



MINISTERO DELLO SVILUPPO ECONOMICO
DIREZIONE GENERALE PER LA TUTELA DELLA PROPRIETÀ INDUSTRIALE
UFFICIO ITALIANO BREVETTI E MARCHI

UIBM

DOMANDA NUMERO	101994900393026
Data Deposito	29/09/1994
Data Pubblicazione	29/03/1996

Sezione	Classe	Sottoclasse	Gruppo	Sottogruppo
H	04	N		

Titolo

PROCEDIMENTO DI MULTIPLAZIONE PER FLUSSI DI SEGNALI AUDIOVISIVI CODIFICATI
SECONDO LO STANDARD MPEG1.

Descrizione dell'invenzione avente per titolo:

**"PROCEDIMENTO DI MULTIPLAZIONE PER FLUSSI DI SEGNALI
AUDIOVISIVI CODIFICATI SECONDO LO STANDARD MPEG1"**

a nome CSELT - Centro Studi e Laboratori Telecomunicazioni S.p.A. - Via
Guglielmo Reiss Romoli, 274 - 10148 Torino - nazionalità italiana.

Inventori: Giovanni CAIRE

Giampaolo MICHIELETTO

Domanda n. TO 94A000753

depositata il 28.3.1994

==.==.==.==

Testo della descrizione

La presente invenzione si riferisce alle applicazioni di registrazione digitale di immagini in movimento e audio associato e in particolare riguarda un procedimento di multiplazione per flussi di segnali audiovisivi codificati secondo lo standard MPEG1.

In questi anni si stanno affermando numerosi servizi, offerti da fornitori sia privati, che pubblici, denominati "multimediali", in grado di fornire informazioni video, comprendenti sequenze di immagini in movimento, immagini statiche, testi, informazioni sonore e altri dati. Fra questi servizi, riveste un'importanza notevole il "Video on demand", che permette all'utente di selezionare da casa propria la visione di filmati in qualunque momento, senza necessità di seguire una programmazione prestabilita. Altri servizi, comprendono la diffusione di filmati pubblicitari, filmati a scopo didattico o di telelavoro, eccetera.

Per queste applicazioni è necessario tradurre in forma digitale i segnali generati dalle diverse sorgenti, il che comporta la necessità di elaborare, memorizzare e trasferire grosse quantità di dati. Date le limitate capacità degli attuali supporti di trasmissione e registrazione, si rende quindi necessario un procedimento di codifica, che permetta di eliminare tutta quella parte di informazione non strettamente necessaria per una buona qualità della presentazione.

A tale scopo, sono stati definiti vari procedimenti di codifica, fra i quali vi è lo standard ISO/IEC 11172, denominato MPEG1, proposto dal gruppo MPEG.

Dovendo presentare in un terminale remoto una sequenza di immagini in movimento e audio associato, registrata in un archivio centralizzato, è necessario trattare le informazioni audio e video contemporaneamente; queste devono perciò essere unite in un solo flusso numerico con un procedimento di multiplazione. All'atto della presentazione all'utente, che avviene su apparati differenti (monitor, altoparlante), le informazioni video e audio devono essere nuovamente separate seguendo un procedimento inverso di demultiplazione.

In generale, volendo per esempio inserire nel flusso completo anche dei sottotitoli da visualizzare durante la presentazione, oppure pensando alla possibilità di offrire all'utente più di un canale audio, i procedimenti di multiplazione/demultiplazione devono prevedere la possibilità di dover operare su più di due flussi in ingresso.

Il sistema completo di produzione di un flusso codificato MPEG1 comprende quindi vari flussi in ingresso, ciascuno relativo a un certo

tipo di informazione. Ciascuno di essi viene generato da una sorgente (Telecamera, microfono, etc) ed è codificato secondo lo standard suddetto. A valle dei codificatori, i singoli flussi vengono multiplati in un unico flusso d'uscita.

Analogamente il sistema di decodifica comprende un demultipatore, che estrae i singoli flussi e li smista ai rispettivi decodificatori. Qui, i diversi segnali sono ritradotti nella loro forma originale e poi passati agli apparati di presentazione.

Il procedimento di multiplazione adottato da MPEG1 è del tipo a divisione di tempo: il flusso complessivo è strutturato come una sequenza di intervalli detti pacchetti, ciascuno dei quali contiene dati di un solo tipo, indicato in testa al pacchetto stesso.

Per ogni intervallo, il multiplatore deve decidere da quale flusso in ingresso prelevare i dati per costruire i pacchetti. Lo standard MPEG1 non impone alcuna restrizione su questa scelta, se non che il flusso prodotto possa essere decodificato senza che un decodificatore modello, definito nello standard, venga a trovarsi nelle condizioni in cui la sua memoria di ingresso abbia una mancanza (underflow) o un eccesso (overflow) di dati su cui operare.

Nel paragrafo 5 dell'Annex A del documento ISO/IEC 11172 che definisce la standard MPEG1, precedentemente citato, è riportato a titolo di esempio un procedimento di multiplazione che utilizza una strategia denominata "Proportional rate". Questa si basa su uno schema fisso di scelta dei flussi da multiplare, che si ripete ciclicamente. All'interno di questo schema, ad ogni flusso elementare viene assegnato un numero di

pacchetti proporzionale alla sua velocità di cifra nominale. In questo modo, la frequenza media di ogni flusso elementare all'interno del flusso completo è uguale alla velocità di cifra nominale su un intervallo di tempo il più piccolo possibile.

Il limite di questo procedimento è dovuto al fatto che le velocità di cifra reali dei flussi in ingresso non sono in realtà costanti, ma fluttuano intorno al loro valore nominale. Ciò è dovuto principalmente al fatto che le immagini codificate hanno dimensioni, in termini di numero di bit, che possono variare notevolmente per ragioni intrinseche al processo di codifica, mentre la loro durata (intervallo di tempo durante il quale un'immagine è visualizzata) è fissa. E' allora chiaro che la presenza di un certo numero di immagini "grandi" vicine all'interno della sequenza fa sì che il decodificatore estragga dalla sua memoria d'ingresso dati a velocità maggiore della velocità di cifra nominale del flusso video, creando problemi di "underflow". Analogamente, la presenza di più immagini "piccole" vicine può portare a problemi di "overflow".

Ovvio a questi inconvenienti il procedimento di multiplazione per flussi di segnali audiovisivi codificati secondo lo standard MPEG1, oggetto della presente invenzione, il quale tiene conto dell'andamento delle quantità di dati nelle memorie di ingresso dei decodificatori evitando così possibili condizioni di underflow e overflow. Inoltre, tale procedimento si applica a un numero qualunque di flussi in ingresso senza introdurre particolari complicazioni nella costruzione del flusso multiplato.

E' particolare oggetto della presente invenzione un procedimento di moltiplicazione per flussi di segnali audiovisivi codificati secondo lo standard MPEG1, come descritto nella parte caratterizzante della rivendicazione 1.

Queste ed altre caratteristiche della presente invenzione saranno meglio chiarite dalla seguente descrizione di una forma preferita di realizzazione della stessa, data a titolo di esempio non limitativo, e dai disegni annessi in cui:

- la Fig. 1 è un diagramma temporale dell'andamento della quantità di dati nella memoria d'ingresso di un decodificatore;
- la Fig. 2 è un diagramma temporale che evidenzia una condizione di futuro underflow;
- la Fig. 3 è un diagramma temporale dove è rappresentata la grandezza "relax";
- la Fig. 4 è un diagramma temporale dove è meglio definita la grandezza "relax";
- la Fig. 5 è un diagramma temporale dove è evidenziato un caso particolare di condizione di underflow.

Nella Fig. 1 sono tracciati in un diagramma temporale, per un generico flusso elementare S^i , le quantità di dati Q in transito attraverso la memoria d'ingresso del decodificatore. In particolare, è indicato con $A^i(t)$ l'andamento dei dati in arrivo dal demultiplicatore e con $E^i(t)$ l'andamento dei dati estratti per la decodifica.

Quando il flusso completo, generato dal moltiplicatore, contiene pacchetti di S^i , l'andamento $A^i(t)$ cresce con pendenza definita dalla

velocità di cifra totale R (intervallo $t < t_1$). Viceversa, quando ci sono pacchetti appartenenti ad un altro flusso elementare, nessun dato entra nella memoria e l'andamento è piatto ($t_1 < t < t_2$). L'andamento $E^i(t)$ è invece verticale ($t = t_3$) quando una "unità di accesso" viene prelevata per la decodifica e piatto in tutti gli altri istanti. Per "unità di accesso" si intende l'equivalente codificato di un insieme di dati da presentare contemporaneamente, ad esempio un'immagine nel caso di un flusso video. Da notare che i tratti verticali di $E^i(t)$ indicano che tutti i dati di una unità di accesso sono estratti contemporaneamente, come specificato nella definizione del decodificatore modello.

La quantità di dati, presente in ogni momento nella memoria d'ingresso del decodificatore, indicata in figura con $BF^i(t)$, è data dalla differenza tra $A^i(t)$ e $E^i(t)$. Se l'andamento delle estrazioni supera quello degli arrivi, $BF^i(t)$ diventa negativa e si ha underflow. Se $BF^i(t)$ cresce fino a superare l'ampiezza della memoria d'ingresso del decodificatore, si ha overflow.

Nella Fig. 2 è rappresentata una condizione di "futuro underflow" per il flusso S^i : all'istante attuale t_c non ci sono problemi, ma, supponendo costante la velocità di cifra d'uscita del multiplatore e quindi la pendenza di $A^i(t)$, sicuramente si verificherà underflow a un istante $t_n^i > t_c$ nella memoria d'ingresso del decodificatore di S^i in quanto $E^i(t_n^i)$ sarà maggiore di $A^i(t_n^i)$.

Nella Fig. 3 è indicata con X^i una grandezza denominata "relax" per il flusso elementare S^i . Questa è definita come il massimo intervallo di tempo che può trascorrere senza che vengano costruiti pacchetti con

dati di S^i prima di entrare nella condizione di "futuro underflow" per S^i .

In termini quantitativi si ha:

$$X^i = (t_n^i - t_c) - [E^i(t_n^i) - A^i(t_c)]/R$$

essendo R la velocità di cifra del flusso completo.

La Fig. 4 evidenzia che l'istante t_n^i , attraverso il quale è definito il relax, è l'istante di decodifica di una qualsiasi tra le prossime unità di accesso da decodificare. Ciascuno di questi istanti t_n^i ($t_1^i, t_2^i, t_3^i, \dots$) definisce un diverso valore di relax $X_1^i, X_2^i, X_3^i, \dots$; il relax effettivo per il flusso S^i sarà dato dal minimo fra questi valori, che in Fig. 4 è X_2^i .

In generale, non possiamo sapere quale tra le prossime unità di accesso da decodificare definisce il relax effettivo, se non facendo una prova esaustiva. In fase di calcolo di X^i occorrerà allora fissare un limite K e considerare gli istanti di decodifica t_k^i , per k che va da 1 a K , delle prossime K unità di accesso da decodificare. Il relax sarà calcolato con la formula:

$$X^i = \min(t_k^i - t_c) - [E^i(t_k^i) - A^i(t_c)]/R$$

Chiaramente, aumentando K , si hanno maggiori garanzie di ottenere il relax effettivo, ma cresce il carico computazionale. Semplici considerazioni sulla forma dei flussi elementari codificati secondo lo standard MPEG1 e numerose prove effettuate portano a concludere che $K = 3$ è un limite sufficiente per flussi in ingresso codificati correttamente.

Possiamo pensare a X^i come a un indicatore dell'urgenza che la memoria d'ingresso del decodificatore ha di ricevere dati per non svuotarsi. Il procedimento oggetto dell'invenzione effettua il calcolo del

relax per ciascun flusso S^i da moltiplicare, prima della costruzione di ogni pacchetto. Il pacchetto conterrà dati del flusso elementare che ha il relax minore, ovvero del flusso che ha più urgenza di ricevere dati per evitare problemi di underflow.

Come si vede, il procedimento mira a prevenire le condizioni di underflow. Per evitare problemi di overflow, è sufficiente controllare se si eccede la capacità totale della memoria d'ingresso del decodificatore di S^i aggiungendo alla quantità di dati $BF^i(t_c)$ attualmente presenti quelli che verrebbero messi nel pacchetto in fase di costruzione. In tal caso il flusso S^i non viene preso in considerazione per la costruzione del pacchetto. Se poi nessuno dei flussi da moltiplicare può essere preso in considerazione per problemi di overflow, verrà inviato un pacchetto fittizio (padding), previsto dallo standard, il cui contenuto non ha significato e sarà scartato dal demoltiplicatore senza perciò andare ad occupare spazio nelle memorie dei decodificatori.

Prima di illustrare in dettaglio il procedimento, è necessario fare alcune osservazioni.

La prima riguarda l'applicabilità di questo metodo: per calcolare X^i e $BF^i(t_c)$ è infatti necessario conoscere gli andamenti delle estrazioni $E^i(t)$, i quali sono gestiti dai decodificatori dei singoli flussi elementari, che stanno a valle del moltiplicatore. Peraltro tali andamenti sono definiti unicamente dalle ampiezze delle varie unità di accesso (ampiezze, i cui valori sono reperibili all'interno dei flussi da moltiplicare) e dagli istanti di decodifica di queste, che il moltiplicatore conosce (Decoding Time Stamps, secondo la definizione contenuta nello standard MPEG1).

La seconda osservazione riguarda alcuni accorgimenti che occorre ancora prendere per evitare problemi di underflow che si potrebbero verificare nel caso particolare illustrato nella Fig. 5. In questa figura sono rappresentati gli andamenti degli arrivi e delle estrazioni per due flussi elementari S^i ed S^j .

Come si può notare, all'istante corrente t_c , X^i è minore di X^j e quindi il moltiplicatore dovrebbe scegliere di costruire il prossimo pacchetto con dati di S^i . Peraltro, la costruzione e l'invio del pacchetto richiedono un certo tempo T , proporzionale alla dimensione P del pacchetto stesso e durante tale intervallo di tempo nessun dato del flusso S^j può essere inserito nel flusso completo. L'andamento $A^j(t)$ degli arrivi per il flusso S^j resterà allora piatto fino all'istante $t_c + T$ e si verificherà perciò underflow sulla memoria d'ingresso del decodificatore di S^j .

Prima di costruire un pacchetto è quindi necessario inserire un ulteriore controllo. Se esiste un flusso S^j tale per cui:

$$E^j(t_c + T) > A^j(t_c) \quad \text{dove } T = P/R$$

si deve costruire il prossimo pacchetto con i dati di S^j e con dimensione $P' = E^j(t_c + T) - A^j(t_c)$, anche se il confronto fra i relax avrebbe indicato il flusso S^i come sorgente di dati da utilizzare.

Da notare che se si mantenesse P come dimensione di pacchetto, si eviterebbe l'underflow per S^j , ma si entrerebbe sicuramente in condizione di futuro underflow per S^i . Così invece, a patto che $T' = P'/R$ sia minore di X^i , si riesce ad evitare lo svuotamento della memoria d'ingresso dei decodificatori di entrambi i flussi.

Il procedimento oggetto dell'invenzione è ora descritto in dettaglio. Esso è strutturato per passi, in ognuno dei quali viene aggiornata la situazione all'istante corrente t_c e viene scelto il flusso elementare da cui prendere i dati per costruire il prossimo pacchetto.

Si definiscono nel seguito alcuni simboli relativi a grandezze d'interesse, alcune delle quali sono già state introdotte:

N è il numero di flussi da moltiplicare;

R è la velocità di cifra del flusso moltiplicato, mantenuta ad un valore costante;

P è un valore conveniente del numero di byte contenuti in ciascun pacchetto, uguale per esempio a 2048;

U^{i_h} è la generica unità di accesso del flusso S^i . A questa sono associati un istante di decodifica t^{i_h} (Decoding Time Stamp) e una dimensione D^{i_h} ;

C^i è la capacità della memoria d'ingresso del decodificatore del flusso S^i .

Le operazioni che il moltiplicatore deve compiere ad ogni passo sono le seguenti. Per ogni flusso in ingresso S^i (i compreso tra 1 e N) provvede ad:

- aggiornare all'istante corrente t_c i dati già estratti dalla memoria d'ingresso del decodificatore

$$E^i(t_c) = \sum D^{i_h} \quad \text{dove } h \text{ deve essere tale che } t^{i_h} < t_c;$$

- aggiornare le prossime K unità di accesso non ancora decodificate all'istante corrente t_c ;

- calcolare la quantità di dati attualmente contenuta nella memoria d'ingresso del decodificatore secondo la formula

$$BF^i(t_c) = A^i(t_c) - E^i(t_c)$$

- se $BF^i(t_c) + P \leq C^i$ calcolare il relax secondo la formula

$$X^i = \min(t_k^i - t_c) - [E^i(t_k^i) - A^i(t_c)] / R \quad \text{per } 1 \leq k \leq K$$

essendo $E^i(t_k^i) = E^i(t_c) + D^i_1 + \dots + D^i_k$

altrimenti il flusso S^i è considerato non valido per problemi di overflow.

Eseguite queste operazioni, se nessun flusso è valido per problemi di overflow, viene costruito un pacchetto fittizio (padding), altrimenti si eseguono i seguenti passi, che consistono nello:

- scegliere il flusso S^i con il relax minore con i dati del quale riempire il pacchetto in costruzione;
- se esiste S^j ($j \neq i$) tale che

$$E^j(t_c + P/R) > A^j(t_c)$$

cambiare il flusso scelto S^i con S^j e modificare la dimensione del pacchetto da costruire a questo passo da P a

$$P' = E^j(t_c + P/R) - A^j(t_c);$$

- detto S^* il flusso scelto al termine di queste operazioni, costruire il pacchetto di dimensione P^* con i dati di S^* e aggiornare l'andamento degli arrivi da considerare al prossimo passo (istante $t_c + P^*/R$) secondo la formula

$$A^*(t_c + P^*/R) = A^*(t_c) + P^*.$$

All'istante $t = 0$, si pone $A^i(0) = 0$, per ogni i , a fini di inizializzazione.

E' evidente che quanto descritto è stato dato a titolo di esempio non limitativo. Varianti e modifiche sono possibili senza per questo uscire dall'ambito di protezione delle rivendicazioni.

Rivendicazioni

1. Procedimento di multiplazione per N flussi di segnali audiovisivi codificati secondo lo standard MPEG1, per ottenere un flusso multiplato con velocità di cifra R e costituito da pacchetti, contenenti P byte ciascuno, da inviare a un demultiplatore remoto in grado di pilotare N decodificatori, ciascuno relativo a uno dei flussi di segnali audiovisivi codificati e fornito di una memoria di capacità C^i , da cui vengono prelevate agli istanti t_h^i unità di accesso U_h^i , intese come gli equivalenti codificati di insiemi di dati di dimensione D_h^i da presentare contemporaneamente, caratterizzato da ciò che, per ogni flusso in ingresso S^i ($1 \leq i \leq N$) provvede ad:

- aggiornare all'istante corrente t_c i dati $E^i(t_c)$ già estratti dalla memoria d'ingresso del decodificatore

$$E^i(t_c) = \sum D_h^i \quad \text{dove } h \text{ deve essere tale che } t_h^i < t_c;$$

- aggiornare le prossime K unità di accesso non ancora decodificate all'istante corrente t_c ;

- calcolare la quantità di dati $BF^i(t_c)$ attualmente contenuta nella memoria d'ingresso del decodificatore secondo la formula

$$BF^i(t_c) = A^i(t_c) - E^i(t_c)$$

dove $A^i(t_c)$ è la quantità di dati arrivati;

- se $BF^i(t_c) + P \leq C^i$ calcolare il relax X^i secondo la formula

$$X^i = \min(t_k^i - t_c) - [E^i(t_k^i) - A^i(t_c)] / R \quad \text{per } 1 \leq k \leq K$$

essendo $E^i(t_k^i) = E^i(t_c) + D_1^i + \dots + D_k^i$

altrimenti il flusso S^i è considerato non valido per problemi di overflow;

- eseguite queste operazioni per tutti gli N flussi S^i , se nessun flusso è valido per problemi di overflow, viene costruito un pacchetto fittizio (padding), altrimenti si eseguono i seguenti passi, che consistono nello:
- scegliere il flusso S^i con il relax minore con i dati del quale riempire il pacchetto in costruzione;
- se esiste S^j ($j \neq i$) tale che

$$E^j(t_c + P/R) > A^j(t_c)$$

cambiare il flusso scelto S^i con S^j e modificare la dimensione del pacchetto da costruire a questo passo da P a

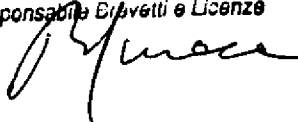
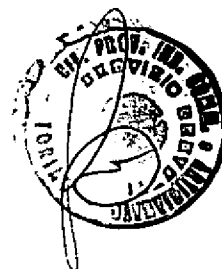
$$P' = E^j(t_c + P/R) - A^j(t_c);$$

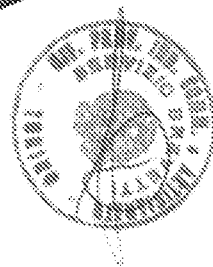
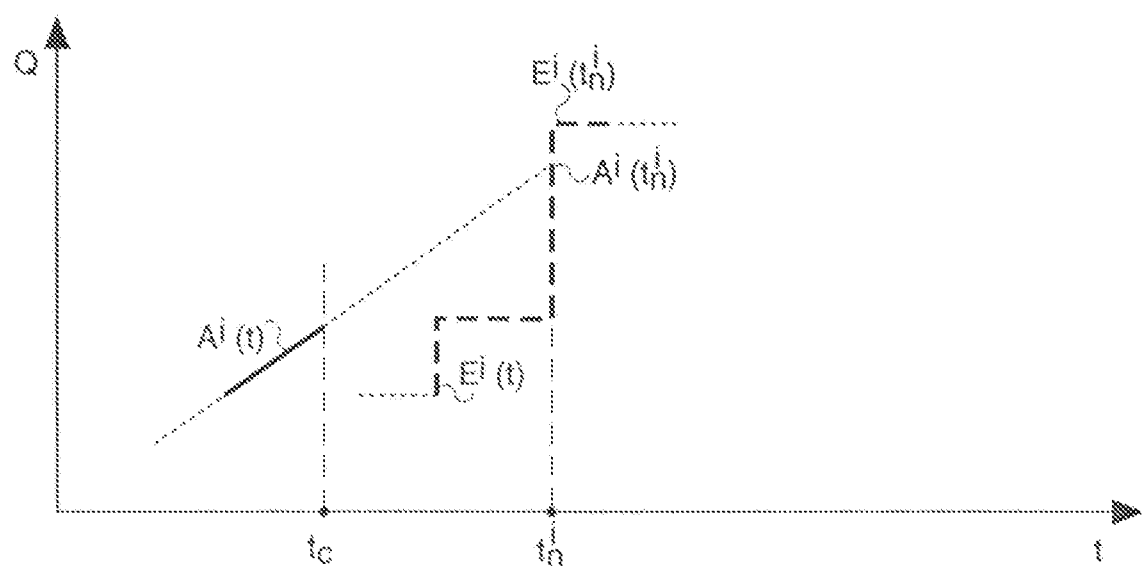
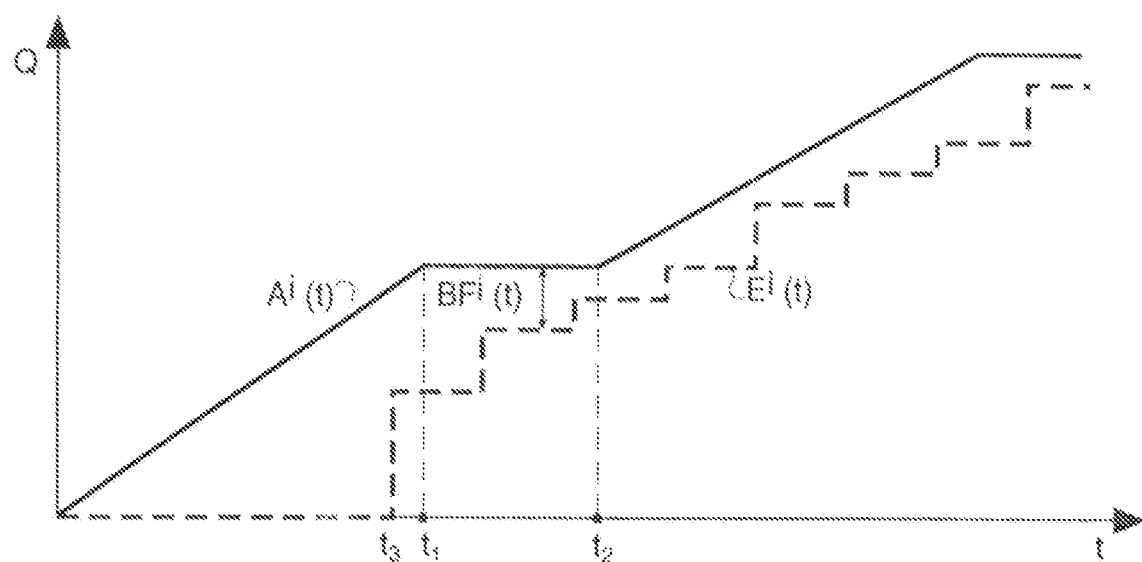
- detto S^* il flusso scelto al termine di queste operazioni, costruire il pacchetto di dimensione P^* con i dati di S^* e aggiornare l'andamento degli arrivi $A^*(t)$ da considerare al prossimo passo (istante $t_c + P^*/R$) secondo la formula

$$A^*(t_c + P^*/R) = A^*(t_c) + P^*.$$

2. Procedimento di moltiplicazione come nella rivendicazione 1, caratterizzato da ciò che all'istante $t = 0$, si pone $A^i(0) = 0$, per ogni i , a fini di inizializzazione.
3. Procedimento di moltiplicazione come nella rivendicazione 1, caratterizzato da ciò che il numero K delle prossime unità di accesso non ancora decodificate all'istante corrente t_c è uguale a 3.
4. Procedimento di moltiplicazione come nella rivendicazione 1, caratterizzato da ciò che il numero P di byte contenuti in ciascun pacchetto è uguale a circa 2048.

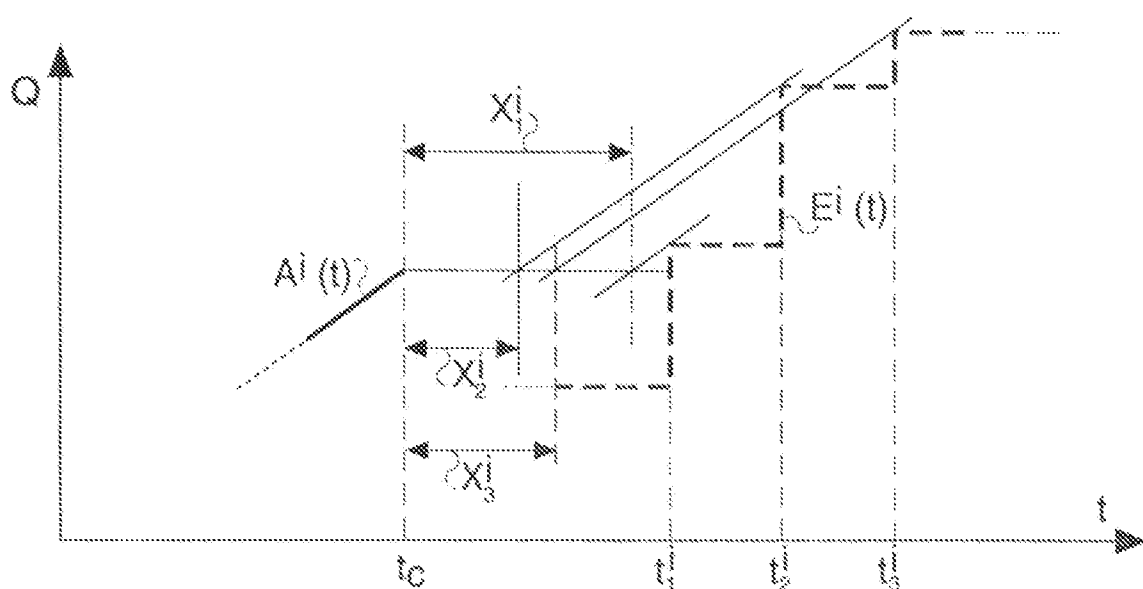
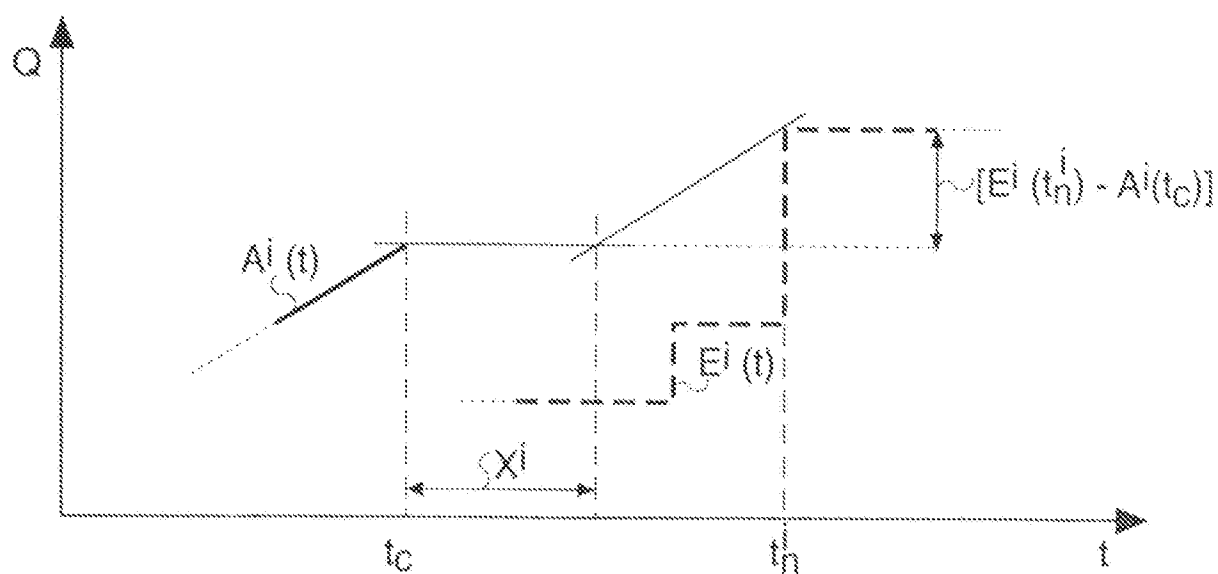
CSELT
Centro Studi e Laboratori Telecomunicazioni S.p.A.
Il Responsabile Brevetti e Licenze



OSSELT
 Centro Studi e Innovazioni Telecomunicazioni S.p.A.
 Il Responsabile Brevetti e Licenze

[Handwritten signature]



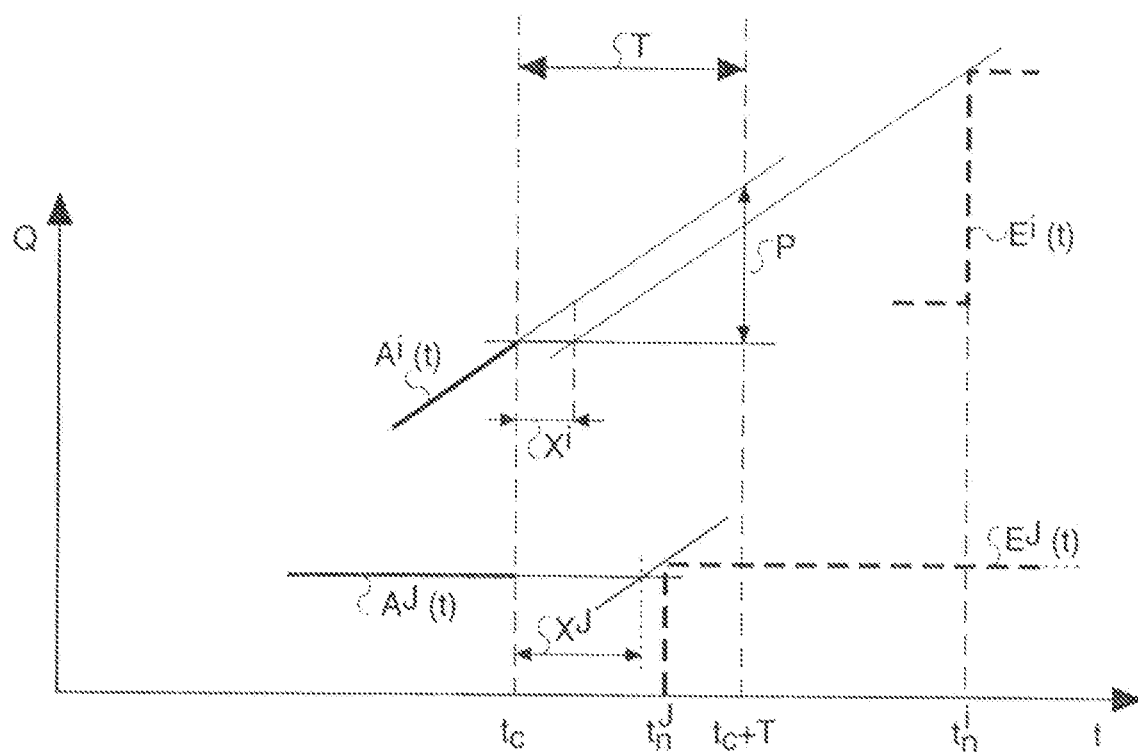


FIG. 5

CSELT
Centro Studi e Laboratori Telecomunicazioni S.p.A.
Il Responsabile Brevetti e Licenze

Refinera

