

P. PALAMIDESE

Rapporto Interno C89-50

**FASI FONDAMENTALI DEL PROCESSO
DI VISUALIZZAZIONE SCIENTIFICA**

FASI FONDAMENTALI DEL PROCESSO DI VISUALIZZAZIONE SCIENTIFICA

Sommario

Progettazione (CAD), pubblicità e cinematografia sono i settori che hanno determinato, in misura maggiore, lo sviluppo delle tecniche di "computer graphics" negli ultimi anni portando alla realizzazione di sistemi grafici completi di buon livello. In anni più recenti l'interesse si è spostato verso le problematiche del mondo tecnico scientifico e verso l'uso della visualizzazione come strumento idoneo all'analisi di fenomeni complessi. In alcuni settori, più che in altri, ad esempio nel campo chimico, vengono usati diffusamente, da tempo, sistemi grafici interattivi di alto livello. Solo recentemente, però, con l'aumentata produzione di dati sia calcolati che sperimentali causata dalla diffusione dei super calcolatori e dal perfezionamento degli strumenti di rilevazione, disporre di risorse grafiche adeguate è diventato un problema generale in vari settori (ingegneria, fisica, astrofisica, etc.). C'è richiesta di programmi semplici da usare e che siano integrati nel calcolo scientifico, ma i software esistenti difficilmente soddisfanno tali requisiti. In questo articolo si analizzano le modalità con cui viene attualmente eseguita una visualizzazione scientifica, le varie fasi del calcolo e si discute della necessità di sviluppare sistemi interattivi e modelli di rappresentazione che consentano la migliore analisi possibile dei fenomeni studiati.

Classificazione Computer Graphics, Visualizzazione Scientifica, Modelli Geometrici

1.0 Introduzione*

Nell'ambito delle scienze computazionali il processo per realizzare una simulazione è molto complesso. Ciascuna fase richiede tempi abbastanza lunghi che possono essere dell'ordine di quelli elencati in Fig. 1 [1].

Le operazioni descritte nei punti 3 e 4 sono quelle in cui vengono creati i dati di input e output che possono essere dell'ordine di 200 Mbytes fino a 24 Gbytes. Ciò comporta, per il ricercatore, diversi tipi di problemi che la tecnologia attuale non riesce a risolvere in modo soddisfacente. Dove, come e a quali costi memorizzare così enormi quantità di dati? Usando i sistemi grafici esistenti si riescono ad ottenere facilmente le immagini volute di un fenomeno? Le immagini di sintesi sono utili solo per presentare e divulgare i risultati ottenuti o consentono di vedere aspetti del fenomeno che potrebbero sfuggire all'esame numerico? Da questi problemi è nato un maggior interesse verso la *visualizzazione*, termine coniato di recente [2] per raggruppare in un'unica area svariate tecniche messe a punto per altri scopi e che appartengono, storicamente, alla *computer graphics*, all'*image processing*, all'*animation*, al *geometric modeling*, etc. Con visualizzazione si intendono i vari tipi di calcolo necessari per creare immagini simboliche di eventi e fenomeni scientifici, con lo scopo di mettere in evidenza il maggior numero possibile di proprietà e di caratteristiche spaziali che temporali dei dati. In questo settore non ha senso parlare di "immagini realistiche" in quanto si tratta spesso di fenomeni non visibili nel mondo reale o di entità non strettamente associabili ad una geometria. L'importanza della visualizzazione,

* Questo lavoro è stato realizzato come attività preliminare all'interno dell'Unità Operativa "Funzionalità di stazioni di lavoro connesse a supercalcolatore per visualizzazione e interazione su grafici e immagini in movimento", del Progetto Finalizzato "Sistemi informativi e calcolo parallelo", sottoprogetto "Calcolo scientifico per grandi sistemi".

come strumento di indagine scientifica, e' stata sottolineata in un rapporto, pubblicato da ACM [2], che e' il risultato di uno studio promosso dalla National Science Foundation (NSF). Da questa analisi emerge l'ipotesi che le tecniche grafiche potrebbero rappresentare un nuovo strumento d'avanguardia per la ricerca scientifica e si suggeriscono obiettivi a breve e a lungo termine per un loro proficuo uso, nonche' gli aspetti su cui e' indispensabile promuovere ulteriori ricerche.

Alcuni vantaggi offerti dalla visualizzazione sono i seguenti:

- possibilita' di esaminare grosse quantita' di dati attraverso una descrizione sintetica
- rappresentazione di fenomeni che non si possono vedere nella realta' (perche' troppo grossi, troppo piccoli, troppo pericolosi)
- analisi dei risultati intermedi di una simulazione allo scopo di verificare la correttezza dei parametri iniziali e guidare il calcolo
- presentazione e divulgazione dei risultati finali

L'aspetto piu' interessante e' l'integrazione della grafica interattiva all'interno del processo di calcolo usato per mettere a punto il modello teorico di un fenomeno fisico o chimico: *definizione di un modello, esecuzione di una simulazione, confronto tra risultati teorici e risultati sperimentali per scegliere parametri piu' corretti, lancio di una nuova simulazione*. Alcuni esempi di questo tipo sono descritti in [3] e [4]. Nel primo caso viene presentata una metodologia per la valutazione dei parametri di cella dello YAG (Yttrium Aluminium Garnet) centrata sull'analisi dello spettro di diffrazione sperimentale ai raggi X; una sequenza di iterazioni grafiche consente di scegliere i parametri corretti. Nel secondo caso e' realizzato un metodo di calcolo per raffinare i parametri di un modello di dispersione di elementi inquinanti che escono da camini, su un territorio non pianeggiante.

Allo stato attuale, queste metodologie di calcolo, hanno molti "colli di bottiglia", dovuti sia all'hardware che al software, che impediscono la realizzazione di processi interattivi semplici da usare. Nella maggior parte dei casi i vari compiti computazionali sono eseguiti da software separati che si trasmettono i dati tramite files. Un processo di simulazione completo, che comprenda anche la visualizzazione, e' un processo molto costoso in termini di risorse hardware e software necessarie, difficilmente alla portata di ogni singolo ricercatore ed esistente come procedimento modello, per certe applicazioni, presso centri specializzati.

Nei seguenti paragrafi ci proponiamo di esaminare la parte che riguarda la visualizzazione in quanto, mentre non sembrano esserci grossi problemi a simulare situazioni molto complesse, data la continua crescita della potenza di calcolo, il tempo per analizzare i dati e' destinato a crescere in modo eccessivo se non si adottano soluzioni automatiche.

2.0 Il processo di visualizzazione

Allo stato attuale, la visualizzazione e' un procedimento "batch", a posteriori rispetto al calcolo ed ha lo scopo di presentare e divulgare i risultati finali di una elaborazione. Le modalita' con cui viene utilizzata la visualizzazione, sono due: *per creare immagini statiche e per creare animazioni*. Il software di visualizzazione trasforma le informazioni numeriche in immagini simboliche, eseguendo una serie di operazioni fondamentali che esamineremo nel seguito. La realizzazione di animazioni, in particolare, e' effettuata con procedimenti sequenziali e laboriosi, che richiedono molto tempo calcolo e sono eseguiti, non dal ricercatore, ma da teams specializzati talvolta affiancati anche da esperti di comunicazione visiva. Animazioni di alto livello tecnico, sono state realizzate all'estero, in "laboratori di visualizzazione" collegati con grossi centri di supercalcolo [1] [5] [6]. Le componenti principali di una architettura hardware tipicamente orientata alla

creazione di animazioni su videocassette, e' illustrata in Fig. 2. Esistono software commerciali che possono eseguire alcune fasi del calcolo, ma e' necessario sviluppare interfacce di collegamento tra pacchetti diversi per creare un sistema integrato, nonche' sviluppare algoritmi che consentano nuovi tipi di rappresentazioni.

La realizzazione di una animazione richiede i seguenti passi :

1- gli scienziati e gli esperti di visualizzazione stabiliscono insieme come tradurre in forma grafica i risultati del calcolo scientifico; cio' significa decidere "quali" dati sono piu' significativi per una corretta interpretazione dei fenomeni in questione e "come" devono essere rappresentati; per una produzione di alto livello, anche estetico, viene definito un vero e proprio "storyboard"

2- e' necessario sviluppare software "ad hoc" per trasformare i dati numerici in primitive geometriche e per associare alle primitive opportuni parametri di visualizzazione (modeling); la modellazione e' preceduta da operazioni di filtro sui dati

3- usando una rappresentazione geometrica schematica (wireframe) e quindi piu' veloce da manipolare, si costruiscono le singole scene e si modificano fino a raggiungere l'effetto desiderato

4- si utilizza un software commerciale (ad esempio Wavefront [22]) che offre diversi modelli di illuminazione e si fanno dei tests per decidere quale modello e' piu' adatto a rappresentare il fenomeno in questione

5- vengono fatte delle prove, su un certo numero di scene campione, per valutare che i parametri per il rendering siano stati scelti correttamente

6- le operazioni di rendering sono eseguite in modo sequenziale per ogni scena; si cerca di utilizzare un hardware veloce, se possibile con processori paralleli tra cui distribuire le singole scene (ad esempio Alliant)

7- le scene calcolate e convertite in immagini video (PAL, NTSC) vengono memorizzate su dischi ad eccesso veloce per la successiva fase di editing e di registrazione su nastro

8- viene fatto l'editing delle immagini per introdurre testi e titoli usando software commerciali (ad esempio: Lumena, Wavefront) e l'animazione e' registrata su cassetta

2.1 Fasi del calcolo
 Sia che si tratti di una animazione o di una rappresentazione statica, il nucleo del software di visualizzazione è costituito da tre classi di operazioni [5]: *data enhancement and enrichment*, *mapping*, *rendering* [Fig. 3].

data enhancement and enrichment

Comprende le operazioni di *interpolazione*, *filtering*, *smoothing*, *new quantities*. Le interpolazioni sono molto usate per calcolare dati regolari partendo da dati di input sparsi. Ciò avviene spesso con i dati sperimentali (ad esempio informazioni geologiche rilette su punti del terreno irregolarmente spazati), ma è vero anche nel caso di dati calcolati solo per sottoinsiemi del dominio. Spesso i dati non sono sufficienti per realizzare delle rappresentazioni geometriche significative. In questo caso, si calcolano altri valori come, ad esempio, la normale alla superficie passante per ogni punto. Tutte queste operazioni vanno usate con cautela perché possono introdurre errori sul dato iniziale.

mapping

Si stabiliscono le regole in base a cui mappare i dati su un *modello fisico* simbolico. Il modello è definito tramite i suoi attributi (*appearance parameters*) che sono la geometria, la trasparenza, il colore, il tempo, etc. Le modalità di mappatura dei dati sono definite tramite *transformation functions* di complessa variabile. Nei casi più semplici la geometria del modello è identica a quella dei dati, ma si possono realizzare trasformazioni più complesse ed efficaci, adottando, ad esempio, tecniche di *image processing* (histogramming). I sistemi più flessibili consentono di scegliere le funzioni di trasferimento in modo da selezionare, di volta in volta, la rappresentazione più significativa.

rendering

Con le operazioni di *rendering* si trasforma il modello fisico in una immagine. Ci sono diversi algoritmi di *rendering* a seconda del modello usato e della rappresentazione che si vuole ottenere. Tipiche operazioni di *rendering* sono *viewing transformations*, *hidden surface removal*, *ray casting*, *lighting*, etc. Il tipo di *rendering* scelto può influire sulla interpretazione dei dati.

Bisogna sottolineare che lo scopo delle operazioni di *mapping* e *rendering* non è quello di ottenere immagini esteticamente suggestive, ma quello di trovare una rappresentazione che faciliti la comprensione dei dati, senza modificarli. Per questo motivo, lo scienziato deve conoscere perfettamente i passaggi e le trasformazioni che il dato subisce durante il processo di visualizzazione.

3.0 Tipi di dati

Consideriamo i dati che sono più comuni a varie aree scientifiche e le modalità di rappresentazione. Si tratta di campi scalari, vettoriali e tensori.

3.1.1 Campi scalari su dominio 2D

La maggior parte del software commerciale consente la rappresentazione di campi scalari ad uno (z_1) o due valori (z_1, z_2) definiti su un dominio bidimensionale (x, y). Si tratta di grandezze come la temperatura, la pressione, la densità, l'altezza, etc. Il software contiene modelli di rappresentazione sotto forma di curve e superfici di livello. Nel caso di due valori scalari per ogni punto, in genere, un valore viene mappato sull'altezza e il secondo sul colore. La rappresentazione di più di due scalari può essere complessa anche dal punto di vista interpretativo. Il modello per superfici di livello consente una rappresentazione sintetica di alcune informazioni numeriche, ma non è la migliore possibile in quanto generalmente alcune parti ne oscurano altre. I software più sofisticati contengono anche modelli di illuminazione.

3.1.2 Campi scalari su dominio 3D
Quando il dominio è tridimensionale e difficile costruire una rappresentazione che permetta di avere, in una sola immagine, la vista completa del volume in esame. Attualmente i metodi più usati per rappresentare campi scalari 3D sono i seguenti:

Cross-section rendering

L'immagine è considerata come un insieme di voxels opachi; l'utente può rimuovere interattivamente parte di voxels per vedere quelli interni; la luce illumina le sezioni o slices [7]. L'utente esamina i campi e ricerca i valori minimi o massimi passando in rassegna tutte le slices e facendo degli abbinamenti mentalmente.

Threshold rendering

Questa tecnica è comune in medicina per esaminare immagini 3D misurate mediante la TAC [8] [9] [10]. In ogni immagine ci sono diversi tipi di materiali ognuno caratterizzato da un particolare valore di intensità. Basta selezionare un valore di soglia per individuare i voxels corrispondenti ad uno specifico materiale (ad esempio ossa) ed applicare l'operazione di rendering solo ad essi. Con questa tecnica vengono rappresentate superfici di isovalore. Alle superfici, così identificate, si può applicare un modello di illuminazione o possono essere applicati i più semplici e veloci algoritmi di Gouraud e Phong [11]. Questo metodo permette di selezionare piccole parti dell'oggetto complessivo e richiede uno sforzo mentale di ricostruzione dello stesso attraverso diverse viste. È adatto al campo medico ma meno ad altre applicazioni come, ad esempio, in geofisica dove ha poco senso avere dei frammenti di immagine.

Osservazioni

Un'altra soluzione adottata è quella di introdurre effetti di trasparenza in modo da vedere più livelli contemporaneamente. Questo effetto, però, non può essere spinto troppo oltre per non rendere difficile l'interpretazione dell'immagine. Si può anche realizzare un programma interattivo con cui l'utente varia in continuazione la superficie rappresentata e riesce, così, a costruire un modello mentale del campo; risultato simile si ottiene tagliando, in modo animato, il volume con dei piani scelti dall'utente. Naturalmente queste operazioni richiedono un hardware veloce. Attualmente non è pensabile di estrarre in tempo reale superfici isopotenziali da dati volumetrici, anche se con opportuni preprocessing dei dati e nuovi algoritmi è possibile raggiungere una qualche forma di interattività.

3.2.1 Campi vettoriali su dominio 2D

Questa classe di dati è costituita da punti con un valore di ampiezza e una direzione come può essere il caso di forze, velocità, flussi. Un vettore è matematicamente individuato da tre o più scalari ma, una rappresentazione tramite scalari, come nel caso precedente, porterebbe ad una immagine poco significativa. Si pensi ad una velocità rappresentata tramite le sue componenti x e y mappate rispettivamente sul colore e sulla dimensione z. Su un dominio bidimensionale si usano spesso *arrows* disegnate in ogni punto con lunghezza e orientamento che dipendono dal valore del vettore in quel punto. Questa rappresentazione per vettori è discreta e può creare dei problemi quando i punti sono troppo fitti o ci sono troppe variazioni nella lunghezza dei vettori. Una soluzione al problema è quella di disegnare *arrows* tutte della stessa lunghezza per indicare l'orientamento e usare un altro attributo, ad esempio il colore, per indicare la misura del vettore.

3.2.2 Campi vettoriali su dominio 3D

Su un dominio tridimensionale è molto difficile interpretare *arrows* posizionate nei vari punti dello spazio; in più, quelle di fronte, nascondono quelle che stanno e dietro. Alcune soluzioni possono essere l'effetto di depth-cues, rotazione e traslazione interattiva, sezione e clipping interattivi del volume considerato.

E' possibile semplificare il problema e rappresentare il campo vettoriale tramite alcune delle sue proprietà piu' importanti selezionate mediante *contraction operations*. Una operazione tipica e' quella che calcola la componente del vettore u in una data direzione v (prodotto scalare $u \cdot v$) con conseguente rappresentazione delle componenti dei vettori in una certa direzione. L'utente puo' interattivamente variare la direzione prescelta e avere una successione di immagini con i valori del campo in varie direzioni. Puo' essere significativo rappresentare il quadrato della grandezza del campo, dato dal prodotto scalare $u \cdot u$.

Le variazioni di forma di un oggetto o campo possono essere rappresentate indirettamente sovrapponendo due diverse geometrie. In alcuni casi e' necessario introdurre un fattore di scala per rendere evidenti variazioni che sarebbero altrimenti troppo piccole. Una rappresentazione animata in cui l'utente varia interattivamente il fattore di scala permette di studiare con accuratezza gli scostamenti dai valori di base.

3.3 Campi tensoriali

Consideriamo campi tensoriali del secondo ordine in quanto i campi di ordine zero si riconducono al caso dei campi scalari e quelli del primo ordine si riconducono al caso dei campi vettoriali.

La rappresentazione di tensori e' molto complessa e si stanno ancora cercando soluzioni soddisfacenti. In genere bisogna ricorrere a *contraction operations* per ottenere valori scalari piu' facilmente rappresentabili. Queste operazioni variano a seconda dell'applicazione: alcune hanno significato nel campo dell'analisi degli sforzi, altre sono utili per problemi di fluidodinamica, etc. Spesso si ricorre a valori scalari che danno la densita' di energia nel punto e si rinuncia a rappresentare le direzioni.

4.0 Altri modelli

Una rappresentazione che contenga un maggior numero di informazioni richiede modelli piu' complessi rispetto a quelli esaminati in precedenza. Algoritmi e tecniche di modellazione avanzati sono stati studiati per ottenere effetti di realismo visivo nell'ambito della sintesi di immagini e nella rappresentazione di fenomeni naturali [12]. I modelli creati per descrivere fenomeni come nubi, gas, acqua, fuoco, etc. possono risultare utili nella rappresentazione di dati multidimensionali. Per rappresentare quei fenomeni che hanno un aspetto irregolare e che non possono essere racchiusi entro superfici ben definite, sono state scartate le primitive tradizionali come poligoni, quadriche e sono stati adottati diversi modelli basati su primitive volumetriche come *voxels* e *particles* [13] [14]. Poiche' i modelli trovati sono tuttora molto complessi dal punto di vista computazionale, l'attivitа' di ricerca continua. L'obiettivo e' quello di trovare delle primitive semplici rispetto alle operazioni di rendering in modo da eseguire, in tempi ragionevoli, questa operazione e costruire immagini complesse costituite da milioni di primitive [15].

Nel seguito vengono descritti alcuni dei modelli piu' significativi.

Particle systems

Sono stati definiti da Reeves [14] [16] per modellare *fuzzy objects* cioe' oggetti che hanno superfici irregolari e variabili nel tempo. Sono stati utilizzati, in particolare, per rappresentare fuoco, esplosioni e nuvole. Le operazioni di rendering possono essere molto complesse se si tiene conto di tutti gli effetti di ombre e trasparenze. Reeves ha proposto algoritmi semplificati per alcuni casi specifici. Nel caso del fuoco ogni particella e' semplicemente una sorgente di luce. Il caso delle nubi e' piu' complesso perche' ogni punto non puo' essere considerato semplicemente sorgente di luce, ma come un oggetto riflettente. I sistemi a particelle sono stati utilizzati anche per particolari applicazioni come la modellazione del pianeta Giove, o la simulazione del moto Browniano.

Nel seguito si descrivono le caratteristiche principali di alcuni modelli che possono essere indicati alla rappresentazione di campi.

Soft objects

Blinn [17] propone un approccio per descrivere modelli molecolari. Prende a prestito dalla chimica quantitativa la funzione di distribuzione della densità elettronica per descrivere il contributo di un atomo di materia; la somma di tali funzioni permette di descrivere la superficie di un oggetto. Su questo modello costruisce un algoritmo di rendering abbastanza semplice che diventa però pesante quando gli atomi sono molti. Generalizzando, il modello è adatto a rappresentare campi scalari. Wyvill [18] dimostra che scegliendo funzioni di campo diverse si possono rappresentare oggetti di forma diversa o *soft objects*. La funzione di campo dipende da una serie di parametri ed è valutata all'interno di un raggio di influenza al di fuori del quale la funzione vale zero. Modelli di questo tipo sono studiati per rappresentare tessuti anatomici o tessuti elastici [19].

Scattering models

Questo modello è stato studiato inizialmente per rappresentare gli anelli di Saturno; la materia è costituita da uno strato che contiene particelle riflettenti di raggio R casualmente distribuite. La caratteristica predominante delle particelle è la diffusione (scattering) della luce che crea un effetto di trasparenza tipico di elementi come le nubi, i gas, il fumo. Questo modello è stato ripreso da Sabella [20] per rappresentare campi scalari 3D e fenomeni simili in particolare, con l'obiettivo di dare una rappresentazione che consenta di "vedere" l'interno di un volume.

5.0 Conclusioni

Abbiamo analizzato quegli aspetti di un software di visualizzazione che riguardano la definizione del modello di rappresentazione dei dati e abbiamo sottolineato la necessità di ulteriori indagini per trovare nuovi metodi di rappresentazione di dati numerici caratterizzati da denso contenuto informativo. Si è visto che la maggior parte delle attività di visualizzazione attuale è rivolta alla rappresentazione dei tipi di dati descritti al paragrafo 3 e che, spesso, si usano modelli geometrici basati sulla poligonizzazione delle superfici. L'approssimazione per poligoni è chiaramente poco efficiente quando si hanno grosse quantità di dati: vanno infatti considerati problemi di memoria, problemi di calcolo nella fase di rendering, nonché il fatto che vengono perse delle informazioni. D'altra parte un approccio per volumi presenta ugualmente problemi: di memoria, se la griglia è troppo fitta; perdita di contenuto informativo, se vengono selezionati solo contorni.

Altri problemi riguardano la rappresentazione di molte variabili nella stessa immagine ed in particolare una conveniente rappresentazione di vettori e tensori. Va sottolineato che esistono, allo stato attuale, molti altri aspetti non risolti. C'è carenza di software grafico interattivo integrato nei sistemi di calcolo scientifico, salvo applicazioni sviluppate in laboratori avanzati e specialistici. Nei grossi centri di supercalcolo l'approccio è quello di creare laboratori di visualizzazione con personale tecnico specializzato che realizza disegni e animazioni concordati con lo scienziato. Per ora questo è il miglior modo di procedere fino a che non ci saranno sistemi facilmente utilizzabili dal ricercatore stesso.

Un buon software di visualizzazione deve avere anche altre caratteristiche che lo rendano facilmente utilizzabile da parte di non informatici. Si sottolineano qui tre requisiti che sono ritenuti, da molti, fondamentali [21]: la presenza di interfacce di alto livello (VIMS), il collegamento con una varietà di dispositivi di output (stampanti, diapositive), la possibilità di scambiare disegni con altri software mediante un metafile standard (CGM [23]).

- [1] Stefan M. Fangmeier, *The Scientific Visualization Process*, National Center for Supercomputing Applications, University of Illinois at Urbana/Champaign, Internal Report, 1987
- [2] *Visualization in Scientific Computing*, Computer Graphics 21(6), 1987
- [3] G. Berti, G. Di Guglielmo, P. Palamidese, *CU K β X-RAY powder diffraction pattern improved analysis*, Powder diffraction: An International Journal of Materials Characterization, 1989 (to be published)
- [4] Pirrone N., *Valutazione di impatto ambientale di sostanze inquinanti*, Tesi di laurea, 1989, Facolta' di Ingegneria-CNUCE, Pisa
- [5] Robert B. Haber, *Visualization in Engineering Mechanics: Techniques, Systems and Issues*, National Center for Supercomputing Applications, University of Illinois at Urbana/Champaign, Internal Report
- [6] E. Clementi, et al., *Computer Graphics Tools for Chemistry and Engineering*, IBM Research Report, KGN-172, April, 1988
- [7] J. Blinn, M. Newell, *Texture and Reflection in Computer Generated Images*, Comm. ACM, Oct. 1976, pp. 542-547
- [8] E. Farrell, *Color Display and Interactive Interpretation of Three Dimensional Data*, IBM J. Res. Develop 27(4), July 1983, pp.356-366
- [9] G. Frieder, et al., *Back-to-front Display of Voxel-based Objects*, IEEE Computer Graphics and Applications 5(1), Jan. 1985, pp. 52-60
- [10] W. Lorenzen, H. Cline, *Marching cubes: A High Resolution 3D Surface Construction Algorithm*, Computer Graphics 21(4), July 1987, pp. 163-169
- [11] J.D. Foley, A. Van Dam, *Fundamentals of Interactive Computer Graphics*, Addison-Wesley
- [12] N. Magnenat-Thalman, D. Thalman, *Image Synthesis*, Springer-Verlag, Tokyo, 1987
- [13] L. Chen, et al., *Surface Shading in the Cuberille Environment*, IEEE Computer Graphics and Applications 5(12), December 1985, pp. 33-43
- [14] Reeves, W.T., *Particle Systems - A Technique for Modelling a Class of Fuzzy Objects*, Computer Graphics, Vol. 17, N. 3, July 1983, pp. 359-376
- [15] J. Blinn, et al., *The simulation of natural phenomena*, Proc. SIGGRAPH, Computer Graphics 17(3), July 1983, pp. 137-139
- [16] W. Reeves and R. Blinn, *Approximate and Probabilistic Algorithms for Shading and Rendering Structured Particle Systems*, Computer Graphics, Vol. 19, N. 3, July 1985, pp. 313-322
- [17] Blinn J., *A Generalization of Algebraic Surface Drawing*, ACM Transactions on Graphics 1(3), 1982, pp. 235-256
- [18] Wyvill G., et al., *Soft objects*, Advanced Computer Graphics, Springer, Tokyo, 1986, pp.113-127

- [19] Don House, *Soft Objects*, RPI Technical Report (to be published)
- [20] P. Sabella, *A rendering Algorithm for Visualizing 3D Scalar Fields*, SIGGRAPH '88 Conference Proceedings, Computer Graphics 22(4), August 1988, pp.51-55
- [21] M. Jern, *Highly sophisticated graphics application software breaks the Visualization Barriers*, UNIRAS, Denmark, Internal Report, January 1989
- [22] *Wavefront Technologies*, Santa Barbara, CA
- [23] *Computer Graphics Metafile for the storage and transfer of picture description information*, International Standard ISO 8632, Geneva, February, 1986

P. Palamidese
CNUCE - Pisa

CALCOLO SCIENTIFICO
E
VISUALIZZAZIONE

Congresso AICA - Trieste 4-6 Ottobre 1989

COSA SIGNIFICA VISUALIZZAZIONE SCIENTIFICA?

VISC = Visualization in Scientific Computing

e' un modo per presentare fenomeni naturali,
anche complessi, che sono stati o modellati
matematicamente o misurati scientificamente,
mediante una rappresentazione grafica in n-
dimensioni che sia significativa e
comprensibile per l'analisi del fenomeno

parole chiave: fenomeni naturali
modelli matematici
misure scientifiche
dati n dimensionali
rappresentazione significativa
analisi

QUALI DATI, QUALE RAPPRESENTAZIONE

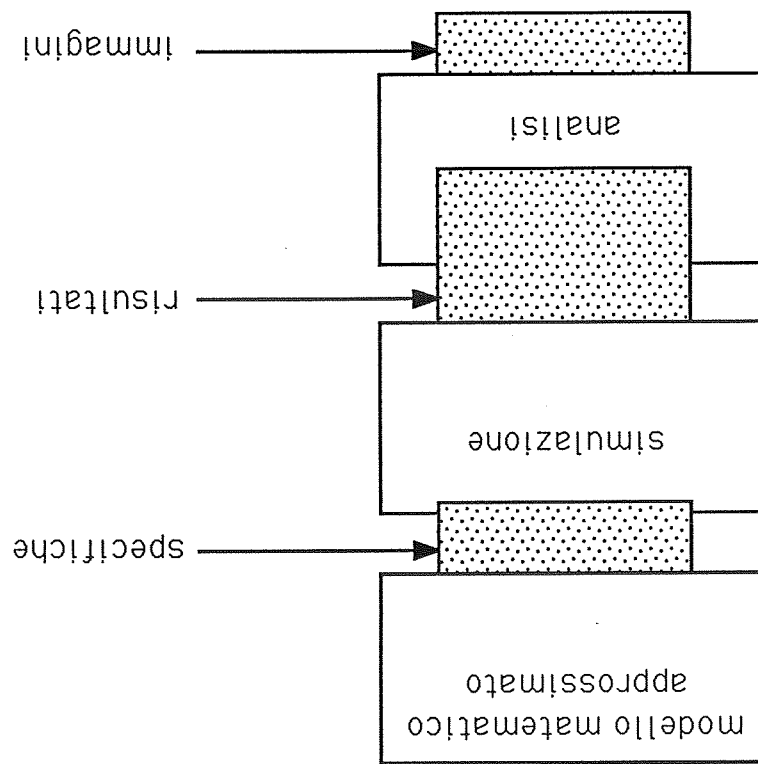
- 1- nella maggior parte dei casi, indipendentemente dalla disciplina, sono dati multidimensionali cioè campi scalari, vettoriali o tensori definiti su domini 2D o 3D (temperatura, pressione, densità, altezza, forze, velocità, flussi,...)
- 2- metodo generale di rappresentazione: le variabili indipendenti sono mappate nelle dimensioni spaziali e le altre variabili o grandezze sono rese tramite gli attributi della visualizzazione (colore, trasparenza, texturing,...)
- 3- molte volte queste grandezze non hanno una forma geometrica, altre volte non è nota a priori; i metodi tradizionali di rappresentazione di superfici si adattano male a questi casi
- 4- se le grandezze sono molte e' necessario selezionare quelle piu' importanti e, comunque rappresentarne poche alla volta

5- molte volte e' utile visualizzare grandezze derivate da quelle date

QUALI SETTORI?

modelli molecolari
immagini mediche
funzioni matematiche
scienze geofisiche
esplorazione dello spazio
astrofisica
fluidodinamica computazionale

IN QUALE FASE DEL CALCOLO ?



A COSA SERVE LA VISUALIZZAZIONE?

- 1 - a presentare i risultati in modo sintetico per farli conoscere ad un pubblico piu' vasto
- 2 - ad analizzare i risultati attraverso la rappresentazione grafica dei vari attributi che caratterizzano un fenomeno
- 3 - per guidare il programma di simulazione attraverso l'analisi visiva dei risultati intermedi

QUALI METODI USA?

- 1 - statico:
presentazione di singole immagini
utile per analizzare dei particolari di un fenomeno
- 2 - dinamico:
presentazione di immagini in movimento
(animazioni)
utile per analizzare l'evoluzione temporale o il fenomeno nel suo complesso
- 3 - interattivo:
interazione sui dati e ricerca di nuove rappresentazioni
utile per analizzare il fenomeno in ogni suo aspetto sia globale che puntuale

tipica "pipeline" di visualizzazione

"se uno fa parte dell'85 % di scienziati che non possiedono strumenti grafici specifici per i loro problemi e non sono programmatori di software grafico, non puo' fare assolutamente nulla con la visualizzazione"

CI SONO GLI STRUMENTI ADATTI PER
VISC?

tipica "pipeline" di visualizzazione

"se uno fa parte dell'85 % di scienziati che non possiedono strumenti grafici specifici per i loro problemi e non sono programmatori di software grafico, non puo' fare assolutamente nulla con la visualizzazione"

CI SONO GLI STRUMENTI ADATTI PER
VISC?

**COME VIENE FATTA LA
VISUALIZZAZIONE NEI LABORATORI
ATTREZZATI?**

- 1- scienziati ed esperti di visualizzazione decidono quali dati rappresentare e come (storyboard)
- 2- uso di vari software commerciali o sviluppati ad hoc e di vari tipi di hardware collegati in rete per eseguire le seguenti operazioni:

a) filtering: dati =====> dati
 •selezione di un sottoinsieme
 didati

•interpolazioni
 •calcolo di grandezze derivate
 •.....

b) mapping: dati =====> geometria
 •trasformazioni da punti a
 primitive
 geometriche come linee,
 triangoli,
 patches,.....

c) rendering: geometria =====>
 immagine
 creazione dell'immagine;
 alle primitive vengono
 associate
 colore,
 shading,
 trasparenza, texturing,

 d)display: immagine =====> output
 riproduzione su un supporto
 schermo o videoregistratore
 come

- 3) altre operazioni
- a) prove delle singole frames con geometrie semplificate (wireframes)
 - b) prove di diversi modelli di illuminazione per scegliere quello piu' significativo per il fenomeno in studio
 - c) il rendering delle singole scene cosi' costruite viene eseguito su calcolatori veloci (supermini) eventualmente con struttura parallela
 - d) postprocessing e editing delle scene e conversione in immagini video

L'IMPORTANZA DELLA "ITERAZIONE" PER L'ANALISI

il procedimento descritto presenta diversi aspetti negativi e molti limiti soprattutto peche' sfugge al controllo diretto dello scienziato; perche' si abbia un processo rivolto piu' all'analisi dei dati che alla presentazione bisogna introdurre un'altra fase

4) iterazione: si ricomincia dalla fase 1) perche'

a) lo scienziato non sempre sa a priori quali dati sono piu' significativi e importanti

b) non sa a priori se la geometria ad una immagine significativa

c) non sa a priori quali parametri di scegliere (colore? trasparenza? shading?...) rendering

L'interattività' e' collegata con
l'interpretabilità'-bisogna esplorare il modo
migliore per rappresentare le proprietà' di un
fenomeno

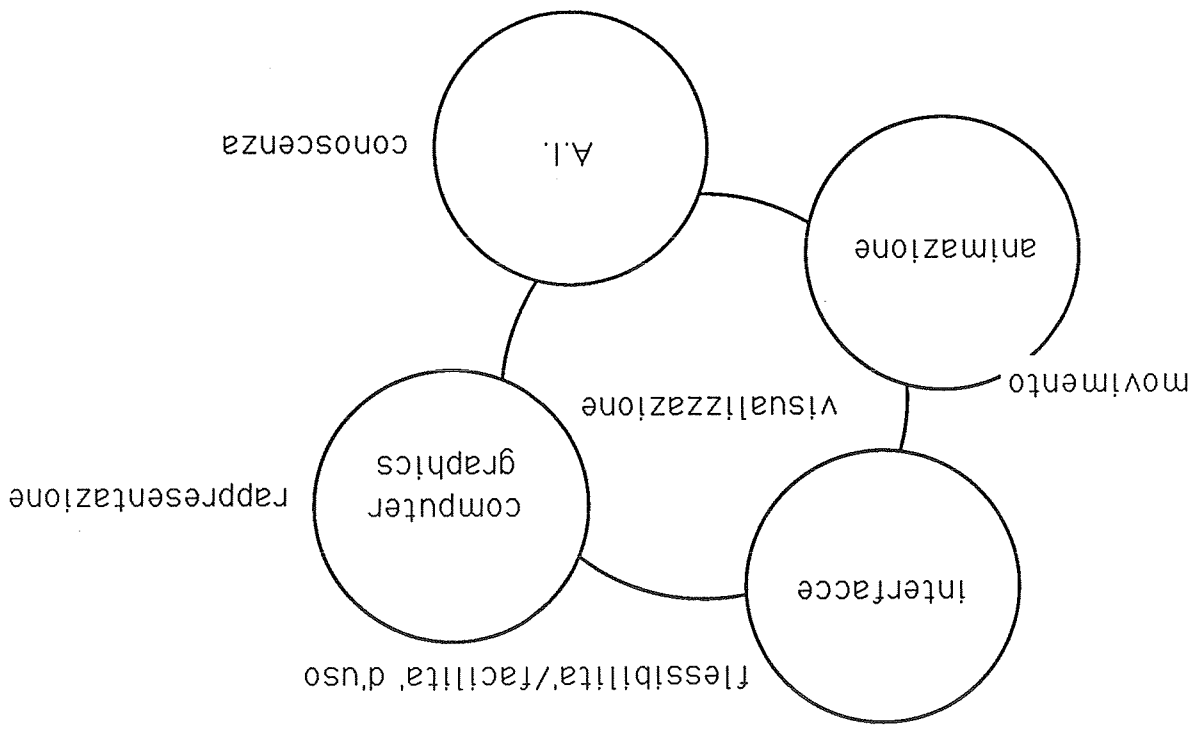
CARATTERISTICHE DEI SW DI VISUALIZZAZIONE NEL PASSATO

1 - sw grafico specialistico legato ad un
problema e difficilmente adattabile ad altri
casi

2 - sw grafico legato all'hw

3 - sw grafico difficile da integrare con
programmi di calcolo

QUALI TECNOLOGIE SONO NECESSARIE?



QUALI TECNICHE DI RAPPRESENTAZIONE?

modelli basati sulla geometria
a) descrivono oggetti in termini di linee,
poligoni, patches e altri concetti
geometrici astratti che sono poi
trasformati in pixels

b) pochi oggetti in natura sono
riconducibili a queste descrizioni!

c) la maggior parte delle visualizzazioni
svilupate fino ad ora si sono basate
sulla rappresentazione di superfici (ad
esempio ricavando involucri all'interno
di volumi)

modelli basati su volumi

a) piu' adatti a molti tipi di dati in campo
medico, geologico, chimica
computazionale, fluidodinamica, ...

b) soddisfano l'esigenza di vedere sia la
parte esterna, che quella interna, che le
variazioni interne locali dei dati

CARATTERISTICHE DEI SW FUTURI

1 - non c'è 'necessità' di sistemi sofisticati e veloci per il rendering di poligoni (richiesti da altre applicazioni) ma sistemi efficienti per:

- a) vedere situazioni specifiche (traiettorie di particelle,...)
- b) vedere interattivamente dati volumetrici
- c) rappresentazione veloce di superfici isovalore

.....

osservazione: fotorealismo versus analisi

- 2) sviluppo di toolkits cioè 'di pacchetti generali orientati alla visualizzazione da utilizzare per costruire applicazioni specifiche
- a) per programmatori
 - b) per scienziati

TENDENZE ATTUALI

1 - si sta passando da una situazione in cui tutto era "calcolo" ad una situazione in cui lo scienziato si rende conto che puo' essere vantaggioso lavorare in un ambiente orientato alla visualizzazione

2 - sta nascendo una nuova generazione di software con caratteristiche specifiche per le esigenze scientifiche

a) livello toolkit cioè 'pacchetti di base per lo sviluppo di applicazioni di visualizzazione sia da parte di programmatori che di scienziati

b) sw orientato piu' a rendering volumetrico che geometrico

c) sw portabile ma nato prevalentemente per superworkstations (Apollo 1000, Ardent Titan, Silicon Graphics IRIS, Stellar GS1000, SUN Taac-1, Pixar Image Computer, ...)

esempi di sw: DORE, Pixar's ChapVolumes, Tracer, A.V.S., VoxelView, Foto,...

c) SW estensibile cioè' strutturato in modo
che sia facile aggiungere funzionalità'
(soprattutto i tipi di dati in input e i tipi
di rappresentazione)
per questa ragione c'è' una certa
tendenza a usare ambienti di sviluppo
object oriented

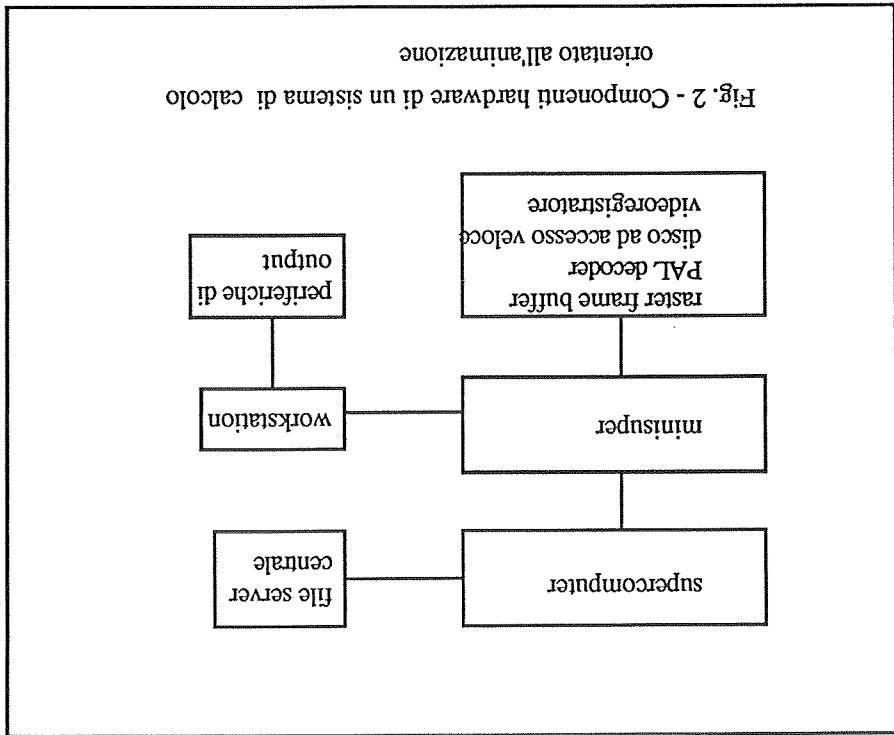
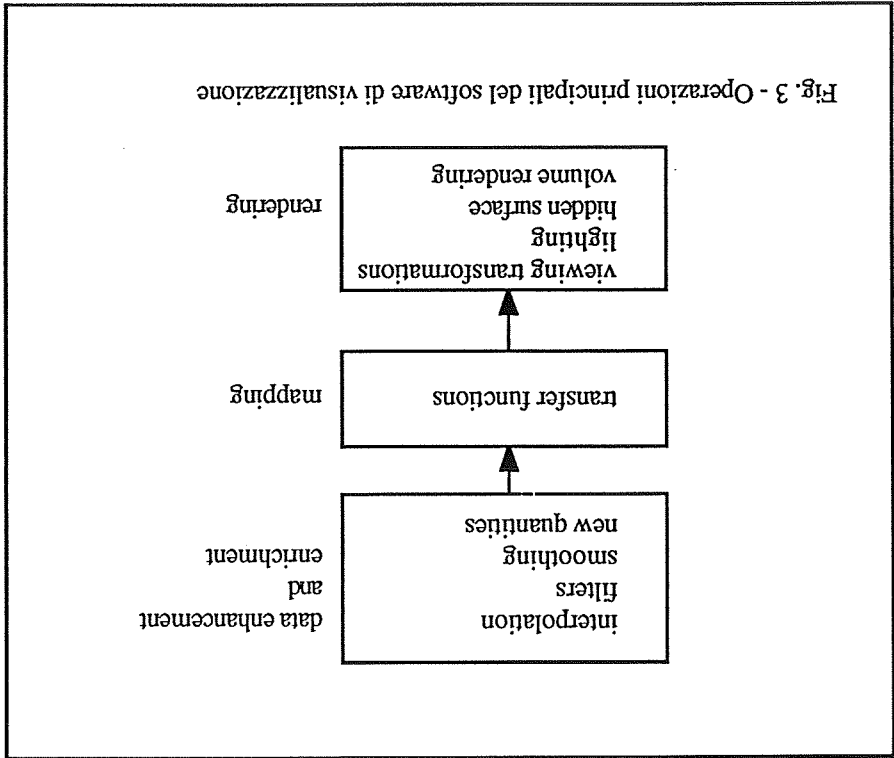


Fig. 1 - Fasi del calcolo scientifico

