptyset)$. Furthermore, for $a \neq b$ and $\{a,b\} \subseteq A$,

$$\textit{no} \text{ string in } A^+ \text{ satisfies } \exists x\, \mathsf{U}_a(x) \wedge \mathsf{U}_b(x) \tag{6}$$

which makes it awkward to extend $\models_A$ to formulas with free variables (requiring variable assignments on top of strings in A^+).

A simple way to accommodate variables is to include them in A and interpret MSO$_A$-formulas not over A^+ but over $(2^A)^+$, lifting $\models_A$ from strings s over A to a predicate $\models^A$ on strings over 2^A such that

$$\mathsf{s} \models_A \varphi \;\text{ iff }\; \eta_A(\mathsf{s}) \models^A \varphi \tag{7}$$

for every MSO$_A$-sentence φ (Fernando 2013a). For all $s \in (2^A)^+$, we set

$$s \models^A \mathsf{S}(x,y) \;\text{ iff }\; \rho_{\{x,y\}}(s) \in \boxed{}^* \boxed{x}\boxed{y} \boxed{}^* \tag{8}$$

for $A \supseteq \{x,y\}$, and

$$s \models^A \mathsf{U}_a(x) \;\text{ iff }\; \rho_{\{a,x\}}(s) \in E_a \boxed{a,x} E_a \tag{9}$$

for $A \supseteq \{a,x\}$, where $E_a := (\boxed{} + \boxed{a})^*$. We must be careful to incorporate into the clauses defining $s \models^A \varphi$ the presupposition that each first-order variable x free in φ occurs uniquely in s — i.e. $s \models^A x = x$ where

$$s \models^A x = y \;\text{ iff }\; \rho_{\{x,y\}}(s) \in \boxed{}^* \boxed{x,y} \boxed{}^* \tag{10}$$

for $x, y \in A$. In particular, we restrict negation $\neg\varphi$ to strings $\models^A$-satisfying $x = x$, for each first-order variable x free in φ. We can then put

$$s \models^A \exists x\, \varphi \;\text{ iff }\; (\exists s')\, \rho_A(s') = \rho_A(s) \text{ and } s' \models^{A \cup \{x\}} \varphi$$

and similarly for second-order existential quantification. The equivalence (5) above then becomes

$$s \models^A \exists x \exists y\, \mathsf{S}(x,y) \;\text{ iff }\; \rho_\emptyset(s) \in \boxed{}\boxed{}^+ \tag{11}$$

and in place of (6), we have

$$s \models^A \exists x\, \mathsf{U}_a(x) \wedge \mathsf{U}_b(x) \;\text{ iff }\; \rho_{\{a,b\}}(s) \in (2^{\{a,b\}})^* \boxed{a,b} (2^{\{a,b\}})^* \tag{12}$$

for $a, b \in A$.

Working back from (7)

$$\mathsf{s} \models_A \varphi \;\text{ iff }\; \eta_A(\mathsf{s}) \models^A \varphi$$

to the Büchi-Elgot-Trakhtenbrot theorem, one can check that for every finite A and MSO$_A$-formula φ, the set

$$\mathcal{L}_A(\varphi) \;:=\; \{s \in (2^A)^+ \mid s \models^A \varphi\}$$

of strings over 2^A that $\models^A$-satisfy φ is regular, using the fact that for all $A' \subseteq A$, the restriction of $\rho_{A'}$ to $(2^A)^*$ is computable by a finite state transducer. But for $A \subseteq \Phi$,[4] $\rho_{A'} \upharpoonright (2^A)^*$ is just $\mathbf{P}_{id}(A, A')$. In recognition of the role of these functions in $\models^A$, we effectivize the presheaf $\mathbf{Q}_{id}$ from §2.3 as follows. Let $\mathbf{R}_\Phi$ be the presheaf on $Fin(\Phi)$ mapping

- $A \in Fin(\Phi)$ to the set of languages over the alphabet 2^A that are regular

$$\mathbf{R}_\Phi(A) \;:=\; \{L \in \mathbf{Q}_{id}(A) \mid L \text{ is regular}\}$$

and

- a $Fin(\Phi)^{op}$-morphism (B, A) to the restriction of $\mathbf{Q}_{id}(B, A)$ to $\mathbf{R}_\Phi(B)$

$$\mathbf{R}_\Phi(B, A) \;:=\; (\lambda L \in \mathbf{R}_\Phi(B))\, \rho_A L.$$

$\int \mathbf{R}_\Phi$-objects are then pairs (A, L) where $A \in Fin(\Phi)$ and L is a regular language over the alphabet 2^A, while $\int \mathbf{R}_\Phi$-morphisms are quadruples $(B, L, A, \rho_A L)$ from (B, L) to $(A, \rho_A L)$ for $A \subseteq B \in Fin(\Phi)$. To account for the Boolean operations in MSO (as opposed to the predications (8)– (10) involving ρ_A), we add inclusions for a category $\mathfrak{R}(\Phi)$ with

- the same objects as $\int \mathbf{R}_\Phi$
- morphisms all of those in $\mathfrak{C}(\Phi, id)$ between objects in $\int \mathbf{R}_\Phi$ — i.e., quadruples (B, L', A, L) such that $A \subseteq B \in Fin(\Phi)$, $L' \subseteq (2^B)^*$ is regular, $L \subseteq (2^A)^*$ is regular, and $\rho_A L' \subseteq L$.

Let us agree to write

$$(B, L') \rightsquigarrow (A, L)$$

[4] Note an MSO$_A$-formula φ is *not* strictly a fluent in Φ but is formed in part from fluents.

to mean (B, L', A, L) is a $\mathfrak{R}(\Phi)$-morphism. Clearly, for $s \in (2^A)^+$, $A' \subseteq A$ and $L \subseteq (2^{A'})^+$,

$$\rho_{A'}(s) \in L \quad \text{iff} \quad (A, \{s\}) \rightsquigarrow (A', L).$$

In particular, for $x \in A$ and $s \in (2^A)^+$,

$$s \models^A x = x \quad \text{iff} \quad (A, \{s\}) \rightsquigarrow (\{x\}, \boxed{}^* \boxed{x} \boxed{}^*)$$

and similarly for $x = x$ replaced by the different MSO$_A$-formulas specified in clauses (8)–(12) above. The MSO$_A$-sentence

$$\mathit{spec}(A) \;:=\; \forall x \bigvee_{a \in A} (\mathsf{U}_a(x) \wedge \bigwedge_{b \in A - \{a\}} \neg \mathsf{U}_b(x))$$

associating a unique $a \in A$ with each string position (presupposed in $\models_A$ but not in $\models^A$) fits the same pattern

$$\begin{aligned} s \models^A \mathit{spec}(A) \quad &\text{iff} \quad \rho_A(s) \in \{\boxed{a} \mid a \in A\}^+ \\ &\text{iff} \quad (A \cup \mathit{voc}(s), \{s\}) \rightsquigarrow \\ &\qquad (A, \{\boxed{a} \mid a \in A\}^+) \\ &\text{iff} \quad \rho_A(s) \in \eta_A A^+. \end{aligned}$$

Let us define a string $s \in \mathit{Fin}(\Phi)^+$ to be

- *A-specified* if $s \models^A \mathit{spec}(A)$
- *A-underspecified* if $\rho_A(s) \in \eta_A({A_\perp}^+ - A^+)$
- *A-overspecified* if $\rho_A(s) \notin \mathit{image}(\eta_A)$

so that for $a \neq a'$ and $A = \{a, a'\}$, $\boxed{a}\boxed{a}$ is A-specified, $\boxed{a}\boxed{}$ is A-underspecified, and $\boxed{a, a'}\boxed{a}$ is A-overspecified. Given a finite automaton $\mathcal{A}$ over A with set Q of states, its set $\mathit{AcRun}(\mathcal{A})$ of accepting runs (Example C) is both A-specified and Q-specified, provided $A \cap Q = \emptyset$ (and otherwise risks being A-overspecified). The language accepted by $\mathcal{A}$ is the η_A^{-1}-image of the language $\rho_A \mathit{AcRun}(\mathcal{A})$ that is Q-underspecified, in accordance with the intuition that the states are hidden. From the regularity of $\mathit{AcRun}(\mathcal{A})$, however, it is clear that we can make these states visible, with $\mathit{AcRun}(\mathcal{A})$ as the language accepted by a finite automaton $\mathcal{A}'$ (over $2^{A \cup Q}$) that may (or may not) have the same set Q of states.

The maps ρ_A and inclusions $\subseteq$ underlying the morphisms of $\mathfrak{R}(\Phi)$ represent the two ways information may grow from $\int \mathbf{R}_\Phi$-objects (A, L) to (B, L') — *expansively* with $A \subseteq B$ and $L = \rho_A L'$, and *eliminatively* with $L' \subseteq L$ and $A = B$. The same notion of f-entailment defined in §2.3 through the sets $[\![A, L]\!]_f$ applies, but we have been careful here to fix f to id, in view of

Proposition 6. *For $A \subseteq B \in \mathit{Fin}(\Phi)$, φ an MSO$_A$-formula and $s \in (2^B)^+$,*

$$s \models^B \varphi \quad \text{iff} \quad \rho_A(s) \models^A \varphi.$$

Proposition 6 says that $s \models^B \varphi$ depends only on the part $\rho_A(s)$ of s mentioned in φ. It is a particular instance of the *satisfaction condition* in *institutions*, expressing the invariance of truth under change of notation (Goguen and Burstall 1992). Proposition 6 breaks down if we replace ρ_A by bc_A or unpad_A, as can be seen with $A = \emptyset$, and $\varphi = \exists x \exists y\, \mathsf{S}(x, y)$, for which recall (11).

3.3 Varying grain and span

Troublesome as they are, the maps bc_A and unpad_A have some use. Just as we can vary temporal grain through bc (Examples A and B in section 1), we can vary temporal span through *unpad*. For instance, we can combine runs of automata $\mathcal{A}_1$ over A_1 and $\mathcal{A}_2$ over A_2 in

$$L(\mathcal{A}_1, \mathcal{A}_2) \;:=\; \mathit{AcRun}(\mathcal{A}_1) \,\&_{\mathit{unpad}}\, \mathit{AcRun}(\mathcal{A}_2)$$

with the subscript *unpad* on $\&$ relaxing the requirement that $\mathcal{A}_1$ and $\mathcal{A}_2$ start and finish together (running in lockstep throughout). For $i \in \{1, 2\}$, and Q_i the state set for $\mathcal{A}_i$,

$$\mathit{AcRun}(\mathcal{A}_i) \;=\; \mathit{unpad}_{A_i \cup Q_i} L(\mathcal{A}_1, \mathcal{A}_2)$$

assuming the sets A_1, A_2, Q_1 and Q_2 are pairwise disjoint. The disjointness assumption rules out any communication (or interference) between $\mathcal{A}_1$ and $\mathcal{A}_2$. As subsets of one large set Φ of fluents, however, it is perfectly natural for these sets to intersect (and communicate through a common vocabulary), and we might express very partial constraints involving them through, for example, MSO-formulas. Recalling the definition $\mathcal{L}_A(\varphi) := \{s \in (2^A)^+ \mid s \models^A \varphi\}$, we can rewrite the satisfaction condition

$$s \models^B \varphi \quad \text{iff} \quad f_A(s) \models^A \varphi$$

on MSO$_A$-formulas φ, $A \subseteq B \in \mathit{Fin}(\Phi)$ and $s \in (2^B)^+$ as

$$\mathcal{L}_B(\varphi) \;=\; \{s \in (2^B)^+ \mid f_A(s) \in \mathcal{L}_A(\varphi)\}.$$

This equation lifts any regular language $\mathcal{L}_A(\varphi)$ to a regular language $\mathcal{L}_B(\varphi)$, provided f is computed by a finite-state transducer (as in the case of bc or *unpad*). Inverse images under such relations are a useful addition to the stock of operations constituting MSO-formulas as well as regular expressions.

References

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *C. ACM*, 26(11):832–843.

Michael Bennett and Barbara Partee. 1972. Toward the logic of tense and aspect in English. Technical report, System Development Corporation, Santa Monica, California. Reprinted in Partee 2008.

Robin Cooper. 2012. Type theory and semantics in flux. In *Handbook of the Philosophy of Science*. Volume 14: Philosophy of Linguistics. pages 271–323.

David R. Dowty. 1979. *Word Meaning and Montague Grammar*. Reidel, Dordrecht.

Tim Fernando. 2004. A finite-state approach to events in natural language semantics. *J. Logic and Computation*, 14(1):79–92.

Tim Fernando. 2007. Observing events and situations in time. *Linguistics and Philosophy* 30(5):527–550.

Tim Fernando. 2008. Branching from inertia worlds. *J. Semantics* 25(3):321–344.

Tim Fernando. 2011. Regular relations for temporal propositions. *Natural Language Engineering* 17(2): 163–184.

Tim Fernando. 2013a. Finite state methods and description logics *Proceedings of the 11th International Conference on Finite State Methods and Natural Language Processing*, pages 63 – 71.

Tim Fernando. 2013b. Dowty's aspect hypothesis segmented. *Proceedings of the 19th Amsterdam Colloquium*, pages 107 – 114.

Danny Fox and Martin Hackl. 2006. The universal density of measurement. *Linguistics and Philosophy* 29(5): 537 – 586.

Joseph Goguen and Rod Burstall. 1992., Institutions: Abstract model theory for specification and programming, *J. ACM*, 39(1):95–146.

Hans Kamp and Uwe Reyle. 1993. *From Discourse to Logic*. Kluwer.

Christopher Kennedy. 2001. Polar opposition and the ontology of degrees. *Linguistics and Philosophy* 24. 33 – 70.

Ewan Klein and Michael Rovatsos. 2011. Temporal Vagueness, Coordination and Communication In *Vagueness in Communication 2009*, LNAI 6517, pages 108–126.

Wolfgang Klein. 2009. How time is encoded. In W. Klein and P. Li, editors, *The Expression of Time*, pages 39 – 81, Mouton De Gruyter.

Saunders Mac Lane and Ieke Moerdijk. 1992. *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. Springer.

Richard Montague. 1973. The proper treatment of quantification in ordinary English. In *Approaches to Natural Language*, pages 221 – 42. D. Reidel, Dordrecht.

Ian Pratt-Hartmann. 2005. Temporal prepositions and their logic. *Artificial Intelligence* 166: 1–36.

Roger Schwarzschild and Karina Wilkinson. 2002. Quantifiers in comparatives: A semantics of degree based on intervals. *Natural Language Semantics* 10(1):1–41.

Stephanie Solt. 2013. Scales in natural language. Manuscript.

Wolfgang Thomas. 1997. Languages, automata and logic. In *Handbook of Formal Languages: Beyond Words*, volume 3, pages 389 – 455. Springer-Verlag.