Temporal Semantics and Temporal Reasoning

Mark Steedman, Informatics, University of Edinburgh

ISC Lyon, February 2000

`www.cogsci.ed.ac.uk/~steedman/papers.html`

Aspectual Coercion and Temporal Reference

(1) It took me ten years to play the Minute Waltz in less than 50 seconds for more than an hour.

(2) Two weeks later, Bonadea had already been his lover for a fortnight.

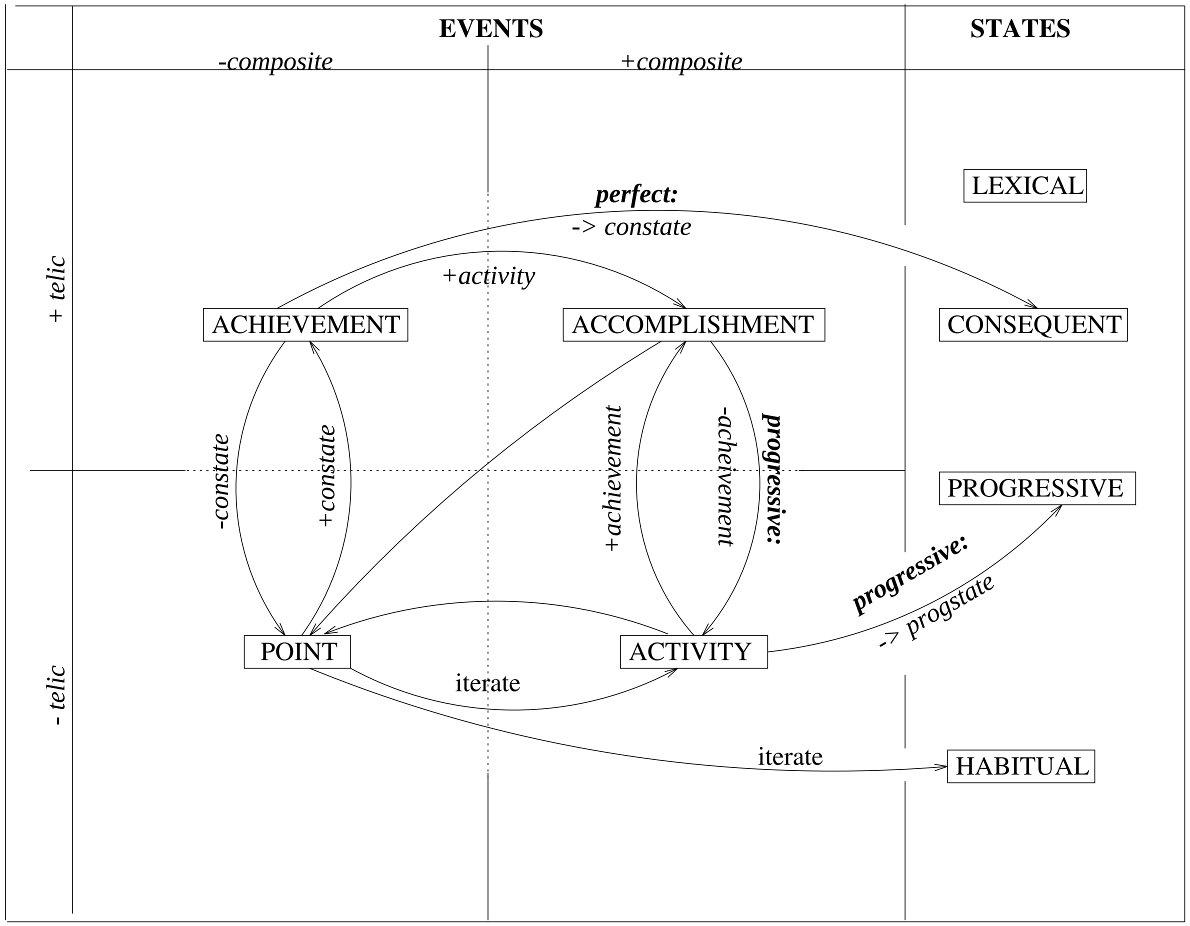

Figure 1: A scheme of aspectual coercion (Moens and Steedman 1988)

The Nuclear Structure of Default Temporal Relations

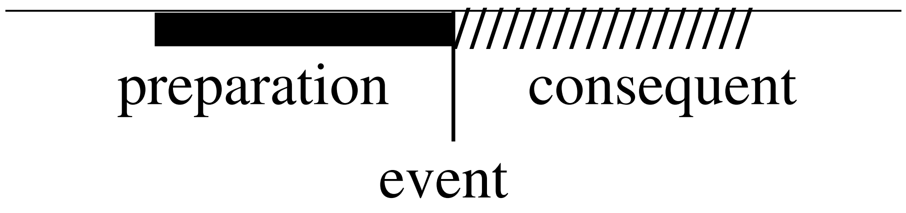

Figure 2: The event nucleus (adapted from Moens and Steedman 1988)

This structure, or “nucleus” can be regarded as composed of the types described in figure 1. Thus the preparation is an activity, the consequent is the same kind of state that the perfect gives rise to, while the event itself is an achievement.

Coercions as Rules

(3) a. $\models \mathit{predicate}(\mathit{accomplishment}(\mathit{act}, \mathit{ach})$
$\rightarrow \mathit{predicate}(\mathit{achievement}(\mathit{ach}))$

b. $\models \mathit{predicate}(\mathit{activity}(\mathit{act}))$
$\rightarrow \mathit{predicate}(\mathit{accomplishment}(\mathit{activity}(\mathit{act}), \mathit{achievement}(\mathit{ach}))$

(4) a. $\models \mathit{in\_progress}(\mathit{act}) \rightarrow \mathit{progressive}(\mathit{activity}(\mathit{act}))$

b. $\models \mathit{preparation}(\mathit{act}, \mathit{ach}) \wedge \mathit{progressive}(\mathit{activity}(\mathit{act}))$
$\rightarrow \mathit{progressive}(\mathit{accomplishment}(\mathit{activity}(\mathit{act}), \mathit{achievement}(\mathit{ach})))$

c. $\models \mathit{preparation}(\mathit{act}, \mathit{ach})$
$\wedge \mathit{progressive}(\mathit{accomplishment}(\mathit{activity}(\mathit{act}), \mathit{achievement}(\mathit{ach})))$
$\rightarrow \mathit{progressive}(\mathit{achievement}(\mathit{ach}))$

Many Strange Verb Phenomena Follow—cf. Thomason 1999

He tried to close the door.

(5) $\models preparation(act, ach) \wedge intend(ach) \wedge do(act)$
$\quad \rightarrow try(accomplishment(act), achievement(ach))$

He failed to close the door

(6) $\models preparation(act, ach) \wedge failed(ach) \wedge intend(ach) \wedge do(act)$
$\quad \rightarrow fail(accomplishment(act), achievement(ach))$

Jeeves deftly trousered the £5 note.

(7) $\models normalstate(£5\_note, trouser, s) \wedge constate(achievement(ach), s)$
$\quad \wedge preparation(act, ach) \wedge do(accomplishment(activity(act), achievement(ach)))$
$\quad \rightarrow trouser(£5\_note)$

Kripke Model

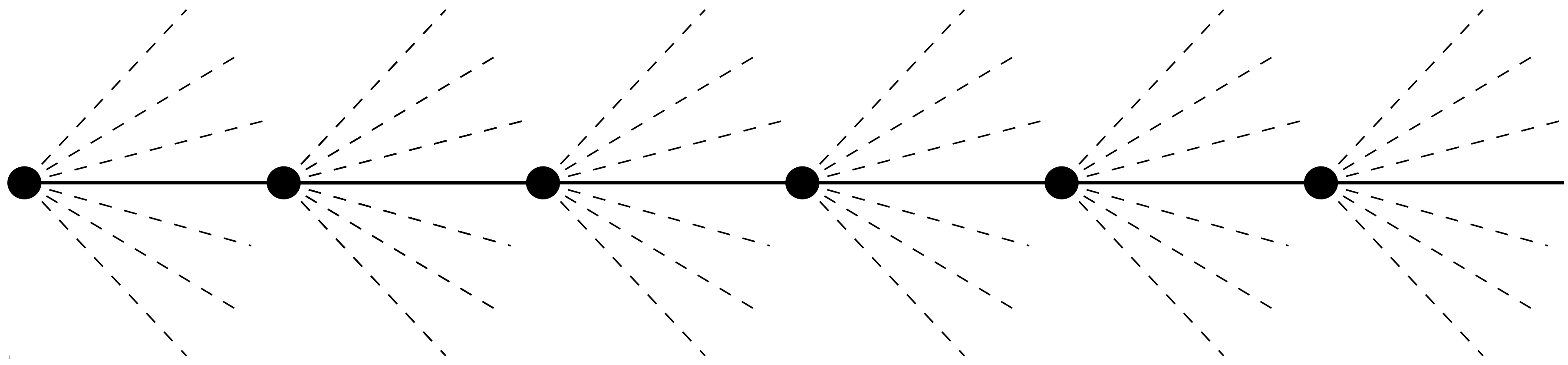

Figure 3: Modal Temporal Structure

The Situation Calculus of McCarthy and Hayes

(8) $\forall s, \forall person, present(person, result(arrive(person), s))$

(9) a. $clear(a, s_0) \wedge clear(b, s_0) \wedge clear(c, s_0)$

b. $\forall x, \forall y, \forall s, clear(x, s) \wedge clear(y, s) \wedge x \neq y$
$\rightarrow clear(x, result(puton(x, y, s)))$
$\wedge \neg clear(y, result(puton(x, y, s)))$
$\wedge on(x, y, result(puton(x, y, s)))$

(10) $on(a, b, result(puton(a, b, result(puton(b, c, s_0)))))$
$\wedge on(b, c, result(puton(a, b, result(puton(b, c, s_0))))$

(11) a. $\forall u, \forall x, \forall y, \forall s, clear(u, s) \wedge u \neq y \rightarrow clear(u, result(puton(x, y, s)))$

b. $\forall u, \forall v, \forall x, \forall y, \forall s, on(u, v, s) \wedge u \neq x \rightarrow on(u, v, result(puton(x, y, s)))$

A Declarative Solution to the Frame Problem (Kowalski)

The situation:

(12) $holds(clear(a), s_0) \wedge holds(clear(b), s_0) \wedge holds(clear(c), s_0)$

"Qualifications" of *puton*

(13) $holds(clear(x), s) \wedge holds(clear(y), s) \wedge (x \neq y)$
$\rightarrow preconditions(puton(x,y), s)$

"Ramifications" of *puton*

(14) a. $holds(on(x,z), s) \rightarrow holds(clear(z), result(puton(x,y), s))$
b. $holds(on(x,y), result(puton(x,y), s))$

"Frame Axiom" for *puton*

(15) $holds(p,s) \wedge (p \neq clear(y)) \wedge (p \neq on(x,z))$
$\rightarrow holds(p, result(puton(x,y),s))$

(16) a. $poss(s_0)$
b. $poss(s) \wedge preconditions(action,s) \rightarrow poss(result(action,s))$

(17) $poss(s) \wedge holds(on(a,b),s) \wedge holds(on(b,c),s)$

(18) $s = result(puton(a,b), result(puton(b,c),s_0))$

"Counterfactuals"

(19) If Nixon had pushed the button, there would have been a nuclear war.

The Dynamic Event Calculus

Dynamic Logic:

(20) $n \geq 0 \rightarrow [\alpha](y = F(n))$

(21) $n \geq 0 \rightarrow \langle\alpha\rangle(y = F(n))$

Sequential Composition:

(22) $[\alpha][\beta]P \rightarrow [\alpha;\beta]P$

Notation: Situation Calculus vs. Dynamic Event Calculus

(23) a. $holds((on(a,b) \wedge on(b,c)), result(puton(a,b), result(puton(b,c), s_0)))$

b. $[puton(b,c); puton(a,b)](on(a,b) \wedge on(b,c))$

Frame Problem Revisited

The start state:

(24) $clear(a) \wedge clear(b) \wedge clear(c)$

Qualifications of *puton*

(25) $\models clear(x) \wedge clear(y) \wedge (x \neq y) \rightarrow possible(puton(x,y))$

Ramifications of *puton*

(26) a. $\models on(x,z) \rightarrow [puton(x,y)]clear(z)$
b. $\models [puton(x,y)]on(x,y)$

Frame Axiom for *puton*

(27) $\models p \wedge (p \neq clear(y)) \wedge (p \neq on(x,z)) \rightarrow [puton(x,y)]p$

Composition:

(28) $\models possible(\alpha) \wedge [\alpha]possible(\beta) \rightarrow possible(\alpha;\beta)$

Goal to find a plan to put a on b on c:

(29) $possible(\alpha) \wedge [\alpha](on(a,b) \wedge on(b,c))$

The plan:

(30) $\alpha = [puton(b,c);puton(a,b)]$

Durative Events in DEC

(31) a. $\models [start(p)]in\_progress(p)$

b. $\models \neg in\_progress(move(x, l_0, l)) \wedge at(x, l_0)$
$\quad \rightarrow possible(start(move(x, l_0, l)))$

c. $\models p \wedge p \neq at(x, l_0) \rightarrow [start(move(x, l_0, l))]p$

(32) $\models in\_progress(p) \rightarrow possible(stop(p))$

(33) $\models p \wedge (p \neq in\_progress(q)) \rightarrow [stop(q)]p$

(34) a. $\models in\_progress(act) \rightarrow progressive(activity(act))$

b. $\models preparation(act, ach) \wedge progressive(activity(act))$

$\rightarrow progressive(accomplishment(activity(act), achievement(ach)))$

c. $\models preparation(act, ach)$

$\wedge progressive(accomplishment(activity(act), achievement(ach)))$

$\rightarrow progressive(achievement(ach))$

Coercion and Strange Verbs Revisited—cf. Thomason 1999

He tried to close the door.

(35) $\models preparation(act, ach) \wedge intend(ach)$
$\rightarrow [start(act)]in\_progress(try(accomplishment(act), achievement(ach)))$

He failed to close the door

(36) $\models preparation(act, ach) \wedge failed(ach) \wedge intend(ach)$
$\rightarrow [start(act)]in\_progress(fail(accomplishment(act), achievement(ach)))$

Jeeves deftly trousered the £5 note.

(37) $\models normalstate(£5\_note, trouser, s) \wedge constate(achievement(ach), s)$
$\wedge\, preparation(act, ach)$
$\rightarrow [start(act)]in\_progress(trouser(£5\_note))$

The Imperfective Paradox

(38) $possible(\alpha) \wedge [\alpha] progressive(achievement(finish(write(keats, sonnet))))$

(39) $\models preparation(achievement(finish(e)), activity(e))$

(40) $\alpha = start(write(keats, sonnet))$

(41) a. $\models [arrive(x)]present(x)$

b. $\models not(present(x)) \rightarrow possible(arrive(x))$

c. $\models p \rightarrow [arrive(x)]p$

(42) $\alpha = start(write(keats, sonnet)); arrive(x)$

$\alpha = start(write(keats, sonnet)); arrive(x); arrive(y)$

$\ldots$

(43) $\alpha = start(write(keats, sonnet)); arrive(x); stop(write(keats, sonnet)$

That Man Keats

(44) $\models (possible(\alpha) \wedge [\alpha]necessary(\beta)) \rightarrow possible(\alpha @ \beta)$

(45) $\models present(x) \wedge (x \neq keats) \wedge in\_progress(write(keats, y))$
$\rightarrow necessary(stop(write(keats, y)))$

(46) a. $\alpha = start(write(keats, y)); arrive(x)$
b. $\alpha = start(write(keats, y))$

The (Solved) Ramification Problem

The consequences of an action may be overridden. Start State (simplified by omitting the individual)

(47) $at(here) \wedge color(green)$

Frame axiom for *move* (similarly simplified):

(48) $\models p \wedge p \neq at(a) \rightarrow [start(move(a,b))]p$

Ramification of spraying something a color d:

(49) $\models [start(spray(d))]color(d)$

Frame axiom for spraying something a color d:

(50) $\models p \wedge p \neq color(c) \wedge c \neq d \rightarrow [start(spray(d))]p$

Then we correctly fail to prove the following:

(51) $[start(move(here,there));start(spray(pink));$
$stop(spray(pink));stop(move(here,there))]color(green)$

(52) $[start(spray(pink));start(move(here,there));$
$stop(move(here,there));stop(spray(pink))]at(here)$

Clock-time

Its easy enough to fix up DEC with temporal indices identifying clock times:

(53) $[_{0} start(move(here, there))_{1}; start(spray(pink))_{1};$
$stop(spray(pink))_{2}; stop(move(here, there))_{2}]_{2} color(green)$

(54) $[_{0} start(spray(pink))_{1}; start(move(here, there))_{1};$
$stop(move(here, there))_{2}; stop(spray(pink))_{2}]_{2} at(here)$

(55) $\models [_{t_0} \alpha]_{t_2} clocktime(t_2)$

“Generation” (Goldman)

(56) $\models [_{t} flip]_{t} necessary(turnon_{t})$

Delayed Action

(57) a. $\models alive \rightarrow possible(die)$

b. $\models [die]\neg alive$

c. $\models p \wedge \neg(p = alive) \rightarrow [die]p$

(58) $\models [_{t_0} drink(A_s)_{t_1}]_{t_2} constate(drink(A_s)_{t_1})$

(59) $\models constate(drink(A_s)_{t_1}) \wedge clocktime(t_2) \wedge t_2 = (t1 + 4)$
$\rightarrow necessary(die_{t_2})$

(60) $possible(\alpha) \wedge [\alpha]actual(die)$

(61) $[_0 drink - A_s(0)]_4 \neg alive$

Temporal Reference

(62) a. When I took your pawn, you took my queen.

b. I took your pawn. You took my queen

(63) When I took your pawn, I did not know it was protected by your knight.

(64) When I took your pawn, I used a rook.

(65) a. If you take my queen, you may win.

b. If you had taken my queen, you might have won.

c. Since you took my queen, you have been winning.

Temporal Reference in Dynamic Event Calculus

(66) $start(write(keats, sonnet)); arrive(chapman) @ stop(write(keats, sonnet))$

(67) $start(write(keats, sonnet)); arrive(chapman)$

(68) a. $\models evoke((\alpha;\beta),(\alpha;\beta))$

b. $\models evoke((\alpha;\beta),\gamma) \rightarrow evoke((\alpha;\beta),(\gamma;\delta))$

(69) a. $\models state(\gamma) \wedge S(history) \wedge evoke((\alpha;\beta), history) \wedge [\alpha;\beta]\gamma$
$\rightarrow when(\beta,\gamma)$

b. $\models event(\varepsilon) \wedge S(history) \wedge evoke((\alpha;\beta@\varepsilon), history) \rightarrow when(\beta,\varepsilon)$

(70) $when((start(write(keats, sonnet)); arrive(chapman)),$
$stop(write(keats, sonnet)))$

(71) a. #When Keats started writing *In Disgust of Vulgar Superstition*, Chapman arrived.

b. Keats started writing *In Disgust of Vulgar Superstition*. Chapman arrived.

(72) a. When Chapman arrived, Keats stopped writing.

b. When Chapman flipped the switch, he turned on the light.

The (Unsolved) Qualification Problem

You can never state all the conditions of an action.

I am sitting in my car and I want to drive. I know that for me to drive the engine must be running, and that for it to be running I must switch it on and that the precondition for switching on is to have the key in the lock and the way to do that is to put it there, for which the only precondition is my being there.

There are several, possibly infinitely many, further preconditions. For example, if there is a potato on the tailpipe of a car, switching on will not result in the engine running. By assumption, this practical precondition is not explicit in the knowledge representation that I am using. What do I do?

either: Some hard scientific reasoning from first principles according to the Hypothetico Deductive Method.

or: Kick the tires.