JA-STIT: the stit way to (public) justification announcements

Grigory Olkhovikov

Ruhr University Bochum

Bochum, 15.12.2017
The main subject of this talk will be the stit logic of justification announcements (JA-STIT), one example in the family of justification stit (jstit) logics.

I will start by outlining the goals and philosophical choices behind the project that lead to this family of logics.

In the technical part, I will first briefly explain the two parent logics of the jstit logics family:

1. Stit logic; and
2. Epistemic justification logic.

I will then explain the jstit ideas as to how one should merge stit and justification structures within one model.

I will then provide an in-depth study of JA-STIT against the backdrop of these contextualizations, contrasting it both with what can be called minimal jstit logic and its numerous extensions.

Throughout the exposition I will be focusing on completeness and definability issues, unpacking the latter both in terms of precise results and (hopefully) revealing examples.
The family of jstit logics contains a number of systems that are united by (1) common goals, (2) underlying philosophical choices and (3) formal characteristics. I will characterize these in turn, starting with goals.

- **Goal 1**: to make sense (formally and explicitly) of the distinction between proofs-as-objects and proofs-as-acts.
  (This was realized, at least partially).

- **Goal 2**: to make sense (formally and explicitly) of the notion of responsibility for doxastic actions in application to proofs.
  (Under construction; this will be the next stage of the project)

The philosophical choices are therefore mainly related to Goal 1.
Speaking of actions, we must keep in mind the distinction between generic and concrete actions.

This dictates the preferred action logic: dynamic logic for generic actions and stit logic for concrete actions.

Speaking of proof objects, one has to keep an eye on the distinction between proofs as abstract objects (theoretical possibilities of constructing a proof) and proofs as realized objects (on paper, on a whiteboard, on a screen, in a brain etc)

The question is, something must be taken as basic and the other as constructed from the basic elements, so we face two dilemmas.

The set of philosophical choices which constitutes the family of jstit logics is predicated on the choice of concrete actions (supplied by stit logic) plus abstract proof objects (supplied by epistemic justification logic).

We will skip possible motivations but please feel free to ask about my views on them during Q & A!
A stit model for a finite community $Ag$ of agents is a structure $\mathcal{M} = \langle Tree, \sqsubseteq, Choice, V \rangle$ where $Tree$ is a non-empty set of moments, $\sqsubseteq$ is a forward-branching partial order on $Tree$ in which every two moments have a common ancestor. It has a causal temporal interpretation.

The set of histories $Hist(\mathcal{M})$ is then defined as the set of maximal $\sqsubseteq$-chains in $Tree$. The set of histories passing through a given moment $m$ is denoted $H_m$. The set $MH(\mathcal{M}) = \{(m, h) \mid m \in Tree, h \in H_m\}$ of moment-history pairs is used to evaluate the formulas so that $V$ returns a subset of $MH(\mathcal{M})$ for a given propositional variable $p$.

$Choice$ is a function on $Tree \times Ag$ such that $Choice(m, j)$ (denoted $Choice^m_j$) is a partition of $H_m$. It is assumed to satisfy the following additional constraints for all $m \in Tree$:

\[
(\forall h, h' \in H_m)(h \approx_m h' \Rightarrow Choice^m_j(h) = Choice^m_j(h'));
\]

\[
(\forall f : Ag \rightarrow 2^{H_m})(\forall j \in Ag)(f(j) \in Choice^m_j) \Rightarrow \bigcap_{j \in Ag} f(j) \neq \emptyset).
\]
For a fixed finite \( Ag \), and the set of propositional variables \( Var \), the set of \( StitForm^Ag \) of stit formulas is defined by the following BNF:

\[
A := p \in Var \mid A_1 \land A_2 \mid \neg A \mid [j]A \mid \Box A
\]

where \( j \in Ag \).

These formulas are interpreted by the following satisfaction clauses:

\[
\begin{align*}
\mathcal{M}, m, h \models p & \iff (m, h) \in V(p); \\
\mathcal{M}, m, h \models [j]A & \iff (\forall h' \in Choice^m_j(h))(\mathcal{M}, m, h' \models A); \\
\mathcal{M}, m, h \models \Box A & \iff (\forall h' \in H_m)(\mathcal{M}, m, h' \models A).
\end{align*}
\]

where \( Choice^m_j(h) \) is the cell in \( Choice^m_j \) to which \( h \) belongs.
The following system $\mathcal{S}$ is a strongly complete axiomatization for this logic:

Classical propositional tautologies \hspace{1cm} (AS0)

$S5$ axioms for $\Box$ and $[j]$ for every $j \in Ag$ \hspace{1cm} (AS1)

$\Box A \rightarrow [j]A$ for every $j \in Ag$ \hspace{1cm} (AS2)

$(\Diamond [j_1]A_1 \land \ldots \land \Diamond [j_n]A_n) \rightarrow \Diamond ([j_1]A_1 \land \ldots \land [j_n]A_n)$ \hspace{1cm} (AS3)

The assumption is that in (AS3) $j_1, \ldots, j_n$ are pairwise different.

The rules of inferences are then as follows:

From $A, A \rightarrow B$ infer $B$; \hspace{1cm} (MP)

From $A$ infer $\Box A$; \hspace{1cm} (Nec$\Box$)
One of central distinctions in stit logic can be summarized as follows:

- **Contingent** events like future sea battles (true in a given moment in a random subset of $H_m$); **No underlying state of affairs, but may get one in future**.

- **Static** events that are accomplished facts and can not be undone by the future events (true in a given moment throughout $H_m$). These events are described by *moment-determinate* statements satisfying $A \rightarrow \Box A$. One sometimes requires that propositional variables the domain of such events. **Underlying state of affairs is given currently and acts as a truth-maker**.

- **Dynamic** events or events in the making typically true throughout the histories in a given choice cell but not necessarily throughout $H_m$. Characterized by formulas like $A \rightarrow [j]A$ or, to draw the line more sharply, $A \rightarrow [j]A \land \neg \Box A$. Statements describing actions are of this type. **Underlying state of affairs is present currently but conditioned by the agent’s choice — it is dynamically unfolding rather than statically given**.
The language of justification logic features two grammatical categories: formulas and proof polynomials.

Set $Pol$ of proof polynomials is defined on the basis of the countable sets $PVar$ of proof variables and $PConst$ of proof constants via the following BNF:

$$\begin{align*}
t &:= x \in PVar \mid c \in PConst \mid s \cdot t \mid s + t \mid !t
\end{align*}$$

where $s \cdot t$ is an application of $s$ to $t$, $s + t$ is the sum of proofs, and $!t$ is the proof checking the correctness of $t$.

Set $JForm$ of formulas is then defined on the basis of $Pol$ and $Var$:

$$\begin{align*}
A &:= p \in Var \mid A \rightarrow B \mid \neg A \mid A \land B \mid A \lor B \mid t:A \mid KA
\end{align*}$$

with $t:A$ meaning $t$ proves $A$ and $KA$ meaning $A$ is known (or maybe $A$ is provable).
The frames for justification logic are just bi-S4 Kripke frames \( \langle W, R, R_e \rangle \) satisfying \( R \subseteq R_e \).

A justification model is then a structure of the form \( \langle W, R, R_e, \mathcal{E}, V \rangle \), \( V \) being the evaluation function for \( \text{Var} \), and \( \mathcal{E} \) being a function which says when a given proof polynomial is \textit{admissible} as an evidence for a given formula. Thus we have \( \mathcal{E} : W \times \text{Pol} \to 2^{\text{JForm}} \).

In a justification model, \( \mathcal{E} \) has to satisfy the following constraints (universally closed):

- Monotonicity of evidence:
  \[
  R_e(w, w') \Rightarrow \mathcal{E}(w, t) \subseteq \mathcal{E}(w', t).
  \]

- Evidence closure properties:
  1. \( A \rightarrow B \in \mathcal{E}(w, s) \& A \in \mathcal{E}(w, t) \Rightarrow B \in \mathcal{E}(w, s \cdot t) \);
  2. \( \mathcal{E}(w, s) \cup \mathcal{E}(w, t) \subseteq \mathcal{E}(w, s + t) \);
  3. \( A \in \mathcal{E}(w, t) \Rightarrow t:A \in \mathcal{E}(w, !t) \);
Satisfaction is then defined as follows (Booleans standard and omitted): $\mathcal{M}, w \models p$ iff $w \in V(p)$, for every $p \in \text{Var}$.  
$\mathcal{M}, w \models KA$ iff for all $u \in W$, if $wRu$, then $\mathcal{M}, u \models A$.  
$\mathcal{M}, w \models t:A$ iff $(A \in E(w, t)$, and for all $u \in W$, if $wReu$, then $\mathcal{M}, u \models A$).

The following system $\mathcal{J}$ is strongly complete for this semantics:

A full set of propositional axioms  
$(s:(A \rightarrow B) \rightarrow (t:A \rightarrow (s \cdot t):B)$  
(t:A \rightarrow (!t:(t:A) \land KA)$  
$(s:A \lor t:A) \rightarrow (s + t):A$  
$S4$ axioms for $K$

The rules of inferences are (MP) plus:

From $A$ infer $KA$;  
(NecK)
It is customary to enrich any logic which has justification part with constant specifications ensuring that we have enough proofs for the axioms. More precisely:

Let $\Gamma$ be a set of formulas in some language. A constant specification for $\Gamma$ any set $CS$ such that:

- $CS \subseteq \{ c_n : \ldots c_1 : A \mid c_1, \ldots, c_n \in PConst \quad A \in \Gamma \}$;
- Whenever $c_{n+1} : c_n : \ldots c_1 : A \in CS$, then also $c_n : \ldots c_1 : A \in CS$.

A constant specification $CS$ for $\Gamma$ is appropriate for $\Delta$ iff for every $A \in \Delta$ there is a $c \in PConst$ such that $c : A \in CS$.

If $\Sigma$ is a Hilbert-style axiomatic system then $CS$ is for $\Sigma$ iff $CS$ is for the set of $\Sigma$’s axioms, and $CS$ is appropriate for $\Sigma$ iff it is both for $\Sigma$ and is appropriate for the set of $\Sigma$’s axioms.
We then say that a model $\mathcal{M}$ is $CS$-normal iff it is true that:

$$(\forall c \in P\text{Const})(\forall w \in W)(\{A \mid c:A \in CS\} \subseteq E(w, c))$$

$\mathcal{J}$ is stable w.r.t. to constant specifications for $\mathcal{J}$ in the sense that to capture $CS$-normal models one only has to add to the basic system the following rule:

If $c_n: \ldots c_1:A \in CS$, infer $c_n: \ldots c_1:A$. \hspace{1cm} (\text{R}_{CS})

Note that all the above results still hold when one restricts attention to the unirelational models/frames satisfying the additional constraint that $R_e \subseteq R$ which leads to the collapse of the two accessibility relations into one.

Also, note how the distinction between explicit and implicit knowledge is crucial in justification logic (in much the same way as the distinction between moment-determinate and dynamic events is crucial for stit logic).
If we want to represent the agents' proving activity, we need three components: (a) the agents presenting the proofs to the community, (b) the proofs to be presented, and (c) an interface for the interaction of agents with proofs.

(a) and (b) are then disposed of by stit and justification logics, respectively.

As for (c), the underlying intuitions here are an idealized and abstracted version of a situation when a group of agents tries to produce a proof working on a common whiteboard. This whiteboard then is their main medium for making different possible proofs epistemically transparent to themselves and their colleagues.

In our formalization of (c) we abstract away from: (1) the other available media (private notes, private messages, etc.), (2) the natural limitations of the actual whiteboard (limited space and necessity to erase old proofs), and (3) the natural limitations of the agents’ communication capacities (bad handwriting or short-sightedness).
Put in more general terms, our interaction model looks as follows:

- A common pool of presented proofs is supposed to exist;
- This pool is the unique medium connecting the realm of abstract proofs to the world in which agents act;
- This pool can hold an unlimited number of proofs;
- Once a proof is in the pool, it stays there forever;
- Once a proof is in the pool, it is immediately understood and recognized by every agent in the community.
We now describe how to merge (a), (b), and (c) above within a single model structure. Again we start by fixing a finite community $Ag$ of agents. We assume the set $Pol$ defined above as given and we assume that we have already fixed a proper set of formulas $Form^{Ag}$ for describing this type of structures which extends both the stit language and the justification language.

A jstit model for $Form^{Ag}$ is then defined as a structure of the form $\mathcal{M} = \langle Tree, \sqsubseteq, Choice, Act, R, R_e, E, V \rangle$.

In this structure, $\langle Tree, \sqsubseteq, Choice, V \rangle$ is a stit model.

$\langle Tree, R, R_e \rangle$ is a justification frame and $E$ is an admissible evidence function relative to this frame and $Form^{Ag}$. (note the different format for $V$!)

These two parts of the jstit model are supposed to be connected by $\sqsubseteq \subseteq R$ — future always matters (epistemically).

Finally, $Act$ is a function which for a given moment-history pair $(m, h)$ returns the set of proof polynomials presented to the community in this moment under this history.
In interpreting these structures we invoke the central stit distinction to tell whether the presence of \( t \) to the community is an accomplished fact, or \( t \) is being dynamically presented to it right now, or else the presentation of \( t \) occurs as a contingency. This explains the constraints placed on \( \text{Act} \):

- **Expansion of presented proofs:**
  \[
  (\forall m, m' \in \text{Tree})(m' \triangleleft m \Rightarrow \forall h \in H_m(\text{Act}(m', h) \subseteq \text{Act}(m, h))).
  \]

- **No new proofs guaranteed:**
  \[
  (\forall m \in \text{Tree})(\text{Act}_m \subseteq \bigcup_{m' \triangleleft m, h \in H_m} (\text{Act}(m', h))).
  \]

- **Presenting a new proof makes histories divide:**
  \[
  (\forall m \in \text{Tree})(\forall h, h' \in H_m)(h \approx_m h' \Rightarrow (\text{Act}(m, h) = \text{Act}(m, h'))).
  \]

- **Presented proofs are epistemically transparent:**
  \[
  (\forall m, m' \in \text{Tree})(R_e(m, m') \Rightarrow (\text{Act}_m \subseteq \text{Act}_{m'})).
  \]
Note that in the justification part of jstit model we required that Tree rather than $MH(\mathcal{M})$ serves as the set of reference points. This reflects the fact that we want to interpret relations between proofs and their proved sentences as accomplished truths not amenable to the agents’ will. Agents may produce a proof, but they may not make a given proof prove something different from what it actually proves.

However, since our semantics is an extension of stit semantics, all of the formulas have to be evaluated at $MH(\mathcal{M})$ including formulas of the form $KA$ and $t:A$. Thus we want to adapt the justification semantics to this new format of evaluation but we ensure these type of formulas end up being moment-determinate, i.e. the reference to histories is vacuous. Therefore, we adopt for them the following clauses:

\[
\mathcal{M}, m, h \models KA \iff \forall m' \forall h'(R(m, m') \& h' \in H_{m'} \Rightarrow \mathcal{M}, m', h' \models A);
\]

\[
\mathcal{M}, m, h \models t:A \iff A \in \mathcal{E}(m, t) \& \forall m'(R_e(m, m') \& h' \in H_{m'} \Rightarrow \mathcal{M}, m', h' \models A).
\]
We now have everything in place to interpret a minimal jstit language, which is the intersection of all the existing jstit languages.

Given an agent community $Ag$, the minimal language $\mathcal{L}_0^{Ag}$ has the set of formulas $Form_0^{Ag}$ given by the following BNF:

$$A := p \in Var \mid \bot \mid A \rightarrow A \mid A \land A \mid A \lor A \mid t: A \mid KA \mid \square A \mid [j]A$$

We can now define justification stit logic as any language $\mathcal{L}^{Ag}$ extending $\mathcal{L}_0^{Ag}$ and interpreted over a class of jstit models for $\mathcal{L}_0^{Ag}$ using the above satisfaction clauses for the modalities of $\mathcal{L}_0^{Ag}$ (i.e. the modalities inherited from parent logics).
\( \mathcal{L}_0^{Ag} \) and its properties

- Strongly complete axiomatization of \( \mathcal{L}_0^{Ag} \) is given by the system \( \mathcal{I}_0 \) extending \( \mathcal{I} \) with (AS1)–(AS3) and the following axiomatic scheme:

\[
KA \rightarrow \Box K \Box A
\]  

(AJS)

representing FAM-constraint

- \( \mathcal{I}_0 \) is stable w.r.t. constant specifications for \( \mathcal{I}_0 \) and also complete w.r.t. unirelational models.

- However, \( \mathcal{I}_0 \) does not have finite model property or even finite history property; the innocent-looking formula \( K(\Diamond p \land \Diamond \neg p) \) is only satisfiable with a model with an infinite history.

- Even though the histories sometimes have to be infinite, they do not need to be too complicated. All of the above completeness claims still hold if we restrict to models based on discrete time (with every history isomorphic to an initial segment of \( \omega \)).
Of course, $\mathcal{L}^A_0$ is not adequate to the potential of jstit models since it has no means to connect with $Act$, the only new element and the center for the whole construction.

In order to complete $\mathcal{L}^A_0$ in this respect, one may add one or more of the following modalities to the language:

<table>
<thead>
<tr>
<th></th>
<th>Implicit</th>
<th>Explicit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Factual</td>
<td>$A$ has been proven</td>
<td>$A$ has been proven by $t$</td>
</tr>
<tr>
<td></td>
<td>$Proven(A)$</td>
<td>$Proven(t, A)$</td>
</tr>
<tr>
<td>Agentive</td>
<td>$j$ proves $A$</td>
<td>$j$ proves $A$ by $t$</td>
</tr>
<tr>
<td></td>
<td>$Prove(j, A)$</td>
<td>$Prove(j, t, A)$</td>
</tr>
<tr>
<td>Chaotic</td>
<td>a proof of $A$ is presented</td>
<td>$t$ is presented</td>
</tr>
<tr>
<td></td>
<td>$EA$</td>
<td>$Et$</td>
</tr>
</tbody>
</table>

Only the basic versions are given above, assuming all modalities are available, many more versions and refinements of these notions can be defined.
The above modalities are interpreted by the following satisfaction clauses:

\[
\begin{align*}
\mathcal{M}, m, h &\models Prove(j, A) \iff (\forall h' \in \text{Choice}^m_j(h))(\exists t \in \text{Act}(m, h'))(\mathcal{M}, m \models t : A) \& \\
&\quad \& (\forall s \in \text{Pol})(\exists h'' \in H_m)(\mathcal{M}, m, h \models s : A \Rightarrow s \notin \text{Act}(m, h''));
\end{align*}
\]

\[
\begin{align*}
\mathcal{M}, m, h &\models Proven(A) \iff (\exists t \in \text{Pol})(\forall h' \in H_m)(t \in \text{Act}(m, h') \& \mathcal{M}, m \models t : A);
\end{align*}
\]

\[
\begin{align*}
\mathcal{M}, m, h &\models Prove(j, t, A) \iff (\forall h' \in \text{Choice}^m_j(h))(t \in \text{Act}(m, h') \& \mathcal{M}, m \models t : A) \\
&\quad \& (\exists h'' \in H_m)(t \notin \text{Act}(m, h''));
\end{align*}
\]

\[
\begin{align*}
\mathcal{M}, m, h &\models Proven(t, A) \iff (\forall h' \in H_m)(t \in \text{Act}(m, h') \& \mathcal{M}, m \models t : A);
\end{align*}
\]

\[
\begin{align*}
\mathcal{M}, m, h &\models Et \iff t \in \text{Act}(m, h);
\end{align*}
\]

\[
\begin{align*}
\mathcal{M}, m, h &\models EA \iff (\exists t \in \text{Pol})(t \in \text{Act}(m, h) \& \mathcal{M}, m, h \models t : A)
\end{align*}
\]
To represent the interplay between proofs-as-objects and proofs-as-acts, one needs to have at least one factual and one agentive modality available.

No axiomatization is known for any interesting combination of explicit and implicit jstit modalities, so the major choice is between the columns of the table.

Choosing the implicit mode typically allows to look no further than $\omega$-ordered histories; but compactness and strong completeness are invariably lost.

Choosing the explicit mode allows to keep strong completeness; however, the complexity of histories in the models goes up.
JA-STIT

- JA-STIT is perhaps one of the most simple-minded extensions of \( L_0^{Ag} \) adding to it the single new jstit ‘modality’ \( Et \).
- \( JS_E \), a strongly complete axiomatization for JA-STIT is obtained by enriching \( JS_0 \) with two new axiomatic schemes:

\[
\square Et \rightarrow K \square Et \tag{AJSE1}
\]

\[
K(\neg \square Et_1 \lor \ldots \lor \neg \square Et_n) \rightarrow (\neg Et_1 \lor \ldots \lor \neg Et_n) \tag{AJSE2}
\]

- Construction of a universal model in the completeness proof shows that we may assume, wlog, that every history is ordered in the type of \((0) \oplus (\omega^* \otimes \omega)\). Moreover, assuming that every copy of \( \omega^* \) is given as the set of negative integers plus 0, we may arrange that histories only branch off at 0’s. Thus every history is at least countable with the branching points along it ordered in the type of \( \omega \).
- To get a system \( JS_{ED} \) complete w.r.t. to the models based on discrete times, one needs to replace (AJSE2) with the following axiom:

\[
K(\neg \square Et_1 \lor \ldots \lor \neg \square Et_n \lor \square Es_1 \lor \ldots \lor \square Es_k) \rightarrow \\
\rightarrow (\neg Et_1 \lor \ldots \lor \neg Et_n \lor Es_1 \lor \ldots \lor Es_k) \tag{AJSE2'}
\]
We may also ask a converse question, i.e. to what extent the adoption of \((AJSE2')\) restricts the underlying frame.

It is not so clear however, which notion of frame will be the right one for this type of enquiry. Since we are speaking about time structure, we will need at least a temporal frame (structure of the type \(\langle Tree, \sqsubseteq \rangle\)). Another natural notion could be stit frame (of the type \(\langle Tree, \sqsubseteq, Choice \rangle\)). Finally, we can consider jstit frames with the structure \(\langle Tree, \sqsubseteq, Choice, R, R_e \rangle\).

\(V\) and \(E\) clearly have to be outside any viable frame notion; we also think that \(Act\) is not a part of any frame, basically since it already allows to evaluate a big part of \(Form^{Ag}\).

If \(F\) is a frame of any of the three above-defined types, we say that \(F\) is a mixed successor frame iff for all \(m, m_1 \in Tree\) it is true that:

\[
[m \triangleleft m_1 \Rightarrow (\exists m_2 \sqsubseteq m_1)(Next(m, m_2))] \lor [(\forall h, g \in H_m)(h \approx_m g)]
\]

If \(F\) is either a stit or a temporal frame, then every model based on \(F\) verifies \((AJSE2')\) iff \(F\) is a mixed successor frame.

However, this is not the case for jstit frames.
The situation with jstit frames is not so clear since epistemic accessibility relations can interact with stit substructures in the most involved ways.

For an $m \in \text{Tree}$, we define $\Theta_m \subseteq 2^{\text{Tree}}$ setting that $S \subseteq \text{Tree}$ is in $\Theta_m$ iff all of the following conditions hold:

1. $m \in S$;
2. $(\forall m_1, m_2 \in \text{Tree})((m_1 \in S \& R_e(m_1, m_2)) \Rightarrow m_2 \in S)$;
3. $(\forall m_1 \in \text{Tree})[(\forall h \in H_{m_1})(\exists m_2 \in h)(\text{Next}(m_1, m_2) \& m_2 \in S) \Rightarrow m_1 \in S]$;
4. $(\forall m_1 \in \text{Tree})[(m_1 \in S \& (\forall m_2 \triangleleft m_1) \exists m_3 (m_2 \triangleleft m_3 \triangleleft m_1)] \Rightarrow (\exists m_4 \triangleleft m_1)(m_4 \in S))$.

We define that a jstit frame $F$ is regular iff the following holds for all $m, m_1 \in \text{Tree}$:

$$\{m \triangleleft m_1 \& (\exists S \in \bigcap_{m \triangleleft m_0 \leq m_1} \Theta_{m_0})(m \notin S \&$$

$$\& (\exists h' \in H_m)((\forall g \in H_{m_1})(h' \not\equiv_m g) \& (\forall m' \in h')[\text{Next}(m, m') \Rightarrow m' \notin S)] \}

\Rightarrow (\exists m_2 \leq m_1)(\text{Next}(m, m_2))$$
The relevance of JA-STIT expressible axioms to the underlying temporal structure of a frame is amazing given that the logic contains no modalities looking outside a given moment, and also that the format of interactions for $Et$ with modalities present in $\mathcal{L}^{Ag}_0$ is rather limited ($Et$ cannot be applied to modalized formulas but can itself be modalized).

However, saying things about the structure of time is not the main goal of JA-STIT. The following slides present some of the more philosophically relevant examples of its expressivity.
Agent $j$ publicly announces $t$: $[j]Et \land \Diamond \neg Et$.

Proofs-as-acts vs proofs-as-objects:

$$Prove(j, t, A) =_{df} [j]Et \land \Diamond \neg Et \land t : A; \quad Proven(t, A) =_{df} \Box Et \land t : A.$$  

so it turns out that the other two explicit jstit modalities are expressible in JA-STIT.

The structure of truth sets for $Et$ is not constrained, but it is natural to view presented proofs as less chaotic than future sea-battles since they are typically due to the agents’ activity. One way to express the dependence of proofs on agents would be to impose the following additional axiom:

$$Et \rightarrow [j_1]Et \lor \ldots \lor [j_n]Et,$$

where $Ag = \{j_1, \ldots, j_n\}$.

The downside will be that the agents cannot collaborate on a proof within the same moment. If one wants to allow for such a collaboration, one needs the group operator $[Ag]$ for the grand coalition to modify the axiom as follows: $Et \rightarrow [Ag]Et$. 

Olkhovikov JA-STIT
Imposing the principle that whenever a sum of proofs is presented to the community, each of the summands is also presented can be captured by the following axiom:

$$E(s + t) \rightarrow Es \land Et.$$  

Imposing the constructive nature of complex proofs (i.e. that we can e.g. apply $s$ to $t$ whenever both $s$ and $t$ have been already constructed), would amount to the following set of principles:

$$(\Box Es \land \Box Et) \rightarrow \Diamond E(s \circ t) \quad \text{for } \circ \in \{+, \cdot\};$$
$$\Box Es \rightarrow \Diamond E(!s).$$

These principles allow for the following strengthening, where we ensure that not only these complex proofs are constructible but also the agents are able to ensure their construction by making the right choices:

$$(\Box Es \land \Box Et) \rightarrow \Diamond [c]_j E(s \circ t) \quad \text{for } \circ \in \{+, \cdot\};$$
$$\Box Es \rightarrow \Diamond [c]_j E(!s).$$
The strong completeness property of all the jstit systems explained above ($\mathcal{JS}_0, \mathcal{JS}_E, \mathcal{JS}_{ED}$) is stable w.r.t. constant specifications and restriction to unirelational models.

Let $L \in \{\mathcal{JS}_0, \mathcal{JS}_E, \mathcal{JS}_{ED}\}$ and let $CS$ be a constant specification for $L$. Then:

- If $A \in \text{StitForm}^{Ag}$, then $A$ is provable in $L(CS)$ iff $A$ is provable in $S$.
- If $CS$ is also a constant specification for $J$ and $A \in J\text{Form}$, then $A$ is provable in $L(CS)$ iff $A$ is provable in $J(CS)$.
- If $CS$ is appropriate for $L$ and $A \in \text{Form}^{Ag}$ is provable in $L(CS)$, then there exists a closed $t \in Pol$, such that $t:A$ is also provable in $L(CS)$.

Under the same assumptions, if

$B_1 \rightarrow \ldots \rightarrow B_n \rightarrow t_1:C_1 \rightarrow \ldots \rightarrow t_m:C_m \rightarrow A$ is provable in $L(CS)$, then for all $s_1, \ldots, s_n \in Pol$ there exists a closed $t_0 \in Pol$ such that:

$s_1:B_1 \rightarrow \ldots \rightarrow s_n:B_n \rightarrow t_1:C_1 \rightarrow \ldots \rightarrow t_m:C_m \rightarrow$

$\rightarrow ((((((t_0 \cdot s_1) \cdot \ldots) \cdot s_n) \cdot t_1) \cdot \ldots) \cdot t_m:A$.

is also provable in $L(CS)$. 
The above hopefully shows that justification stit logics have interesting potential.

Now that we have a bunch of axiomatizations, a more nuanced work, both formally and philosophically, may commence.

As far as forging of further axiomatizations is concerned, the following tasks seems to be the most promising:

**Task 1.** Axiomatizing an interesting logic combining implicit and explicit modalities. The natural candidate for the first effort is the logic induced by the two chaotic jstit modalities $Et$ and $EA$.

- This is my current work in progress;
- Enhances expressivity in an interesting way, e.g. allows to express Disjunction Property: $E(A \lor B) \rightarrow EA \lor EB$.
- Further extension with the logic of Priorean past time allows to express all the 6 jstit modalities from the above table.
Task 2. Extending the existing axiomatizations with group operators.

Task 3. Adapting the existing axiomatizations other types of semantics for justification logic, e.g. to the semantics introduced by M. Fitting for the quantified logic of evidence. In this way the compactness of the axiomatizations for implicit modalities is likely to be restored.
Thank you very much for your attention!
S. Artemov.
Operational modal logic.  

S. Artemov and E. Nogina.
Introducing justification into epistemic logic.  

Alternative axiomatics and complexity of deliberative stit theories.  

N. Belnap, M. Perloff, and M. Xu.
M. Fitting.
A quantified logic of evidence

G. Olkhovikov and H. Wansing.
Inference as doxastic agency. Part I: The basics of justification stit logic.

G. Olkhovikov and H. Wansing.
Inference as doxastic agency. Part II: Ramifications and refinements.
It is a typical situation in justification logics that every new logical operation or principle is mirrored in the realm of proof polynomials.

In jstit logic this correspondence is not maintained since all the new modalities and notions do not seem to have their proof polynomial companions.

This is partly justified for $[j]$ and the agentive half of proving modalities, as is witnessed by the following validities:

\[ \neg t: ([j]A \land \neg \Box A); \neg t: (Prove(j, s, A)); \neg t: (Prove(j, A)). \]

For some other modalities, like $\Box$ and the moment-determinate proving modalities the situation is less clear, since they lead to sentences that can be thought of as objects of proving activity. We briefly illustrate the available options for $\Box$ which is, perhaps, the simplest of these cases.

The main question here is: when we have a proof of $A$, does this same proof count as a proof of $\Box A$ or the latter will require a different proof.
A modal logician could say that a proof of $\Box A$ would require an additional application of necessitation rule and thus must be different. This approach would lead to introducing an operation $\Box$ on proof polynomials with the following additional constraint on $E$:

$$A \in E(m, t) \Rightarrow \Box A \in E(m, \Box(t)).$$

However, a different solution is also possible. Classical mathematics has often been construed along the lines that whatever is proven, is proven to hold necessarily, so that the proof that $A$ and the proof that, necessarily, $A$ would be one and the same proof. This idea amounts to the following additional restriction on $E$:

$$A \in E(m, t) \Rightarrow \Box A \in E(m, t),$$

for every formula $A$ and proof polynomial $t$.

Note that some validities associated with this case are $t:A \rightarrow \Box t:A$ and $t:A \rightarrow \Box A$. We have the same validities for $K$: $KA \rightarrow \Box KA$ and $KA \rightarrow \Box A$. However, for $K$ we also have $KA \rightarrow \Box K \Box A$. Now the second option also ensures that $t:A \rightarrow \Box t \Box A$ is a validity introducing a sort of nice symmetry.
Appendix 2: more axiomatizations 1 of 4

A complete axiomatization for implicit jstit logic (induced over $\mathcal{L}^{Ag}_0$ by $Prove(j, A)$ and $Proven(A)$) is given by extending $JS_0$ with the following axioms:

\begin{align*}
Prove(j, A) & \rightarrow (\neg Proven(A) \land [j]Prove(j, A) \land KA) \quad (AI1) \\
\Box Prove(j, A) & \rightarrow \Box Prove(i, A) \quad (AI2) \\
Proven(A) & \rightarrow (KProven(A) \land KA) \quad (AI3) \\
\neg K(\bigvee_{l=1}^{n} \langle K \rangle \Diamond Prove(j_l, A_l)) & \quad (AI4) \\
\neg Prove(j, A) & \rightarrow \langle j \rangle( \bigwedge_{i \in Ag} \neg Prove(i, A)) \quad (AI5) \\
K(\bigvee_{i=1}^{n} \neg Proven(B_i)) & \rightarrow \bigvee_{i=1}^{n} (\bigwedge_{j \in Ag} \neg Prove(j, B_i)) \quad (AI6)
\end{align*}
This axiomatization has what may be called *restricted strong completeness* w.r.t. the class of jstit models in the following sense. For a given \( X \subseteq PVar \) let \( \text{Form}^A_X \) be the set of formulas in which only proof variables from \( X \) occur. Then:

- Let \( X \subseteq PVar \) be such that \( PVar \setminus X \) is countably infinite. Then an arbitrary \( \Gamma \subseteq \text{Form}^A_X \) is consistent iff it is satisfiable in a jstit model.
- From this result, weak completeness easily follows.
- This result cannot be strengthened to a full strong completeness while staying within a finitary notion of proof since implicit jstit logic lacks *compactness*. E.g. the set \( \{\text{Proven}(p)\} \cup \{\neg t : p \mid t \in Pol\} \) is finitely satisfiable but not satisfiable.
- It follows from the proof of the above result that implicit jstit logic is also complete w.r.t. the class of jstit models based on discrete time.
A complete axiomatization for implicit jstit logic (induced over $\mathcal{L}_0^{Ag}$ by $Prove(j, t, A)$ and $Proven(t, A)$) is given by extending $\mathcal{JS}_0$ with the following axioms:

\[
Prove(j, t, A) \to (\neg Proven(t, A) \land [j]Prove(j, t, A) \land \\
\neg \Box Prove(j, t, A) \land t:A) \quad (AE1)
\]

\[
(Prove(j, t, A) \land t:B) \to Prove(j, t, B) \quad (AE2)
\]

\[
Proven(t, A) \to (KProven(t, A) \land t:A) \quad (AE3)
\]

\[
(Proven(t, A) \land t:B) \to Proven(t, B) \quad (AE4)
\]

\[
\neg Prove(j, t, A) \to \langle j \rangle (\bigwedge_{i \in Ag} \neg Prove(i, t, A)) \quad (AE5)
\]

\[
K(\bigvee_{i=1}^{n} \neg Proven(t_i, B_i)) \to \bigvee_{i=1}^{n} (\bigwedge_{j \in Ag} \neg Prove(j, t_i, B_i)) \quad (AE6)
\]
This axiomatization is strongly complete w.r.t. the class of jstit models (and the logic is thus compact).

However, unlike its implicit counterpart, explicit jstit logic can tell the difference between the general class of models and the class of models based on discrete time. The above system is therefore not complete w.r.t. to discrete-time models.

Here is one example of a formula which is valid if discrete time structure is assumed, but not valid generally:

\[ K(\neg \text{Proven}(x, p) \lor \text{Proven}(y, q)) \rightarrow (\neg \text{Prove}(j, x, p) \lor \]
\[ \lor (y : q \rightarrow (\text{Proven}(y, q) \lor \text{Prove}(j, y, q))) \]

We do not know at the moment how one can extend the above system to capture the set of explicit validities over discrete-time models.
Appendix 3: alternative versions of jstit modalities and their definitions 1 of 2

\[ M, (m, h) \models Prove'(j, A) \iff \]
\[ \iff (\forall h' \in \text{Choice}_j^m(h)) (\exists t \in \text{Act}(m, h')) (M, (m, h') \models t : A) \& \]
\[ \& (\exists h'' \in H_m) (\forall s \in \text{Act}(m, h'')) (M, (m, h'') \not\models s : A). \]

(1)

\[ M, (m, h) \models Prove'(j, t, A) \iff \]
\[ \iff (\forall h' \in \text{Choice}_j^m(h)) (t \in \text{Act}(m, h') \& M, (m, h') \models t : A) \& \]
\[ \& (\forall s \in \text{Pol}) (\exists h'' \in H_m) (M, (m, h'') \models s : A \Rightarrow s \notin \text{Act}(m, h'')). \]

(2)

\[ M, (m, h) \models Prove''(j, t, A) \iff \]
\[ \iff (\forall h' \in \text{Choice}_j^m(h)) (t \in \text{Act}(m, h') \& M, (m, h') \models t : A) \& \]
\[ \& (\exists h'' \in H_m) (\forall s \in \text{Act}(m, h'') (M, (m, h'') \not\models s : A). \]

(3)
Appendix 3: alternative versions of jstit modalities and their definitions 2 of 2

\[ \text{Prove}'(j, A) =_{df} \text{Prove}(j, A) \land \Diamond \neg \text{Prove}(j, A); \]

\[ \text{Prove}'(j, t, A) =_{df} \text{Prove}(j, t, A) \land \neg \text{Proven}(A); \]

\[ \text{Prove}''(j, t, A) =_{df} \text{Prove}'(j, t, A) \land \Diamond \neg \text{Prove}(j, A). \]
One way to motivate the choice of concrete actions plus abstract proof objects is to begin with the principle:

**Proofs-as-acts are proofs in virtue of realizing proof objects**

Then note that generic actions not so much realize the objects/states-of-affairs themselves as are *co-realized* with those objects/states-of-affairs by concrete actions.

Thus, action type *walking* can only lead to ‘Jill getting to her office’ if some concrete action by Jill happens both to bring Jill to her office and be of type *walking*.

In this way, action types only realize objects in some secondary sense, not unlike the sense in which *human* and *horse* are substances in a secondary sense as compared to concrete humans and horses.

In this way, one may decide to choose concrete actions, rather than action types, as primary.
Turning now to proof objects, existence of realized objects cannot be conceived without actions or processes realizing them. On the other hand, abstract objects, as theoretical possibilities of a proof, may exist without there being any processes or actions at all.

In this way, once the choice is made for concrete actions, their appropriate complement would be abstract proof objects, whereas realized proof objects are necessarily construed as results of interaction between these two categories (and represented by modalities like $\text{Proven}(t, A)$ and $\text{Proven}(A)$).