to infer |a||c| from |b||c| ...

ETall C 3 & Elbilc 13 -> interals

4 LØ = Elallb | C |, | a|b| | C |, | a|ab | C |,

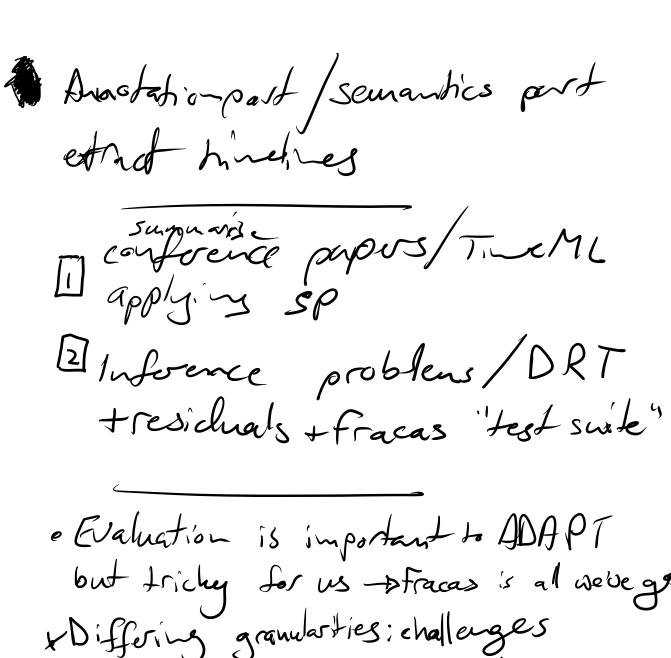
1 a | a, b | a | C |, | a | a, b | b | | C |, | 1 a, b | C |,

1 a, b | a | C |, | a, b | b | | C |, | b | a, b | b | | c |,

1 b | a, b | a | | c |, | b | a | b | b | c |,

1 b | a | b | a | c |, | b | a | c | 3

 $S \in L \beta = |a||b||c|, vac(s) = \{a,b,c\}$ $V \in Voc(s) = \{a,b\}$ $T_{v}(s) = |a||b| = p$ P(s) = |a||b||c||3 = p $P(s) = |a||c||, voc(m) = \{a,c\} = van$ $T_{ym}(H) = m$ Tym (II) == m Iall bol == lall bol v Ef lall bol is a residual, gap (L, L') -> L=premises, L'=conclusions ·L&L'={s"|seL,s'eL',3&s'=5"} p(L, L') = P(L, L)= N=ETTV(s") | Ys" & L&L', YY = vocabulary (s")) $\beta = \alpha \& L$ $\begin{cases}
8 = \alpha \& L \\
8 = \epsilon & \alpha \mid \alpha \in \alpha, \forall s' \in L', \pi_{voc(s')}(b) = s'
\end{cases}$ $(\forall r \in \pi_{V}(s') \& s)(\forall s' \in L') \pi_{VOC(s')}(r) \equiv s')$ minî={r|reî, \Vcvc(r), \tau(r) \Eî} gap (IbleI), IaleII)= [[a,b]] + [a]a,b]] + [[b]a,b]] + [[a]c][Note: Tim's gap.pdf has the premises + conclusion the other way around.



but tricky for us -> fracas is all we've get x Differing granularties; challenges . Formal evaluation? se. Lormal sumarbies

- Impro ~ TimeML/TLINKS - Literature Review

- DRT 7 existing -> input DRT becomes output strings -> Need to evaluate correctioness of output LINT PMB examples & adding whe 5 Manual evaluation

-> objective decision on correctness of output -> can my manipulations produce something incarrect? No -> can I construe a proof of this Fracas -> : faccess to productive boxer -> Do be got what Fracas says be should? This is also evaluation of a sort
Doesn't have to be a corpany evaluation
TEA? -) Manual evaluation? -) manually convert to strings and show the informees nold -) denorstrates the aspectuality
Zebra/scheduling? -) Primer on finite state temporality ch. like lit. review -> shaving operations in action as

much as possible + reninders to help the examiners as often as possible

Make T.C.C.

Applications -> code in appendix

Ly Linhed w/ formal definitions

Ly show implementation

Ly evaluate xorretness of code

demanstrably -> maybe not formal

correctness proofs

-> test suites? would be good.