

**PROGETTO FINALIZZATO
SISTEMI INFORMATICI E CALCOLO PARALLELO**

SOTTOPROGETTO 7
Sistemi di Supporto al Lavoro Intellettuale
Coordinatore Cesare Maioli

Francesco Giuffrida(o)

NOTE TECNICHE DEL MODULO

JAM+

N.M/23

7/91

Rapporto Tecnico Serie MUSIC

- (o) DIST - Laboratorio di Informatica Musicale
Dipartimento di Informatica Sistemistica e Telematica
Università degli studi di Genova

Indice

1	Introduzione	4
2	Struttura del Sistema	4
3	Rappresentazione della Conoscenza	6
4	Strutture musicali gestite da Jam+	8
4.1	Sviluppo dell'Armonia	9
4.2	Sviluppo dell'Improvvisazione	9
5	Architettura di Controllo a Blackboard	17

Indice delle Figure

1	Schema a blocchi del sistema	5
2	Esempio di Rete	7
3	Sviluppo dell'Armonia	10
4	Moduli Relativi allo Sviluppo dell'Improvvisazione	11
5	Strategia di Pianificazione 1	13
6	Strategia di Pianificazione 2	15
7	Schema del blackboard system	18

1 Introduzione

Il Sistema Jam+ è un tool di supporto alla composizione ed allo sviluppo di improvvisazioni basato su tecniche di intelligenza artificiale. Jam+ può essere inoltre utilizzato come valido strumento di ausilio per analisi musicale. Jam+ è stato sviluppato in Linguaggio Common Lisp ed utilizza, per le parti relative all'interfaccia grafica la libreria X-Windows versione 11. L'implementazione è avvenuta in principio su Workstation Hp, e successivamente su workstation Sun Sparc utilizzando i packages Lucid Common Lisp e Dcl.

2 Struttura del Sistema

Il sistema è di tipo modulare e risulta composto di quattro blocchi principali: il **Jam+ kernel**, l'**Interfaccia utente**, il **Modulo di Traduzione in formato Adagio** e il **Sistema di Segmentazione per l'estrazione di nuovi stili**. I quattro moduli dialogano tra di loro tramite opportune connessioni, interagiscono con l'utente tramite il modulo di interfaccia grafica. Il sistema è interattivo.

Vediamo ora più in dettaglio le caratteristiche dei singoli moduli:

Jam+ kernel È il cuore del sistema di improvvisazione. Il suo scopo è di assemblare strutture musicali (basi armoniche), su cui poi costruire linee melodiche. Oltre a ciò svolge la funzione di gestore delle comunicazioni tra i moduli del sistema.

Interfaccia Utente Rappresenta il "Front end" del sistema, gestisce le comunicazioni con l'utente tramite un sistema a finestre autoesplicativo e di facile uso basato su un'applicazione costruita in ambiente X-Windows.

Modulo di Traduzione Essendo il sistema sviluppato su macchine UNIX, che come è noto molto raramente sono dotate di interfaccia Midi, è stato necessario realizzare un modulo per il dialogo con il mondo esterno. Questo è stato realizzato utilizzando il linguaggio **adagio**, sviluppato presso la **Carnegie Mellon University** per la scrittura di partiture. Adagio è stato implementato su diversi elaboratori utilizzando sia il sistema operativo Unix che Dos, questi ultimi più facilmente dotati di

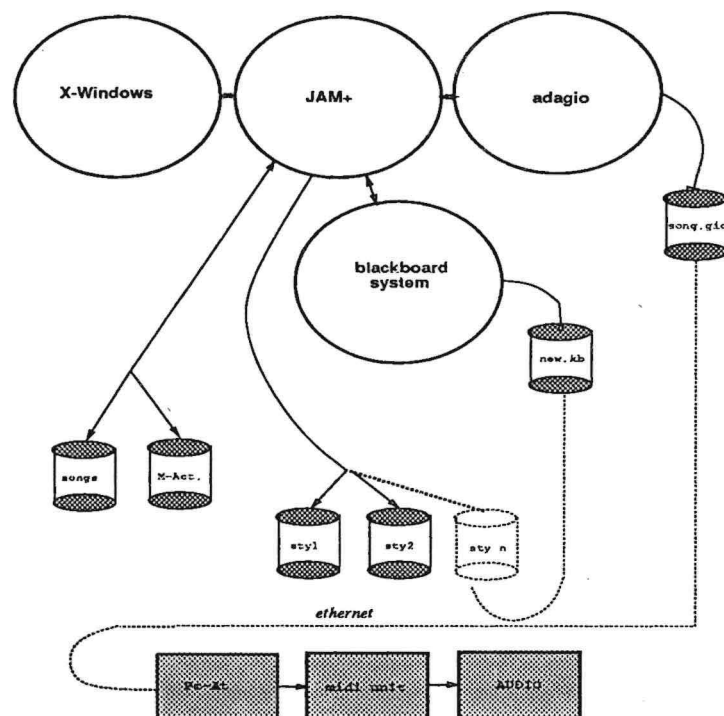


Figure 1: Schema a blocchi del sistema

interfaccia midi. In tal modo si ottiene un sistema di ascolto , anche se off-line.

Sistema di Segmentazione Questo modulo, basato su di un pianificatore a lavagna, permette di costruire una nuova base di conoscenza (kb), a partire da file relativi ad improvvisazioni reali, come ad esempio temi estratti da standard di Jazzistici o trascrizioni di improvvisazioni. Tale nuova kb comprende le frasi fondamentali utilizzate all'interno dell'esecuzione. Ciò è utile poichè velocizza la fase di istruzione all'utilizzatore del sistema, senza la quale il sistema risulta essere difficilmente utilizzabile.

3 Rappresentazione della Conoscenza

La rappresentazione della conoscenza è fondata sul concetto di **frame**. Tipicamente un frame è costituito da una collezione di caselle (slot) che definiscono una classe di oggetti: queste caselle a loro volta possono "riferirsi ad" altri frame. In una casella è possibile anche inserire dei valori di default. In una slot è possibile inserire parti di codice come la *procedure if-added* e la *procedure if-needed* che determinano una serie di operazioni da eseguire nel caso di assegnazione od estrazione di un valore da una slot. Essendo possibile effettuare operazioni di composizione all'interno di frames, sono state definite le relazioni:

- ISA - Specifica la tassonomia gerarchica, consente meccanismi di eredità
- PART-OF - Determina il legame tra un oggetto e le sue componenti.

L'informazione simbolica immagazzinata nella base di conoscenza è una rappresentazione astratta di un particolare mondo musicale.

La base di conoscenza risulta formata da:

- Una Memoria a Lungo Termine - dove è memorizzato il mondo delle conoscenze sotto forma di azioni, attori e classi.
- Una Memoria di Lavoro - che contiene le azioni e le istanze create durante l'uso del sistema.

Con il termine *attore* si considera sia l'attività atomica (procedura che implementa un dato algoritmo di analisi e sintesi) contenuta nel singolo frame, che l'azione, che è l'attività compiuta da uno o più attori.

Ogni azione è un sotto-albero la cui radice è un frame collegato a tutti i nodi (frames) attori.

Nell'architettura della base di conoscenza possiamo individuare un certo numero di strutture base, che servono per rappresentare l'armonia, la struttura degli accordi, le strutture ritmiche e melodiche. Tutte queste strutture si riconducono al frame padre di tutta la base di conoscenza: la classe delle **M-Actions**. La classe M-Actions definisce tutte le proprietà generali legate allo sviluppo dei processi musicali; in generale si può distinguere tra due categorie:

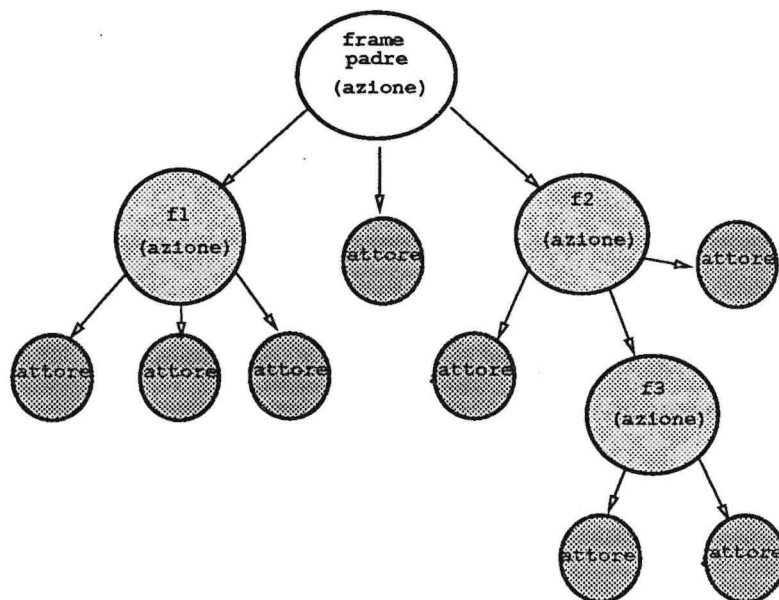


Figure 2: Esempio di Rete

locali - relative alle caratteristiche melodiche, armoniche e ritmiche del brano,

connette - relative alla struttura temporale e globale (stesura) del brano.

Queste seconde proprietà servono per collegare le varie sezioni del brano e gestiscono il flusso di controllo all'interno dei vari processi.

L'architettura della base di conoscenza fonda le relazioni temporali tra gli oggetti della rete sui wires (fili), proprietà connettive che permettono il passaggio di messaggi tra processi, messaggi che possono comportarsi anche come segnali di attivazione e disattivazione di processi. Il frame padre da cui parte l'esecuzione della rete è una azione che attiva un Link-Simbolico che fa puntare il frame stesso ad una sequenza di azioni che devono essere eseguite. Ogni frame può essere azione oppure attore. Nel primo caso ciò porterà alla valutazione di un certo numero di condizioni e quindi all'esecuzione di un altro ramo della rete che potrà essere composta a sua volta di azioni od attori, nel secondo si tratterà di un ramo terminale della rete, che dopo la sua esecuzione porterà il flusso di controllo ad un livello precedente per poi seguire un nuovo link. Con ramo terminale si intende un ramo che

non contiene azioni, ma può contenere links a uno o più attori.

4 Strutture musicali gestite da Jam+

Una delle caratteristiche fondamentali dell'improvvisazione jazzistica è l'importanza relativa che viene data allo spartito, che a differenza di altri ambiti come ad esempio la musica classica, non indica nota per nota l'armonia e la melodia del brano, ma rappresenta uno schema armonico ed un percorso melodico su cui l'esecutore, sfruttando la sua creatività e la sua esperienza si muove. Il compositore non impone ma suggerisce, personalizzando l'armonia con soluzioni alternative, e la melodia con movimenti improvvisati e rielaborazioni ritmiche. Il disegno melodico, pur rivisitato assume in questo modo un aspetto particolare, conservando sempre l'immutato movimento armonico. La realizzazione di un brano può essere divisa in due fasi principali:

- Lo Sviluppo dell'Armonia
- Lo Sviluppo dell'Improvvisazione

Nella prima fase viene costruito lo scheletro fisico della song, ossia il frame relativo alla song, e quindi vengono impostate le azioni che portano alla costruzione delle parti di cui è composta, i giri di cui sono composte le parti, ed infine gli accordi presenti nei giri. Al momento in cui risultano presenti tutte le caratteristiche del brano, viene generata la M-Action relativa, cioè il frame di azioni ed attori in cui, tramite l'esecuzione della rete, inserire gli accordi, che portano alla definizione della struttura armonica, ed infine alla costruzione della linea melodica.

In questa seconda fase viene sviluppata l'improvvisazione facendo uso dei meccanismi di valutazione e pianificazione.

4.1 Sviluppo dell'Armonia

Lo sviluppo della parte armonica del brano può essere fatto in diversi modi:

- Costruzione di song nuova
- Costruzione a partire da struttura già esistente
- Caricamento di song già utilizzata precedentemente

Nel primo caso intervengono i moduli che permettono la costruzione di una nuova M-Action, che portano alla costruzione dello scheletro della song. Nel secondo caso, essendo già presente il file relativo alla M-Action, si giunge a questo punto tramite la selezione della song da caricare. A questo punto in entrambi i casi la rete prescelta viene eseguita, tramite le azioni presenti all'interno della M-Action, e tramite l'ausilio dell'interfaccia grafica viene inserita la struttura armonica relativa, ottenendo così un aggiornamento della base di conoscenza presente. Nel terzo caso si giunge a questo stadio semplicemente tramite il caricamento della song prescelta, che rappresenta un congelamento della base di conoscenza presente nel sistema in questo istante, relativo però ad un'altra seduta di lavoro in cui la parte armonica era stata costruita utilizzando la prima opzione.

4.2 Sviluppo dell'Improvvisazione

Costruire un'improvvisazione può sembrare un controsenso, ma in realtà non lo è.

Molti musicisti che sono entrati a fare parte della storia del jazz, come Charlie Parker o più tardi il grande Jaco Pastorius avevano una concezione improvvisativa profondamente legata a patterns, ossia vedevano l'improvvisazione come una sequenza di frasi note della loro struttura, e non costruite sul momento basandosi su istinto e sentimento. Con ciò questa filosofia improvvisativa non è assolutamente da disprezzare poiché fonda le sue radici su valide regole di armonia funzionale. Il musicista che affronta l'improvvisazione con questa tecnica può muoversi liberamente sulla melodia e sugli accordi che definiscono l'armonia del brano. Per la costruzione dell'improvvisazione vengono usati un certo numero di moduli legati tra loro; al motore inferenziale del sistema Jam+, che poi non è altro che un modulo che permette l'esecuzione della rete generata al passo precedente, si affianca il modulo di pianificazione

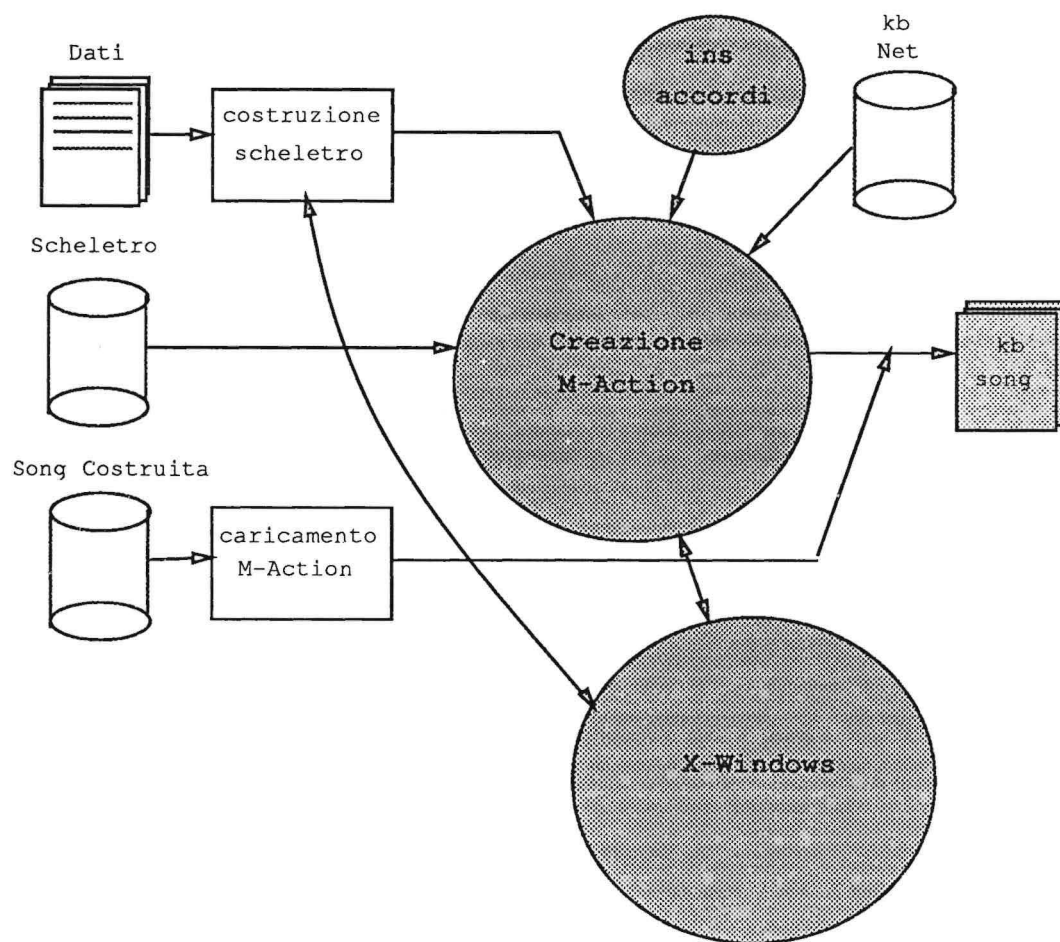


Figure 3: Sviluppo dell'Armonia

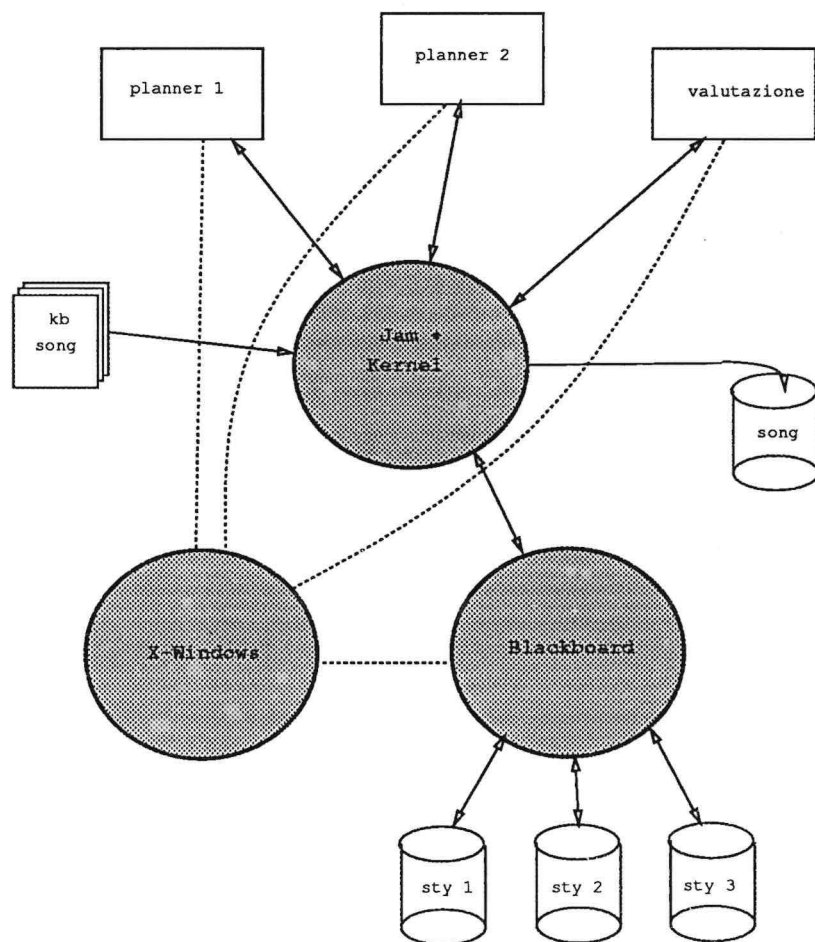


Figure 4: Moduli Relativi allo Sviluppo dell'Inprovvisazione

a lavagna che in questa fase si occupa solo di gestire il collegamento con le basi di conoscenza relative agli stili disponibili, ed il modulo di valutazione, che note le caratteristiche armoniche del brano effettua una operazione di valutazione dei frammenti presenti all'interno delle basi di conoscenza prescelte. La fase di pianificazione e quella di esecuzione sono, come è già stato detto, strettamente correlate, questo è dovuto alle caratteristiche degli eventi che si verificano all'interno del funzionamento del sistema stesso, che a loro volta sono legati alle decisioni del pianificatore al passo precedente; queste sono le ragioni che portano alla definizione di una fase di replanning ossia alla modifica delle regole di pianificazione tra un passo ed il successivo.

Il sistema Jam+ in questa fase ha diverse possibilità di funzionamento:

Sistema Esperto - l'esecuzione è totalmente automatica, l'utente deve semplicemente al principio dell'improvvisazione inizializzare la scelta delle regole da utilizzare nei vari blocchi; poi i blocchi stessi, durante il funzionamento, modificano l'attivazione di queste regole.

Sistema Parzialmente Guidato - Solo parte dei sistemi di valutazione e pianificazione sono attivi, il resto del lavoro viene svolto dall'utente tramite gli ausili forniti dall'interfaccia grafica.

Sistema Guidato - I sistemi di valutazione e pianificazione sono esclusi, tutte le scelte devono essere effettuate dall'utente.

Le frasi considerate sono frammenti musicali dotati di senso compiuto, con il caricamento all'interno della memoria di lavoro. Ogni frase definisce un frame. Oltre ai dati che sono contenuti nel file, viene effettuata una prima operazione che porta all'estrazione di parametri che, sarebbe inutile e dispendioso calcolarsi nei punti del programma in cui vengono utilizzati. Alla fase di caricamento delle basi di conoscenza di frasi, segue la fase di costruzione dell'improvvisazione vera e propria: alla base di ciò vi è il sistema di valutazione. L'obiettivo del modulo è di effettuare una prima scelta di frasi a seconda del contesto in cui ci si trova, da una base di conoscenza che può contenere un grande numero di frasi (in media un centinaio), estrae, nove frasi che rispecchiano i caratteri richiesti dalle regole applicate. Il sistema di valutazione è l'unico modulo che viene utilizzato in tutti i modi di funzionamento del sistema. La valutazione viene effettuata utilizzando un gruppo di regole euristiche che penalizzano il voto che viene assegnato alla frase.

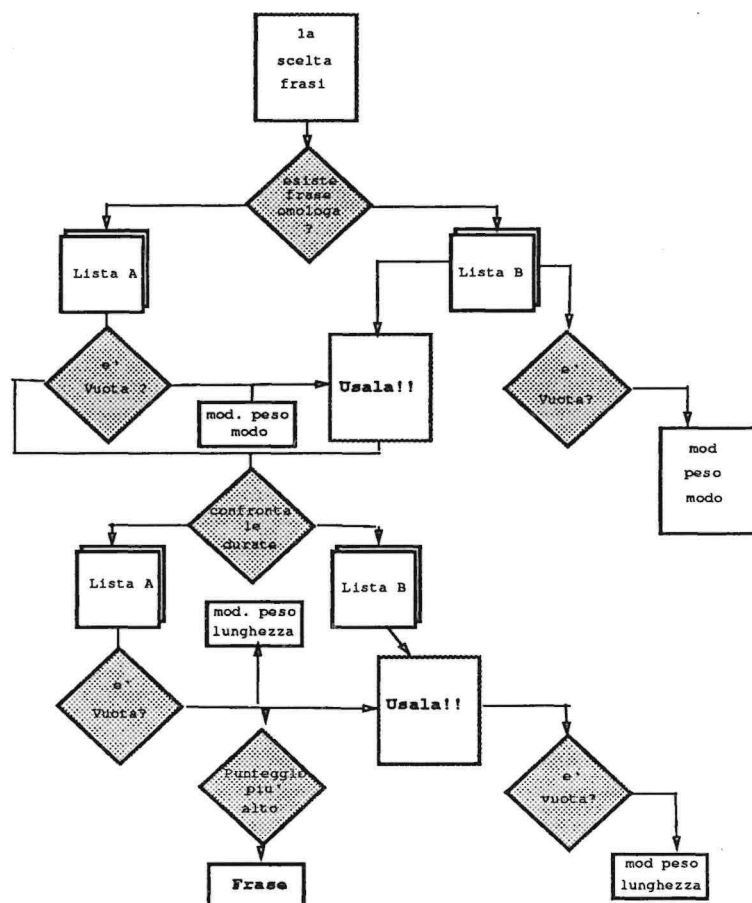


Figure 5: Strategia di Pianificazione 1

Ogni regola effettua una analisi sulla frase e fornisce un giudizio di penalità. Ad ogni regola è assegnato un peso che quindi fa variare l'importanza della regola stessa all'interno del contesto di valutazione:

$$\text{Voto} = 100 - \alpha * \text{rule1} - \beta * \text{rule2} - \gamma * \text{rule3} \dots\dots$$

A questo punto vengono estratte dall'elenco le frasi che hanno ottenuto un punteggio superiore ad un certo limite e di queste, dopo avere ordinato la lista, vengono scelte quelle che si trovano al centro. Effettuata la prima scelta, in caso di funzionamento automatico, la lista contenente le frasi è passata al primo pianificatore. Il primo test che viene effettuato sulla lista riguarda le caratteristiche modali delle frasi: vengono divise in una prima

scelta, contenente frasi dello stesso modo dell'accordo corrente e una seconda scelta composta di frasi costruite su altri modi. Nel caso che una delle due liste resti vuota, viene effettuata una correzione nella pesatura della regola modale della valutazione: se non risultano essere presenti frasi dello stesso modo dell'accordo, viene aumentato il peso: in caso che non risultino presenti le altre, viene ridotto. Considerando la lista di seconda scelta solo come un ripiego, ossia solo se non esistono scale di prima scelta, le scale vengono ancora suddivise in due gruppi, quelle che hanno una durata simile all'accordo su cui devono suonare, e le altre. Giunti qui avremo a disposizione due nuovi insiemi di frasi, il sistema andrà a questo punto a scegliere una frase, utilizzando come criterio di valutazione il voto, provando prima a cercare una frase ovviamente nel primo insieme e, solo in caso di totale mancanza, nel secondo.

Ne caso che uno dei due insiemi sia vuoto viene alterato il peso relativo alla regola di valutazione che si occupa della lunghezza delle frasi, riducendolo in caso che la prima scelta sia vuota, aumentandolo in caso contrario.

La frase scelta tramite il primo pianificatore o, in caso di funzionamento guidato, dall'utente tramite l'interfaccia grafica, passa al secondo pianificatore che ne confronta la durata con lo spazio disponibile, che è rappresentato dalla durata dell'accordo più eventuali eredità dovute alle durate delle frasi precedentemente scelte.

Si possono verificare tre possibilità :

- Frase della stessa durata dello spazio a disposizione.
- Frase corta rispetto alla durata a disposizione.
- Frase lunga.

Nel primo caso la frase viene direttamente inserita nella struttura della song, senza effettuare alcuna modifica sui pesi del sistema di valutazione per il passo successivo, poichè questo è stato un passo con tutte le caratteristiche del successo, quindi i pesi dovrebbero essere tarati adeguatamente.

Nel secondo caso si presentano tre alternative: se la frase è molto corta (con l'aggettivo molto si intende che la lunghezza della frase è al massimo pari ad 1/6 dello spazio disponibile), allora viene attivato, tramite procedura ricorsiva, il sistema di valutazione, che si occupa di scegliere un giusto seguito. In questa operazione vengono anche inserite le regole che si occupano della

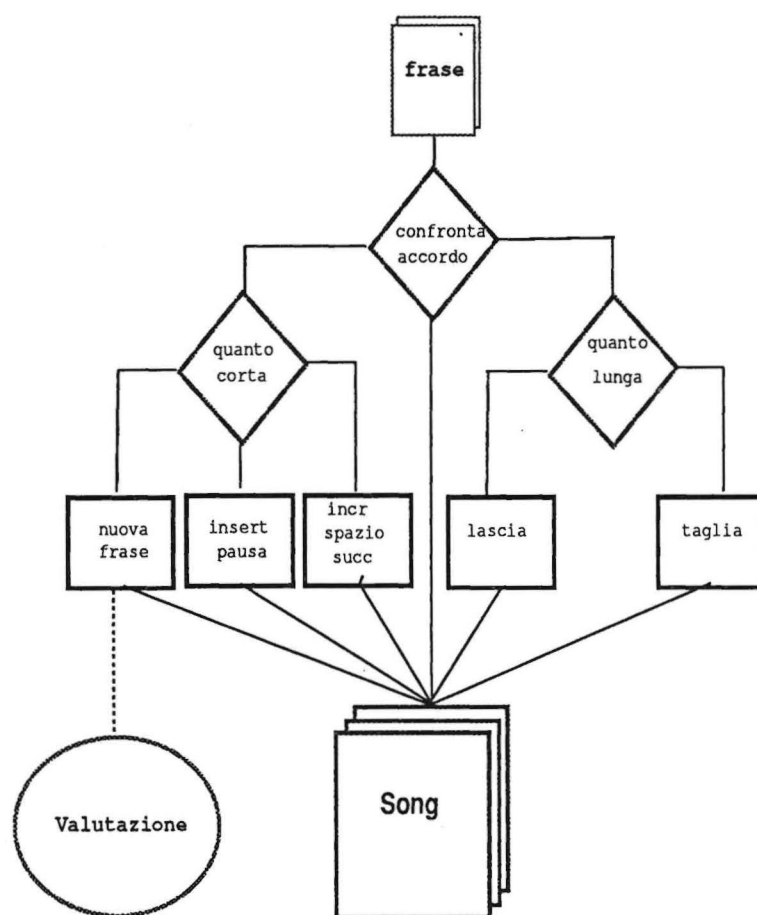


Figure 6: Strategia di Pianificazione 2

lunghezza delle frasi da scegliere, in modo da ottenere il più possibile frasi di lunghezze tali da inserirsi nello spazio disponibile.

Se lo spazio lasciato dalla frase è minore di $1/3$ della durata della frase stessa, viene inserita una pausa e vengono disattivate le regole che si occupano della valutazione della lunghezza delle frasi stesse. Se lo spazio lasciato è breve (minore di $1/6$ di quello totale a disposizione), viene inserita la frase nella struttura della song e fatto procedere il passo seguente, penalizzando però la scelta di frasi di lunghezza diversa dallo spazio a disposizione.

Nel terzo caso si presentano due alternative: se la frase risulta molto lunga, ossia la sua durata è maggiore di quella dello spazio a disposizione sommato ad $1/3$ della durata dell'accordo successivo, la frase viene tagliata. L'operazione di taglio viene effettuata cercando di non fare perdere il carattere alla frase stessa ossia il senso ; per fare ciò non vengono semplicemente eliminate delle note in coda alla frase stessa, ma ridotte le durate in modo proporzionale alla durata da ridurre, una sorta di compressione temporale. Oltre a ciò, vengono eliminate dal sistema di valutazione le regole che condizionano la lunghezza delle frasi.

Nel caso in cui lo spazio lasciato risulta minore, la frase viene lasciata nella sua totalità ed inserita nella song. Ovviamente questo andrà a condizionare le scelte del sistema di valutazione al passo successivo, sia perchè lo spazio a disposizione verrà ridotto, sia perchè saranno inserite le regole di valutazione relative alla lunghezza delle frasi.

Le regole su cui è costruito questo pianificatore sono, come per il precedente, totalmente euristiche e si basano su criteri estetici di valutazione delle frasi musicali, in parte mie ed in parte tratte dal testo scritto da Lerdhal e Jackendoff: "A Generative Theory of Tonal Music". La scelta che il secondo pianificatore modifichi l'azione delle regole del meccanismo di valutazione in modo ON-OFF e anch'essa una scelta personale, legata solo all'intento di fare sì che i vari moduli abbiano la possibilità di interagire tra loro e poter operare sul sistema a regole a più livelli.

5 Architettura di Controllo a Blackboard

L'architettura a blackboard è stata sviluppata la prima volta per l' HEARSAY-II , un sistema di comprensione del linguaggio naturale real-time, ed è poi stato utilizzato in molti altri campi.

La logica di funzionamento risulta fondata sull'ottimizzazione della ricerca e dell'assemblaggio intelligente di soluzioni parziali precedentemente sviluppate. Tutti gli elementi candidati a fare parte della soluzione vengono registrati in un database globale strutturato denominato blackboard . La struttura a blackboard organizza gli elementi facenti parte della soluzione su due assi: intervalli di soluzione e livelli di astrazione. Gli intervalli di soluzione rappresentano regioni di soluzione, ossia le possibili soluzioni parziali. Gli elementi facenti parte della soluzione vengono generati, registrati ed elaborati sulla blackboard da processi indipendenti chiamati sorgenti di conoscenza o ksars (knowledge source activation record) , questi processi hanno la forma:

If condition then action

dove la condizione descrive la situazione in cui il processo può contribuire allo

sviluppo della soluzione; l'azione specifica il comportamento della sorgente di conoscenza, solo i processi di cui sono soddisfatte le condizioni vengono attivate.

Ogni mutamento nel contenuto della blackboard costituisce un evento che, in presenza di specifiche condizioni, può attivare altri processi. In generale le ksars sono indipendenti tra loro e non hanno la possibilità di attivarsi a vicenda, ma cooperano, nel senso che agiscono, in tempi diversi sulla stessa base di conoscenza.

L'organizzazione del sistema è gestita tramite un sistema di schedulazione di jobs e tre ksars fondamentali. Lo sviluppo della soluzione avviene tramite un meccanismo ciclico. Ad ogni ciclo di soluzione del problema lo schedatore sceglie una singola sorgente di conoscenza perchè esegua la sua azione. Generalmente ad ogni ciclo più processi possono avere le condizioni soddisfatte per fare eseguire la propria azione. Per evitare conflitti è stato inserito un meccanismo di schedulazione che usando un insieme di criteri di valutazione sceglie il processo da attivare. L'esecuzione di una sorgente di conoscenza applica vari metodi di inferenza che portano alla generazione della soluzione parziale, che poi viene nuovamente inserita nella blackboard. Alla base del

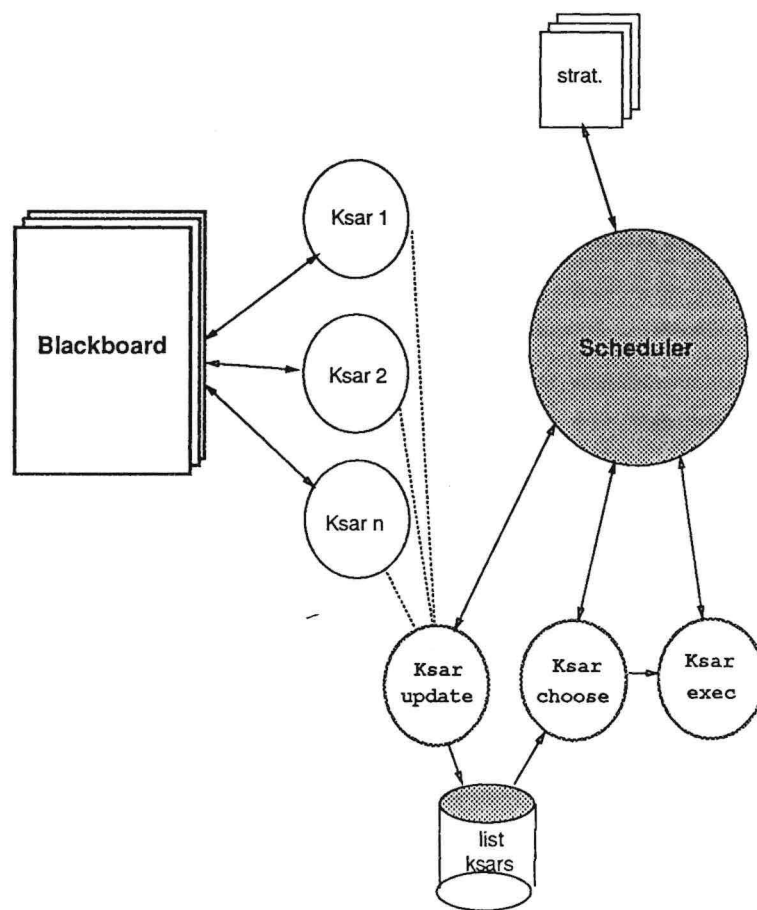


Figure 7: Schema del blackboard system

funzionamento del meccanismo di schedulazione vi sono le tre ksars fondamentali attive ad ogni ciclo, che si occupano dell'enumerazione dei processi attivabili, della scelta del processo attivabile e dell'esecuzione del processo stesso. Queste ksars, a differenza delle altre, non hanno conoscenze specifiche, ma semplicemente confrontano conoscenze proprie del dominio controllando la blackboard. Pur essendo definite anch'esse come sorgenti di conoscenza, a differenza di tutte le altre si influenzano a vicenda, nel senso che la loro attivazione è serializzata ed i risultati delle elaborazioni non portano direttamente alla modifica della blackboard, ma all'attivazione di altri processi:

1. **update-to-do-set** : verifica tra le ksars presenti nel sistema quali sono quelle che hanno le condizioni di attivazione necessarie verificate al passo presente, e crea una lista di preferenza delle stesse a seconda del punteggio che è stato calcolato nella valutazione.
2. **choose-ksar** : sceglie la ksar da eseguire al passo corrente tra quelle identificate dalla (1) utilizzando una strategia (un insieme di regole di valutazione delle ksars che variano durante l'esecuzione del processo)
3. **exec-ksar** : esegue la ksar scelta al passo precedente.

Lo schedulatore si occupa della sincronizzazione del sistema, l'algoritmo è parzialmente sequenziale e parzialmente parallelo. In particolare, la valutazione delle ksars, sempre che l'elaboratore utilizzato lo permetta, può essere fatta totalmente in modo parallelo, mentre le fasi di esecuzione delle ksars devono necessariamente essere svolte in modo sequenziale. Le operazioni svolte dallo schedulatore sono quindi:

1. La generazione del ciclo, poichè il sistema svolge le sue operazioni per passi successivi, ad ogni ciclo viene effettuata l'operazione di valutazione, tramite la strategia presente viene scelta una ksar che quindi viene eseguita.
2. Il controllo della condizione di arresto, che a questo stadio dello sviluppo del sistema è rappresentata dal raggiungimento di un certo numero massimo di passi, lo svuotamento totale della blackboard oppure la mancanza di condizioni per la realizzabilità di alcuna ksar.
3. L'aggiornamento della strategia, svolto in funzione del numero di passi effettuati.

4. L'esecuzione in sequenza delle tre ksar fondamentali ossia l' update-to-do-set , la choose-ksar e la exec-ksar .

Le *knowledge sources activation record* rappresentano il cuore del blackboard system. Escludendo per un attimo le tre ksars fondamentali che hanno struttura differente, la struttura di tutte le ksars può essere divisa in due parti: condizioni necessarie per l'attivazione e azione. Difatti le ksars ad ogni passo, in fase di ricerca, vengono tutte attivate nella loro prima parte, ossia ogni ksar va ad analizzare i segmenti presenti nella blackboard per vedere esistono le condizioni per permetterne l'attivazione e con quale punteggio. Tramite la exec-ksar viene poi attivata l' azione della ksar prescelta. Quando il trigger scatta crea una immagine di attivazione che descrive le ksars che dovrebbero essere attivate e l'evento specifico che ha fatto scattare l'innescio; questa seconda informazione può essere usata per focalizzare l'attenzione della ksar quando viene attivata realmente. Le ksar fondamentali hanno invece una struttura differente: non esistono condizioni di attivazione, ma solo azioni. La loro attivazione è stabilita dallo schedulatore.

