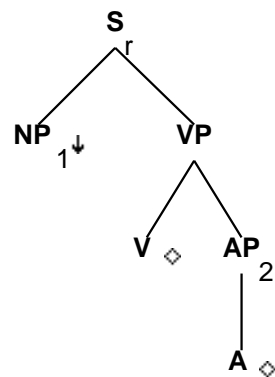


# Family "TREnx1VA2"

March 5, 2008

## 1 Tree "alphaREnx1VA2"

### 1.1 graphe



### 1.2 comments

Adjectival resultative w/ ergative verb:

'The river froze solid'

### 1.3 features

```
S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<comp> = nil

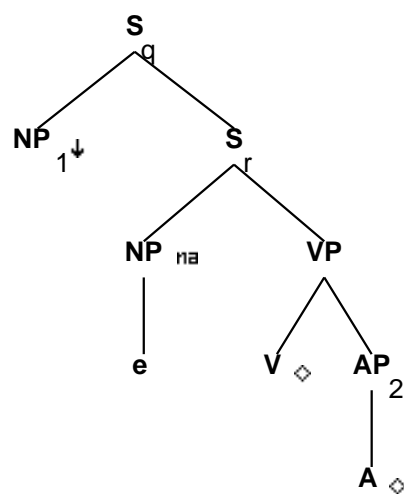
S_r.b:<mode> = VP.t:<mode>
S_r.b:<tense> = VP.t:<tense>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>
```

S\_r.b:<wh> = NP\_1.t:<wh>  
 S\_r.b:<control> = NP\_1.t:<control>  
 NP\_1.t:<wh> = -  
 NP\_1.t:<agr> = S\_r.b:<agr>  
 NP\_1.t:<case> = S\_r.b:<assign-case>

VP.b:<compar> = -  
 VP.b:<passive> = V.t:<passive>  
 VP.b:<agr> = V.t:<agr>  
 VP.b:<assign-case> = V.t:<assign-case>  
 VP.b:<assign-comp> = V.t:<assign-comp>  
 VP.b:<tense> = V.t:<tense>  
 VP.b:<mode> = V.t:<mode>  
 VP.b:<mainv> = V.t:<mainv>  
 V.t:<passive> = -  
 V.t:<punct struct> = nil  
 AP\_2.b:<wh> = A.t:<wh>  
 AP\_2.b:<compar> = A.t:<compar>  
 AP\_2.b:<equiv> = A.t:<equiv>

## 2 Tree "alphaREW1nx1VA2"

### 2.1 graphe



### 2.2 comments

Wh on the subject of an adjectival resultative w/ ergative verb:

'What froze solid'

## 2.3 features

```
S_q.b:<comp> = nil
S_q.b:<extracted> = +

S_q.b:<mode> = S_r.t:<mode>
S_q.b:<inv> = S_r.t:<inv>
S_q.b:<wh> = NP_1.t:<wh>
NP_1.t:<wh> = +
NP_1.t:<wh> = NP.t:<wh>
NP_1.t:<trace> = NP.t:<trace>
NP_1.t:<agr> = NP.t:<agr>
NP_1.t:<case> = NP.t:<case>
S_r.t:<comp> = nil

S_r.t:<conj> = nil
S_r.b:<assign-comp> = inf_nil/ind_nil/ecm

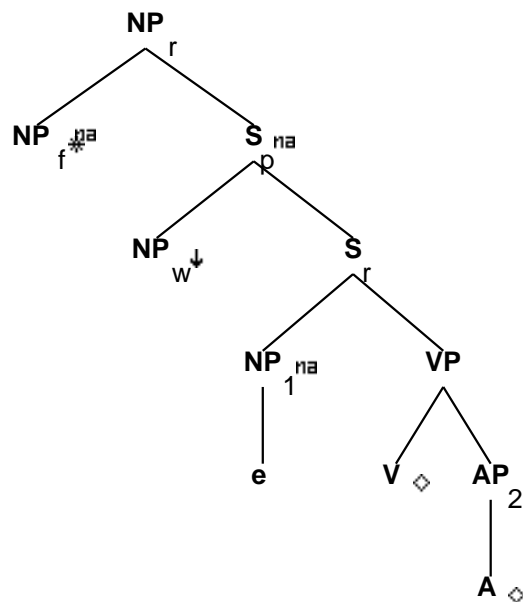
S_r.b:<comp> = nil
S_r.b:<inv> = -

S_r.b:<mode> = VP.t:<mode>
S_r.b:<tense> = VP.t:<tense>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<assign-case> = NP.t:<case>
S_r.b:<agr> = NP.t:<agr>

VP.b:<compar> = -
VP.b:<passive> = V.t:<passive>
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<tense> = V.t:<tense>
VP.b:<mode> = V.t:<mode>
VP.b:<mainv> = V.t:<mainv>
V.t:<punct struct> = nil
V.t:<passive> = -
AP_2.b:<wh> = A.t:<wh>
AP_2.b:<compar> = A.t:<compar>
AP_2.b:<equiv> = A.t:<equiv>
```

### 3 Tree "betaREN1nx1VA2"

#### 3.1 graphe



#### 3.2 comments

(wh) Relative clause on subject of  
ergative adjectival resultative:

'(I saw) the river which froze solid'

#### 3.3 features

NP\_r.b:<rel-clause> = +  
NP\_r.b:<wh> = NP\_f.t:<wh>  
NP\_r.b:<agr> = NP\_f.t:<agr>  
NP\_r.b:<case> = NP\_f.t:<case>

NP\_r.b:<pron> = NP\_f.t:<pron>  
NP\_f.b:<case> = nom/acc  
NP\_f.b:<refl> = -  
NP\_w.t:<wh> = +  
NP\_w.t:<trace> = NP\_1.t:<trace>  
NP\_w.t:<case> = NP\_1.t:<case>  
NP\_w.t:<agr> = NP\_1.t:<agr>

S\_r.t:<conj> = nil  
S\_r.t:<comp> = nil  
S\_r.t:<inv> = -  
S\_r.t:<mode> = ind

```

S_r.b:<comp> = nil

S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<tense> = VP.t:<tense>
S_r.b:<mode> = VP.t:<mode>

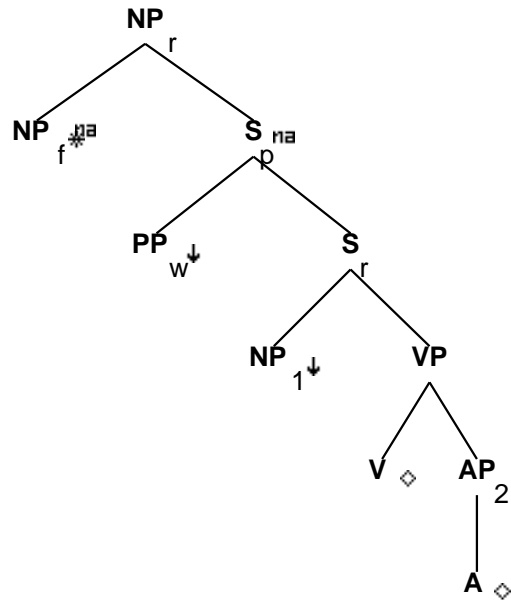
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<agr> = NP_1.t:<agr>
S_r.b:<assign-case> = NP_1.t:<case>

VP.b:<compar> = -
VP.b:<agr> = V.t:<agr>
VP.b:<passive> = V.t:<passive>
VP.b:<tense> = V.t:<tense>
VP.b:<mode> = V.t:<mode>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mainv> = V.t:<mainv>
V.t:<punct struct> = nil
V.t:<passive> = -
AP_2.b:<wh> = A.t:<wh>
AP_2.b:<compar> = A.t:<compar>
AP_2.b:<equiv> = A.t:<equiv>

```

## 4 Tree "betaRENpxnx1VA2"

### 4.1 graphe



## 4.2 comments

(wh) Relative clause (pied-piped) adjunct PP of an adjectival ergative resultative:

'The day on which the river froze solid'

## 4.3 features

```
NP_r.b:<rel-clause> = +
NP_r.b:<wh> = NP_f.t:<wh>
NP_r.b:<agr> = NP_f.t:<agr>
NP_r.b:<case> = NP_f.t:<case>
NP_r.b:<pron> = NP_f.t:<pron>

NP_f.b:<case> = acc/nom
PP_w.t:<wh> = +

S_r.t:<inv> = -
S_r.t:<comp> = nil
S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<comp> = nil

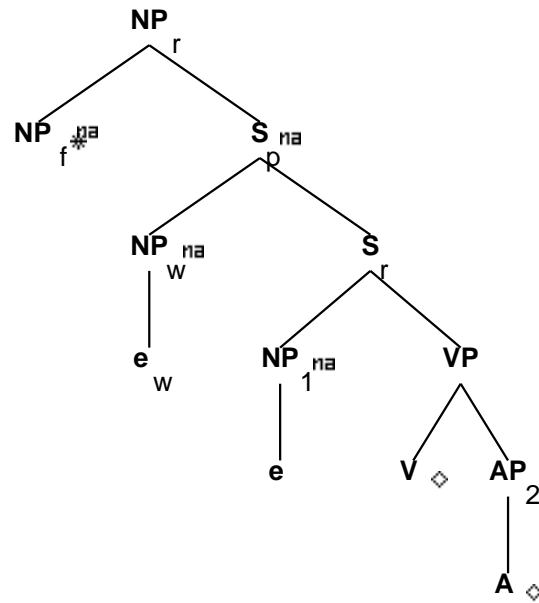
S_r.b:<mode> = VP.t:<mode>
S_r.t:<mode> = ind/inf
S_r.b:<tense> = VP.t:<tense>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<control> = NP_1.t:<control>
NP_1.t:<agr> = S_r.b:<agr>
NP_1.t:<case> = S_r.b:<assign-case>
NP_1.t:<wh> = -

VP.b:<compar> = -
VP.b:<passive> = V.t:<passive>
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<tense> = V.t:<tense>
VP.b:<mode> = V.t:<mode>
VP.b:<mainv> = V.t:<mainv>
V.t:<punct struct> = nil
V.t:<passive> = -
AP_2.b:<wh> = A.t:<wh>
```

AP\_2.b:<compar> = A.t:<compar>  
 AP\_2.b:<equiv> = A.t:<equiv>

## 5 Tree "betaREnc1nx1VA2"

### 5.1 graphe



### 5.2 comments

(COMP) relative clause on the subject  
 of ergative adjectival resultative:

'(I crossed) the river that froze solid'

### 5.3 features

NP\_r.b:<rel-clause> = +  
 NP\_r.b:<wh> = NP\_f.t:<wh>  
 NP\_r.b:<agr> = NP\_f.t:<agr>  
 NP\_r.b:<case> = NP\_f.t:<case>  
  
 NP\_r.b:<pron> = NP\_f.t:<pron>  
 NP\_f.b:<case> = nom/acc  
 NP\_w.t:<trace> = NP\_1.t:<trace>  
 NP\_w.t:<case> = NP\_1.t:<case>  
 NP\_w.t:<agr> = NP\_1.t:<agr>  
 S\_r.t:<nocomp-mode> = inf/ger  
  
 S\_r.t:<conj> = nil

```

S_r.t:<inv> = -
S_r.t:<mode> = inf/ger/ind

S_r.b:<comp> = nil
S_r.b:<agr> = NP_1.t:<agr>
S_r.b:<assign-case> = NP_1.t:<case>
S_r.b:<nocomp-mode> = S_r.b:<mode>

S_r.b:<tense> = VP.t:<tense>
S_r.b:<mode> = VP.t:<mode>
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<agr> = VP.t:<agr>
VP.t:<assign-comp> = that/ind_nil/inf_nil/ecm

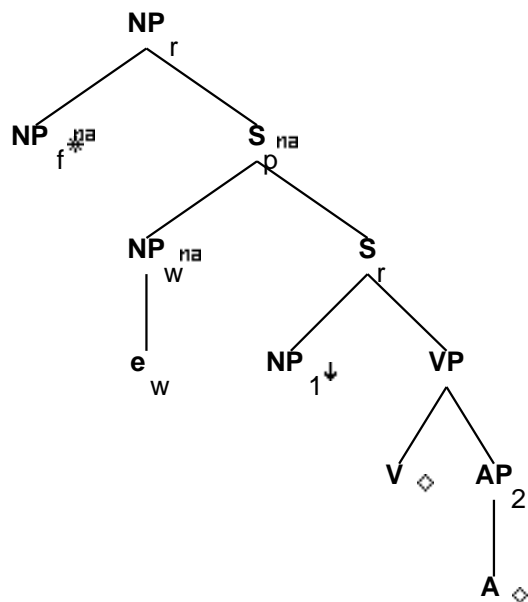
VP.b:<compar> = -
VP.b:<agr> = V.t:<agr>
VP.b:<passive> = V.t:<passive>
VP.b:<tense> = V.t:<tense>
VP.b:<mode> = V.t:<mode>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<mainv> = V.t:<mainv>
V.t:<punct struct> = nil
V.t:<passive> = -
AP_2.b:<wh> = A.t:<wh>
AP_2.b:<compar> = A.t:<compar>
AP_2.b:<equiv> = A.t:<equiv>

```



## 6 Tree "betaREncnx1VA2"

### 6.1 graphe



### 6.2 comments

(COMP) relative clause on adjunct  
of ergative adjectival resultative.

'The day that the river froze solid'

### 6.3 features

NP\_r.b:<rel-clause> = +  
NP\_r.b:<wh> = NP\_f.t:<wh>  
NP\_r.b:<agr> = NP\_f.t:<agr>  
NP\_r.b:<case> = NP\_f.t:<case>

NP\_r.b:<pron> = NP\_f.t:<pron>  
NP\_f.b:<case> = nom/acc  
S\_r.t:<inv> = -

S\_r.t:<mode> = ind/inf  
S\_r.t:<nocomp-mode> = ind  
VP.t:<assign-comp> = that/for/ind\_nil  
S\_r.b:<nocomp-mode> = S\_r.b:<mode>  
S\_r.b:<extracted> = -  
S\_r.b:<inv> = -

S\_r.b:<comp> = nil

```

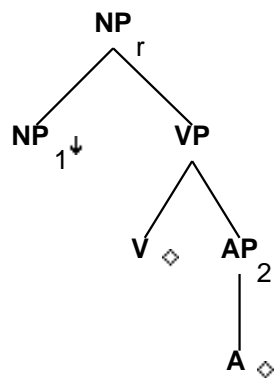
S_r.b:<mode> = VP.t:<mode>
S_r.b:<tense> = VP.t:<tense>
S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<assign-comp> = VP.t:<assign-comp>
S_r.b:<control> = NP_1.t:<control>
NP_1.t:<agr> = S_r.b:<agr>
NP_1.t:<case> = S_r.b:<assign-case>
NP_1.t:<wh> = -
VP.b:<compar> = -

VP.b:<passive> = V.t:<passive>
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<tense> = V.t:<tense>
VP.b:<mode> = V.t:<mode>
VP.b:<mainv> = V.t:<mainv>
V.t:<punct struct> = nil
V.t:<passive> = -
AP_2.b:<wh> = A.t:<wh>
AP_2.b:<compar> = A.t:<compar>
AP_2.b:<equiv> = A.t:<equiv>

```

## 7 Tree "alphaREGnx1VA2"

### 7.1 graphe



### 7.2 comments

Ergative NP gerund tree with an adjectival resultaive::

'The river's freezing' disturbed everyone.

### 7.3 features

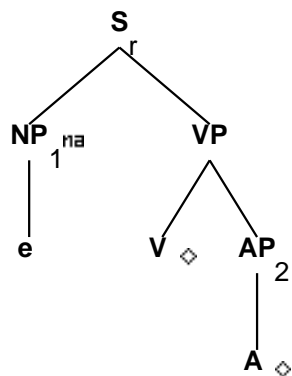
NP\_r.b:<gerund> = +  
 NP\_r.b:<agr pers> = 3  
 NP\_r.b:<case> = nom/acc  
 NP\_r.b:<agr num> = sing  
 NP\_r.b:<agr 3rdsing> = +

NP\_r.b:<wh> = NP\_1.t:<wh>  
 NP\_r.b:<compar> = NP\_1.t:<compar>  
 NP\_1.t:<case> = acc/none/gen  
 VP.b:<compar> = -  
 VP.t:<mode> = ger

VP.b:<mode> = V.t:<mode>  
 VP.b:<passive> = V.t:<passive>  
 V.t:<passive> = -  
 V.t:<punct struct> = nil  
 AP\_2.b:<wh> = A.t:<wh>  
 AP\_2.b:<compar> = A.t:<compar>  
 AP\_2.b:<equiv> = A.t:<equiv>

## 8 Tree "alphaREInx1VA2"

### 8.1 graphe



### 8.2 comments

Imperative ergative resultative.

'Freeze solid'

### 8.3 features

```
S_r.t:<assign-comp> = inf_nil/ind_nil

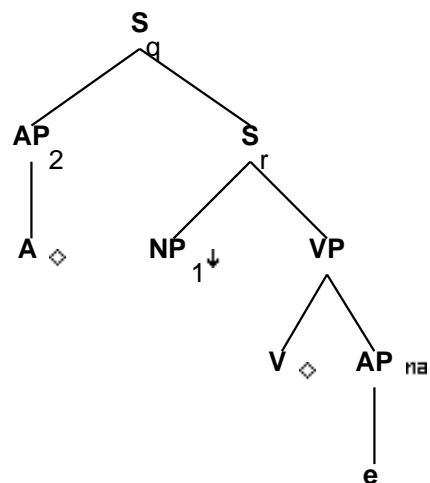
S_r.b:<extracted> = -
S_r.b:<inv> = -
S_r.b:<comp> = nil
S_r.b:<assign-comp> = VP.t:<assign-comp>

S_r.b:<progressive> = VP.t:<progressive>
S_r.b:<perfect> = VP.t:<perfect>
S_r.b:<passive> = VP.t:<passive>
S_r.b:<mainv> = VP.t:<mainv>
S_r.b:<tense> = VP.t:<tense>
S_r.b:<wh> = NP_1.t:<wh>
S_r.b:<agr> = VP.t:<agr>
S_r.b:<assign-case> = VP.t:<assign-case>
S_r.b:<mode> = imp
NP_1.t:<wh> = -
NP_1.t:<agr pers> = 2
NP_1.t:<agr 3rdsing> = -
NP_1.t:<agr num> = plur/sing
NP_1.t:<case> = nom
NP_1.t:<agr> = S_r.b:<agr>
NP_1.t:<case> = S_r.b:<assign-case>
VP.t:<tense> = pres
VP.t:<neg> = -
VP.t:<mode> = base
VP.b:<mode> = V.t:<mode>

VP.b:<compar> = -
VP.b:<passive> = V.t:<passive>
VP.b:<agr> = V.t:<agr>
VP.b:<assign-case> = V.t:<assign-case>
VP.b:<assign-comp> = V.t:<assign-comp>
VP.b:<tense> = V.t:<tense>
VP.b:<mainv> = V.t:<mainv>
V.t:<passive> = -
V:<punct struct> = nil
AP_2.b:<wh> = A.t:<wh>
AP_2.b:<compar> = A.t:<compar>
AP_2.b:<equiv> = A.t:<equiv>
```

## 9 Tree "alphaREWA2VA2"

### 9.1 graphe



### 9.2 comments

NIL

### 9.3 features

S<sub>r</sub>.b:<inv> = -

S<sub>r</sub>.b:<comp> = nil

S<sub>r</sub>.b:<mode> = VP.t:<mode>

S<sub>r</sub>.b:<tense> = VP.t:<tense>

S<sub>r</sub>.b:<agr> = VP.t:<agr>

S<sub>r</sub>.b:<assign-case> = VP.t:<assign-case>

S<sub>r</sub>.b:<assign-comp> = VP.t:<assign-comp>

S<sub>r</sub>.b:<progressive> = VP.t:<progressive>

S<sub>r</sub>.b:<perfect> = VP.t:<perfect>

S<sub>r</sub>.b:<passive> = VP.t:<passive>

S<sub>r</sub>.b:<mainv> = VP.t:<mainv>

S<sub>r</sub>.b:<control> = NP<sub>1</sub>.t:<control>

NP<sub>1</sub>.t:<agr> = S<sub>r</sub>.b:<agr>

NP<sub>1</sub>.t:<case> = S<sub>r</sub>.b:<assign-case>

VP.b:<compar> = -

VP.b:<passive> = V.t:<passive>

VP.b:<agr> = V.t:<agr>

VP.b:<assign-case> = V.t:<assign-case>

VP.b:<assign-comp> = V.t:<assign-comp>

VP.b:<tense> = V.t:<tense>

VP.b:<mode> = V.t:<mode>

```

VP.b:<mainv> = V.t:<mainv>
V.t:<passive> = -
V.t:<punct struct> = nil
AP_2.b:<wh> = A.t:<wh>
AP_2.b:<compar> = A.t:<compar>
AP_2.b:<equiv> = A.t:<equiv>
S_q.b:<extracted> = +

S_q.b:<inv> = S_r.t:<inv>
S_q.b:<inv> = S_q.b:<invlink>
S_q.b:<wh> = AP_2.t:<wh>
S_q.b:<mode> = S_r.t:<mode>
S_q.b:<comp> = nil
S_r.t:<comp> = nil
S_r.t:<conj> = nil
AP.t:<trace> = AP_2.t:<trace>
AP.t:<wh> = AP_2.t:<wh>
AP_2.t:<wh>=+

```