# TOWARDS RELIABLE GENERAL-PURPOSE SPEECH EDITING AGENTS: FORMULATION, DESIGN, AND EMPIRICAL ANALYSIS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

General-purpose speech editing should satisfy requested changes while preserving content and acoustic properties that should remain unchanged. Specialized speech models provide complementary capabilities, but reliably composing them requires accurate observation, robust interpretation of complex editing intent, executable tool composition, and assessment of realized outcomes. We formulate agentic speech editing as a partially observable Markov decision process (POMDP), separating latent editing states, observations, belief, actions, environment transitions, reward, and policy revision. Guided by this formulation, we introduce X·EDIT, a training-free speech editing agent that combines active evidence acquisition, optional belief materialization, structured tool planning, outcome evaluation, and feedback-guided contextual policy revision. On all 426 MMAE speech tasks, X·EDIT achieves 54.30% instruction following, 93.05% consistency, and 20.89% exact match. Requiring both complete delivery and passing all native checks yields 20.19% delivered success. Controlled studies further show that acquired conditioning evidence improves fine-grained observation fidelity, explicit belief helps retain complex interacting requirements, and higher-level action representations reduce the burden of composing fragmented speech tools. Contextual policy revision substantially improves delivery, while gains in complete fulfillment remain smaller, highlighting reliable preservation, reward estimation, and candidate selection as remaining challenges. Together, our formulation, system design, and empirical analysis provide a principled framework and practical guidance for building reliable general-purpose speech editing agents.[1]

## 1 INTRODUCTION

Speech synthesis has achieved high naturalness and controllability (Hu et al., 2026a; Li et al., 2023; Yang et al., 2024; Chen et al., 2025), yet editing an existing recording remains substantially more difficult. A single request may involve content, speaker, prosody, acoustic events, or audio composition, while requiring unrelated parts of the recording to remain unchanged. For example, removing one speaker's cough requires identifying the speaker and event, localizing their temporal relation, and modifying the event without damaging the speech. Such requirements become even harder across multiple turns and audio inputs.

No single speech model reliably covers this full range of operations. Instead, recognition, localization, separation, generation, and signal processing are supported by increasingly capable but specialized models and tools (Yan et al., 2025a;b; Ma et al., 2026a; Wang et al., 2026a). This makes an agentic approach attractive: an agent can acquire task-relevant information, compose complementary tools, and revise its decisions after observing their outcomes (Kuan et al., 2024; Wang et al., 2024; Yao et al., 2023; Shinn et al., 2023). However, general speech editing then becomes a sequential decision problem. The true editing state is only partially observed, observations can be

[1] Anonymous code: https://anonymous.4open.science/r/X-Edit-Anonymous-3FAB; anonymous demo: https://speech-editing-examples.anonymous-demo-20260926.workers.dev.



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

incomplete or incorrect, complex user intent must be translated into executable tool compositions, and failures must be interpreted before they can influence subsequent decisions.

We therefore formulate agentic general-purpose speech editing as a partially observable Markov decision process (POMDP) (Young et al., 2013). The formulation separates the latent editing state, observations, actions, state transitions, reward, belief, and policy. For speech editing, actions include information extraction, audio operations, and control. Extraction can provide additional conditioning information to the observation process before the agent commits to an edit. Rather than explicitly maintaining the full Bayesian posterior, an agent may use neural models to summarize decision-relevant state information and to approximate task-relevant observation and outcome models. The effective policy is induced jointly by its models, contexts, tools, and workflow, and can therefore be revised through context without changing model parameters. This provides a common formulation for perception, decision making, tool execution, evaluation, and subsequent policy revision.

Guided by this formulation, we develop X·EDIT, a training-free general-purpose speech editing agent. Perceptor realizes the observation process using an audio-language model together with optional specialist outputs or targeted perceptual queries. Reasoner can explicitly materialize a language-level belief summary, while Planner converts the current interpretation into executable editing programs. Executor and speech tools realize state transitions. After execution, task-specific rubrics and Evaluators assess the realized outcome, and Guider converts detected failures into contextual revisions of subsequent policies. Thus, X·EDIT implements the POMDP interaction as a bounded agentic editing loop without requiring parameter updates.

Our experiments follow the same decision process. ***RQ1*** asks whether composing heterogeneous speech capabilities through a complete agent provides an effective route to general-purpose editing, jointly improving requested edits and preservation. ***RQ2*** studies partial observability: whether targeted evidence acquisition improves task-relevant observations compared with simply reviewing existing information. ***RQ3*** studies decision construction: how explicit belief materialization and action abstraction affect the translation of complex editing intent into executable compositions of fragmented speech tools. ***RQ4*** studies contextual policy revision: whether realized failures can improve subsequent decisions, and how runtime, semantic, and audio feedback address different failure modes.

On all 426 MMAE (Ma et al., 2026b) speech tasks, X·EDIT achieves the strongest instruction-following, consistency, and exact-match results among the evaluated references, reaching 54.30% IFR, 93.05% CR, and 20.89% EMR with up to three attempts, together with 87.56% delivery. The controlled studies further clarify the mechanisms behind these results. Targeted conditioning evidence improves fine-grained event localization and speaker-attributed transcription, whereas additional review provides no consistent benefit. Explicit belief improves the retention of complex interacting requirements in controlled settings, although this does not translate uniformly to better end-to-end editing. Higher-level action representations reduce the mechanical burden of composing fragmented tools and improve structural executability, while correct compilation cannot repair an incorrect editing decision. Finally, contextual policy revision improves overall reliability, with runtime, semantic, and audio feedback providing complementary signals for different classes of failure.

Our contributions are threefold.

**A unified formulation for agentic speech editing.** We formulate general-purpose speech editing as a POMDP that connects partial observation, belief approximation, information acquisition, tool-based state transitions, outcome assessment, and contextual policy revision within one sequential decision process.

**A formulation-guided speech editing agent.** We instantiate this formulation in X·EDIT, a training-free agent that integrates specialist perception, optional explicit belief reasoning, structured action construction, heterogeneous speech tools, and feedback-guided policy revision in a bounded interaction loop.

**Empirical insights into general-purpose speech editing agents.** Through complete-agent evaluation and controlled studies corresponding to observation, decision construction, and policy revision, we identify where targeted information, explicit state summarization, action abstraction, and outcome feedback improve general-purpose speech editing and where errors remain.

## 2 PROBLEM FORMULATION

### 2.1 PRELIMINARY

We start from a standard partially observable Markov decision process (POMDP) (Smallwood & Sondik, 1973; Kaelbling et al., 1998):

$$\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, R, \mathcal{O}, Z, \gamma, b_0). \tag{1}$$

Here $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces. $T(s' \mid s, a)$ governs state transitions, while $Z(o \mid s', a)$ governs observations of the resulting state. $R(\cdot \mid s, a)$ denotes the reward law, $\gamma \in [0, 1]$ the discount factor, and $b_0$ the initial belief. The usual scalar reward function can be recovered as the expectation of $R$.

At decision step $t$, the environment occupies a latent state $s_t$. The agent maintains a belief $b_t$ over possible states and selects $a_t$ accordingly. The action induces a transition to $s_{t+1}$ and a reward $r_{t+1}$, after which the agent receives an observation $o_{t+1}$ of the resulting state and updates its belief for the next decision. Under the Markov assumption, the transition and observation depend only on the current state, action, and resulting state as specified by $T$ and $Z$.

### 2.2 POMDP-BASED FORMULATION OF SPEECH EDITING AGENTS

**State, observation, and action.** For speech editing, $s_t$ represents the current audio assets and task-relevant state of the editing process(e.g. source audios and user instructions). The agent does not directly access the complete $s_t$; it acts on observations generated through $Z$. Partial observability therefore includes not only noisy measurements, but also missing or ambiguous information about variables such as speaker identity, event location, temporal overlap, or the realized acoustic effect.

We divide the action space into

$$\mathcal{A} = \mathcal{A}_{\text{extract}} \cup \mathcal{A}_{\text{operate}} \cup \mathcal{A}_{\text{control}}. \tag{2}$$

Extraction actions acquire information for the observation process, for example by invoking specialist models, retrieving structured acoustic evidence, or issuing a targeted perceptual query. Operation actions modify or generate audio through editing tools, while control actions restore a recorded state or terminate the episode.

The environment interaction follows

$$s_{t+1} \sim T(\cdot \mid s_t, a_t), \qquad o_{t+1} \sim Z(\cdot \mid s_{t+1}, a_t). \tag{3}$$

An extraction action need not modify the underlying waveform. Instead, it can change the information available to the observation process, allowing the next observation to expose previously unresolved task-relevant properties. The policy can therefore acquire additional evidence before committing to an editing operation.

**Belief approximation.** In a classical POMDP, the belief is a probability distribution over the latent state, $b_t(s) = \Pr(s_t = s)$, ais updated through Bayesian filtering using the transition and observation models.

We use this as an ideal inference reference rather than explicitly representing the complete posterior state space. Instead, the agent uses a neural belief estimator $\mathcal{B}_\phi$ to summarize decision-relevant information:

$$\hat{b}_{t+1} \sim \mathcal{B}_\phi(\cdot \mid \hat{b}_t, a_t, o_{t+1}). \tag{4}$$

The resulting $\hat{b}_t$ is a language-level representation of the current editing state, not a calibrated probability distribution. In the explicit mode this representation is materialized; in the implicit mode the same inference role is absorbed into Planner without exposing a separate $\hat{b}_t$.

Section 3 instantiates the observation and reward models as $\widehat{Z}_{\theta,\phi}$ and $\widehat{\mathcal{R}}_{\theta,\phi}$.

**Agent system and policy.** An agent is determined jointly by its models, contexts, tools, and control workflow. Following ang Gao et al. (2026), We write the agent system as

$$\Pi_t = (\Gamma_t, \{\psi_{i,t}\}, \{C_{i,t}\}, \{\mathcal{W}_{i,t}\}), \tag{5}$$

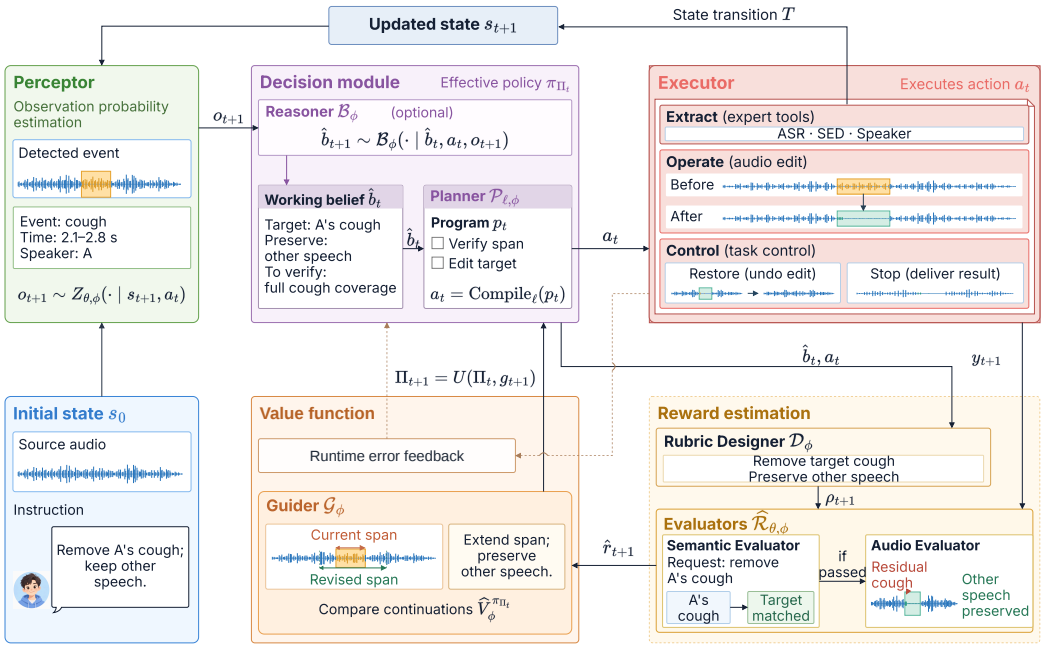



Figure 1: The overview framework of X·EDIT.

where $\Gamma_t$ denotes the workflow, $\psi_{i,t}$ the model parameters, $C_{i,t}$ the context available to each component, and $\mathcal{W}_{i,t}$ its available tools. These components jointly induce the effective policy

$$a_t \sim \pi_{\Pi_t}(\cdot \mid \hat{b}_t). \tag{6}$$

Policy revision therefore does not require updating model weights. Let $g_{t+1}$ denote a revision signal obtained after the realized outcome, and let $U$ be the agent-system update operator:

$$\Pi_{t+1} = U(\Pi_t, g_{t+1}). \tag{7}$$

In general, $U$ may modify model parameters, context or memory, available tools, or the control workflow, each of which can induce a different effective policy. In X·EDIT, Section 3 specializes this update to contextual revision within an episode.

**Reward and value.** After action $a_t$, the environment produces a reward $r_{t+1} \in \mathcal{R}$ according to

$$r_{t+1} \sim R(\cdot \mid s_t, a_t), \tag{8}$$

where $\mathcal{R}$ may be scalar, multidimensional, or structured to encode task-relevant properties of the realized outcome.

In the standard scalar case, value $V$ summarizes expected future reward. However, for structured rewards, value can instead be represented by a task-dependent function of the induced future-outcome distribution and need not reduce to a single scalar.

Our editing controller does not numerically accumulate evaluator scores from successive candidates or retries. Each action produces a new state and realized outcome, which is assessed again; a superseded candidate's score is not added to the score of the next candidate.

## 3 X·EDIT: A POMDP-BASED SPEECH EDITING AGENT

X·EDIT instantiates the preceding formulation with fixed audio-model weights $\theta$ and text-model weights $\phi$. Executor and editing tools realize state transitions; Perceptor provides observations; Reasoner and Planner interpret observations and choose actions; Rubric Designer and Evaluators assess realized outcomes; and Guider revises subsequent decisions. These modules approximate

the formal objects in Section 2 through conditional generative mappings rather than explicit numerical POMDP inference or calibrated beliefs, rewards, and values. Algorithm 1 in Appendix B.6 summarizes the editing process.

### 3.1 Perceptor: realizing the environment observation

Perceptor realizes the observation model $Z$ with fixed audio-language and text models. Although general audio-language models provide broad acoustic and semantic understanding, fine-grained speech editing often requires more precise speaker, event, and temporal grounding; direct perception can therefore omit or mislocalize decision-relevant information.

We denote the implemented observation model by $\widehat{Z}_{\theta,\phi}$. When evidence is insufficient, the policy may select $a_t^{\text{extract}} \in \mathcal{A}_{\text{extract}}$ to provide additional conditioning evidence $e_{t+1}$:

$$e_{t+1} \sim \mathcal{E}(\cdot \mid s_t, a_t^{\text{extract}}), \qquad o_{t+1} \sim \widehat{Z}_{\theta,\phi}(\cdot \mid s_{t+1}, a_t^{\text{extract}}, e_{t+1}). \tag{9}$$

Here $e_{t+1}$ is an implementation-level conditioning variable, not an additional POMDP observation. It may contain specialist outputs such as transcripts, alignments, speaker segments, and event proposals, or a task-specific perceptual query directing the same audio-language model toward a missing state property; without extraction, the same observation process operates without $e_{t+1}$.

For example, direct perception may detect a cough without reliably identifying its speaker or temporal extent, while event proposals, speaker-attributed transcription, or a focused query can provide the missing grounding. Extraction can therefore be invoked iteratively when decision-relevant uncertainty remains unresolved. Unlike additional review, which reprocesses existing evidence, extraction changes the conditioning information available to $\widehat{Z}_{\theta,\phi}$. RQ2 evaluates whether this improves task-relevant observation fidelity.

### 3.2 Reasoner and Planner: approximating belief and policy

In the explicit mode, Reasoner maintains a language-level belief approximation:

$$\hat{b}_0 \sim \mathcal{B}_\phi(\cdot \mid s_0), \tag{10}$$

$$\hat{b}_{t+1} \sim \mathcal{B}_\phi(\cdot \mid \hat{b}_t, a_t, o_{t+1}), \tag{11}$$

where $a_t$ reflects the previous transition and $o_{t+1}$ provides evidence about the resulting state. The resulting $\hat{b}_t$ is a language representation rather than a calibrated probability distribution.

Planner maps the resulting interpretation to an executable action:

$$p_{t+1} \sim \mathcal{P}_{\ell,\phi}(\cdot \mid \hat{b}_{t+1}), \qquad a_{t+1} = \text{Compile}_\ell(p_{t+1}), \tag{12}$$

$$p_{t+1} \sim \mathcal{P}^{\text{implicit}}_{\ell,\phi}(\cdot \mid \hat{b}_t, a_t, o_{t+1}), \qquad a_{t+1} = \text{Compile}_\ell(p_{t+1}), \tag{13}$$

where the second line gives the implicit mode, in which belief revision and program generation occur in the same Planner call. Thus, the two modes differ in whether the belief approximation is explicitly materialized before action construction.

FunctionIR, GraphIR, and Command provide three action representations. FunctionIR specifies editing effects, scope, and composition; GraphIR represents operation dependencies; and Command directly lists ordered tool calls. Compilation maps each representation to an executable action.

### 3.3 Reward estimation and contextual policy revision

After an operation, the environment reaches a new state with an underlying reward $r_{t+1}$ that is not directly observable by X·EDIT. For a delivered result, Rubric Designer first constructs task-specific criteria and Evaluators then estimate the realized outcome:

$$\rho_{t+1} \sim \mathcal{D}_\phi(\cdot \mid \hat{b}_t, a_t), \tag{14}$$

$$\hat{r}_{t+1} \sim \widehat{\mathcal{R}}_{\theta,\phi}(\cdot \mid \rho_{t+1}, y_{t+1}, a_t). \tag{15}$$

In the implicit mode, the rubric is generated from the task information available directly to Planner without an explicit $\hat{b}_t$. Fixing the rubric before assessment separates what should be checked from how the result is judged. Semantic evaluation checks the intended edit and supporting operations; audio evaluation, when required, further checks the realized acoustic effect and preservation constraints. Thus, $\hat{r}_{t+1}$ estimates the task outcome but need not equal $r_{t+1}$.

Guider uses this estimate to compare continuations under the current policy. Its implicit continuation-value estimate need not be scalar and is realized through language-level comparison of possible continuations:

$$\widehat{V}_{\phi}^{\pi_{\Pi_t}}(\hat{b}_t, \hat{r}_{t+1}), \tag{16}$$

$$g_{t+1} \sim \mathcal{G}_{\phi}(\cdot \mid \hat{b}_t, \hat{r}_{t+1}, \pi_{\Pi_t}). \tag{17}$$

The resulting $g_{t+1}$ revises the information available to subsequent decisions rather than model parameters. If execution fails before delivery, Rubric Designer, Evaluators, and Guider are bypassed and

$$g_{t+1} = G_{\text{err}}(e_{t+1}^{\text{err}}). \tag{18}$$

Finally,

$$\Pi_{t+1} = U(\Pi_t, g_{t+1}), \qquad \pi_{\Pi_t} \to \pi_{\Pi_{t+1}}. \tag{19}$$

Within an episode, model weights, available tools, and the control workflow remain fixed; revision is realized through the context supplied to subsequent model calls.

### 3.4 Bounded interaction and candidate selection

The preceding components operate within a bounded interaction loop. At each step, the current policy selects an extraction, operation, or control action; execution realizes the transition, Perceptor provides the resulting observation, and outcome feedback may revise the subsequent policy. Interaction ends when the agent stops, the budget is exhausted, or no further revision is available.

Each completed result is stored as $y \in \mathcal{E}$. For delivered candidates, Evaluators provide $\hat{r}(y)$, from which the fixed selection proxy $\widehat{R}(y)$ is derived. At termination,

$$\mathcal{E}_{\text{elig}} = \{y \in \mathcal{E} : \text{Eligible}(y) = 1\}, \tag{20}$$

$$\hat{y} = \arg\max_{y \in \mathcal{E}_{\text{elig}}} \widehat{R}(y). \tag{21}$$

If $\mathcal{E}_{\text{elig}} = \varnothing$, the source audio is returned with an incomplete status rather than counted as successful delivery.

The protocol permits one initial complete attempt and at most two retries, each following the same loop under the policy revised by previous outcome feedback. Candidates across attempts remain in $\mathcal{E}$, and final selection occurs only after policy revision terminates. Thus, policy revision controls how later candidates are generated, whereas $\widehat{R}$ only selects among realized eligible candidates.

## 4 Experiments

### 4.1 Experimental setup

**Models, data, and protocol.** In X-EDIT, Qwen3.8-27B (Qwen Team, 2026) serves as Reasoner, Planner, Semantic Evaluator, Rubric Designer, and Guider, while Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025) serves as Perceptor and Audio Evaluator. Unless otherwise specified, the agent allows the initial perception and planning round plus at most two additional rounds, and one initial complete execution plus at most two retries. The main evaluation uses the 426-task MMAE speech subset (Ma et al., 2026b). Module inputs and decoding settings are detailed in Table 3 and Appendix B.1.

Table 1: **Overall performance on MMAE speech.** † denotes published results (Ma et al., 2026b). All values are percentages (%).

| System | DR | IFR | CR | EMR | $D \cap E$ |
|---|---|---|---|---|---|
| *Single models* | | | | | |
| Step-Audio-EditX† | 100.00 | 43.52 | 77.27 | 4.69 | 4.69 |
| Ming-UniAudio† | 100.00 | 34.13 | 76.01 | 7.04 | 7.04 |
| AuK | 100.00 | $40.32 \pm 1.13$ | $85.03 \pm 0.31$ | $7.67 \pm 0.82$ | $7.67 \pm 0.82$ |
| AuK-Flash | 100.00 | $39.18 \pm 0.47$ | $86.23 \pm 0.84$ | $7.43 \pm 0.36$ | $7.43 \pm 0.36$ |
| *Agentic frameworks* | | | | | |
| AuK+PromptEnhancer | 66.43 | $46.16 \pm 1.33$ | $85.93 \pm 0.33$ | $11.74 \pm 1.88$ | $9.78 \pm 2.00$ |
| **X·EDIT (Ours, *single attempt*)** | 81.69 | $53.14 \pm 0.59$ | $92.25 \pm 0.44$ | $20.58 \pm 0.49$ | $19.72 \pm 0.62$ |
| **X·EDIT (Ours, *up to 3 attempts w. revision*)** | 87.56 | $54.30 \pm 0.07$ | $93.05 \pm 0.26$ | $20.89 \pm 1.24$ | $20.19 \pm 1.24$ |

**Diagnostic data and controls.** RQ2 combines 256 constructed speech–event mixtures with real meeting excerpts: 597 from AMI (Carletta et al., 2006) and 689 from AliMeeting (Yu et al., 2022). Event reference intervals follow isolated tracks and mixing offsets; meeting annotations supply speaker-attributed transcription references. We measure event mIoU, AMI cpWER, and AliMeeting cpCER (Appendix B.2).

RQ3 complements MMAE trials with 200 constructed decision contexts from 20 template groups, equally divided between English and Chinese. Each specifies instructions, version history, and observed content. We check target and plan consistency under fixed inputs without generating audio. The action-representation study compares MMAE continuations from a shared planning prefix. RQ4 compares actual initial MMAE outputs with selected results after at most two feedback-guided retries. The matched inputs, evaluation rules, and candidate-selection protocol are specified in Appendix B.3.

**Baselines and metrics.** We compare with published Step-Audio-EditX (Yan et al., 2025b) and Ming-UniAudio (Yan et al., 2025a) results and local runs of AuK (Ma et al., 2026a), AuK-Flash, and AuK+PromptEnhancer. MMAE scoring follows the official questions and aggregation protocol. Fixed outputs are scored in three independent batches, and we report the mean and batch standard deviation for instruction-following rate (IFR), consistency rate (CR), and exact-match rate (EMR), together with delivery rate (DR).

Because native scoring can evaluate source audio when a required output is not delivered, EMR alone does not imply successful delivery. We therefore additionally report $S = \frac{1}{N}\sum_{i=1}^{N} D_i E_i (N = 426)$, where $D_i$ indicates complete delivery and $E_i$ indicates passing all native checks. We denote this delivered-and-passed metric by $D \cap E$.

### 4.2 RQ1: Complete-agent editing performance

RQ1 asks whether an agentic framework can effectively combine heterogeneous speech-editing tools to satisfy requested edits while preserving content that should remain unchanged. Table 1 shows that X·EDIT consistently improves the quality metrics most closely related to these two objectives. Under the three-attempt budget, it reaches 54.30% IFR, 93.05% CR, and 20.89% EMR, the strongest results among the evaluated references.

These gains indicate that composing specialized capabilities through an agent can improve both edit fulfillment and preservation beyond what the individual editing models achieve in isolation. The result is important because the agent does not introduce a new general-purpose editing model; instead, it obtains broader effective editing capability by selecting, grounding, and composing existing tools under a shared decision process.

The remaining gap is reliability rather than editing coverage alone. Although the three-attempt configuration reaches 87.56% delivery, only 20.19% of all tasks are both delivered and pass every native check. Thus, agentic composition is a promising route toward general-purpose speech editing, but reliable perception, decision construction, and policy revision remain necessary before the integrated tool capability can be consistently realized. RQ2–RQ4 analyze these three sources of failure in turn.

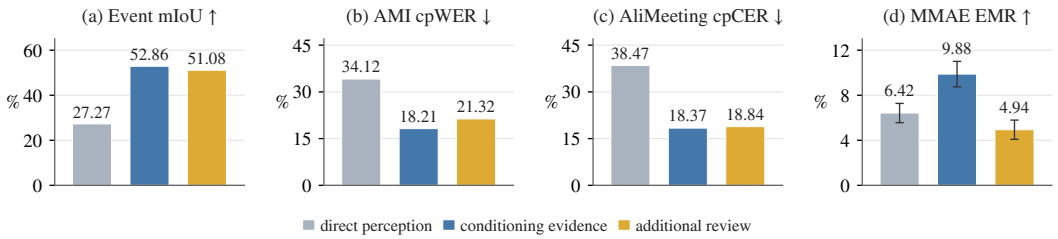



Figure 2: **Observation quality under direct perception, conditioning evidence, and additional review.**

Table 2: **Effect of explicit belief materialization.** All values are percentages (%).

| Belief | *MMAE editing (426)* | | | | | *Fixed-context (200)* | |
|---|---|---|---|---|---|---|---|
| | Delivery | IFR | CR | EMR | $D \cap E$ | TC | PC |
| Implicit | 81.92 | $53.80 \pm 1.14$ | $91.67 \pm 0.42$ | $19.41 \pm 0.98$ | $18.70 \pm 1.20$ | 82.50 | 76.50 |
| Explicit | 79.11 | $52.60 \pm 0.67$ | $92.82 \pm 0.30$ | $19.41 \pm 0.68$ | $18.15 \pm 0.75$ | 86.50 | 83.00 |

### 4.3 RQ2: Active Evidence Acquisition under Partial Observability

RQ2 asks whether an agent can reduce task-relevant partial observability when direct audio-language perception lacks the evidence required for an editing decision. Extraction can either provide structured conditioning evidence from a specialist model or issue a targeted perceptual query for an unresolved state property. We contrast these targeted acquisitions with additional review, which reprocesses an existing observation without explicitly acquiring the missing information.

**Targeted evidence acquisition improves observation fidelity.** Figure 2 shows that conditioning Perceptor on task-relevant evidence substantially improves both event localization and speaker-attributed transcription. Importantly, useful conditioning information can remain below the decision threshold of a standalone specialist, yet still improve the observation when exposed to $\widehat{Z}_{\theta,\phi}$. By contrast, additional review is not consistently beneficial and can corrupt an already useful observation.

Detailed observation metrics, threshold analysis, and revision statistics are reported in Appendix B.2.

**Iterative extraction can recover missing information.** Complete-agent runs further show that allowing an additional extraction-and-decision opportunity can recover information missing from the initial observation and substantially improve downstream editing. Further rounds, however, provide no monotonic gain.

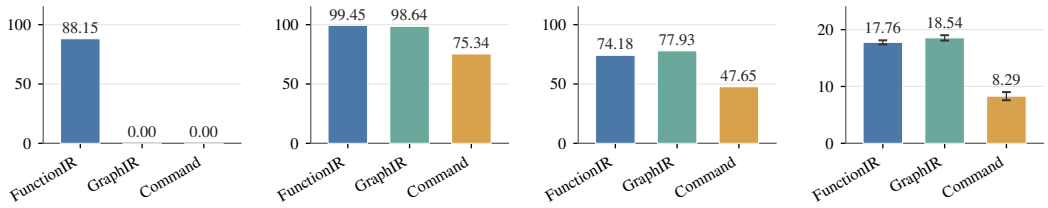
Because extraction also creates another planning opportunity, these runs measure the value of additional opportunity to *observe and act*, rather than observation alone. The full budget sweep and grounding-transfer analysis are provided in Appendix C.2.

### 4.4 RQ3: From Complex Editing Intent to Executable Tool Composition

Speech editing decisions must bridge two difficulties: observations may be incomplete or inaccurate, while editing capabilities are fragmented across tools that must be composed into executable programs. RQ3 therefore studies how the agent represents complex editing intent before planning and how much mechanical tool composition should be delegated to the execution interface.

On the 200 fixed contexts, target consistency (TC) counts decisions whose declared target text and acoustic parameters satisfy all active requirements. Plan consistency (PC) checks whether the selected version, operations, and actual parameters jointly express that target, including agreement between edit tags and source text. Each is the percentage of contexts that pass its complete check, with all 200 contexts in the denominator. Equivalent programs and synonymous targets are accepted; missing plans and pending observation requests do not pass PC. These are program-level checks, not audio-quality scores (Appendix B.3).

**Explicit belief helps organize complex editing intent.** Under matched observations and tool information, explicitly materializing the belief improves consistency on decisions involving interacting targets, versions, undo operations, and preservation requirements.




(a) Delegated bindings (%) (b) Compilation (%) (c) Delivery (%) (d) Delivered success $(D \cap E)$ (%)

Figure 3: **Action representation from compiler delegation to editing outcome.** Panels (a) and (b) use applicable programs; (c) and (d) use all 426 tasks. Whiskers show scoring SD.

Explicit belief is therefore most useful as an interface for organizing complex instruction state, rather than as a universal accuracy booster. The complete category-level diagnosis is reported in Appendix B.3.

**Action abstraction reduces tool-composition burden.** Figure 3 shows that fragmented speech tools create a substantial construction burden when Planner must directly specify low-level calls, bindings, and dependencies. Higher-level representations transfer much of this deterministic work to the compiler and substantially improve structural validity. However, the resulting differences in compilation are much larger than the differences in final editing quality.

Reliable decisions require both stable task interpretation and an action interface that delegates deterministic composition whenever possible. Detailed compilation failures and delegation statistics are provided in Appendix B.3.

### 4.5 RQ4: Contextual Policy Revision for Reliable Editing

RQ4 studies whether realized failures can be converted into contextual guidance for subsequent policies, thereby improving editing reliability.

As shown in Table 1, policy revision enables X-EDIT to reach 87.56% delivery while maintaining the strongest IFR, CR, and EMR among the evaluated configurations. This suggests that outcome-conditioned revision is a useful mechanism for improving the reliability of the complete agent.

Representative trajectories further show that different feedback sources address different failure modes. Runtime errors constrain invalid execution paths; semantic feedback exposes missing evidence or inappropriate operations; and audio feedback identifies failures in the realized acoustic effect that are not visible from the program alone. Guider converts these signals into additional conditions for subsequent policies, enabling targeted revision of the failed decision. Representative cases and detailed feedback routes are provided in Appendices C.4 and C.8; model-based outcome assessment is detailed in Appendix B.4.

## 5 Conclusion and Discussion

X-EDIT organizes fixed models and heterogeneous tools into a general-purpose speech editing agent. Its POMDP formulation separates uncertain observation, belief interpretation, executable action, and contextual policy revision. It attains the strongest MMAE editing scores among evaluated references, yet only 20.19% of tasks are delivered and pass every native check. The diagnostics explain this gap: evidence acquisition improves grounding while review can damage supported reports; explicit belief organizes interacting requirements but can propagate errors; and compilation removes structural work without correcting intent. Policy revision most clearly resolves execution barriers. Reliable agents should revise unresolved requirements while preserving supported evidence and decisions.

## Reproducibility statement

Anonymous code and a listening demo are linked in the abstract Appendix B specifies checkpoints, decoding and interaction budgets, module inputs, tool interfaces, diagnostic data, and scoring and listening protocols. We distinguish fixed-output scoring variation from repeated agent execution, retain failures in the reported denominators, and separate source fallback from valid delivery. The framework code uses Apache-2.0; external models and data retain their own licenses.

## Ethics statement

Speech editing can misrepresent a person's words, identity, or intent. Edited examples are research artifacts, not evidence of authentic utterances; a perceptual identity match does not establish consent. Use and redistribution must respect recording permissions and the licenses of the underlying datasets and models. Our experiments use benchmark recordings and constructed scenes; source provenance and construction are described in Appendix B.2.

## AI assistance disclosure

Generative AI assists research framing, mathematical formulation, code implementation, analysis, and writing.

## References


Keyu An, Qian Chen, Chong Deng, Zhihao Du, Changfeng Gao, Zhifu Gao, Yue Gu, Ting He, Hangrui Hu, Kai Hu, Shengpeng Ji, Yabin Li, Zerui Li, Heng Lu, Haoneng Luo, Xiang Lv, Bin Ma, Ziyang Ma, Chongjia Ni, Changhe Song, Jiaqi Shi, Xian Shi, Hao Wang, Wen Wang, Yuxuan Wang, Zhangyu Xiao, Zhijie Yan, Yexin Yang, Bin Zhang, Qinglin Zhang, Shiliang Zhang, Nan Zhao, and Siqi Zheng. FunAudioLLM: Voice understanding and generation foundation models for natural interaction between humans and LLMs, 2024. URL https://arxiv.org/abs/2407.04051.

Huan ang Gao, Jiayi Geng, Wenyue Hua, Mengkang Hu, Xinzhe Juan, Hongzhang Liu, Shilong Liu, Jiahao Qiu, Xuan Qi, Qihan Ren, Yiran Wu, Hongru WANG, Han Xiao, Yuhang Zhou, Shaokun Zhang, Jiayi Zhang, Jinyu Xiang, Yixiong Fang, Qiwen Zhao, Dongrui Liu, Cheng Qian, Zhenhailong Wang, Minda Hu, Huazheng Wang, Qingyun Wu, Heng Ji, and Mengdi Wang. A survey of self-evolving agents: What, when, how, and where to evolve on the path to artificial super intelligence. *Transactions on Machine Learning Research*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=CTr3bovS5F. Survey Certification.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, Guillaume Lathoud, Mike Lincoln, Agnes Lisowska, Iain McCowan, Wilfried Post, Dennis Reidsma, and Pierre Wellner. The AMI meeting corpus: A pre-announcement. In *Machine Learning for Multimodal Interaction*, volume 3869 of *Lecture Notes in Computer Science*, pp. 28–39. Springer Berlin Heidelberg, 2006. doi: 10.1007/11677482_3. URL https://doi.org/10.1007/11677482_3.

Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, Jian Zhao, Kai Yu, and Xie Chen. F5-TTS: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025. URL https://aclanthology.org/2025.acl-long.313/.

Thomas Palmeira Ferraz, Kartik Mehta, Yu-Hsiang Lin, Haw-Shiuan Chang, Shereen Oraby, Sijia Liu, Vivek Subramanian, Tagyoung Chung, Mohit Bansal, and Nanyun Peng. LLM self-correction with DeCRIM: Decompose, critique, and refine for enhanced following of instructions with multiple constraints. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7773–7812, 2024. URL https://aclanthology.org/2024.findings-emnlp.458/.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. CRITIC: Large language models can self-correct with tool-interactive critiquing. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Sx038qxjek.

Jiarui Hai, Helin Wang, Weizhe Guo, and Mounya Elhilali. FlexSED: Towards open-vocabulary sound event detection. In *2025 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 1–5. IEEE, 2025. doi: 10.1109/waspaa66052.2025.11230975. URL https://doi.org/10.1109/waspaa66052.2025.11230975.

Hangrui Hu, Xinfa Zhu, Ting He, Dake Guo, Bin Zhang, Xiong Wang, et al. Qwen3-TTS technical report, 2026a. URL https://arxiv.org/abs/2601.15621.

Jingbin Hu, Huakang Chen, Linhan Ma, Dake Guo, Qirui Zhan, Wenhao Li, Haoyu Zhang, Kangxiang Xia, Ziyu Zhang, Wenjie Tian, Chengyou Wang, Jinrui Liang, Shuhan Guo, Zihang Yang, Bengu Wu, Binbin Zhang, Pengcheng Zhu, Pengyuan Xie, Chuan Xie, Qiang Zhang, Jie Liu, and Lei Xie. VoiceSculptor: Your voice, designed by you, 2026b. URL https://arxiv.org/abs/2601.10629.

Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1–2):99–134, 1998. URL https://people.csail.mit.edu/lpk/papers/aij98-pomdp.pdf.

Chun-Yi Kuan, Chih-Kai Yang, Wei-Ping Huang, Ke-Han Lu, and Hung-Yi Lee. Speech-Copilot: Leveraging large language models for speech processing via task decomposition, modularization, and program generation. In *2024 IEEE Spoken Language Technology Workshop (SLT)*, pp. 1060–1067. IEEE, 2024. doi: 10.1109/SLT61566.2024.10832184. URL https://doi.org/10.1109/SLT61566.2024.10832184.

Kuan-Yi Lee, Tsung-En Lin, and Hung-Yi Lee. Audio-Maestro: Enhancing large audio-language models with tool-augmented reasoning, 2025. URL https://arxiv.org/abs/2510.11454.

Yinghao Aaron Li, Cong Han, Vinay Raghavan, Gavin Mischler, and Nima Mesgarani. StyleTTS 2: Towards human-level text-to-speech through style diffusion and adversarial training with large speech language models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 19594–19621. Curran Associates, Inc., 2023. doi: 10.52202/075280-0860. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/3eaad2a0b62b5ed7a2e66c2188bb1449-Abstract-Conference.html.

Jinhua Liang, Huan Zhang, Haohe Liu, Yin Cao, Qiuqiang Kong, Xubo Liu, Wenwu Wang, Mark D. Plumbley, Huy Phan, and Emmanouil Benetos. WavCraft: Audio editing and generation with large language models. In *LLM Agents Workshop at ICLR 2024*, 2024. URL https://openreview.net/forum?id=xJw7x2ZBex.

Ziyang Ma, Yinghao Ma, Yanqiao Zhu, Chen Yang, Yi-Wen Chao, Ruiyang Xu, Wenxi Chen, Yuanzhe Chen, Zhuo Chen, Jian Cong, Kai Li, Keliang Li, Siyou Li, Xinfeng Li, Xiquan Li, Zheng Lian, Yuzhe Liang, Minghao Liu, Zhikang Niu, Tianrui Wang, Yuping Wang, Yuxuan Wang, Yihao Wu, Guanrou Yang, Jianwei Yu, Ruibin Yuan, Zhisheng Zheng, Ziya Zhou, Haina Zhu, Wei Xue, Emmanouil Benetos, Kai Yu, Eng-Siong Chng, and Xie Chen. MMAR: A challenging benchmark for deep reasoning in speech, audio, music, and their mix. In *Advances in Neural Information Processing Systems*, volume 38, pp. 74837–74868. Curran Associates, Inc., 2025. doi: 10.52202/085713-2252. URL https://doi.org/10.52202/085713-2252.

Ziyang Ma, Zhikang Niu, Wenming Tu, Tianrui Wang, Ruiqi Yan, Junxi Liu, et al. AuK technical report: An open-source foundational model for speech generation and editing, 2026a. URL https://arxiv.org/abs/2609.08936.

Ziyang Ma, Ruiqi Yan, Ruiyang Xu, Jie Fang, Zhikang Niu, Yi-Wen Chao, et al. Mmae: A massive multitask audio editing benchmark, 2026b. URL https://arxiv.org/abs/2606.07229.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. Self-Refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46534–46594. Curran Associates, Inc., 2023. doi: 10.52202/075280-2019. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/91edff07232fb1b55a505a9e9f6c0ff3-Abstract-Conference.html.

MOSI.AI. MOSS transcribe diarize technical report, 2026. URL https://arxiv.org/abs/2601.01554.

Sylvie C. W. Ong, Shao Wei Png, David Hsu, and Wee Sun Lee. POMDPs for robotic tasks with mixed observability. In *Robotics: Science and Systems*, 2009. URL https://www.comp.nus.edu.sg/~leews/publications/rss09.pdf.

Karol J. Piczak. ESC: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia*, pp. 1015–1018. ACM, 2015. doi: 10.1145/2733373.2806390. URL https://doi.org/10.1145/2733373.2806390.

Plachtaa. Seed-VC: Zero-shot voice conversion. https://github.com/Plachtaa/seed-vc, 2024.

Alexander Polok, Dominik Klement, Samuele Cornell, Matthew Wiesner, Jan Černocký, Sanjeev Khudanpur, and Lukáš Burget. SE-DiCoW: Self-enrolled diarization-conditioned whisper. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 16712–16716. IEEE, 2026. doi: 10.1109/icassp55912.2026.11461785. URL https://doi.org/10.1109/icassp55912.2026.11461785.

Qwen Team. Qwen3.8-Max: A new bar for coding and cowork, August 2026. URL https://qwen.ai/blog?id=qwen3.8.

Xian Shi, Xiong Wang, Zhifang Guo, Yongqi Wang, Pei Zhang, Xinyu Zhang, Zishan Guo, Hongkun Hao, Yu Xi, Baosong Yang, Jin Xu, Jingren Zhou, and Junyang Lin. Qwen3-ASR technical report, 2026. URL https://arxiv.org/abs/2601.21337.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: Language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 8634–8652. Curran Associates, Inc., 2023. doi: 10.52202/075280-0377. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1b44b878bb782e6954cd888628510e90-Abstract-Conference.html.

Richard D. Smallwood and Edward J. Sondik. The optimal control of partially observable Markov processes over a finite horizon. *Operations Research*, 21(5):1071–1088, 1973. doi: 10.1287/opre.21.5.1071.

Jayakumar Subramanian, Amit Sinha, Raihan Seraj, and Aditya Mahajan. Approximate information state for approximate planning and reinforcement learning in partially observed systems. *Journal of Machine Learning Research*, 23(12):1–83, 2022. URL https://www.jmlr.org/papers/v23/20-1165.html.

Suno. Bark: Text-prompted generative audio model. https://github.com/suno-ai/bark, 2023.

Ye Tao, Wen Wu, Chao Zhang, Mengyue Wu, Shuai Wang, and Xuenan Xu. Mmedit: A unified framework for multi-type audio editing via audio language model, 2025. URL https://arxiv.org/abs/2512.20339.

Hankun Wang, Bohan Li, Shi Lian, Xiaoyu Gu, Jing Peng, Da Zheng, Yiwei Guo, Colin Zhang, and Kai Yu. dots.tts.edit: Precisely controlled speech editing with a continuous autoregressive model, 2026a. URL https://arxiv.org/abs/2608.02673v2.

Yucheng Wang, Jing Peng, Hanqi Li, Chenghao Wang, Wenming Tu, Yu Xi, Zhaokai Sun, Kai Yu, and Shuai Wang. Audio-Mind: An auditable agentic framework for audio understanding, 2026b. URL https://arxiv.org/abs/2605.28480.

Zixuan Wang, Chi-Keung Tang, and Yu-Wing Tai. Audio-agent: Leveraging llms for audio generation, editing and composition, 2024. URL https://arxiv.org/abs/2410.03335.

Yuqing Wen, Yukai Huang, Qianqian Xie, Jiangtao Wu, Yibin Lin, Yikai Gu, Jialu Chen, Yuanxing Zhang, and Jiaheng Liu. AVE-Compass: Towards holistic evaluation for audio-video editing abilities, 2026. URL https://arxiv.org/abs/2607.24821.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-Omni technical report, 2025. URL https://arxiv.org/abs/2509.17765.

Junichi Yamagishi, Christophe Veaux, and Kirsten MacDonald. CSTR VCTK corpus: English multi-speaker corpus for CSTR voice cloning toolkit (version 0.92). University of Edinburgh, The Centre for Speech Technology Research (CSTR), 2019. URL https://doi.org/10.7488/ds/2645.

Canxiang Yan, Chunxiang Jin, Dawei Huang, Haibing Yu, Han Peng, Hui Zhan, et al. MingUniAudio: Speech LLM for joint understanding, generation and editing with unified representation, 2025a. URL https://arxiv.org/abs/2511.05516.

Chao Yan, Boyong Wu, Peng Yang, Pengfei Tan, Guoqiang Hu, Li Xie, Yuxin Zhang, et al. Step-Audio-EditX technical report, 2025b. URL https://arxiv.org/abs/2511.03601.

Dongchao Yang, Songxiang Liu, Rongjie Huang, Chao Weng, and Helen Meng. InstructTTS: Modelling expressive TTS in discrete latent space with natural language style prompt. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 32:2913–2925, 2024. doi: 10.1109/TASLP.2024.3402088. URL https://doi.org/10.1109/TASLP.2024.3402088.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=WE_vluYUL-X.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D. Williams. POMDP-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179, 2013. doi: 10.1109/JPROC.2012.2225812. URL https://mi.eng.cam.ac.uk/~sjy/papers/ygtw13.pdf.

Fan Yu, Shiliang Zhang, Yihui Fu, Lei Xie, Siqi Zheng, Zhihao Du, Weilong Huang, Pengcheng Guo, Zhijie Yan, Bin Ma, Xin Xu, and Hui Bu. M2MeT: The ICASSP 2022 multi-channel multi-party meeting transcription challenge. In *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6167–6171. IEEE, 2022. doi: 10.1109/icassp43922.2022.9746465. URL https://doi.org/10.1109/icassp43922.2022.9746465.

Kunlun Zhu, Zijia Liu, Bingxuan Li, Muxin Tian, Yingxuan Yang, Jiaxun Zhang, et al. Where LLM agents fail and how they can learn from failures, 2025. URL https://arxiv.org/abs/2509.25370.

# A Related Work

**Speech editing.** Instruction-driven editors map requests to audio (Yan et al., 2025b;a; Ma et al., 2026a; Wang et al., 2026a; Tao et al., 2025); synthesis, conversion, and infilling offer narrower operations (Hu et al., 2026a; Plachtaa, 2024; Suno, 2023; Chen et al., 2025). We study how an agent combines these capabilities to make the requested change while preserving the rest of a recording.

**Audio tool agents.** Audio agents organize tools for understanding and editing (Kuan et al., 2024; Lee et al., 2025; Wang et al., 2026b; Liang et al., 2024; Wang et al., 2024), with related evaluations of audio and video editing (Wen et al., 2026). Our analysis follows errors from acoustic evidence through target selection and program construction to the final edit.

**Feedback and partial observability.** Feedback methods revise decisions through critique and diagnosis (Madaan et al., 2023; Shinn et al., 2023; Gou et al., 2024; Ferraz et al., 2024; Zhu et al., 2025). Dialogue POMDPs (Young et al., 2013), mixed observability (Ong et al., 2009), and approximate information states (Subramanian et al., 2022) distinguish the scene, observations, and interpretation. We use this distinction to analyze how feedback changes editing decisions and whether the revised output fulfills the complete request.

# B Implementation Details

## B.1 Models, decoding, and module inputs

Text decisions use fixed Qwen3.8-27B (Qwen Team, 2026)[2] weights; audio perception and assessment use Qwen3-Omni-30B-A3B-Instruct (Xu et al., 2025)[3]. Expert priors come from Qwen3-ASR-1.7B[4], Qwen3-ForcedAligner-0.6B (Shi et al., 2026)[5], SE-DiCoW (Polok et al., 2026)[6] for English speaker-attributed transcription, MOSS (MOSI.AI, 2026)[7] for Chinese, and FlexSED (Hai et al., 2025)[8] for sound events. Models are not updated during an editing episode.

Text generation uses temperature 0, seed 0, and at most 8192 output tokens. Ordinary audio perception uses temperature 0 and at most 1536 output tokens. The default permits two additional perception rounds and three planning rounds; recovery permits three complete attempts, including the initial attempt. R0–R3 allow zero to three additional perception rounds and one to four planning rounds, respectively. These settings change the opportunities to obtain evidence and revise a decision together.

Table 3 summarizes the input to each model role. Reasoner receives a task-specific view, rather than a copy of the full Planner input. Executor and candidate selection are deterministic procedures, not additional free-form planning models.

Direct AuK (Ma et al., 2026a)[9] uses 32 steps and CFG 2; AuK-Flash[10] uses 4 steps and CFG 0. Both use BF16 and seed 44. AuK+PromptEnhancer additionally uses Qwen3.8-27B and SenseVoiceSmall (An et al., 2024)[11] to prepare the editing request. Published Step-Audio-EditX and Ming-UniAudio scores are taken from the MMAE comparison (Ma et al., 2026b).

[2] https://huggingface.co/Qwen/Qwen3.8-27B
[3] https://huggingface.co/Qwen/Qwen3-Omni-30B-A3B-Instruct
[4] https://huggingface.co/Qwen/Qwen3-ASR-1.7B
[5] https://huggingface.co/Qwen/Qwen3-ForcedAligner-0.6B
[6] https://huggingface.co/BUT-FIT/SE-DiCoW
[7] https://huggingface.co/OpenMOSS-Team/MOSS-Transcribe-Diarize/tree/4a1af868018e7974197f4f018730758012b28c27
[8] https://huggingface.co/Higobeatz/FlexSED
[9] https://huggingface.co/tencent/AuK
[10] https://huggingface.co/tencent/AuK-Flash
[11] https://huggingface.co/FunAudioLLM/SenseVoiceSmall

Table 3: Information supplied to each model role.

| Role | Main input |
|---|---|
| Perceptor | Audio or crop, evidence question, asset version and time coordinates; expert transcripts, speaker segments, or event candidates when enabled. |
| Reasoner | Instructions, public interaction history, asset roles, observations, unresolved questions, and relevant capabilities. |
| Planner | Instructions and evidence, tool capability cards and argument contracts, representation specification, remaining budget, and any Reasoner memo or retry feedback. |
| Semantic Reviewer | Instructions, evidence, task interpretation, generated program, and actual tool calls. |
| Rubric Designer and Audio Evaluator | Task requirements for question design; source, output, and necessary reference audio for assessment, with atomic questions and comparison intervals. |
| Guider | Review findings, unresolved requirements, and relevant tool capabilities. Execution failures instead follow the direct error-feedback branch. |

### B.2 Data and simulation

**Complete editing tasks.** MMAE speech contains 426 tasks: 240 single-step, 62 multi-hop, 51 multi-part, 43 multi-audio, and 30 multi-turn tasks. Development additionally uses 143 tasks, comprising 64 existing tasks and 79 authored tasks from 31 constructed scenes. The new scenes combine recordings from 42 VCTK (Yamagishi et al., 2019) speakers and 8 event sources (Piczak, 2015), covering multiple speakers, speech mixed with background events, and reference-audio inputs. Source tracks and mixing offsets provide reference structure; instructions specify which components to change and preserve.

**Event localization.** Two batches of 128 scenes provide 256 event-localization examples. They share 32 source groups. Reference intervals follow activity in the isolated event track and its placement in the mixture. Activity is extracted with a relative RMS threshold of $-35$ dB and shifted by the mixing offset, providing a construction-based reference rather than manually annotated perceptual boundaries. Pure FlexSED converts frame scores of at least 0.5 into intervals, with an 80-ms minimum duration and an 80-ms merging gap. These parameters follow the historical service defaults and are not tuned against the evaluation references. Expert-prior integration instead exposes candidate evidence to Omni, including soft candidates that need not survive this threshold. Thus, pure detection and integration differ in how they use the same expert evidence.

In Figure 2, additional review denotes one review stage; its model-call count follows the task-specific protocol. The initial observation uses one audio-model call. Review sees the proposed answer, whereas reobservation obtains another report without that proposal. Extended conditions use three audio-model calls in total; expert calls are additional. Review and reobservation are matched in audio-model calls.

**Speaker-attributed transcription.** AMI (Carletta et al., 2006) contributes 597 excerpts from 16 meetings, and AliMeeting (Yu et al., 2022) contributes 689 excerpts from 20 meetings. Excerpts are approximately 60 seconds. SE-DiCoW and MOSS provide the English and Chinese expert reports, respectively. Review receives the report and recording; cropped review can request up to three slices, and full-recording reobservation matches its actual model-call count.

Event localization is measured by continuous-time IoU. Transcription uses micro cpWER for English words and micro cpCER for Chinese characters, minimizing the edit count over speaker permutations within each excerpt. Invalid transcripts count as empty hypotheses. Transcription errors can exceed one because of insertions; they are not clipped or treated as bounded IoU losses. Both diagnostics compare the same target before and after an additional observation, allowing reductions and increases in error to be measured separately. Intervals are grouped by event source or meeting, rather than treating related excerpts as independent.

**Observation revision and budget.** For the same target and a fixed error measure $L$, define expected correction and corruption through

$$\mathbb{E}[L_{k+1} - L_k] = \mathcal{H}_k - \mathcal{C}_k, \quad \mathcal{H}_k = \mathbb{E}[(L_{k+1} - L_k)_+], \quad \mathcal{C}_k = \mathbb{E}[(L_k - L_{k+1})_+]. \tag{22}$$

where $(x)_+ = \max(x, 0)$. The identity applies to both bounded localization loss and nonnegative transcription error when the expectations exist. With $L = 1 - \mathrm{IoU}$, event review improves 11 reports, worsens 21, and leaves 224 unchanged; mean correction is 0.0087 and corruption 0.0265. These are changes in reports, not terminal editing rewards.

Figure 4 retains the complete budget sweep. Explicit and implicit GraphIR conditions use expert prior and disable report review and outer recovery. R0–R3 jointly allow zero to three additional extraction rounds and one to four planning decisions; lines connect separate runs, not successive states of one trajectory.

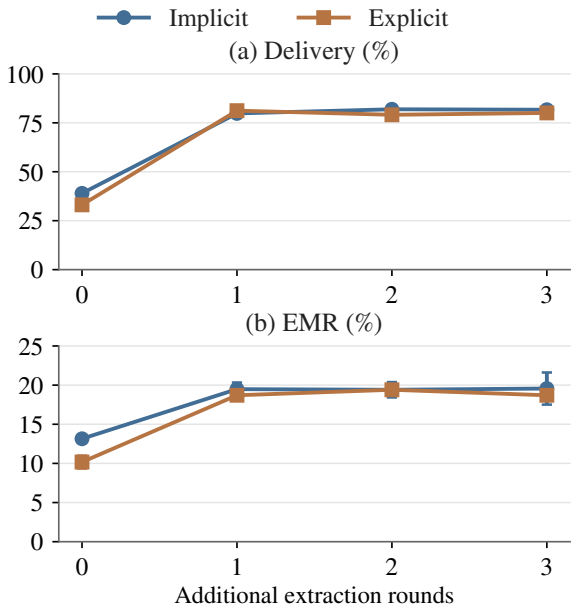



Figure 4: Extraction and decision budget on all 426 MMAE tasks. Error bars show three-batch scoring SD on fixed outputs.

**Editing tasks that request expert evidence.** The MMAE observation analysis retains the union of tasks requesting either relevant expert interface in any of the three compared conditions: 129 speaker-related tasks and 8 event-related tasks, with 2 shared tasks, giving 135 in total. All these tasks remain in the denominator, including failed deliveries.

### B.3 Controlled decision studies and assessment

The matched explicit and implicit belief conditions use the same model, GraphIR interface, tools, expert priors, and R2 allowance, without outer recovery. The explicit condition adds a separate Reasoner memo. It can change later evidence acquisition as well as planning, so the full-chain comparison measures the joint effect of this configuration.

**Constructed decision contexts.** The separate 200-context study fixes the public evidence and compares reasoning and planning without generating audio. It contains 100 English and 100 Chinese contexts from 20 template groups, covering numerical transformations, indirect targets, selective undo, historical-version selection, unrealized earlier requirements, and cancellation of earlier requirements. Each context specifies available versions, observed content, and the instruction history. Template instances vary target values and requested revisions; recorded tool completion and observed content may disagree, allowing desired and realized states to be tested separately. Equivalent programs and synonymous targets are accepted. Checks distinguish a correct declared target from a program that correctly binds versions, operations, scope, and parameters. Requests for additional observations remain pending within this fixed-decision evaluation.

Table 4: **Belief materialization across two datasets.** MMAE GraphIR quality is percentage mean $\pm$ three-batch scoring SD, with delivery a count. Synthetic cells give declared-target / full-plan consistent counts; overall percentages use each task once. Explicit adds a Reasoner; implicit interpretation remains inside the Planner.

| Measure or task category | $N$ | Implicit | Explicit |
|---|---|---|---|
| *MMAE speech: complete editing (426 tasks)* | | | |
| Delivered | 426 | 349 | 337 |
| IFR (%) | 426 | $53.80 \pm 1.14$ | $52.60 \pm 0.67$ |
| CR (%) | 422 | $91.67 \pm 0.42$ | $92.82 \pm 0.30$ |
| EMR (%) | 426 | $19.41 \pm 0.98$ | $19.41 \pm 0.68$ |
| $D \cap E$ (%) | 426 | $18.70 \pm 1.20$ | $18.15 \pm 0.75$ |
| *Synthetic fixed contexts (200)* | | | |
| Single-turn conversion | 40 | 39 / 39 | 33 / 33 |
| Indirect target and scope | 40 | 26 / 26 | 35 / 35 |
| Selective undo | 40 | 36 / 24 | 39 / 32 |
| Historical version selection | 40 | 40 / 40 | 40 / 40 |
| Active requirement not realized | 20 | 4 / 4 | 6 / 6 |
| Cancellation and protection | 20 | 20 / 20 | 20 / 20 |
| Total target / plan | 200 | 165 / 153 | 173 / 166 |
| Overall consistency (%) | 200 | 82.5 / 76.5 | 86.5 / 83.0 |

**Fixed-input action representations.** The representation study freezes the evidence, assets, history, and Reasoner memo from the GraphIR R2 planning prefix. FunctionIR, GraphIR, and Command then generate continuations with identical text-model settings and no additional perception or recovery. Prefixes are available for 421 of the 426 tasks; the remaining 5 remain failures in each condition. Each condition also has 5 state-only deliveries, where the requested result is already available. For a compiled program $p$, let $n(p)$ count concrete calls and $n_{\text{auto}}(p)$ those added by lowering. Let $m(p)$ count intermediate output-to-input bindings and $m_{\text{auto}}(p)$ those supplied deterministically. Structural delegation is measured by

$$A_{\text{call}}(p) = \frac{n_{\text{auto}}(p)}{n(p)}, \qquad A_{\text{bind}}(p) = \frac{m_{\text{auto}}(p)}{m(p)}. \tag{23}$$

The binding ratio requires $m(p) > 0$. We average the ratios over valid applicable programs, rather than dividing pooled counts.

FunctionIR, GraphIR, and Command compile 364 of 366, 363 of 368, and 281 of 373 submitted programs, respectively. Their binding ratios use 64, 39, and 21 programs with intermediate bindings. Command has 89 syntax failures already present in the raw model response and another 62 tasks with invalid operation arguments. These diagnoses concern generated programs, not the expressive coverage of the interfaces.

**Native assessment and candidate selection.** MMAE scoring uses Qwen3-Omni at temperature 0.7 with a 4096-token output limit. Each question takes a majority vote over three valid judgments, with at most ten attempts to obtain them. Three independent scoring batches evaluate the fixed outputs; their standard deviation measures assessment variation. CR applies to 422 tasks, while other main rates retain the 426-task denominator.

Online feedback follows a different purpose: execution errors are forwarded directly; delivered outputs receive semantic review, followed by audio review after the semantic gate passes. Audio grades range from 0 to 3, with an explicit unknown state. The selector normalizes grades, considers the weakest required dimension, and treats unknown items conservatively. Eligibility checks delivery and recorded preservation requirements. Recovery retains its actual initial output, so recovery gains are not computed against a different first-attempt run.

### B.4 Blind listening study

The listening set contains 176 atomic questions on 48 editing scenes: 32 use MMAR (Ma et al., 2025) speech recordings and 16 are constructed tasks. Questions concern a specific requested

change or preservation requirement and are attached to actual source and edited audio. The interface presents them in eight groups of 22 questions. Related questions remain grouped by scene.

Listeners hear the source and output, read the request and current question, and choose a grade: 0 for an unmet or reversed requirement, 1 for partial completion with substantial defects, 2 for mostly satisfying the requirement with minor defects, and 3 for full satisfaction without a relevant audible defect. They can abstain when the audio or question is insufficient; abstention is not scored as zero. Confidence, an optional explanation, and prior exposure are recorded separately.

Figure 5 shows the actual study page. Model judgments remain hidden until the group is complete. The interface preserves ratings submitted before disclosure separately from later revisions. Analysis uses the last eligible rating before disclosure and excludes previously exposed items. One listener provides 175 eligible ratings; a second provides 48 overlapping ratings from 13 source groups. The overlap measures inter-listener agreement, rather than constituting a second complete evaluation of all questions.

Calibration compares model grades with these blind ratings using Spearman correlation, mean absolute error, and agreement at a grade threshold of 2. Reverse-question grades are mapped to $3 - g$ before averaging with positive-question grades. This calibration protocol is distinct from the historical online feedback used in recovery. Model answers are comparison outputs, not ground truth for the listener.

Table 5 compares the numeric positive-plus-reverse grader with the blind ratings. Human agreement on the 48-item overlap has quadratic weighted kappa 0.679. Over 175 eligible H1 items, one positive judgment has Spearman correlation 0.495 and MAE 0.909; positive-plus-reverse grading lowers MAE to 0.757, with correlation 0.483. It falsely rejects 47 of 140 H1-accepted items. The 48-item and 175-item populations have different coverage and should not be pooled.

Table 5: **Blind rubric agreement.** Model: numeric positive + reverse Qwen; H1/H2: blind listeners. Overlap is the same 48 items.

| Comparison | $N$ | $\rho$ | MAE |
|---|---|---|---|
| Model / H1 | 175 | 0.483 | 0.757 |
| Model / H1 overlap | 48 | 0.467 | 0.656 |
| Model / H2 overlap | 48 | 0.473 | 0.781 |
| H1 / H2 overlap | 48 | 0.678 | 0.438 |

### B.5 Tools and execution

The catalog contains 17 tools exposing 31 editing operations. Table 6 groups these operations by their role in the edit. Perception interfaces additionally provide speech context, speaker review, semantic-span review, general audio questions, timestamped captions, and event queries.

Table 6: Editing capabilities available to the agent. Counts refer to operations, not separately loaded models.

| Family | Operations | Capabilities |
|---|---|---|
| Artifact operations | 10 | Beep, concatenate, extract, mute, reconstruct, remove, reorder, insert silence, speaker-scoped editing, and gain. |
| Speech edit pipeline | 9 | Text, prosody, emotion and style, pause, rate, normalization, pitch, boundary smoothing, and assembly. |
| Other editing interfaces | 12 | Audio effects, emphasis, intra-audio transfer, reference-free voice conversion, Qwen3-TTS (Hu et al., 2026a), Seed-VC (Plachtaa, 2024), spatial reverberation, VoiceSculptor (Hu et al., 2026b), enhancement, separation, foreground–background remix, and AuK (Ma et al., 2026a). |

FunctionIR declares effects, scope, and composition; its compiler generates crops, reconstruction, and dependencies. GraphIR specifies tool nodes and their dependencies directly. Command ex-




Figure 5: Actual blind-listening interface, shown in its original Chinese. The page displays the request, one atomic question, source and edited audio, four anchored grades, an abstention option, and confidence. Model results are hidden while the listener submits the group. The screenshot contains no submitted human rating.

presses ordered calls in a restricted Python form that is parsed and validated before execution. All three use the same tool contracts.

Checkpoint reuse requires agreement on the goal, input assets and versions, parameters, and dependencies of the completed prefix. An incompatible prefix is executed again. The observed opportunities and limits of reuse are illustrated in Appendix C.8.

### B.6 EDITING ALGORITHM

Algorithm 1 summarizes the process described in Section 3.

**Algorithm 1** X·EDIT speech editing process.

**Require:** $s_0, \Pi_0, H, \ell$

$\mathcal{E} \leftarrow \varnothing$; $\hat{b}_0 \sim \mathcal{B}_\phi(\cdot \mid s_0)$ ▷ optional explicit Reasoner; Eq. 10

**for** $t = 0, \ldots, H-1$ **do**

  $p_t \sim \mathcal{P}_{\ell,\phi}(\cdot \mid \hat{b}_t)$; $a_t \leftarrow \mathrm{Compile}_\ell(p_t)$ ▷ Planner; Eq. 12

  $s_{t+1} \sim T(\cdot \mid s_t, a_t)$; $o_{t+1} \sim Z_{\theta,\phi}(\cdot \mid s_{t+1}, a_t)$ ▷ Executor → Perceptor

  $\hat{b}_{t+1} \sim \mathcal{B}_\phi(\cdot \mid \hat{b}_t, a_t, o_{t+1})$ ▷ optional explicit Reasoner; Eq. 11

  **if** $y_{t+1}$ is delivered **then**

    $\rho_{t+1} \sim \mathcal{D}_\phi(\cdot \mid \hat{b}_t, a_t)$; $\hat{r}_{t+1} \sim \widehat{\mathcal{R}}_{\theta,\phi}(\cdot \mid \rho_{t+1}, y_{t+1}, a_t)$ ▷ Rubric Designer → Evaluators

    $\mathcal{E} \leftarrow \mathcal{E} \cup \{(y_{t+1}, \hat{r}_{t+1})\}$; $g_{t+1} \sim \mathcal{G}_\phi(\cdot \mid \hat{b}_t, \hat{r}_{t+1}, \pi_{\Pi_t})$ ▷ Guider; Eq. 17

  **else**

    $g_{t+1} \leftarrow G_{\mathrm{err}}(e^{\mathrm{err}}_{t+1})$ ▷ Eq. 18

  **end if**

  $\Pi_{t+1} \leftarrow U(\Pi_t, g_{t+1})$ ▷ Eq. 19

**end for**

$\mathcal{E}_{\mathrm{elig}} \leftarrow \{y : (y, \hat{r}) \in \mathcal{E}, \mathrm{Eligible}(y) = 1\}$; $\hat{y} \leftarrow \arg\max_{y \in \mathcal{E}_{\mathrm{elig}}} \widehat{R}(y)$

**return** $\hat{y}$ ▷ source audio with incomplete status if $\mathcal{E}_{\mathrm{elig}} = \varnothing$

# C Case Study

We show how an instruction becomes an editing operation, and where this process can fail. Following the trajectory presentation of ReAct (Yao et al., 2023), the cards distinguish observations, interpretations, actions, and feedback. Text is condensed from execution records; it is not a verbatim transcript. Colored waveforms are measured from the actual audio. "Pass" refers to native automatic assessment of fixed outputs in batches A, B, and C, not a human listening judgment.

## C.1 From instructions to edits

The following 6 cases use an explicit Reasoner and GraphIR in the first-attempt run with 341 deliveries among 426 MMAE tasks. They illustrate this execution path, rather than the different implicit configuration selected for Table 1. No online evaluator or outer recovery was used in these cases; targeted perception occurred within the first attempt. Scored audio hashes match the recorded outputs.

**Q1a Change the question, preserve the answer**

| | |
|---|---|
| **Request** | Replace the first sentence with "What is the capital of Germany?" |
| **Observed** | The transcript begins "What is the capital of France? Paris." and continues with further questions and answers. |
| **Interpret** | Only the first sentence is in scope. Preserve "Paris" and the subsequent dialogue; obtain the missing sentence boundary. |
| **Query** | Planner asks for the first sentence's start and end. Alignment returns 0.00–1.36 s. |
| **Action** | One text-edit call replaces the question through a substitution tag and retains the rest of the transcript. |
| **Result** | IFR 4/4; CR 8/8; complete pass in all 3 batches. The operation respects the requested scope: it does not also change Paris to Berlin. |

**Q1b Account for every occurrence**

| | |
|---|---|
| **Request** | Replace all instances of "eight dollars" with "ten dollars." |
| **Interpret** | The transcript contains repeated targets; the initial interpretation remains uncertain about the last one. |
| **Query** | Planner asks how many occurrences there are and where each begins and ends. Returned ranges: 1.20–1.84, 1.92–2.48, and 12.32–13.28 s. |
| **Action** | One text-edit call carries 3 substitution tags, covering all returned occurrences. |
| **Result** | IFR 3/3; CR 5/5; complete pass in all 3 batches. Coverage comes from enumerating the targets, not from making a separate generation call for each. |

**Q1c Turn a semantic category into acoustic spans**

| | |
|---|---|
| **Request** | "Extract only the questions and remove the answers." |
| **Interpret** | From the alternating dialogue, Reasoner selects "Hello, why are you here?", "What is your name?", and "Do you have questions about this job?" |
| **Query** | Planner asks for these utterances' boundaries. Alignment supplies [0.00, 1.68], [6.32, 7.20], and [10.64, 12.16] s. |
| **Action** | `extract_spans` concatenates the original spans in order with 10-ms crossfades. It does not resynthesize the questions or leave silenced answers in place. |
| **Result** | 15.404 s becomes 4.060 s. IFR 6/6; CR 1/1; complete pass in all 3 batches. |

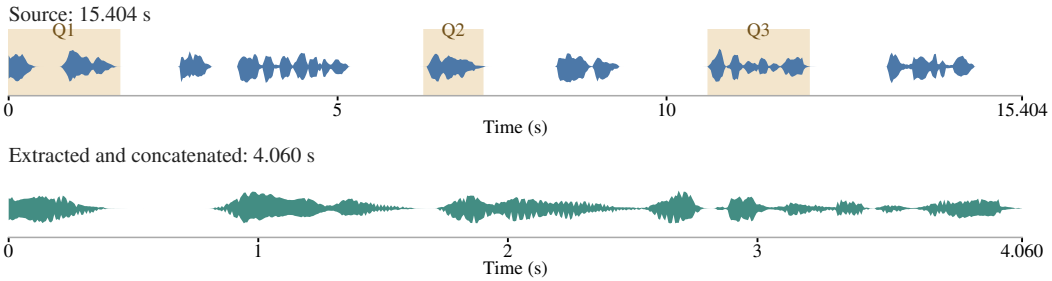



Figure 6: Measured source and extracted audio for Q1c. Shading marks the executed question spans in source coordinates. The output has its own time axis.

**Q1d Continue on the edited version**

| | |
|---|---|
| **Request** | Turn 1 extracts the phrase meaning "I will not say it." Turn 2 asks to lower the extracted voice into a deep demon voice. |
| **Query** | In turn 1, Planner requests the phrase's exact boundaries: 7.20–8.16 s in the source. Turn 2 requests no further targeted perception. |
| **Action** | `extract_spans` creates a 0.960-s asset; `pitch_shift` then reads that asset with PSOLA and $-5$ semitones. |
| **Result** | IFR 3/3; CR 5/5; complete pass in all 3 batches. The source interval becomes the whole short asset; the next user turn operates on that version, not on source-relative coordinates. |

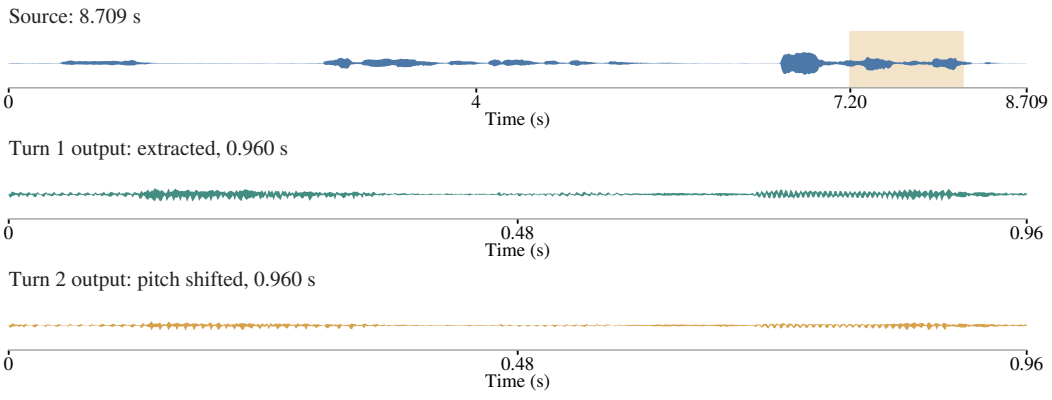



Figure 7: Measured audio for Q1d. Extraction maps the source range 7.20–8.16 s to a new 0.960-s asset, which is the input to pitch shifting. Waveforms show amplitude, not pitch correctness.

**Q1e Ask about the sound that the transcript cannot describe**

| | |
|---|---|
| **Request** | Convert the female speaker's whisper to natural voiced speech while preserving the linguistic content. |
| **Observed** | ASR returns "You have a gentle strength inside you." This establishes words, not phonation or background sound. |
| **Query** | Planner asks whether the voice is whispered, whether it matches the described speaker, and whether the 7.325-s recording contains other sound. Perception reports breathy speech over 0–5.5 s, then silence, with faint low-frequency noise. |
| **Action** | An AuK call edits the complete source asset with instructions to restore voiced speech and preserve content. No crop or source separation is executed. |
| **Result** | IFR 1/1; CR 4/4; complete pass in all 3 batches. Targeted sound observation supports the tool choice. |

**Q1f Bind two assets without unnecessary queries**

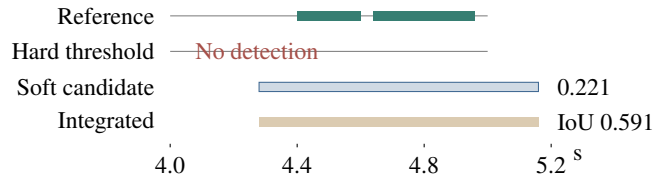
| | |
|---|---|
| **Request** | "Concatenate the two audio clips into one." |
| **Interpret** | The two source assets and their order are available. Reasoner retains that order; Planner requests no targeted perception. |
| **Action** | `concat` takes the first and then the second source. Their durations are 1.974 and 3.671 s; the output is 5.645 s. |
| **Result** | IFR 4/4; CR 7/7; complete pass in all 3 batches. A clear asset-order instruction can proceed directly to execution. |

### C.2 Observation gains must reach the right operation

These constructed scenes separate the recovery of missing event evidence from the choice of which sound to modify.

**Q2a A soft candidate survives an empty hard detection**

| | |
|---|---|
| **Request** | Locate a sneeze in a constructed recording. |
| **Observed** | Direct perception and FlexSED thresholded at 0.5 both give IoU 0. A retained candidate spans 4.28–5.16 s with peak score 0.221. |
| **Action** | Conditioning Perceptor on the expert prior recovers this same interval; IoU becomes 0.591. The construction reference is [4.40, 4.60] and [4.64, 4.96] s. |
| **Result** | The gain recovers a missed candidate, rather than refining its boundaries. |




**Q2b The right time range can still select the wrong sound**

| | |
|---|---|
| **Request** | "Remove the breathing sound at the beginning of the recording." |
| **Observed** | In the constructed mixture, speech overlaps breathing. Expert prior improves localization IoU from 0.316 to 0.708. |
| **Action** | Both plans mute the mixed source: 0–1.5 s without the prior, 0–2.6 s with it. The larger interval contains about 91% of clean-speech energy, versus 40% for the smaller interval. |
| **Result** | Both execute. Better timing exposes more protected speech to muting because the program does not separate the target component. |

### C.3 Interpretation and program construction

Q3a belongs to the 200 fixed-context decision study and has no audio execution. Q3b uses actual MMAE continuations with a shared observation and Reasoner prefix, without further perception or recovery. These controls isolate different decisions and should not be read as the same experiment.

**Q3a Undo one property while retaining another**

| | |
|---|---|
| **Observed** | Constructed history: original "72 boxes"; change to "74 boxes"; reduce volume by 6 dB. The final request restores the wording but retains the volume change. |
| **Implicit** | Planner selects a version still saying 74 and lowers its volume, leaving the text requirement unmet. |
| **Explicit** | Reasoner notes that the current version is already quieter. Planner replaces 74 with 72 in that version. |
| **Result** | The explicit plan passes the program check; the implicit plan does not. No audio is generated. The useful decomposition is current content, active requirements, and the remaining change. |

**Q3b A correct target still needs a valid program**

| | |
|---|---|
| **Request** | Replace "terrible" with "excellent." Both representations receive the same observation history and Reasoner memo. |
| **Command** | Command embeds a transcript containing `he's` and `that's` in an improperly escaped Python string. A `SyntaxError` prevents delivery. |
| **GraphIR** | GraphIR uses structured arguments with `<sub targ="excellent">terrible</sub>`. It delivers and passes all 3 native batches. |
| **Result** | The first failure is program construction, not target interpretation. In a second task, "Spain" to "Canada," Command parses but its unquoted target attribute fails the tool contract; GraphIR again delivers and passes all 3 batches. |

The distinction also reveals a shared weakness: an earlier request changes an end time from 15:00 to 16:00, but the current transcript still says 15:00. When asked to restore the start time and retain the end-time change, both modes can preserve the observed 15:00 as though the requirement were already satisfied. Recorded tool completion is not evidence that the desired state has been reached.

### C.4 Feedback, new candidates, and final selection

The following 4 trajectories come from the separate checkpoint recovery run, which increased delivery from 337 to 378 among 426 tasks. They illustrate repair, regression, and unchanged final selection; they are not task-level examples from the main 341-to-373 comparison. The main-run accounting follows in Appendix C.5.

**Q4a An error opens a path back to execution**

| | |
|---|---|
| **Request** | Replace “S” with “Serena” and “b” with “Blair.” |
| **Feedback** | Initial perception stops with “Contextual recognition lacks a completed observation receipt.” No edited output exists. |
| **Action** | The actual error enters the next attempt's context. The retry executes the replacements and its audio is selected. No nonexistent initial output is sent for audio review. |
| **Result** | IFR changes from 0/4 on fallback audio to 4/4 in every batch. Final CR is 5/6, 6/6, 6/6; complete acceptance is 0, 1, 1. Execution is restored, while preservation remains imperfect. |

**Q4b Semantic feedback changes what the agent observes**

| | |
|---|---|
| **Request** | “Reduce the mechanical feel of the human voice's timbre.” |
| **Feedback** | The initial AuK output exists. Semantic review asks for evidence of mechanical timbre and nonvocal content before a global voice transformation. |
| **Action** | The retry makes 2 targeted observations of the source, then uses AuK with a revised instruction to naturalize the voice and preserve Japanese content, timing, and surroundings. |
| **Result** | The changed output is selected. IFR changes from 0/1 to 1/1; CR stays 6/6 in all 3 batches. The online proxy changes from 0 to $1/6$. Source observations support the revision; this was not triggered by output audio review. |

**Q4c A higher proxy can select a regression**

| | |
|---|---|
| **Request** | "Change all 'Dan' to 'Bill'." |
| **Observed** | The initial output passes all checks in all 3 batches: IFR 3/3 and CR 8/8. |
| **Feedback** | Semantic review questions timing evidence. Renewed observation also changes unrelated words in the transcript passed to the editing tool. |
| **Result** | The selector prefers the new output as its proxy rises from 0 to $1/3$, but every native batch now fails. Batches A and C fail an IFR-labeled check about other names; batch B fails a CR check about the opening sentence. |

**Q4d Detecting a defect is not the same as selecting a repair**

| | |
|---|---|
| **Request** | Increase speech speed while retaining each phrase's onset. |
| **Feedback** | After semantic review passes, audio review reports slower speech: duration changes from about 18.45 to 21.1 s. Guidance asks whether the parameter expresses speed or duration. |
| **Action** | The next plan changes the rate from 0.85 to 1.15. Two audio-guided retries generate candidates, but preservation checks reject them. |
| **Result** | The original audio is retained: IFR 0/5, CR 6/6, and no complete pass in all 3 batches. |

### C.5 WHICH FEEDBACK REACHES THE SELECTED RESULT?

In the main paired run, 297 of 426 tasks retry, producing 519 retry events. Error records occur in 117 tasks; 373 tasks enter semantic review and 262 enter audio review. Table 7 separates this exposure from feedback that actually starts another attempt. Task counts overlap because a task can follow different routes.

Table 7: Feedback routes in the main recovery run. Metric changes are percentage points on the full cohort, averaged over 3 scoring batches. Changes are grouped by the feedback immediately preceding the selected candidate; they are not independent causal effects.

| Feedback | Tasks retrying | Retries | Selected | ΔIFR | ΔEMR | $\Delta(D \cap E)$ |
|---|---|---|---|---|---|---|
| Raw error | 106 | 169 | 34 | +1.79 | +0.86 | +1.56 |
| Semantic | 183 | 246 | 28 | +0.09 | −0.31 | −0.31 |
| Audio | 86 | 104 | 4 | −0.02 | 0.00 | 0.00 |
| Total | 297 | 519 | 66 | +1.86 | +0.55 | +1.25 |

All 32 newly delivered tasks first regain delivery after raw-error feedback, recovering 37.65% of the 85 initial failures. One then selects a further audio-guided revision. No initially delivered task loses final delivery. Thus, the clearest gain is removal of an execution barrier. Semantic and audio review start from existing outputs: 27 semantic-guided and 12 audio-guided next attempts fail to deliver, although retaining older candidates prevents a corresponding loss of final delivery. Review can identify a concern without yielding a better selected edit.

Across batches A, B, and C, error-preceded selected candidates add 6, 6, and 8 delivered complete passes and lose none. Semantic-preceded candidates add 1, 1, and 0 but lose 2 in each batch. Audio-preceded candidates add 0, 0, and 1 and lose 1, 0, and 0. The final gain therefore depends on both repairing failures and protecting already correct results. Earlier feedback may also shape a selected candidate.

### C.6 Evidence and intent are different questions

**An acoustic observation supplies an unsupported intent restriction.** The request is "Replace the spoken word 'dog' with 'cat', matching the same voice and tone." Both FunctionIR conditions initially have a transcript with two occurrences. The explicit chain asks which occurrence the user means, while also requesting that genuine ambiguity be reported. Alignment returns both candidates at 2.00–2.40 and 6.88–7.12 seconds. The audio model marks the first as matching and the second as not matching, assigning confidence 0.95 to each. The report labels the target resolved; the Reasoner accepts the first occurrence, and the final program replaces it alone. The instruction contains no first-occurrence restriction. The implicit program replaces both. The explicit output fails native exact match in all three batches, while the implicit output passes all three. The observed failure is an unsupported restriction promoted from a perception judgment into the task interpretation, not a failure to locate the words.

### C.7 Choosing the editing operation

**Correct compilation preserves content the user asked to remove.** The request is "Extract only the singing voice from the middle part and remove everything else, including the spoken voice." Both FunctionIR conditions obtain additional observations. The explicit condition identifies 2.0–6.0 seconds and specifies separation followed by extraction. The implicit condition identifies 2.0–6.5 seconds but selects a local edit with outside-region preservation. The compiler expands it into crop, processing, and reconstruction, restoring unwanted surroundings. The explicit output passes native exact match in all three batches; the implicit output fails all three. Both operations are expressible in FunctionIR, so the case identifies wrong operation selection, not absent interface coverage.

### C.8 A checkpoint is useful only if the next plan preserves it

Table 8 describes the 463 retries in this separate checkpoint run. Route counts refer to retry events; the same task can retry more than once. Next-attempt delivery does not imply passing all quality checks.

Table 8: **Sequential feedback routes.** Separate checkpoint run; the next attempt delivered, irrespective of quality.

| Route | Retries | Next D |
|---|---|---|
| Raw error | 177 | 67 |
| Semantic | 165 | 134 |
| Audio | 121 | 108 |
| Total | 463 | 309 |

For making speech slower and calmer, emotion editing completes before rate editing fails. The next attempt receives the completed emotion step as a possible checkpoint. Its new plan changes that operation, so execution must repeat it. The final attempt produces audio but does not pass all native checks.

Across this run, 9 of 463 retries receive an eligible prefix. Five stop during planning; the other four change the prefix or the step structure. No tool call is skipped. The restriction first limits when saved work is available, and revised planning then determines whether it remains usable. A checkpoint therefore needs a corresponding planning constraint: retain the accepted prefix and plan the suffix, unless new evidence justifies revoking that prefix.