

Elaborazione di segmenti temporali

Pierre Dutilleux, Giovanni De Poli, Adrian von dem Knesebeck e Udo Zölzer*

(Traduzione ricevuta il 7 gennaio 2016¹)

Sommario

Il testo si occupa delle elaborazioni del suono nel dominio del tempo discreto, considerando una fase di segmentazione, una successiva di analisi, ed una strategia di ricomposizione del segnale temporale al fine di ottenere trasformazioni sull'audio risintetizzato. I vari algoritmi sono presentati illustrando le tecniche analogiche storiche oltre a sviluppi e implementazioni mediante le tecniche dell'audio digitale. Sono altresì citati brani antologici dove compositori e musicisti utilizzano tali elaborazioni, contribuendo ad illustrare l'evoluzione delle tecniche della musica elettronica e dell'elaborazione sonora.

1 Introduzione

In questo capitolo discutiamo diversi algoritmi nel dominio del tempo in quanto combinazioni di blocchi di elaborazione più piccoli, come modulatori di ampiezza/fase, filtri e linee di ritardo. Questi effetti influenzano principalmente l'altezza e la durata temporale del segnale audio. Prima introdurremo alcuni effetti di base come la riproduzione a velocità variabile ed il ricampionamento con controllo dell'altezza. Questi effetti sono basati sulla modulazione mediante linee di ritardo e sulla modulazione di ampiezza. Poi parleremo di due approcci per la dilatazione temporale (scalatura temporale) di segnali audio. Essi si basano su una fase di analisi, in cui il segnale di ingresso è diviso in segmenti (blocchi) di lunghezza fissa o variabile, e una fase di sintesi in cui i blocchi della fase di analisi vengono ricombinati da procedure di sovrapposizione e somma. Queste tecniche di dilatazione temporale ottengono una scalatura temporale senza modificare l'altezza del segnale. La quarta sezione si concentra sulla modifica dell'altezza, e introduce tre tecniche: elaborazione del blocco in base alla dilatazione temporale e ricampionamento, modulazione della linea di ritardo ed elaborazione del blocco sincronizzata con il periodo. L'elaborazione del blocco basato sulla modulazione della linea di ritardo ottiene la modifica dell'altezza scalando l'involuppo spettrale di ogni blocco. L'elaborazione del blocco sincronizzata con il periodo ottiene la modifica dell'altezza dal ricampionamento dell'involuppo spettrale di ogni blocco, preservando l'involuppo spettrale originale. L'ultima sezione presenta un uso più creativo della elaborazione nel tempo dei segmenti quali la ricombinazione nel tempo e la granulazione. Brevi segmenti del segnale di ingresso sono liberamente assemblati e posizionati nell'asse temporale del segnale di uscita.

*Energy Engineering GmbH, DEI – Università di Padova, Department of Signal Processing and Communication – Helmut-Schmidt University di Hamburg, Department of Signal Processing and Communication – Helmut-Schmidt University di Hamburg.

¹Il presente articolo è la traduzione, a cura di Roberto Neri, di Pierre Dutilleux et al. (2011). «Time-segment processing». In: *Digital Audio Effects*. A cura di Udo Zölzer. 2^a ed. John Wiley & Sons, Ltd. Cap. 6, pp. 185–217. Revisione della traduzione di Giovanni De Poli ricevuta il 31 marzo 2016.

In questo caso il suono in ingresso può essere molto meno riconoscibile in quello di uscita. L'ampia scelta di strategie di organizzazione dei segmenti temporali implica un atteggiamento alla composizione sonora da parte dell'utente.

2 Riproduzione a velocità variabile

Introduzione

I registratori audio analogici permettono la riproduzione con un ampio intervallo [range] di velocità. In particolare, l'avvolgimento veloce in avanti o indietro, se il monitoraggio del segnale audio è possibile, può essere utilizzato per localizzare un suono. Durante la riproduzione a maggiore velocità, l'altezza del suono cresce e durante la riproduzione a minore velocità, l'altezza del suono diminuisce. Con questa tecnica la durata del suono è più lunga se il registratore riproduce più lentamente ed è più corta se la velocità del registratore viene aumentata. La Fig. 1 illustra un segmento di suono che è stato allungato ed accorciato, e gli spettri corrispondenti.

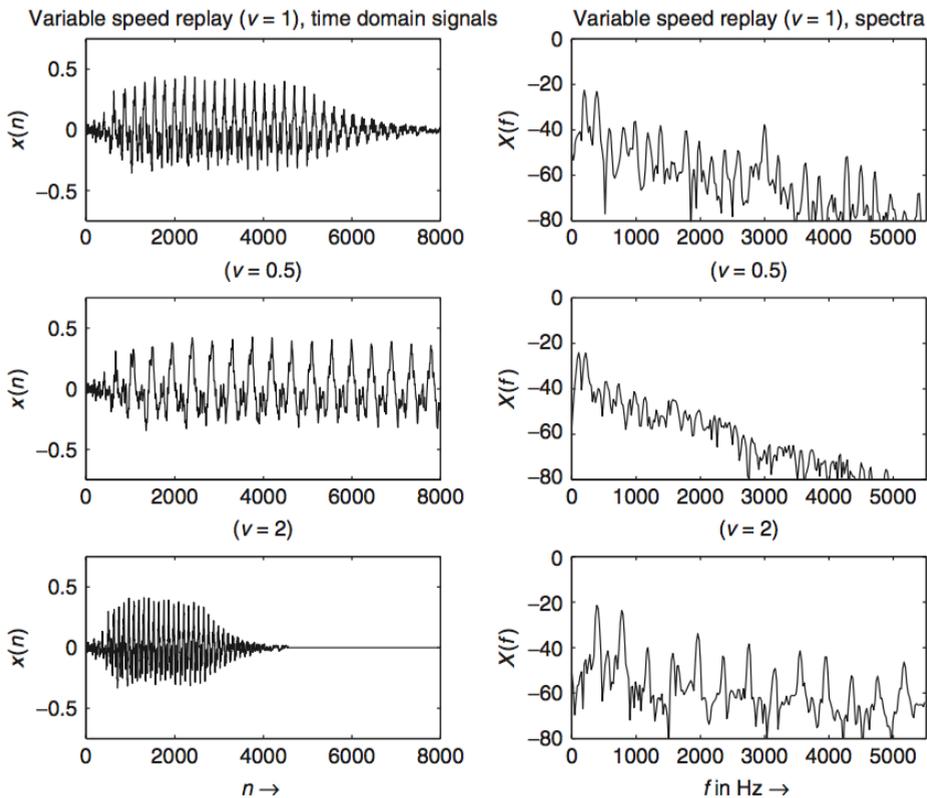


Figura 1: Pitch shifting: La riproduzione a velocità variabile comporta una compressione/espansione temporale e compressione ed espansione dell'involuppo spettrale.

Elaborazione del segnale

La frase “riproduzione a velocità variabile” denota che quello che è successo inizialmente durante l’istante $nT_{s,in}$ sta adesso accadendo nel tempo

$$nT_{s,replay} = nT_{s,in}/v \quad (1)$$

alla velocità relativa v , dove $nT_{s,in}$ e $nT_{s,replay}$ sono i periodi di campionamento iniziale e di riproduzione. L’espansione temporale corrisponde a $v < 1$. Un semplice metodo per implementare la riproduzione a velocità variabile è perciò la modifica della frequenza di campionamento durante la riproduzione del suono in accordo con

$$f_{s,replay} = f_{s,in} \cdot v \quad (2)$$

dove $f_{s,in}$ and $f_{s,replay}$ sono le frequenze di campionamento iniziale e di riproduzione. Si dovrebbe distinguere il caso in cui l’uscita sia digitale o analogica. Se l’uscita è analogica, allora il reale metodo effettivo è quello di modificare la frequenza di campionamento del DAC di uscita. Lo spettro del segnale è scalato di v ed il filtro analogico di ricostruzione dovrebbe essere realizzato per rimuovere le immagini spettrali dopo la conversione (Gaskell 1987; Massie 1998).

Se è richiesta una uscita digitale, allora la frequenza di conversione deve essere regolata tra la frequenza di riproduzione desiderata $f_{s,replay}$ e la frequenza di campionamento di uscita $f_{s,out}$, che di solito è uguale a $f_{s,in}$.

Se $v < 1$ (espansione temporale) allora $f_{s,in} > f_{s,replay} < f_{s,out}$ e servono più campioni di uscita di quelli disponibili dal segnale di ingresso. Il segnale di uscita è una versione interpolata (sovra-campionata) di un fattore $1/v$ del segnale di ingresso. Se $v > 1$ (compressione temporale) allora $f_{s,in} < f_{s,replay} > f_{s,out}$ e sono necessari meno campioni di uscita di quelli disponibili dal segnale di ingresso. Il segnale di ingresso è decimato di un fattore v . Prima della decimazione la larghezza di banda del segnale in ingresso deve essere ridotta a $f_{s,replay}/2$ da un filtro passa basso digitale (McNally 1984). La qualità della conversione della frequenza di campionamento dipende fortemente dal filtro di interpolazione usato. Un metodo molto diffuso è l’interpolazione lineare tra due campioni adiacenti. Una disamina sui metodi di interpolazione può essere trovata in Crochiere e Rabiner (1983) e Massie (1998).

Una implementazione in tempo discreto può essere ottenuta incrementando o decrementando il tasso di trasferimento al DAC di un segnale audio digitale registrato, in modo da cambiare la frequenza di campionamento di uscita rispetto alla frequenza di campionamento di registrazione. Se il segnale di uscita dovrà essere ancora in formato digitale, dovremmo ricampionare il segnale analogico a velocità variata con la corrispondente frequenza di campionamento. Una implementazione tempo-discreto senza una conversione DA e una nuova conversione AD è stata proposta in McNally (1984) ed è mostrata nella Fig. 2. Questa fa uso di tecniche multi frequenza di elaborazione del segnale e si avvale di una approssimazione della conversione DA/AD. Un ulteriore algoritmo di elaborazione del segnale per ottenere il risultato acustico di una riproduzione a velocità variabile è una modulazione di una linea di ritardo con un cambiamento costante di altezza, che sarà discusso nella Sezione 4.3.

Applicazioni musicali e controllo

Così come per l’editing audio nei magnetofoni, anche nei sistemi audio digitali la riproduzione con velocità variabile è molto usata. Vedi Wishart (1994, ID 2.9 e 2.10) per una chiara

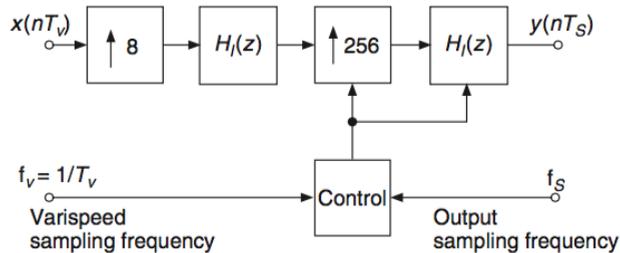


Figura 2: Schema per la riproduzione a velocità variabile.

dimostrazione dell'effetto su un segnale vocale.

L'effetto di trasposizione tramite la velocità del magnetofono è stata usata da Les Paul nel brano intitolato *Whispering* nel 1951 (Lee 1972). Questo metodo è stato usato molto frequentemente nella musica elettroacustica quando l'altezza di un suono concreto non può essere controllata durante la registrazione. P. Schaeffer progetta il *Phonogène chromatique* per trasporre in ciascuno dei 12 gradi della scala temperata. I pinch rollers potevano essere controllati con una tastiera simile ad un pianoforte. Un ingranaggio aggiuntivo estendeva il range delle operazioni a due ottave (Geslin 2000; Moles 1961; Roads 1996). Jacques Poullin aveva sviluppata un'altra versione, il *Phonogène à coulisse*, che permetteva una modifica di velocità continua. Una coppia di coni, con una ruota di frizione in mezzo, costituiva un collegamento meccanico a rapporto variabile tra il motore ed il capstan del magnetofono. La posizione della frizione, e quindi la velocità di riproduzione, era controllata a livello meccanico. Stockhausen, in *Hymnen* traspose il suono dell'orchestra per conferirle un carattere travolgente ed apocalittico (Chion 1982).

Anche nella computer music, la riproduzione a velocità variabile fornisce un efficace sistema di trasposizione. J. C. Risset dice: «miscelando un suono con la trasposizione di se stesso con piccole differenze di frequenza (diciamo un ventesimo di Hertz) si possono convertire toni a periodo costante in un pattern dove gli armonici del tono crescono e cambiano con tassi differenti, proporzionalmente al grado dell'armonico» (Risset 1987, 1998). In *The gates of H* Ludger Brümmer esplora la proprietà che la riproduzione a velocità variabile ha di modificare sia l'altezza sia la durata di un campione (Brümmer 1993, 14'40"-17'25"). Sette copie della stessa frase suonate simultaneamente alle velocità di 7.56, 4.49, 2.24, 1.41, 0.94, 0.67, 0.42, 0.31 vengono sovrapposte. Il suono risultante inizia con una struttura complessa ed uno spettro esteso. Al proseguire del suono, le copie più veloci svaniscono e le versioni più lente emergono l'una dopo l'altra. La struttura del suono si semplifica progressivamente e si evolve verso i registri più bassi.

Il carattere dei suoni trasposti è modificato in quanto tutti gli aspetti dello spettro scalano simultaneamente. Le formanti sono scalate in alto, arrivando ad un effetto 'Topolino', o sono scalate verso il basso, come se i suoni fossero prodotti da oggetti ingigantiti. La struttura del tempo è altresì modificata. I transitori sono allungati o contratti. Un vibrato iniziale nel suono perderà il suo carattere ed apparirà come una modulazione o più lenta o più veloce. I suoni possono essere anche riprodotti a velocità negative. Una velocità -1 produce un suono con lo stesso spettro medio, anche se suona in maniera molto diversa. Pensate ad esempio ad un parlato o ad un suono percussivo riprodotti al contrario. Altri schemi di trasposizione che sono

liberi da questi ostacoli possono essere ottenuti da metodi più sofisticati².

Una particolare applicazione fu richiesta dal compositore Kiyoshi Furukawa. Lui volle un campionatore per il quale la velocità fosse controllata dall'ampiezza di uno strumento acustico. Un suono è memorizzato in un campionatore ed è suonato in loop. Nel frattempo, l'RMS dell'ampiezza di un segnale di controllo in ingresso è calcolato e mediato nel tempo con costanti di tempo di attacco e decadimento indipendenti. L'ampiezza è convertita in *dB* e scalata prima di controllare la velocità del campionatore. I parametri devono essere accordati in una maniera tale che la velocità rimanga all'interno di un intervallo valido e le variazioni di velocità siano intimamente correlate al volume ed alla dinamica dello strumento (vedi Fig. 3).

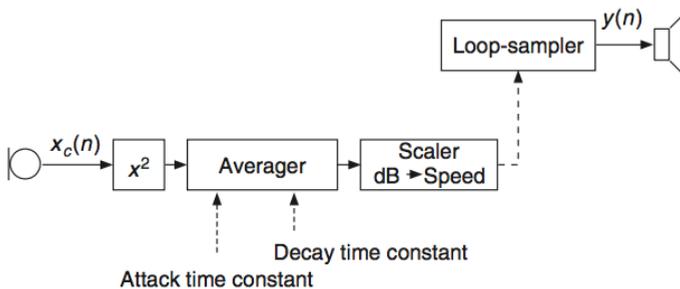


Figura 3: Un loop-sampler controllato da un segnale acustico.

Questo effetto è controllato da un clarinetto in *Swim, swan* e da una viola in *Den Ungebornen Gotten* (Furukawa 1993, 1997). L'altezza dello strumento acustico seleziona parole da un insieme predefinito, mentre il volume controlla la velocità di riproduzione di queste parole.

3 Time Stretching [Allungamento o accorciamento temporale]

Introduzione

Al fine di comprendere il time stretching, prendiamo l'esempio di un segnale la cui durata non combacia con la porzione temporale che è stata riservata per il suo uso. Pensate ad uno speaker che ha già registrato 33 secondi di parlato, ma il cui contributo ad una pubblicità non dovrebbe essere superiore a 30 secondi. Se non vuole registrare nuovamente il suo discorso, il fonico può artificialmente contrarre la durata del suo parlato del 10%. Con il termine "time stretching" ci riferiamo alla contrazione o espansione della durata di un segnale audio. Abbiamo visto nel paragrafo precedente un metodo che altera la durata del suono, la riproduzione a velocità variabile ma questo ha l'inconveniente di trasporre contestualmente il suono. L'*Harmonizer* potrebbe essere usato per trasporre il suono in direzioni opposte e la combinazione di entrambi i metodi porta ad un algoritmo di time stretching.

L'obiettivo principale degli algoritmi di time stretching è di accorciare o allungare il file sonoro di M campioni in una nuova lunghezza desiderata $M' = \alpha \cdot M$, dove α è il fattore di

²Tali metodi sono descritti nelle sezioni successive del libro da cui è tratto il presente testo (Zölzer 2011). N. d. T.

scala. Per realizzare algoritmi di time stretching il file sonoro deve essere disponibile in un supporto di memorizzazione come ad esempio un campionario, un DAT o un Hard Disk. Il time stretching di una sequenza di campioni audio è mostrata in Fig. 4. Il segnale originale è mostrato nella figura in alto.

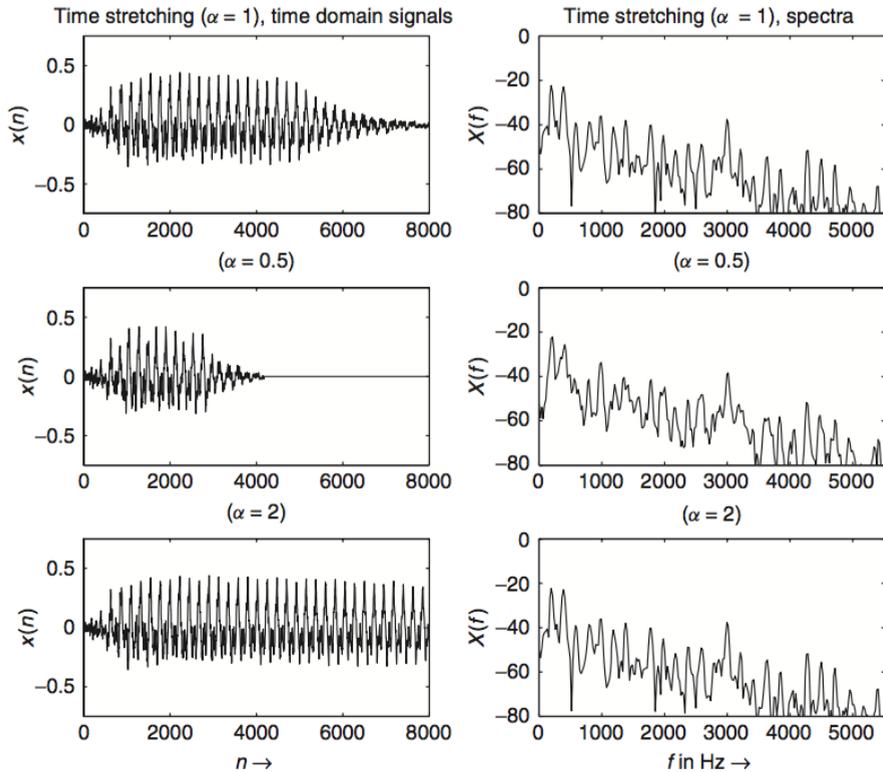


Figura 4: Time stretching con fattore $\alpha = 0.5, 2$.

Il grafico centrale mostra la sequenza ridotta di un fattore $\alpha = 0.5$ e il grafico più in basso mostra la sequenza secondo un fattore $\alpha = 2$.

Elaborazione del segnale

La contrazione temporale desiderata non corrisponde alla contrazione temporale matematica come è stata realizzata tramite la variazione di velocità. Piuttosto abbiamo bisogno di scalare gli attributi temporali percepiti, come ad esempio la velocità del parlato, senza perturbare gli attributi frequenziali percepiti, come ad esempio l'altezza. Si può dire che vogliamo che si percepisca la versione scalata temporalmente di un segnale acustico con la stessa sequenza degli eventi acustici come se il segnale originale fosse riprodotto secondo un modello temporale scalato. Gli algoritmi di time stretching non dovrebbero interferire con l'altezza o con il contenuto frequenziale dei segnali processati. Questo è dimostrato dagli spettri corrispondenti (primi 2000 campioni) del segnale a tempo discreto nella Fig. 4. Per confronto,

la tecnica tradizionale di time stretching basata sulla riproduzione a velocità variabile introduce un'alterazione dell'altezza (vedi Sezione 2 e Fig. 1). L'idea di base del time stretching tramite l'elaborazione di segmenti temporali è quella di dividere il suono in ingresso in segmenti. Quindi se poi il suono dovrà essere allungato, alcuni segmenti dovranno essere ripetuti; al contrario se il suono deve essere accorciato, alcuni segmenti dovranno essere scartati. Un possibile problema è la discontinuità dell'ampiezza e della fase agli estremi dei segmenti. Le discontinuità di ampiezza saranno evitate da sovrapposizioni parziali dei blocchi, quelle di fasi saranno evitate da un appropriato allineamento dei blocchi. Due strategie differenti saranno presentate nelle Sezioni 3.2 e 3.3.

Applicazioni

Macchine speciali come il *Phonogène universel* di Pierre Schaeffer o il *Tempophon* utilizzati da Herbert Eimert consentono di modificare la durata così come l'altezza dei suoni. Il *Phonogène* ha trovato molte applicazioni nella *musique concrète* come un "regolatore di tempo". Nella composizione *Épithap für Aikichi Kuboyama*, Eimert (1962), utilizza il *Tempophon* per iterare frammenti parlati. Il dispositivo ha permesso la scansione di sillabe, vocali e plosive, per renderle più brevi, più lunghe o iterarle a piacimento (Haller 1995, p. 13).

Come accennato nell'introduzione, l'allungamento dei segnali può essere utilizzato per abbinare la loro durata ad uno slot di tempo riservato. Nella musica Techno, diversi brani musicali vengono riprodotti uno dopo l'altro come un flusso continuo. Si suppone che questo flusso abbia solamente transizioni molto sfumate di tempo o bpm (battiti per minuto), anche se i brani musicali di solito non hanno lo stesso tempo. Per regolare i tempi tra loro, il disc jockey modifica la velocità di riproduzione al passaggio da un brano all'altro. Questo metodo porta a modificazioni di altezza temporanee che potrebbero essere discutibili. L'uso di metodi di time stretching potrebbe eliminare questo problema.

Dopo una breve presentazione della tecnologia del *Phonogène*, le seguenti sezioni discutono due tecniche di elaborazione del segnale che svolgono time stretching senza modificare l'altezza.

3.1 Metodi storici – *Phonogène*

Fairbanks, Everitt e Jaeger parlano già nel 1954 di un registratore a nastro modificato per compressione-espansione temporale o frequenziale del parlato (Laroche 1998; Lee 1972)³. Springer sviluppa una macchina simile (Springer 1955; Springer 1959) e Pierre Schaeffer vanta una macchina chiamata *Phonogène universel* che è stata progettata come una combinazione del *Phonogène chromatique* e del *Phonogène à coulisse* di cui abbiamo parlato precedentemente, con il tamburo a testina rotante di Springer (Moles 1961, pp. 71-76), (Schaeffer 1973, pp. 47-48), (Schaeffer e Reibel 1998, CD2, ID. 50-52), (Geslin 2000; Poullin e Sinclair 1957; Poullin 1954).

Il registratore modificato ha diverse testine di riproduzione montate su un tamburo a testina rotante. La velocità assoluta del nastro al capstan determina la durata del suono, mentre la velocità relativa delle testine rispetto a quella del nastro determina l'entità della trasposizione. Elettricamente, con la somma delle uscite delle diverse testine, si ottiene un suono continuo (Fig. 5).

³Si veda anche, in questo numero, Di Scipio, p. 30. N. d. T.

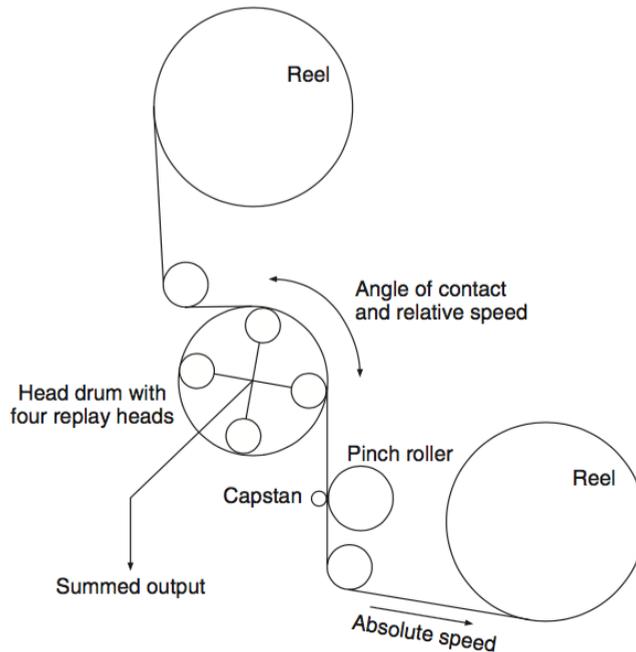


Figura 5: Sistema di compressione/espansione basato su magnetofono, vedi Moles (1961).

Moles riporta una tipica gamma operativa da +10% a -40% (Moles 1961, p. 76). La macchina Springer era anche conosciuta col nome di *Laufzregler* o *Zeitdehner* (Cary 1992, pp. 479-480), (Enders 1997).

3.2 Sovrapposizione e somma sincrona (SOLA) [Synchronous OverLap and Add]

Un semplice algoritmo per l'allungamento del tempo basato su tecniche di correlazione è proposto in Makhoul e El-Jaroudi (1986) e Roucos e Wilgus (1985). La Fig. 6 illustra l'algoritmo. Il segnale di ingresso $x(n)$ è diviso in blocchi sovrapposti di lunghezza fissa N (vedi Fig. 6a). S_a è la dimensione del passo di analisi che determina la progressione dei blocchi successivi lungo il segnale di ingresso. In un secondo momento questi blocchi sovrapposti vengono spostati in base al fattore di scalatura temporale α che portano alla dimensione del passo di sintesi $S_s = \alpha S_a$ (vedi Fig. 6b). Poi le somiglianze in un intervallo di lunghezza L all'interno della zona di sovrapposizione vengono cercate con uno scarto temporale discreto k_m di massima somiglianza. Lo spostamento di due blocchi successivi viene corretto da questo scarto temporale per sincronizzare i blocchi e ridurre gli artefatti (vedi Fig. 6c). Infine a questo punto di massima somiglianza i blocchi sovrapposti sono pesati da una funzione fade-in e fade-out e sommati campione per campione (vedi Fig. 6d).

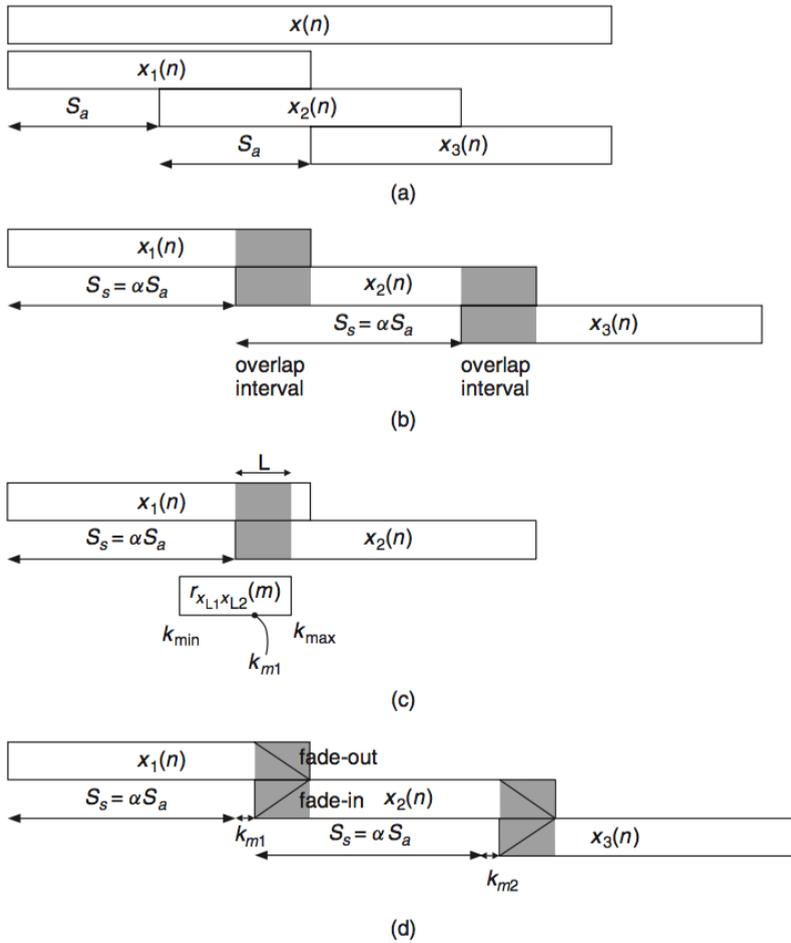


Figura 6: (a) segmentazione del segnale in ingresso, (b) riposizionamento dei blocchi, (c) calcolo della correlazione incrociata e sincronizzazione dei blocchi, (d) fading e sovrapposizione ed somma.

Descrizione dell'algoritmo:

1. Segmentazione del segnale di ingresso in blocchi di lunghezza N con uno slittamento temporale di S_a campioni.
2. Riposizionamento di blocchi con slittamento temporale $S_s = \alpha S_a$ con fattore di scala α .
3. Calcolo della correlazione incrociata

$$r_{x_{L1}x_{L2}} = \begin{cases} \frac{1}{L} \cdot \sum_{n=0}^{L-k-1} x_{L1}(n)x_{L2}(n+k), & 0 \leq k \leq L-1 \\ r_{x_{L2}x_{L1}}(-k), & -L+1 \leq k \leq 0 \end{cases} \quad (3)$$

tra $X_{L1}(n)$ e $X_{L2}(n)$, che sono i segmenti di $x_1(n)$ e $x_2(n)$ nell'intervallo di sovrapposizione (overlap interval) di lunghezza L .

4. Estrazione dello scarto tempo discreto k_m , dove la correlazione incrociata $r_{x_{L1}x_{L2}}(k_m) = r_{max}$ ha il suo valore massimo.
5. Fade-out $x_1(n)$ e fade-in $x_2(n)$. Sovrapporre e aggiungere di $x_1(n)$ e $x_2(n)$ per il nuovo segnale di uscita.

L'implementazione di SOLA porta alla scalatura temporale con poca complessità, dove i parametri S_a , N , L sono indipendenti dal periodo del tono del segnale di ingresso.

Ci sono un certo numero di approcci analoghi all'algoritmo SOLA. Il SOLAFS, come proposto in Hejna e Musicus (1991), realizza la modifica della scalatura temporale con un passo di sintesi fisso, mentre la dimensione del passo di analisi viene variata per ottenere la massima somiglianza nella zona di sovrapposizione tra l'uscita e la nuova finestra. Un approccio simile è proposto in Verhelst e Roelands (1993) sotto forma di un algoritmo di sovrapposizione e somma per la similarità della forma d'onda (WSOLA).

Il listato **M-file 1** realizza l'implementazione dell'algoritmo di scalatura temporale SOLA.

3.3 Sovrapposizione e somma sincronizzata con il periodo (PSOLA) [Pitch-Synchronous OverLap and Add]

Una variante dell'algoritmo SOLA per time stretching è la sovrapposizione e somma sincronizzata con il periodo (PSOLA) proposta da Hamon, Moulines e Charpentier (1989) e Moulines e Charpentier (1990) in particolare per l'elaborazione vocale. Si basa sull'ipotesi che il suono in ingresso sia caratterizzato da un'altezza, come, per esempio, la voce umana e gli strumenti musicali monofonici.

In questo caso PSOLA può sfruttare la conoscenza dell'altezza per sincronizzare correttamente i segmenti di tempo, evitando discontinuità d'intonazione. Quando eseguiamo il time stretching di un suono in ingresso, la variazione temporale del periodo di intonazione $P(t)$ deve essere variata di conseguenza. Se $\tilde{t} = \alpha t$ descrive la funzione di scalatura temporale o la funzione di time-warping che associa il tempo t del segnale di ingresso nel tempo \tilde{t} del segnale di uscita, il periodo locale del segnale di uscita $\tilde{P}(\tilde{t})$ sarà definito da $\tilde{P}(\tilde{t}) = \tilde{P}(\alpha t) = P(t)$.

M-file 1 (TimeScaleSOLA.m)

```

% TimeScaleSOLA.m
% Authors: U. Zölzer, G. De Poli, P. Dutilleux
% Time Scaling with Synchronized Overlap and Add
%
% Parameters:
%
% analysis hop size      Sa = 256 (default parameter)
% block length          N = 2048 (default parameter)
% time scaling factor    0.25 <= alpha <= 2
% overlap interval      L = 256*alpha/2

clear all,close all

[DAFx_in,Fs] = wavread('x1.wav');
DAFx_in = signal';

% Parameters:
Sa = 256; % Sa must be less than N
N = 2048;
alpha = 1; % 0.25 <= alpha <= 2
Ss = round(Sa*alpha);
L = 128; % L must be chosen to be less than N-Ss

% Segmentation into blocks of length N every Sa samples
% leads to M segments
M = ceil(length(DAFx_in)/Sa);

DAFx_in(M*Sa+N)=0;
Overlap = DAFx_in(1:N);

% **** Main TimeScaleSOLA loop ****
for ni=1:M-1
    grain=DAFx_in(ni*Sa+1:N+ni*Sa);
    XCORRsegment=xcorr(grain(1:L),Overlap(1,ni*Ss:ni*Ss+(L-1)));
    [xmax(ni),km(ni)]=max(XCORRsegment);

    fadeout=1:(-1/(length(Overlap)-(ni*Ss-(L-1)+km(ni)-1))):0;
    fadein=0:(1/(length(Overlap)-(ni*Ss-(L-1)+km(ni)-1))):1;
    Tail=Overlap(1,(ni*Ss-(L-1))+ ...
        km(ni)-1:length(Overlap)).*fadeout;
    Begin=grain(1:length(fadein)).*fadein;
    Add=Tail+Begin;
    Overlap=[Overlap(1,1:ni*Ss-L+km(ni)-1) ...
        Add grain(length(fadein)+1:N)];
end;
% **** end TimeScaleSOLA loop ****
% Output in WAV file
sound(Overlap,44100);
wavwrite(Overlap,Fs,'x1_time_stretch');

```

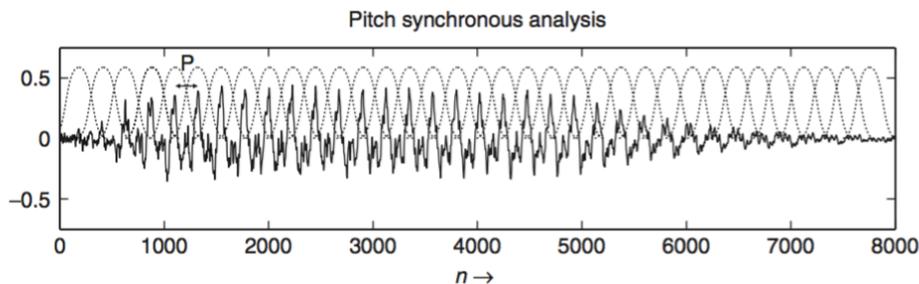


Figura 7: PSOLA: Analisi dell'altezza e finestatura.

Più in generale, quando il fattore di scala α non è costante, una funzione di scalatura temporale non lineare può essere definita come $\tilde{t} = \mathcal{T}(t) = \int_0^t \alpha(\tau) d\tau$ e usato al posto di $\tilde{t} = \alpha t$.

L'algoritmo si compone di due fasi: la prima fase analizza e segmenta il suono in ingresso (vedi Fig. 7), mentre la seconda fase sintetizza una versione time stretching mediante sovrapposizione e aggiungendo segmenti temporali estratti con l'algoritmo di analisi. Algoritmo di analisi (vedi Fig. 8):

1. Determinazione del periodo di intonazione $P(t)$ del segnale di ingresso e di istanti (riferimenti di intonazione) t_i . Questi riferimenti di intonazione (pitch marks) sono definiti in corrispondenza della massima ampiezza o di impulsi della glottide ad un tasso sincronizzato con l'altezza durante la parte periodica del suono e definiti ad un tasso costante durante le porzioni non vocalizzate. In pratica $P(t)$ è considerato costante $P(t) = P(t_i) = t_{i+1} - t_i$ sull'intervallo di tempo (t_i, t_{i+1}) .
2. Estrazione di un segmento centrata a ogni riferimento di intonazione t_i utilizzando una finestra Hanning con lunghezza $L_i = 2P(t_i)$ (due periodi di riferimento di intonazione) per assicurare fade-in e fade-out.

Algoritmo di sintesi (vedi Fig. 9): per ogni riferimento di intonazione \tilde{t}_k

1. Scelta del segmento di analisi corrispondente i (identificato dalla marca temporale t_i) minimizzando la distanza temporale.
2. Sovrapposizione e somma del segmento selezionato. Si noti che alcuni segmenti di ingresso saranno ripetuti per $\alpha > 1$ (espansione temporale) o scartati quando $\alpha < 1$ (compressione temporale).
3. Determinazione dell'istante di tempo \tilde{t}_{k+1} in cui sarà centrato il successivo segmento di sintesi, al fine di preservare l'altezza locale tramite la relazione

$$\tilde{t}_{k+1} = \tilde{t}_k + \tilde{P}(\tilde{t}_k) = \tilde{t}_k + P(t_i)$$

L'algoritmo di sintesi PSOLA di base può essere implementato in MATLAB® come nel listato **M-file 2**.

I fattori di allungamento tipicamente variano da $\alpha = 0.25$ a 2 per il parlato. Artefatti udibili appaiono in suoni non vocalizzati quando valori più grandi sono applicati, a causa della ripetizione regolare di segmenti di input identici. Per evitare che l'algoritmo introduca

M-file 2 (psola.m)

```

function out=psola(in,m,alpha,beta)
% Authors: G. De Poli, U. Zölzer, P. Dutilleux
%   in   input signal
%   m    pitch marks
%   alpha time stretching factor
%   beta pitch shifting factor

P = diff(m); %compute pitch periods

if m(1)<=P(1), %remove first pitch mark
    m=m(2:length(m));
    P=P(2:length(P));
end

if m(length(m))+P(length(P))>length(in) %remove last pitch mark
    m=m(1:length(m)-1);
    else
        P=[P P(length(P))];
end

Lout=ceil(length(in)*alpha);
out=zeros(1,Lout); %output signal

tk = P(1)+1; %output pitch mark

while round(tk)<Lout
    [minimum i] = min( abs(alpha*m - tk) ); %find analysis segment
    pit=P(i);
    gr = in(m(i)-pit:m(i)+pit) .* hanning(2*pit+1);
    iniGr=round(tk)-pit;
    endGr=round(tk)+pit;
    if endGr>Lout, break; end
    out(iniGr:endGr) = out(iniGr:endGr)+gr; %overlap new segment
    tk=tk+pit/beta;
end %while

```

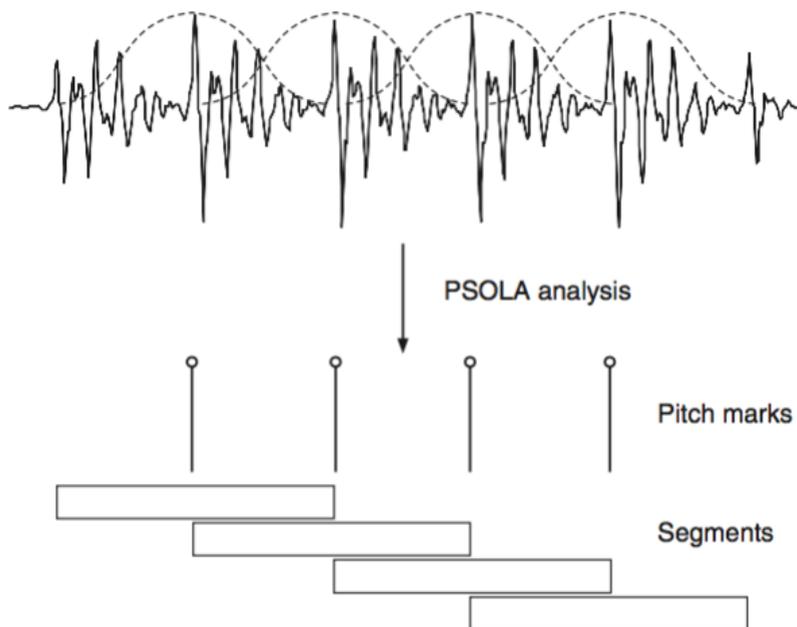


Figura 8: PSOLA Analisi dell'altezza.

tale correlazione artificiale a breve termine nel segnale di sintesi, è opportuno invertire l'asse temporale di ogni versione ripetuta di un segmento non vocalizzato. Con un tale artificio, il parlato può essere rallentato di un fattore quattro, anche se qualche effetto tonale si incontra in fricative sonore, che uniscono regioni di frequenze vocalizzate e non vocalizzate e che quindi non possono essere invertite nel tempo.

È possibile sfruttare ulteriormente la fase di analisi. Infatti, l'applicazione uniforme del time stretching può produrre qualche artefatto sulle parti non periodiche del suono. Ad esempio una consonante occlusiva può essere ripetuta se l'algoritmo di sintesi seleziona due volte il segmento temporale che contiene la consonante. L'analisi può essere quindi estesa al fine di rilevare la presenza di transizioni veloci (Knesebeck, Ziraksaz e Zölzer 2010; Peeters e Rodet 1999). Durante la sintesi, la scala temporale non verrà modificata in questi punti, quindi non si ripeteranno i segmenti. Questo approccio può essere generalizzato per i suoni non vocalizzati in cui un grande cambiamento nella scala temporale durante le transizioni (ad esempio, gli attacchi) cambierebbe radicalmente l'identità del timbro. Anche in questo caso è possibile limitare l'allungamento temporale durante le transizioni e applicarlo principalmente alla porzione stazionaria del suono in ingresso. Questa tecnica è di solito applicata agli strumenti musicali digitali basati sulla sintesi wavetable. D'altra parte, la deformazione delle parti transitorie può essere considerato una trasformazione timbrica interessante e può essere apprezzato come un effetto audio musicale creativo.

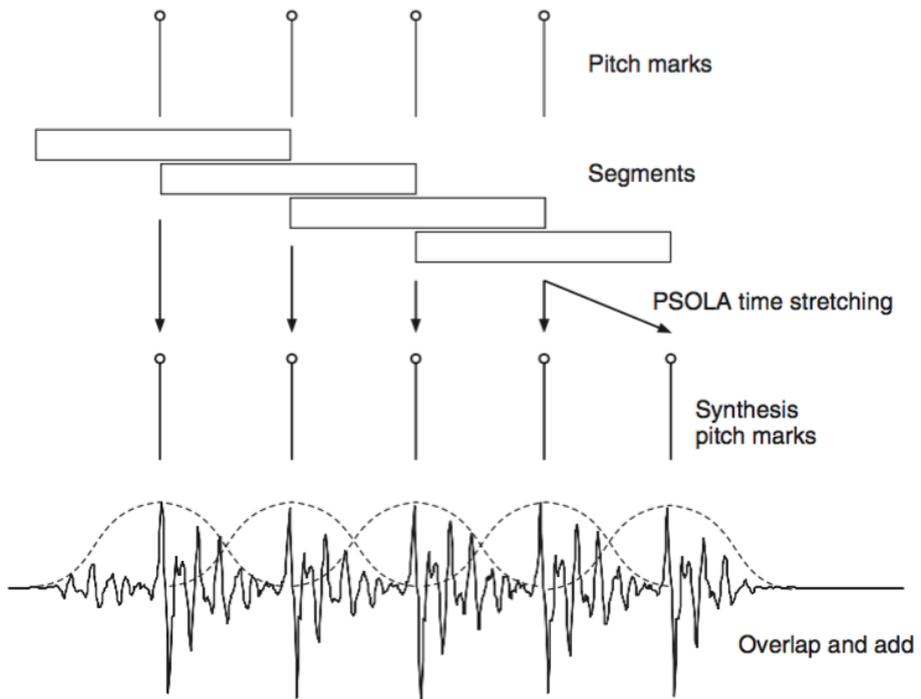


Figura 9: PSOLA Sintesi per allungamento temporale.

Contrasegni di intonazione [Pitch marks]

Va notato che la determinazione del campo e della posizione dei contrasegni di intonazione non è un problema banale e potrebbe essere difficile da implementare in maniera robusta⁴. Una volta che i periodi di intonazione sono noti, la posizione dei contrasegni di intonazione può essere determinata. La qualità del suono risultante dalla modifica mediante l’algoritmo PSOLA dipende essenzialmente dal posizionamento dei contrasegni di intonazione, poiché i contrasegni forniscono i centri delle finestre di segmentazione del PSOLA.

Ovviamente i contrasegni possono essere posizionati solo in corrispondenza al periodo di intonazione per le parti di segnali vocalizzate. Come accennato prima, un modo per gestire le parti non vocalizzate è quello di mantenere il “periodo di intonazione” ad un valore costante fino alla successiva parte vocalizzata. Un metodo che determina i contrasegni di intonazione di una porzione vocalizzata completa è proposto in Lin e Jang (2004) e Mattheyses, Verhelst e Verhoeve (2006). Il picco massimo globale della forma d’onda viene cercato ed il primo contrassegno di intonazione t_i si trova in questa posizione. Poi i contrasegni di intonazione a destra t_i sono posizionati utilizzando il periodo di intonazione come una prima stima delle posizioni dei contrasegni di intonazione, seguito da un affinamento della posizione ricercando il massimo entro una determinata regione. Il nuovo contrassegno di intonazione viene determinato come $t_{i+1} = \max([t_i + \delta P_0, t_i + (2 - \delta)P_0])$, dove P_0 è il periodo di intonazione e δ è un fattore compreso tra 0.5 e 0.9. Questa procedura viene ripetuta fino a raggiungere l’estremità destra della porzione. Poi i contrasegni di intonazione a sinistra di t_i sono posizionati rispettivamente procedendo con la ricerca del massimo nella regione $t_{i-1} = \max([t_i - \delta P_0, t_i - (2 - \delta)P_0])$, fino al raggiungimento dell’estremità sinistra della porzione. Questo metodo è fondamentalmente un approccio di tipo *peak-search* [basato sulla ricerca dei valori di picco].

Una funzione analoga per posizionare i contrasegni di intonazione può essere implementata in una maniera basata sull’analisi di finestre temporali [frame-based] con sovrapposizione di porzioni di finestra di lunghezza fissa. Con la conoscenza del periodo di intonazione del frame corrente e la posizione dell’ultimo contrassegno di intonazione nel frame precedente, il nuovo contrassegno di intonazione può essere prima stimato e successivamente raffinato. Questo rende possibile l’utilizzo in tempo reale dell’approccio *peak-picking* [basato sulla ricerca dei valori di picco in frequenza].

Per i segnali periodici stazionari, questo approccio produce buoni risultati. Ma nei casi in cui il segnale ha un ricco contenuto armonico, il picco massimo può saltare da un periodo all’altro. Questo porta ad un errato posizionamento del contrassegno di intonazione, cioè la distanza di due contrasegni di intonazione successivi non corrisponde al vero periodo di intonazione. Il risultato è la comparsa di artefatti nel segnale risintetizzato. Pertanto, i contrasegni di intonazione devono essere attentamente posizionati e i salti rapidi dei contrasegni di intonazione dovrebbero essere evitati. Poiché è desiderabile posizionare la finestra in modo tale che essa contenga una elevata quantità di energia del segnale, usare un approccio basato sull’individuazione del centro di energia (Knesebeck, Ziraksaz e Zölzer 2010) invece del *peak-picking*, permette un più corretto posizionamento del contrassegno di intonazione. Nei casi in cui il periodo del tono fondamentale all’interno di un frame, l’approccio al centro di energia consente di seguire la variazione dei contrasegni di intonazione in modo continuo, piuttosto che saltare al successivo picco massimo. Ci sono metodi per determinare le posizioni

⁴I periodi di intonazione possono essere determinati con uno degli algoritmi di estrazione presentati nel Capitolo 9.2 di Zölzer (2011). N. d. T.

dei contrassegni di intonazione utilizzando le informazioni spettrali. Un metodo basato sulla somma pesata di un gruppo di componenti frequenziali ritardate, che assicura anche che i marcatori siano posizionati vicino al massimo locale di energia, è descritto in Peeters e Rodet (1999).

Un algoritmo di base per il posizionamento dei contrassegni di intonazione è implementato in MATLAB[®] nel listato **M-file 3**. I contrassegni di intonazione sono posizionati in modo frame-based ad un tasso costante corrispondente al periodo fondamentale di intonazione del frame corrente. In caso di porzioni non sonore viene utilizzato il periodo di intonazione della porzione sonora precedente.

4 Pitch shifting [Trasposizione]

Introduzione

La trasposizione è uno degli strumenti di base dei musicisti. Quando pensiamo di fornire questo effetto mediante elaborazione del segnale, abbiamo bisogno di pensare ai vari aspetti di tale elaborazione. Per un musicista, trasporre significa ripetere una melodia dopo averla traslata di un intervallo fisso. Ogni volta che l'esecutore traspone la melodia, fa uso di un altro registro del suo strumento. In questo modo, non solo l'altezza del suono è modificata, ma anche il timbro viene influenzato.

Nel regno dei DAFX, è una questione di scelta trasporre senza tener conto della modifica del timbro o se il timbro caratteristico dello strumento deve essere mantenuto in ciascuno dei suoi registri. Il primo metodo potrebbe essere chiamato "trasposizione a timbro variabile", mentre il secondo approccio sarebbe chiamato "trasposizione a timbro costante". Per ottenere una panoramica del problema dobbiamo prendere in considerazione le origini fisiche del segnale audio.

Il timbro di un suono dipende fortemente dall'organizzazione del suo spettro. Un modello può essere derivato dallo studio della voce cantata. L'altezza di una voce cantata è determinata dalle corde vocali e può essere correlata con la serie di frequenze disponibili nello spettro. Il timbro della voce è determinato principalmente dal tratto vocale. L'effetto di quest'ultimo è quello di evidenziare alcune parti dello spettro, che sono chiamate formanti. Un modello del segnale può essere definito osservando dove una parte di eccitazione è modificata da una parte di risonanza. Nel caso della voce, l'eccitazione è fornita dalle corde vocali, quindi è legata alle frequenze dello spettro, mentre le risonanze corrispondono ai formanti. Quando un cantante traspone una melodia ha, in una certa misura, la possibilità di modificare altezza e formanti indipendentemente. In un'implementazione di elaborazione del segnale attenta a questo effetto, entrambi questi aspetti devono essere considerati.

Se solo lo spettro dell'eccitazione è espanso o contratto, si ottiene una trasposizione d'altezza in alto o in basso con un timbro costante. Se sono espanse o contratte solo le risonanze, l'altezza rimane la stessa, ma il timbro varia. Il canto armonico si basa su questo effetto. Se sia l'eccitazione che la risonanza vengono deliberatamente e in maniera indipendente alterate, allora entriamo nel campo degli effetti che possono essere percepiti come innaturali, ma che potrebbero avere un grande potenziale musicale.

La separazione di un suono nella sua parte di eccitazione e di risonanza è un processo complesso⁵. Presenteremo qui metodi che alterano contemporaneamente entrambi gli aspetti,

⁵Detto processo è affrontato nel Capitolo 8 di Zölzer (ibid.). N. d. T.

M-file 3 (findpitchmarks.m)

```

function [m] = findPitchMarks(x,Fs,F0,hop,frameLen)
% Author: A. von dem Knesebeck
% x      input signal
% Fs     sampling frequency
% F0     fundamental frequencies
% hop    hop size of F0 detection
% frameLen length of frame

% Initialization
m      = 0;      % vector of pitch mark positions
F0     = zeros (1,length(F0));
index  = 1;
local_m = [];   % local pitch marker position

% processing frames i
for i = 1:length(F0);
% set pitch periods of unvoiced frames
    if (i==1 && F0(i)==0); F0(i) = 120;      % 120Hz in case no preceding pitch
    elseif (F0(i)==0);    F0(i) = F0(i-1);
    end
    P0(i) = round(Fs/F0(i));                % fundamental period of frame i
    frameRange = (1:frameLen) + (i-1)*hop; % hopping frame
    last_m = index;                        % last found pitch mark

% beginning periods of 1st frame
    j = 1; %period number
    if i==1
        % define limits of searchFrame
        searchUpLim = 1 * P0(i);
        searchRange = (1 : searchUpLim);
        [pk,loc] = max(x(searchRange));
        local_m(j) = round(loc);

% beginning periods of 2nd - end frame
    else
        searchUpLim = searchUpLim + P0(i);
        local_m(j) = last_m + P0(i);
    end % beginning periods of 1st - end frame

% remaining periods of 1st - end frame
    index = local_m(1);
    j = 2; % grain/period number
    while( searchUpLim + P0(i) <= frameRange(end))
        % define range in which a marker is to be found
        searchUpLim = searchUpLim + P0(i);
        local_m(j) = local_m(j-1) + P0(i);
        index = local_m(j);
        j = j+1;
    end %while frame end not reached
    m = [m local_m];
end % processing frames i

% finishing calculated pitch marks
m = sort(m);
m = unique(m);
m = m(2:end);

```

come l'harmonizer o Pitch Shifting tramite la modulazione della linea di ritardo nella Sezione 4.3. Un metodo più raffinato basato su PSOLA, che consente la correzione dell'altezza con la conservazione delle formanti, verrà discusso nella Sezione 4.4⁶.

Applicazioni musicali

Tipiche applicazioni di pitch shifting nella musica pop sono la correzione dell'altezza degli strumenti o cantanti, così come la produzione di un effetto simile a un coro. Quando la voce di un cantante è mixata con copie di se stesso leggermente trasposte, appare un sottile effetto che dà l'impressione che si stia ascoltando un coro, invece di un solo cantante.

Anche l'armonizzatore può produrre effetti sorprendenti, come un uomo che parla con una piccola voce stridente o una donna con un grintoso tono basso. Possono essere prodotte sonorità estreme come il profondo suono del rullante in *Let's Dance* di David Bowie (White 1999). È stato anche utilizzato per la scomposizione e ricomposizione del parlato (Giordano, Rothman e Hollien 1973). In combinazione con una linea di ritardo con feedback del suono trasposto in ingresso, può essere prodotta una sorta di spirale in cui il suono viene sempre recepito più alto o più basso ad ogni iterazione.

Un effetto sottile, simile al phasing, può essere realizzato con una serie di harmonizers (Dutilleux 1988) accoppiati in parallelo e mixati al suono in ingresso, come mostrato nella Fig. 10. Il rapporto di trasposizione dell' n^{th} harmonizer potrebbe essere impostato su $1 + nr$, dove r è dell'ordine di $1/3000$. Se f_0 è l'altezza del suono, le uscite dell' n^{th} harmonizer

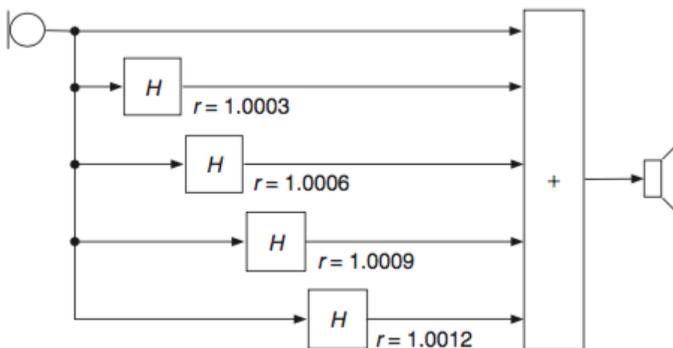


Figura 10: Un set di armonizzatori che produce un effetto tipo phasing. È particolarmente efficace per toni bassi (tipicamente 100 Hz) di lunga durata.

forniranno un'intonazione di $f_0 + n\Delta f$, dove $\Delta f = rf_0$. Se f è abbastanza piccolo (pochi centesimi di Hz) le interferenze tra le varie uscite degli harmonizer saranno chiaramente udibili. Quando viene applicato, ad esempio, ad un suono grave di tuba, sarà enfatizzato un armonico dopo l'altro. Effetti di flanging e di chorus possono essere ottenuti anche impostando il controllo di intonazione per una piccola quantità di trasposizione (ad esempio, $1/10$ a $1/5$ di un semitono) e l'aggiunta di rigenerazione (Anderton 1995, p. 53). Qui ci sembra che l'altezza di un effetto audio è molto dipendente dal suono elaborato. Succede spesso che l'altezza deve essere regolata per ogni nuovo suono o ogni nuova altezza.

⁶Per i metodi più avanzati di pitch-shifting si rimanda ai Capitoli 7-11 di Zölzer (2011). N. d. T.

Hans Peter Haller describe in *Das Experimental Studio der Heinrich-Strobel-Stiftung des Südwestfunks Freiburg 1971- 1989*, pp. 51-55, alcune applicazioni dell'armonizzatore per la produzione di opere musicali di Luigi Nono e André Richard.

4.1 Metodi Storici – *Harmonizer*

Le macchine basate su nastro descritte in 3.1 sono state anche in grado di modificare l'altezza dei suoni pur mantenendo la loro durata iniziale. Il *Phonogène universel* era ingombrante e non riusciva a trovare una vasta diffusione, ma nel bel mezzo degli anni settanta del Novecento, apparso un dispositivo digitale che fu chiamato *Harmonizer*. Questo implementava nel dominio digitale un processo simile a quello del *Phonogène universel*. Da qui l'effetto è diventato molto popolare. Poiché *Harmonizer* è un marchio della società Eventide, altre aziende offrono dispositivi simili con nomi come *pitch transposer* o *pitch shifter*.

La principale limitazione nell'uso dell'armonizzatore è la qualità caratteristica che dà ai suoni elaborati. Moles afferma che il raggio di azione del *Phonogène universel*, utilizzato come regolatore d'intonazione, funzionava da -4 a $+3$ semitoni (Moles 1961, p. 74). Geslin stima che le macchine disponibili alla fine degli anni sessanta hanno trovato applicazione nella musica concreta con rapporti di trasposizione molto più grandi (Geslin 2000).

Le implementazioni digitali in forma di *harmonizer* potrebbero consentire una migliore qualità, ma ci sono ancora importanti limitazioni. Per trasposizioni dell'ordine di un semitono, quasi nessuna alterazione dei suoni può essere udita. Appena il rapporto di trasposizione diventa più grande, in pratica nella gamma di più o meno due ottave, il timbro del suono in uscita ottiene un carattere che è specifico dell'armonizzatore.

Questa modifica può essere ascoltata sia nel dominio della frequenza sia nel dominio del tempo ed è dovuta alla modulazione del segnale da parte della finestra di analisi. Lo spettro del segnale di ingresso è infatti convoluto con quello della finestra. La modulazione nel dominio del tempo può essere caratterizzata dalla sua velocità e dallo spettro della finestra, che dipende dalla sua forma e dalle sue dimensioni. Più grande è la finestra, minore è il tasso e quindi la risoluzione dello spettro della finestra e meno disturba la modulazione. L'effetto di una finestra trapezoidale sarà più forte di quello di una più dolce, ad esempio la finestra coseno rialzato.

D'altra parte, una finestra più grande tende a fornire, attraverso il processo di sovrapposizione e somma, udibili copie iterate dei segnali di ingresso. Per la trasposizione di suoni percussivi è necessario ridurre la dimensione della finestra. Inoltre, per riprodurre in modo accurato transienti e non renderli troppo dolci, la finestra dovrebbe avere transizioni nette. Vediamo che un bilanciamento tra la modulazione udibile dello spettro sonoro e transitori iterati deve essere valutato per ogni tipo di suono. I musicisti che usano il computer come uno strumento musicale potrebbero sfruttare queste peculiarità dell'algoritmo per dare al loro *sound* un sapore unico.

4.2 Pitch Shifting mediante dilatazione temporale e ricampionamento

La riproduzione a velocità variabile discussa nella Sezione 2 porta ad una compressione o espansione della durata di un suono e ad un cambio di intonazione. Questo si ottiene attraverso il ricampionamento nel dominio del tempo. La Fig. 1 illustra i segnali a tempo discreto e gli spettri corrispondenti. Lo spettro del suono viene compresso o espanso nell'asse frequenziale. Le relazioni armoniche

$$f_i = i \cdot f_{\text{fondamentale}} \quad (4)$$

del suono non sono modificate, ma scalate secondo

$$f_i^{\text{new}} = \alpha \cdot f_i^{\text{old}} \quad (5)$$

Le ampiezze delle armoniche rimangono le stesse $a_i^{\text{new}} = a_i^{\text{old}}$. Al fine di riscalarlo il suono trasposto alla lunghezza originale, un ulteriore algoritmo di time-stretching può essere applicato al suono. Il risultato di trasposizione seguito da un algoritmo di time-stretching è illustrato nella Fig. 11.

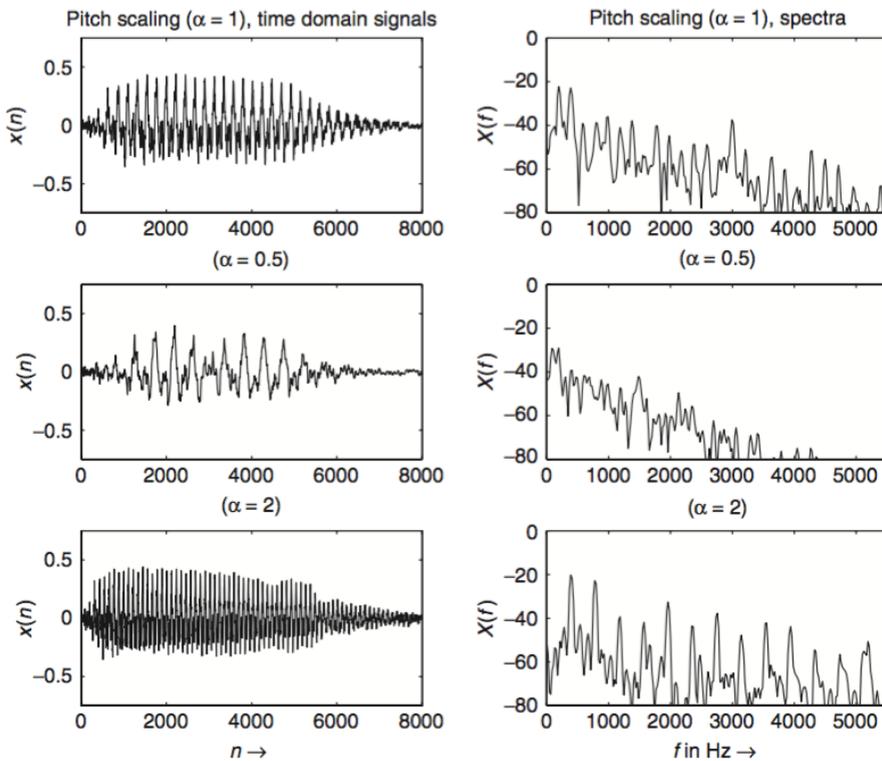


Figura 11: Pitch shifting seguito da correzione temporale.

L'ordine di cambio di intonazione e di scalatura temporale può essere modificato, come mostrato nella Fig. 12. Innanzitutto, un algoritmo di scalatura temporale espande il segnale di ingresso da lunghezza N_1 a lunghezza N_2 . Poi un'operazione di ricampionamento con rapporto inverso N_1/N_2 effettua il cambio di intonazione e una riduzione della lunghezza N_2 alla lunghezza N_1 . Il listato **M-file 4** realizza l'implementazione di scalatura temporale con algoritmo SOLA e algoritmo di cambio di intonazione.

M-file 4 (PitchScaleSOLA.m)

```

% PitchScaleSOLA.m
% Authors: G. De Poli, U. Zölzer, P. Dutilleux
% Parameters:
%   analysis hop size      Sa = 256 (default parmater)
%   block length          N = 2048 (default parameter)
%   pitch scaling factor  0.25 <= alpha <= 2
%   overlap interval      L = 256*alpha/2
clear all,close all
[signal,Fs] = wavread('x1.wav');
DAFx_in = signal';

Sa=256;N=2048;          % time scaling parameters
M=ceil(length(DAFx_in)/Sa);

n1=512;n2=256;        % pitch scaling n1/n2
Ss=round(Sa*n1/n2);
L=256*(n1/n2)/2;

DAFx_in(M*Sa+N)=0;
Overlap=DAFx_in(1:N);

% ***** Time Stretching with alpha=n2/n1*****
..... % include main loop TimeScaleSOLA.m
% ***** End Time Stretching *****

% ***** Pitch shifting with alpha=n1/n2 *****
lfen=2048;lfen2=lfen/2;
w1=hanningz(lfen);w2=w1;

% for linear interpolation of a grain of length lx to length lfen
lx=floor(lfen*n1/n2);
x=1+(0:lfen-1)'*lx/lfen;
ix=floor(x);ix1=ix+1;
dx=x-ix;dx1=1-dx;
%
lmax=max(lfen,lx);
Overlap=Overlap';
DAFx_out=zeros(length(DAFx_in),1);

pin=0;pout=0;
pend=length(Overlap)-lmax;
% Pitch shifting by resampling a grain of length lx to length lfen
while pin<pend
    grain2=(Overlap(pin+ix).*dx1+Overlap(pin+ix1).*dx).* w1;
    DAFx_out(pout+1:pout+lfen)=DAFx_out(pout+1:pout+lfen)+grain2;
    pin=pin+n1;pout=pout+n2;
end;

```



Figura 12: Pitch shifting mediante scalatura temporale e ricampionamento.

4.3 Cambio di altezza mediante modulazione della linea di ritardo

Il cambio di altezza o trasposizione basata su elaborazione di segmenti è descritto in diverse pubblicazioni. In Bogdanowicz e Blecher (1989) è proposto un cambio di altezza basato su uno schema di sovrapposizione e somma con due linee di ritardo variabili nel tempo (vedi Fig. 13). Un blocco cross-fade combina le uscite di due linee di ritardo mediante una funzione di dissolvenza incrociata.

Il segnale viene diviso in piccoli pezzi. I pezzi vengono letti più velocemente per produrre intonazioni più alte o più lentamente per produrre intonazioni più basse. Al fine di produrre un segnale continuo di uscita, due pezzi vengono letti simultaneamente con un ritardo pari alla metà della lunghezza del blocco. È quindi costruita una dissolvenza incrociata da un pezzo all'altro fino a ciascuna estremità di ciascun pezzo (Warstat e Görne 1994, pp. 257-259).

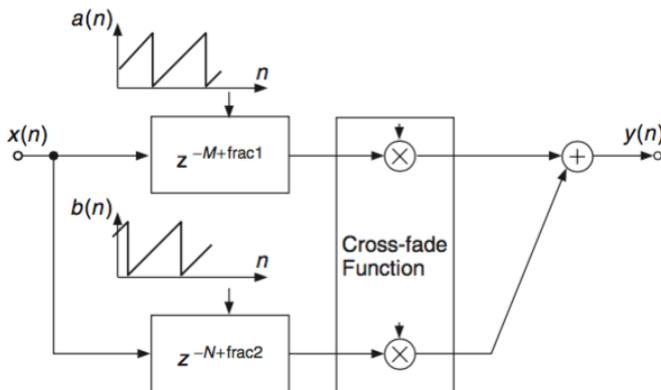


Figura 13: Pitch shifting.

La lunghezza delle linee di ritardo viene modulata da una funzione a dente di sega. Un approccio simile è proposto in Dattoro (1987), in cui viene utilizzata la stessa configurazione per compressione del tempo ed espansione. Un algoritmo di rilevamento della periodicità viene utilizzato per calcolare la funzione di dissolvenza incrociata per evitare cancellazioni durante le dissolvenze.

Un metodo avanzato per la trasposizione dei segnali audio è presentato in Disch e Zölzer (1999). Il metodo si basa su uno schema di sovrapposizione e somma e non necessita di alcuna stima della frequenza fondamentale. La differenza da altre applicazioni è il modo in cui i blocchi sono modulati e combinati per il segnale di uscita. Il sistema di trasposizione potenziato

si basa su uno schema di sovrapposizione e somma con tre linee di ritardo tempo-varianti parallele (vedi Fig. 15).

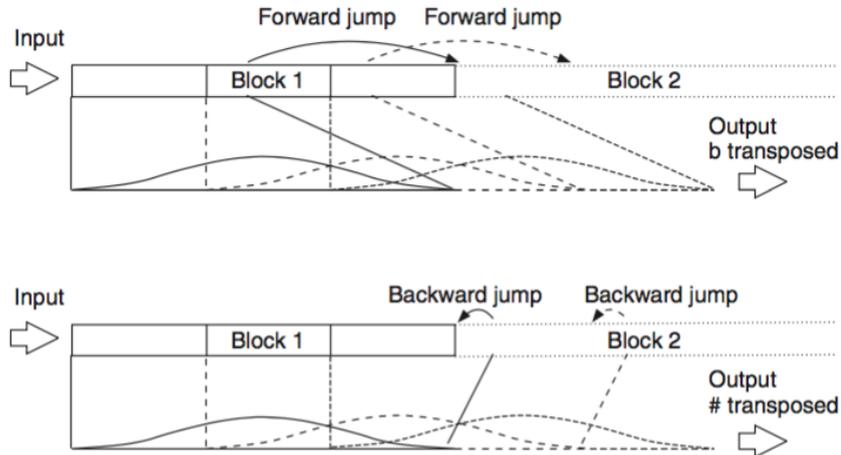


Figura 14: Traspositore avanzato: elaborazione dei blocchi, dilatazione temporale, sovrapposizione ed somma.

La Fig. 14 illustra come il segnale di ingresso è diviso in blocchi, che sono ricampionati (modulazione di fase con un segnale a rampa), modulati in ampiezza e sommati ottenendo un segnale di uscita della stessa lunghezza del segnale di ingresso. Blocchi adiacenti si sovrappongono con $2/3$ della lunghezza del blocco.

I segnali di modulazione formano un sistema di tre funzioni coseno rialzato sfasati di 120° . La somma di queste funzioni è costante per tutti gli argomenti. La Fig. 15 mostra anche la topologia del traspositore di altezza. Dal momento che un coseno completo viene utilizzato per la modulazione, la qualità sonora percepita del segnale elaborato è molto migliore di quella prodotta utilizzando quelle applicazioni con una semplice duplice sovrapposizione e somma che utilizzano diverse finestre. La modulazione di ampiezza produce solo frequenze somma e differenza con la frequenza base del segnale di modulazione, che può essere molto bassa (6-10 Hz). Le armoniche non sono presenti nel segnale di modulazione e quindi non possono generare frequenze somma o differenza di ordine superiore. Gli artefatti percepiti sono simili all'effetto di phasing sono meno fastidiosi rispetto alle discontinuità locali di altre applicazioni basate su metodi di sovrapposizione e somma duplice.

Se vogliamo cambiare l'altezza di un segnale controllato da un altro segnale o da un involuppo di segnale possiamo anche fare uso della modulazione mediante linea di ritardo. L'effetto può essere ottenuto eseguendo una modulazione di fase del segnale registrato secondo $y(n) = x(n - D(n))$. Il fattore modulante $D(n) = M + PROFONDIT \cdot X_{mod}(n)$ è ora dipendente da un segnale modulante $X_{mod}(n)$. Con questo approccio l'altezza del segnale di ingresso $x(n)$ viene cambiata secondo l'involuppo del segnale modulante (vedi Fig. 16).

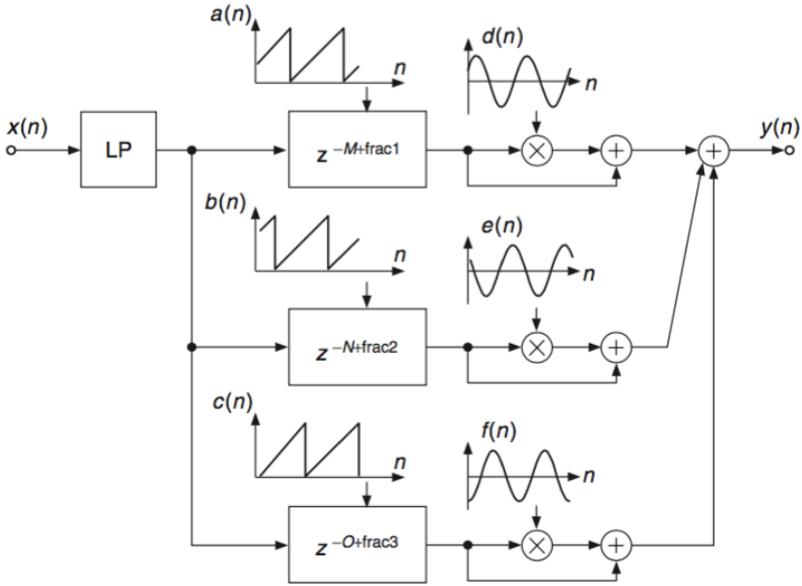


Figura 15: Traspositore avanzato: diagramma a blocchi.

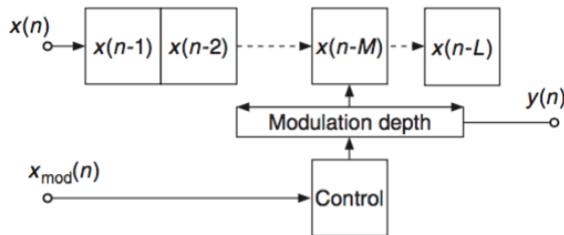


Figura 16: Altezza controllata dall'involuppo del segnale $x_{\text{mod}}(n)$.

4.4 Cambio di intonazione mediante PSOLA e conservazione delle formanti

Questa tecnica è l'operazione duale del ricampionamento nel dominio del tempo ma in questo caso è eseguito un ricampionamento dell'involuppo spettrale ottenuto dall'analisi di Fourier a tempo breve. L'involuppo spettrale a tempo breve descrive una curva in frequenza passando attraverso tutte le ampiezze delle armoniche. Questo è dimostrato nella Fig. 17, dove si vede l'involuppo spettrale. Le armoniche sono ancora scalate in base alla nuova $f_i^{new} = \beta \cdot f_i^{old}$, ma le ampiezze delle armoniche $a_i^{new} = env(f_i^{new}) \neq a_i^{old}$ sono ora determinate campionando l'involuppo spettrale. Si può notare qualche differenza di ampiezza rispetto all'involuppo originale. Questo dipende dalla scelta dell'algoritmo di cambio di altezza.

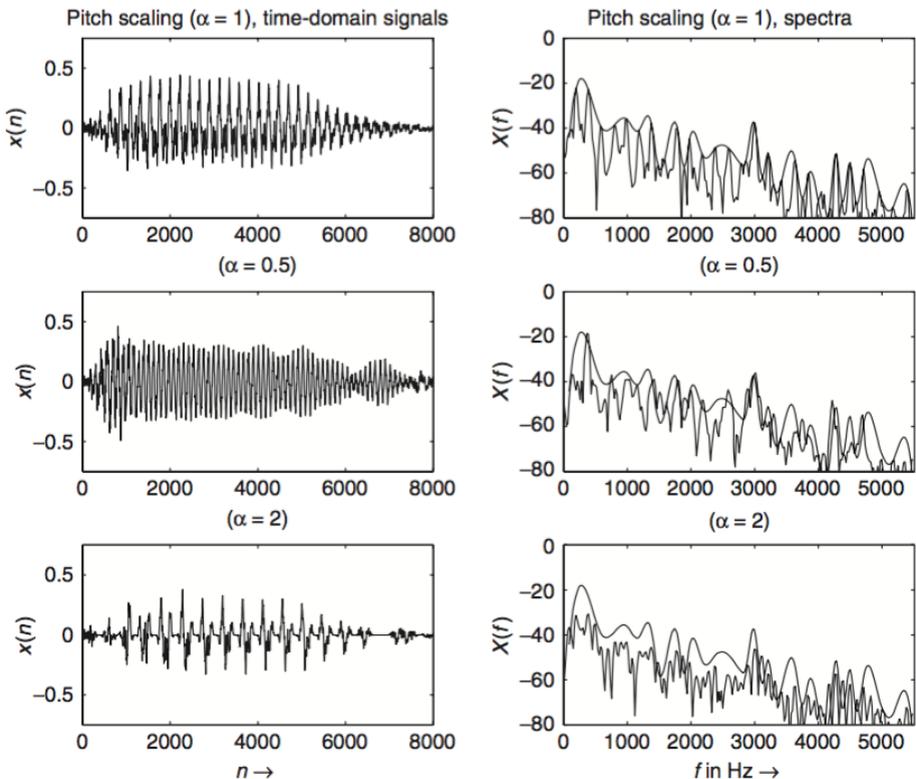


Figura 17: Pitch shifting con il metodo PSOLA: ricampionamento delle frequenze dell'involuppo spettrale.

L'algoritmo PSOLA può essere convenientemente utilizzato per il cambio di intonazione di un suono vocale mantenendo la posizione delle formanti, e così l'identità vocale (Bristow-Johnson 1995; Moulines e Laroche 1995). L'idea di base consiste nello stiramento temporale della posizione dei marcatori di intonazione, mentre il segmento della forma d'onda non viene modificato. Il modello del segnale sottostante la produzione del parlato è un treno di impulsi

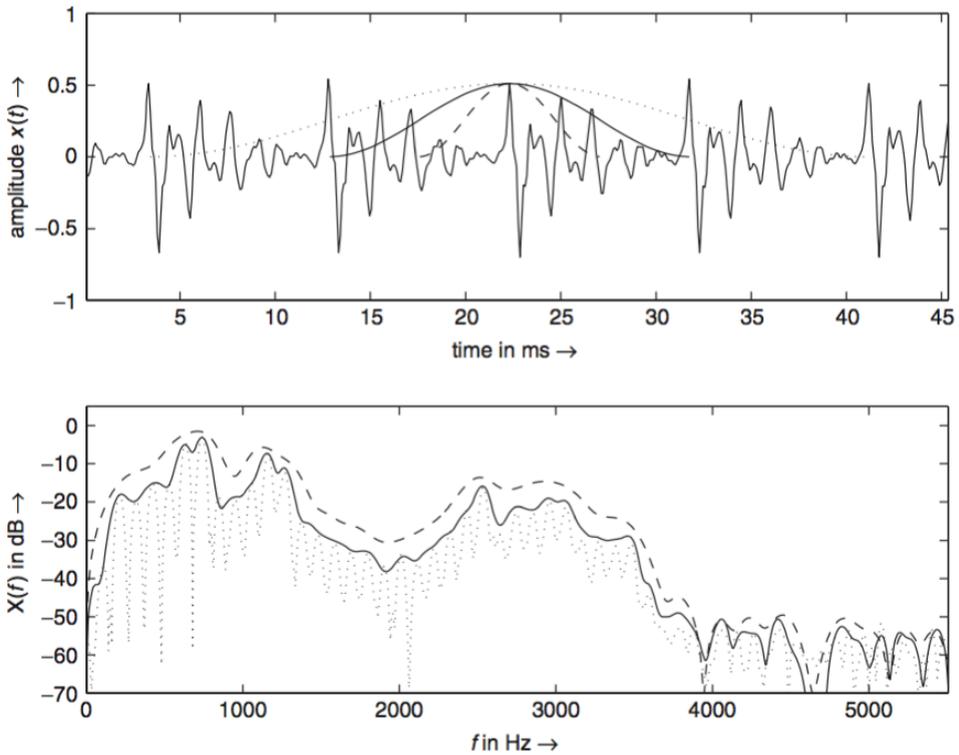


Figura 18: Spettro di segmenti estratti da una vocale /a/ usando una finestra di Hanning lunga rispettivamente 4 (linea punteggiata), 2 (linea continua) e 1 (linea tratteggiata) periodi di intonazione. Si può notare che la linea continua approssima l'involucro spettrale locale.

filtrato attraverso un filtro variabile nel tempo corrispondente al tratto vocale. Il segmento di ingresso corrisponde alla risposta impulsiva del filtro e determina la posizione delle formanti. Pertanto, tale risposta, non dovrebbe essere modificata. Al contrario, la distanza tra i marcatori di intonazione determina il periodo del parlato e, quindi, tale distanza deve essere modificata di conseguenza. Lo scopo dell'analisi PSOLA è quello di estrarre la risposta impulsiva locale del filtro. Come si può vedere nella Fig. 18, lo spettro di un segmento estratto utilizzando una finestra di Hanning con una lunghezza di due periodi approssima l'involucro spettrale locale. Finestre più lunghe tendono ad evidenziare la struttura fine delle righe dello spettro, mentre finestre più corte tendono ad offuscare la struttura delle formanti dello spettro. Così se non estendiamo il segmento, la posizione delle formanti viene mantenuta. L'operazione di sovrapposizione dei segmenti alla posizione del nuovo marcatore di intonazione ricamperà l'involucro spettrale alla frequenza di intonazione desiderata. Quando desideriamo un cambio di intonazione di un fattore β , definito come il rapporto tra la frequenza del tono di sintesi locale rispetto a quello originale $\beta = \tilde{f}_0(\tilde{t})/f_0(t)$, il nuovo periodo di intonazione sarà dato da $\tilde{P}(\tilde{t}) = P(t)/\beta$, dove in questo caso $\tilde{t} = t$ perché il tempo non è allungato.

L'algoritmo di analisi è lo stesso di quello visto in precedenza per allungamento temporale PSOLA in Sezione 3.3 (vedi Fig. 8). L'algoritmo di sintesi viene modificato (vedi Fig. 19) secondo le seguenti fasi:

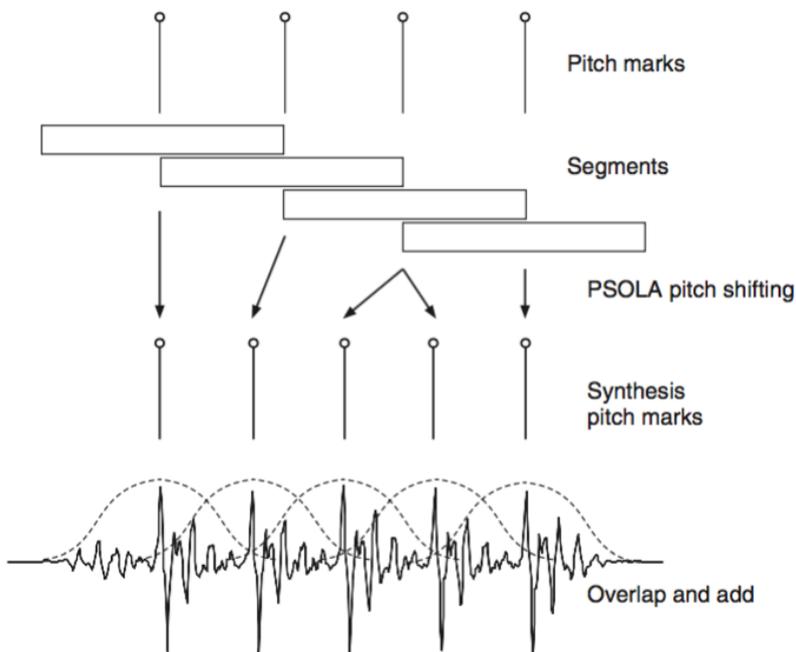


Figura 19: PSOLA: algoritmo di sintesi per cambio di altezza.

- Per ogni marcatore di intonazione nella sintesi \tilde{t}_k :

1. Scelta del corrispondente segmento di analisi i (identificato dalla marca temporale t_i) minimizzando la distanza temporale $|t_i - \tilde{t}_k|$.
2. Sovrapporre e aggiungere il segmento selezionato. Si noti che alcuni segmenti di ingresso saranno ripetuti per $\beta > 1$ (altezza più alta) o scartati quando $\beta < 1$ (altezza più bassa).
3. Determinazione dell'istante di tempo \tilde{t}_{k+1} dove il prossimo segmento di sintesi sarà centrato, in modo da preservare l'altezza locale, con la relazione

$$\tilde{t}_{k+1} = \tilde{t}_k + \tilde{P}(\tilde{t}_k) = \tilde{t}_k + P(t_i/\beta)$$

- Per grandi cambiamenti di altezza è consigliabile, per compensare la variazione di ampiezza introdotta dalla maggiore o minore sovrapposizione di segmenti, moltiplicare il segnale di uscita per $1/\beta$.

È possibile combinare l'allungamento temporale di un fattore α con il cambio di altezza. In questo caso, per ogni marcatore di intonazione sintetizzato \tilde{t}_k il primo passo dell'algoritmo di sintesi presentato sopra sarà modificato dalla scelta del segmento corrispondente di analisi i (identificato dalla marcatura temporale t_i), minimizzando la distanza temporale $|\alpha t_i - \tilde{t}_k|$.

L'algoritmo PSOLA è molto efficace per la elaborazione della voce ed è computazionalmente molto efficiente una volta che il suono sia stato analizzato. Quindi è ampiamente usato per la sintesi vocale da un database di difoni, per la modifica della prosodia, per macchine automatiche telefoniche, ecc. Per un'ampia variazione dell'altezza presenta alcuni artefatti. D'altra parte, la necessità di una fase preliminare di analisi per ottenere un profilo di intonazione rende difficoltosa l'implementazione in tempo reale di una modifica del segnale di ingresso. Anche la stima di impulsi della glottide può essere difficile. Una soluzione consiste nel posizionare i marcatori di intonazione ad un tasso sincrono con il periodo, indipendentemente dalla posizione reale degli impulsi glottali. La qualità di sintesi risultante sarà solo leggermente diminuita (si veda, ad esempio, la Fig. 20).

Un ulteriore effetto che può essere ottenuto da una variazione di PSOLA è una scalatura lineare delle frequenze formanti (vedi Fig. 21). Infatti, abbiamo visto che la scala temporale di un segnale corrisponde ad una scala di frequenza inversa. Così quando eseguiamo una scalatura temporale della risposta all'impulso di un filtro, si scala inversamente la frequenza delle formanti. In termini PSOLA, questo corrisponde alla scalatura temporale dei segmenti di ingresso selezionati prima del passo di sintesi di sovrapposizione e somma, senza alcuna modifica nel calcolo dei marcatori di intonazione. Per aumentare le frequenze dei formanti di un fattore γ , ogni segmento dovrebbe essere ridotto di un fattore $1/\gamma$ attraverso il ricampionamento. Ad esempio, le frequenze medie delle formanti di femmine adulte sono circa il 16% superiori a quelle dei maschi adulti, e le formanti dei bambini sono circa il 20% superiori rispetto alle formanti femminili. Si deve fare attenzione quando le frequenze aumentano in modo da evitare foldover. Idealmente dovrebbe essere usato un campionamento a banda limitata.

Il cambio di altezza mediante PSOLA può essere utilizzato per sintetizzare più voci da un vero cantante per creare un effetto di *coro virtuale* (Schnell et al. 2002).

Il listato **M-file 5** implementa dell'algoritmo base di sintesi PSOLA. Si basa sull'algoritmo di stiramento temporale PSOLA mostrato nella Sezione 3.3.

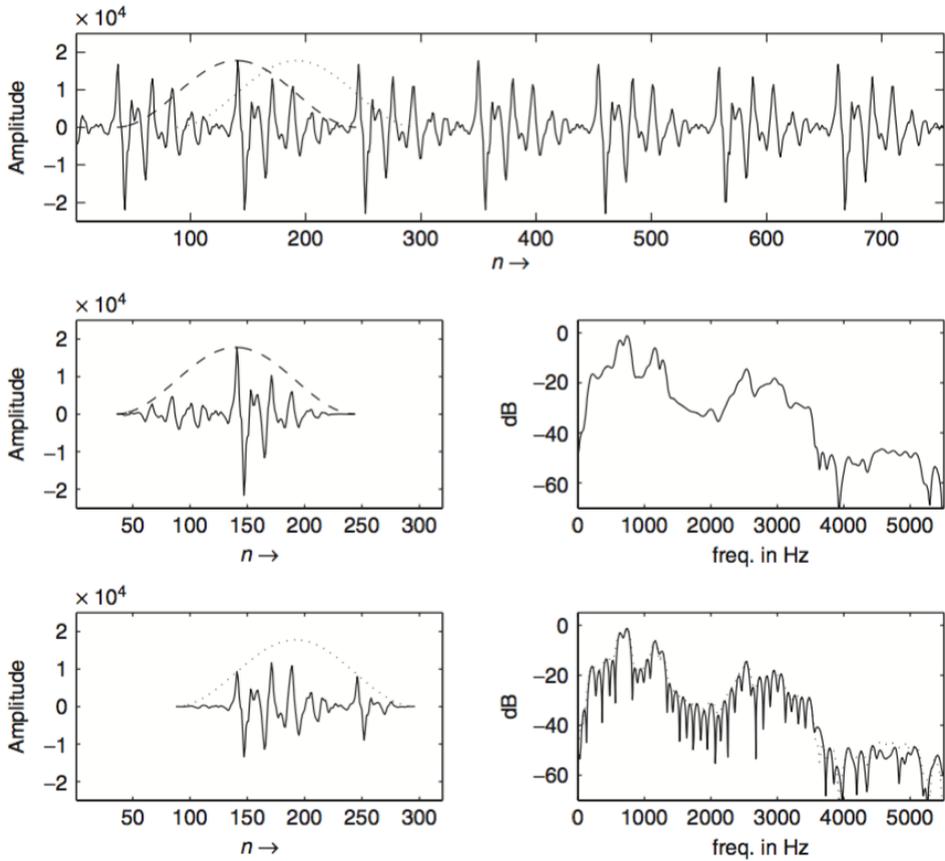


Figura 20: Confronto di un segmento estratto in corrispondenza di un impulso glottale con uno estratto tra impulsi glottali

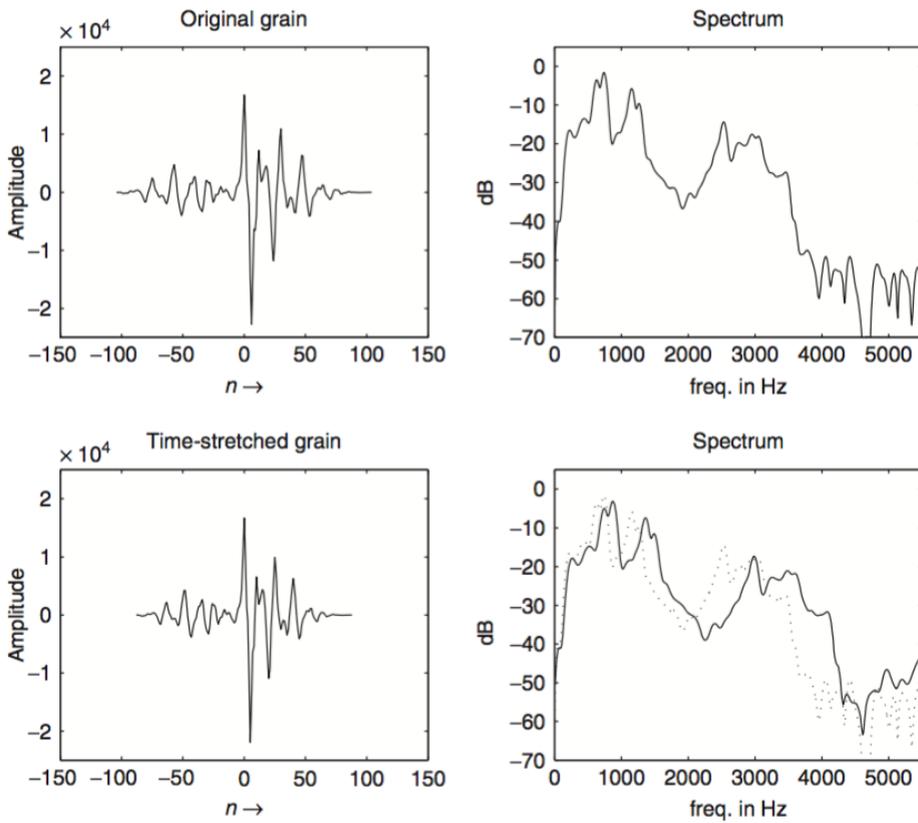


Figura 21: PSOLA: variazione di PSOLA per scalatura lineare delle formanti.

M-file 5 (psolaf.m)

```

function out=psolaF(in,m,alpha,beta,gamma)
% Authors: G. De Poli, U. Zölzer, P. Dutilleux
%
%   . . .
%   gamma newFormantFreq/oldFormantFreq
%   . . .
%   the internal loop as
tk = P(1)+1;           %output pitch mark
while round(tk)<Lout
  [minimum i]=min(abs(alpha*m-tk) );    % find analysis segment
  pit=P(i);pitStr=floor(pit/gamma);
  gr=in(m(i)-pit:m(i)+pit).*hanning(2*pit+1);
  gr=interp1(-pit:1:pit,gr,-pitStr*gamma:gamma:pit);% stretch segm.
  iniGr=round(tk)-pitStr;endGr=round(tk)+pitStr;
  if endGr>Lout, break; end
  out(iniGr:endGr)=out(iniGr:endGr)+gr; % overlap new segment
  tk=tk+pit/beta;
end % end of while

```

5 Ricombinazione temporale e granulazione

5.1 Ricombinazione temporale

Introduzione

La *musique concrète* ha fatto un uso intensivo di ricombinazione di piccoli elementi di nastro magnetico. Quando ben controllato, questo assemblaggio di centinaia di frammenti di diverse decine di millisecondi permette una fusione di materiali sonori eterogenei, al limite della soglia di discriminazione temporale. Questa operazione manuale, chiamata micro-montaggio, è molto laboriosa. Bernard Parmegiani suggerì nel 1980 al Groupe de Recherches Musicales (GRM) che questa operazione può essere svolta dal computer. Una prima versione del software è stata prodotta nei primi anni ottanta. Dopo essere riscritta, migliorata e ricompilata più volte, alla fine si è reso disponibile su personal computer sotto forma di un programma chiamato *brassage* in francese che verrà tradotto qui come *ricombinazione temporale* (Geslin 1998, 2000).

Elaborazione del segnale

Descriviamo qui un algoritmo elementare per la ricombinazione temporale che si basa sulla sovrapposizione di due segmenti di tempo che vengono raccolti in modo casuale dal segnale di ingresso (vedi Fig. 22):

1. Siano $x(n)$ e $y(n)$ i segnali di ingresso e di uscita.
2. Si assegni la durata d dei frammenti e la durata $D \geq d$ del periodo di tempo $[n - D, n]$ da cui verranno selezionati i segmenti di tempo.
3. Memorizzare il segnale in ingresso $x(n)$ in una linea di ritardo di lunghezza D .

4. Scegliere a caso il tempo di ritardo τ_1 con $d \leq \tau_1 \leq D$.
5. Selezionare il segmento del segnale x_{1d} di durata d che inizia a $x(n - \tau_1)$.
6. Seguire la stessa procedura (punti 4 e 5) per un secondo segmento x_{2d} .
7. Leggere x_{1d} e x_{2d} e applicare un involuppo di ampiezza W per ciascuno di loro, al fine di appianare le discontinuità ai bordi.
8. Quando la lettura di x_{1d} e x_{2d} è terminata, iterare la procedura per ciascuno di essi.
9. Calcolare l'uscita come sovrapposizione e somma delle sequenze x_{1d} e x_{2d} con un ritardo temporale di $d/2$.

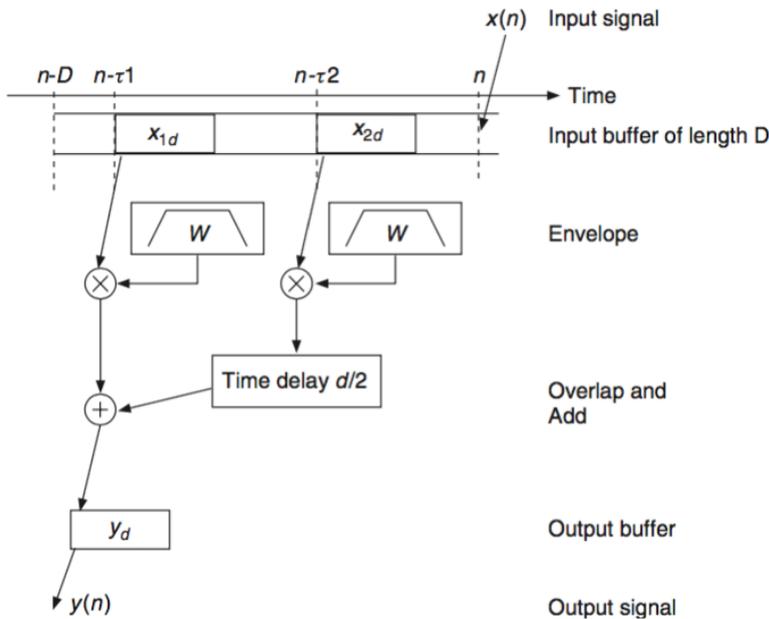


Figura 22: Ricombinazione temporale: due segmenti di ingresso, selezionati casualmente dal segnale in ingresso trascorso, sono sovrapposti e aggiunti per produrre un segmento temporale di uscita. Quando un segmento di ingresso è terminato, viene selezionato uno nuovo.

Applicazioni musicali e controllo

La versione sopra descritta introduce disturbi locali nella evoluzione temporale del segnale, preservando la continuità globale della sua sequenza temporale. Molti ulteriori perfezionamenti di questo algoritmo sono però possibili. Un coefficiente di ampiezza casuale potrebbe essere applicato a ciascuno dei segmenti di ingresso per modificare la densità del materiale sonoro. La forma dell'involuppo potrebbe essere modificata al fine di mantenere la struttura temporale

d'ingresso o, dall'altra parte, per smussare e fondere diversi eventi tra loro. La velocità di riproduzione dei segmenti potrebbe essere variata per produrre trasposizione o glissando.

In un'epoca in cui gli strumenti informatici non erano ancora disponibili, Bernard Parmegiani ha magnificamente illustrato la tecnica di micro-tessitura basata su nastro in opere come *Violostries* (1964) o *Dedans-Dehors* (1977) (Parmegiani 1964, 1977). L'algoritmo elementare presentato sopra può essere utilizzato in tempo reale, ma sono state implementate altre versioni che offrono numerose caratteristiche ulteriori. Esse hanno la possibilità di unire i frammenti, campionati casualmente e di qualsiasi dimensione, da alcuni campioni a diversi minuti. Così, oltre a generare fenomeni di fusione, per cui l'algoritmo è stato concepito, il software è stato in grado di produrre dissolvenza incrociata di suoni tessiturali ed anche accordi sostenuti, con infinitamente piccole variazioni nella stabilità del segnale, e nell'interpolazione di frammenti con silenzio o suoni di altri tipi (Geslin 1998). Jean-Claude Risset ha usato questo effetto per eseguire evoluzioni sonore di suoni brevi, come pietre e campanelli di metallo (Risset 1987, *Sud-I*, 3'44"-4'38"); (Geslin 2000; Risset 1998) e per la produzione di un pianoforte "balzubiente", ulteriormente elaborato da modulazione ad anello (Risset 1987, *Sud-I*, 4'30", 5'45"). A partire da "oggetti trovati" come il canto degli uccelli, Jean-Claude Risset li ha riorganizzati in modo compositivo per ottenere prima un rendering puntilistico, poi un episodio simil-stretto (ibid., 1'42"-2'49").

5.2 Granulazione

Introduzione

Nelle sezioni precedenti sulla modulazione di altezza e sulla dilatazione temporale, abbiamo proposto algoritmi che presentano limitazioni per quanto riguarda il loro scopo iniziale. Al di là di una gamma limitata di modifica di altezza o di durata, appaiono evidenti artefatti. Il metodo della ricombinazione temporale considera questi artefatti da un punto di vista artistico e li dà per scontati. Tra le possibilità offerte dai metodi e dai loro limiti, questi propongono di creare nuove strutture sonore. Considerando che l'effetto di ricombinazione temporale sfrutta le possibilità di uno specifico sistema software, che potrebbe essere considerato qui come "strumento musicale", l'idea di costruire un suono complesso da un grande insieme di suoni elementari potrebbe essere un orizzonte più ampio.

Il fisico Dennis Gabor ha proposto nel 1947 l'idea del *quantum* sonoro, un'unità indivisibile di informazione dal punto di vista della psicoacustica. Secondo la sua teoria, una rappresentazione granulare potrebbe descrivere qualsiasi suono. La sintesi granulare è stata suggerita come una tecnica della computer music per la produzione di suoni complessi da Iannis Xenakis (1971) e Curtis Roads (1978). Questa tecnica accumula eventi acustici provenienti da migliaia di grani sonori. Un grano sonoro è molto breve (tipicamente da 1 a 100 *ms*) e si avvicina alla durata, alla frequenza e all'ampiezza dell'evento minimo percepibile (Roads 1996, 1998; Truax 2000a).

L'effetto di granulazione è un'applicazione di sintesi granulare in cui il materiale di cui sono costituiti i grani è un segnale di ingresso. Barry Truax per primo ha sviluppato questa tecnica (Truax 1988a,b) implementandola in tempo reale e utilizzandola ampiamente nelle sue composizioni.

Elaborazione del segnale

Siano $x(n)$ e $y(n)$ i segnali di ingresso e di uscita. I grani $g_k(i)$ sono estratti dal segnale di ingresso con l'aiuto di una funzione finestra $w_k(i)$ di lunghezza L_k tramite

$$g_k(i) = x(i + i_k)w_k(i) \quad (6)$$

con $i = 0, \dots, L_k - 1$. L'istante di tempo i_k indica il punto in cui il segmento viene estratto; la lunghezza L_k determina la quantità di segnale estratto; la finestra $w_k(i)$ dovrebbe garantire fade-in e fade-out ai bordi del grano e accentuare il contenuto in frequenza del grano. Grani lunghi tendono a mantenere l'identità timbrica della porzione del segnale di ingresso, mentre quelli corti acquisiscono una qualità simile all'impulso. Quando il grano è lungo, la finestra ha una cima piatta e viene utilizzata solo per fade-in e fade-out ai bordi del segmento.

I listati **M-file 6** e **7** mostrano l'estrazione di grani corti e lunghi.

M-file 6 (grainSh.m)

```
function y = grainSh(x,init,L)
% Authors: G. De Poli
% extract a short grain
% x    input signal
% init first sample
% L    grain length (in samples)
y=x(init:init+L-1).*hanning(L)';
```

M-file 7 (grainLn.m)

```
function y = grainLn(x,iniz,L,Lw)
% Authors: G. De Poli
% extract a long grain
% x    input signal
% init first sample
% L    grain length (in samples)
% Lw   length fade-in and fade-out (in samples)
if length(x) <= iniz+L , error('length(x) too short. '), end
y = x(iniz:iniz+L-1);           % extract segment
w = hanning(2*Lw+1)';
y(1:Lw) = y(1:Lw).*w(1:Lw);    % fade-in
y(L-Lw+1:L) = y(L-Lw+1:L).*w(Lw+2:2*Lw+1); % fade-out
```

La formula di sintesi è data da

$$y(n) = \sum_k a_k g_k(n - n_k) \quad (7)$$

dove a_k è un eventuale coefficiente ampiezza e n_k è l'istante di tempo in cui il grano viene inserito nel segnale di uscita. Si noti che i grani possono sovrapporsi. Per sovrapporre un grano g_k (grano) all'istante $n_k = (\text{iniOLA})$ con a_k ampiezza, possono essere utilizzate le seguenti istruzioni MATLAB®

```
endOLA = iniOLA + lunght(grano) -1;
y(iniOLA: endOLA) = y (iniOLA:endOLA) + ak * grano;
```

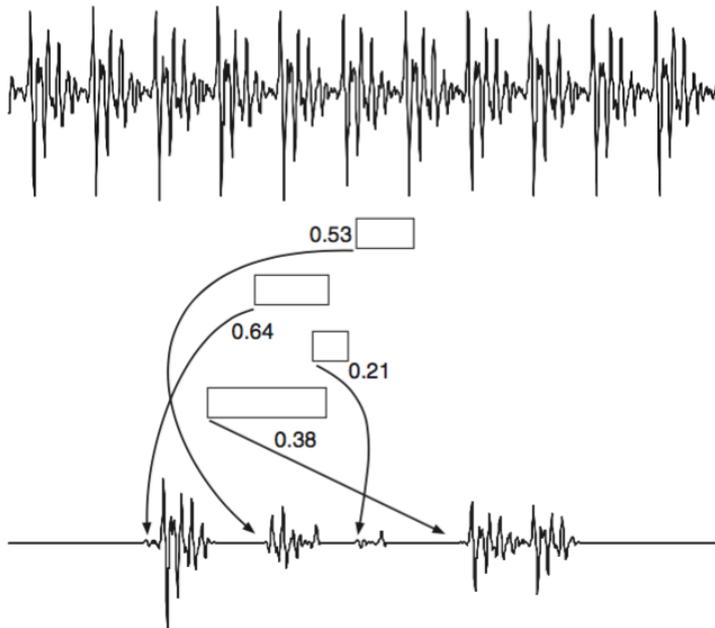


Figura 23: Esempio di granulazione

Un esempio di granulazione con valori casuali dei parametri di punto di inizio del grano e lunghezza, punto di uscita e l'ampiezza è mostrato nella Fig. 23. Il file **M-file 8** mostra l'implementazione dell'algoritmo di granulazione.

Questa tecnica è molto generale e può essere impiegata per ottenere differenti effetti sonori. Il risultato è fortemente influenzato dal criterio utilizzato per scegliere gli istanti n_k . Se questi punti sono regolarmente distanziati nel tempo e la forma d'onda del grano non cambia troppo, la tecnica può essere interpretata come un treno di impulsi filtrati, cioè produce un suono periodico il cui involuppo spettrale è determinato dalla forma d'onda del grano interpretato come una risposta all'impulso. Un esempio è l'algoritmo PSOLA mostrato nelle precedenti Sezioni 3.3 e 4.4. Quando la distanza tra due grani successivi è molto maggiore rispetto a L_k , il suono si tradurrà in grani separati da interruzioni o silenzi, con un carattere specifico. Quando molti grani brevi si sovrappongono (cioè la distanza è inferiore a L_k) si ottiene un effetto di tessitura sonora.

Le strategie per la scelta degli istanti di sintesi possono essere raggruppate in due categorie semplificate: strategie sincrone, principalmente basate su funzioni deterministiche, e asincrone,

M-file 8 (granulation.m)

```

% granulation.m
% Authors: G. De Poli
f=fopen('a_male.m11');
x=fread(f,'int16');
fclose(f);

Ly=length(x); y=zeros(1,Ly);           %output signal
% Constants
nEv=4; maxL=200; minL=50; Lw=20;
% Initializations
L = round((maxL-minL)*rand(1,nEv)+minL; %grain length
initIn = ceil((Ly-maxL)*rand(1,nEv));   %init grain
initOut= ceil((Ly-maxL)*rand(1,nEv));   %init out grain
a = rand(1,nEv);                        %ampl. grain
endOut=initOut+L-1;
% Synthesis
for k=1:nEv,
    grain=grainLn(x,initIn(k),L(k),Lw);
    y(initOut(k):endOut(k))=y(initOut(k):endOut(k))+a(k)*grain;
end

```

basate su funzioni stocastiche. I grani possono essere organizzati in flussi. Ci sono due variabili di controllo principali: il ritardo tra i grani per un singolo flusso, e il grado di sincronicità tra i grani in flussi differenti. Dato che lo spettro locale incide sulla struttura globale del suono, è possibile utilizzare il suono in ingresso che può essere analizzato in grani senza alterare le caratteristiche complessive del suono originale, come gocce d'acqua per suoni tipo corso d'acqua.

È inoltre possibile modificare la forma d'onda del grano con una trasformazione temporale, come modulazione per spostamento di frequenza o dilatazione temporale che si estende per scalatura frequenziale (De Poli e Piccialli 1991). I principali parametri di granulazione sono: durata del grano, ordine di selezione dei grani dal suono in ingresso, ampiezza dei grani, pattern temporale per la sintesi e la densità del grano (cioè grani per secondo). La densità è un parametro primario, in quanto determina la tessitura globale, se frammentaria o continua. Si noti che è possibile estrarre grani da differenti file sonori per creare tessiture ibride, per esempio, che evolvono da una tessitura all'altra.

Applicazioni Musicali

Esempi dell'effetto [di granulazione] possono essere ascoltati in Wishart (1994). Barry Truax ha utilizzato la tecnica della granulazione per elaborare il suono campionato come materiale compositivo. In *The Wings of Nike* (1987) egli ha elaborato solo brevi frammenti "fonemici", ma sequenze più lunghe di suoni ambientali sono state utilizzate in brani come *Pacific* (1990). In ciascuno di questi lavori, il materiale granulato viene dilatato temporalmente da varie quantità e produce così una serie di modifiche percettive che sembrano provenire da dentro il suono (Truax 1995, 2000b).

In *Le Tombeau de Maurice*, Ludger Brümmer utilizza la tecnica della granulazione per ottenere modifiche timbriche, ritmiche e armoniche (Brümmer 1997). Una transizione dalla colorazione del suono originale di un campione orchestrale verso impulsi di rumore si ottiene riducendo progressivamente la dimensione dei grani. Con dimensione dei grani intermedia, l'altezza del suono originale è ancora riconoscibile, sebbene la struttura temporale sia scomparsa (ibid., 3'39"-4'12"). Una melodia può essere suonata selezionando grani ad altezze diverse e variando il ritmo con cui i grani vengono riprodotti (ibid., 8'38"-9'10"). Nuove melodie possono anche apparire come uscita da una granulazione a due stadi. Una prima serie di grani è definita dal campione originale, mentre la seconda è una granulazione della prima. A causa della separazione operata dall'apparato uditivo, la ritmica ed il raggruppamento armonico dei grani sono in continua evoluzione (ibid., 9'30"-10'33").

6 Conclusioni

Gli effetti descritti in questo capitolo sono basati sulla divisione del suono in ingresso in brevi segmenti. Questi segmenti sono trattati con metodi semplici quali la dilatazione temporale tramite ricampionamento, o moltiplicazione dell'ampiezza con un involuppo. La forma d'onda del segmento non viene modificata, mantenendo così la caratteristica del segnale sorgente.

Si possono ottenere due categorie di effetti, a seconda della strategia utilizzata per posizionare i segmenti di tempo durante la sintesi. Se l'ordine e l'organizzazione dei segmenti estratti vengono accuratamente mantenuti, possono essere ottenuti dilatazione temporale o modifica dell'altezza. Metodi di base, SOLA e PSOLA, sono presentati e sono discusse le loro caratteristiche. Questi effetti hanno lo scopo di produrre suoni che vengono percepiti come simili all'originale, ma vengono modificati nella durata o nell'altezza. Come spesso accade con effetti audio digitali, i suoni elaborati prodotti da questi metodi possono essere utilizzati come metodo per deformazione del suono in ingresso, pur mantenendo le sue caratteristiche principali. La bassa complessità computazionale di elaborazione di un segmento temporale consente applicazioni in tempo reale efficienti. Tuttavia, questi algoritmi producono artefatti che limitano l'ambito di applicazione.

La seconda categoria di effetti cambia l'organizzazione e l'ordine dei segmenti in una larga area, e quindi porta alla ricombinazione temporale ed alla granulazione. In questo caso, il suono in ingresso può essere molto meno riconoscibile nell'output. L'elemento centrale diventa il grano con il suo involuppo di ampiezza ed organizzazione temporale. Queste tecniche possono produrre risultati da grani sparsi una tessitura densa, con un rapporto molto libero con il suono originale. Si noti che l'ampia scelta di strategie per l'organizzazione dei grani implica un atteggiamento alla composizione sonora da parte dell'utente. Così la granulazione è diventata una sorta di metafora per la composizione musicale partendo dal micro livello.

Suoni e musica

- Brümmer, Ludger (1993). *The Gates of H*. CD. Computer music. 1995. In: CRI, *The Listening Room*, edel 0014522TLR. Hamburg: CCRMA.
- (1997). *Le Tombeau de Maurice, for computer-generated tape*. CD. In: *Computer Music at CCRMA*. CCRMAV02. CCRMA.

- Eimert, Herbert (1962). *Epitaph für Aikichi Kuboyama*. Electronic music composition. Studio-Reihe neuer Musik, Wergo, LP WER 60014. Riedizione su CD, Koch/Schwann, 1996.
- Furukawa, K. (1993). *Swim, Swan, composition for clarinet and live-electronics*. CD. ZKM.
- (1997). *Den ungeborenen Göttern*. CD. Multimedia-Opera. ZKM.
- Parmegiani, Bernard (1964). *Violostries*. CD. IDEAMA 051 Target Collection, 1996. ZKM e CCRMA.
- (1977). *Dedans-Dehors*. INA-GRM.
- Risset, Jean-Claude (1987). *Risset*. C 1003. CD. Contiene: *Sud* (tracce 1-3), *Dialogues* (traccia 4), *Inharmonique* (traccia 5), *Mutations* (traccia 6). Paris: INA-GRM.
- Schaeffer, Pierre e Guy Reibel (1998). *Solfège de l'objet sonore*. Con 3 CD-DA. Paris: INA-GRM.
- Truax, Barry (1995). *Granular time-shifting and transposition composition examples*. CD. In: *Computer Music Journal*, vol. 19, index 6.
- Wishart, Trevor (1994). *Audible Design*. Sound examples. CD. York.

Bibliografia

- Ander-ton, C. (1995). *Multieffects for Musicians*. Amsco Publications.
- Bogdanowicz, K. e R. Blecher (1989). «Using multiple processors for real-time audio effects». In: *Proc. AES 7th Int. Conf.* Pp. 337–342.
- Bristow-Johnson, R. (1995). «A detailed analysis of a time-domain formant-corrected pitch shifting algorithm». In: *J. Audio Eng. Soc.* 43.5, pp. 340–352.
- Cary, Tristram (1992). *Illustrated Compendium of Musical Technology*. Faber e Faber.
- Chion, Michel (1982). *La Musique Electroacoustique*. QSJ No 1990. PUF.
- Crochiere, R. E. e L. R. Rabiner (1983). *Multirate Digital Signal Processing*. Prentice-Hall.
- Dattoro, J. (1987). «Using digital signal processor chips in a stereo audio time compressor/expander». In: *Proc. 83rd AES Convention, Preprint 2500*.
- De Poli, Giovanni e Aldo Piccialli (1991). «Pitch-synchronous granular synthesis». In: *Representations of musical signals*. A cura di Giovanni De Poli, Aldo Piccialli e Curtis Roads. Cambridge, Mass.: MIT Press, pp. 391–412.
- Disch, S. e U. Zölzer (1999). «Modulation and delay line based digital audio effects». In: *Proc. DAFX-99 Digital Audio Effects Workshop*, pp. 5–8.
- Dutilleux, Pierre (1988). *Mise en œuvre de transformations sonores sur un système temps-reel*. Technical report, Rapport de stage de DEA. CNRS-LMA.
- Dutilleux, Pierre et al. (2011). «Time-segment processing». In: *Digital Audio Effects*. A cura di Udo Zölzer. 2ª ed. John Wiley & Sons, Ltd. Cap. 6, pp. 185–217.
- Enders, B. (1997). *Lexikon Musikelektronik*. Atlantis Schott.
- Gaskell, P. S. (1987). «A hybrid approach to the variable speed replay of digital audio». In: *J. Audio Eng. Soc.* 35, pp. 230–238.
- Geslin, Y. (1998). «Sound and music transformation environments: A twenty-year experiment at the *Groupe de Recherches Musicales*». In: *Proc. DAFX-98 Digital Audio Effects Workshop*, pp. 241–248.
- (2000). «About the various types of *Phonogènes*». Personal communication.
- Giordano, T. A., H. B. Rothman e H. Hollien (1973). «Helium speech unscramblers - a critical review of the state of the art». In: *IEEE Trans. Audio and Electroacoust.* Vol. AU-21. 5.

- Haller, Hans Peter (1995). *Das Experimental Studio der Heinrich-Strobel-Stiftung des Südwestfunks Freiburg 1971- 1989. Die Erforschung der Elektronischen Klangumformung und ihre Geschichte*. Nomos.
- Hamon, C., E. Moulines e F. Charpentier (1989). «A diphone synthesis system based on time-domain prosodic modifications of speech». In: *Proc. ICASSP*, pp. 238–241.
- Hejna, Don e Bruce R. Musicus (1991). *The SOLAFS time-scale modification algorithm*. Technical report, BBN.
- Knesebeck, A. von dem, P. Ziraksaz e U. Zölzer (2010). «High quality time-domain pitch shifting using PSOLA and transient preservation». In: *Proc. of the 129th AES Convention*.
- Laroche, J. (1998). «Time and pitch scale modifications of audio signals». In: *Applications of Digital Signal Processing to Audio and Acoustics*. A cura di M. Kahrs e K.-H. Brandenburg. Kluwer, pp. 279–309.
- Lee, F. F. (1972). «Time compression and expansion of speech by the sampling method». In: *J. Audio Eng. Soc.* 20.9, pp. 738–742.
- Lin, C. Y. e J. S. R. Jang (2004). «A two-phase pitch marking method for TD-PSOLA synthesis». In: *8th Int. Conf. Spoken Lang. Process.*
- Makhoul, J. e A. El-Jaroudi (1986). «Time-scale modification in medium to low rate speech coding». In: *Proc. ICASSP*, pp. 1705–1708.
- Massie, D. C. (1998). «Wavetable sampling synthesis». In: *Applications of Digital Signal Processing to Audio and Acoustics*. A cura di M. Kahrs e K.-H. Brandenburg. Kluwer, pp. 311–341.
- Mattheyses, W., W. Verhelst e P. Verhoeve (2006). «Robust pitch marking for prosodic modification of speech using TD-PSOLA». In: *Proc. IEEE Benelux/DSP Valley Signal Process. Symp.*
- McNally, G. W. (1984). «Variable speed replay of digital audio with constant output sampling rate». In: *Proc. 76th AES Convention, Preprint 2137*.
- Moles, Abraham (1961). *Les Musiques expérimentales*. Trad. da D. Charles. Zurigo: Cercle d'art contemporain.
- Moulines, E. e F. Charpentier (1990). «Pitch synchronous waveform processing techniques for text-to speech synthesis using diphones». In: *Speech Commun.* 9.5/6, pp. 453–467.
- Moulines, E. e J. Laroche (1995). «Non-parametric technique for pitch-scale and time-scale modification of speech». In: *Speech Commun.* 16, pp. 175–205.
- Peeters, G. e X. Rodet (1999). «SINOLA: A new analysis/synthesis method using spectrum peak shape distortion, phase and reassigned spectrum». In: *Proc. ICMC*, pp. 153–156.
- Poullin, J. e D. A. Sinclair (1957). *The Application of Recording Techniques to the Production of New Musical Materials and Forms. Application to Musique Concrète*. Technical Translation TT-646. National Research Council of Canada, pp. 1–29.
- Poullin, Jacques (1954). «L'apport des techniques d'enregistrement dans la fabrication de matières et formes musicales nouvelles. Applications à la musique concrète». In: *L'Onde Électrique* 324, pp. 282–291.
- Risset, Jean-Claude (1998). «Example of the musical use of digital audio effects». In: *Proc. DAFX-99 Digital Audio Effects Workshop*, pp. 254–259.
- Roads, Curtis (1996). *The Computer Music Tutorial*. Cambridge, London: The MIT Press.
- (1998). «Micro-sound, history and illusion». In: *Proc. DAFX-98 Digital Audio Effects Workshop*, pp. 260–269.
- Roucos, S. e A. M. Wilgus (1985). «High quality time-scale modification for speech». In: *Proc. ICASSP*, pp. 493–496.

- Schaeffer, Pierre (1973). «La Musique Concrète». In: *QJ* 1287.
- Schnell, N. et al. (2002). «Synthesizing a choir in real-time using pitch synchronous overlap add (PSOLA)». In: *Proc. 1st IEEE Benelux Workshop Model based Proc. Coding Audio*.
- Springer, Anton (1955). «Ein akustischer Zeitregler». In: *Gravesaner Blätter* 1, pp. 32–37.
- Springer, Anton M. (1959). «Akustischer Tempo- und Tonlagenregler». In: *Gravesaner Blätter* 13.80, pp. 32–37.
- Truax, Barry (1988a). «Discovering inner complexity: Time-shifting and transposition with a real-time granulation technique». In: *Computer Music Journal* 18.2, pp. 38–48.
- (1988b). «Real-time granular synthesis with a digital signal processor». In: *Computer Music Journal* 12.2, pp. 14–26.
- (2000a). *Granular synthesis*. URL: <http://www.sfu.ca/truax/gran.html> (visitato il 07/01/2016).
- (2000b). *Granulation of sampled sounds*. URL: <http://www.sfu.ca/truax/gsample.html> (visitato il 07/01/2016).
- Verhelst, W. e M. Roelands (1993). «An overlap-add technique based on waveform similarity (WSOLA) for high quality time-scale modification of speech». In: *Proc. ICASSP*, pp. 554–557.
- Warstat, M. e T. Görne (1994). *Studiotechnik - Hintergrund und Praxiswissen*. Elektor-Verlag.
- White, P. (1999). *Creative Recording, Effects and Processors*. Sanctuary Publishing.
- Zölzer, Udo, cur. (2011). *Digital Audio Effects*. 2^a ed. John Wiley & Sons, Ltd.