# Компютърна графика

С термина компютърна графика обозначаваме генерирането, представянето, обработката(оценката) и визуализацията на графични(геометрични) обекти с помощта на компютър,както и установяването на отношения(взаимовръзки) между графичните обекти и неграфична информация.

Под неграфична информация разбираме знание,което по някакви причини не можем или не искаме да представим във вид на изображение.

Извод:неграфична информация няма!

Термина изображение запазваме за картинките(чертежите),а всяко друго представяне на такива графични обекти ще наричаме представяне.

Под термина графичен обект ще разбираме някакво множество от точки и отсечки.От гледна точка на целите,за които се отработват геометрични обекти в КГ са обособени три направления:

1. Анализ на изображение – в това направление се обработват дигитализирани фотоснимки.Задачите,които се решават са разнородни:от подобряване на някакви качества на снимката,до търсене и откриване на заснети обекти в тази снимка.Съществена задача от това направление е изчисляването на някои свойства на заснети обекти.
2. Анализ на сцени(перцептивна КГ)- това направление обикновено се отнася към т.нар. изкуствен интелект и първоначално е свързано с моделиране на зрението на робот.Най-общо задачите тук са за търсене,но не на конкретни обекти,а на обекти от даден тип.Например в една снимка да се преброят заснетите триъгълници.
3. Генеративна изобразителна КГ-В това направление се решава въпросът за създаването на геометрични модели и тяхното изобразяване(рисуване).

# 2.Обща постановка

Как в исторически план са се развили системите за графично моделиране?За да имаме „рисуване” с помощта на компютър е необходимо периферно устройство,притежаващо способността да рисува.

Програма за ПУ е написана на език от ниско ниво-машинен или асемблер.При тази схема на работа потребителят на такава система за графично моделиране трябва да бъде системен програмист и за него:

1. Произведената от ПУ картинка е изображение
2. Дисплейният файл е представяне

Последен етап-целта е да се моделира в някаква степен работата на художника,т.е. потребителят на някаква система,т.е. при добавяне и отстраняване на графични примитиви да се изгражда изображението в непосредствен контакт с изображението.

Инструментите за работа с изображение са:

1. Екрана
2. Мишката

Важното е,че за да се обсъдят тези проблеми трябва да се изходи от възможностите на ПУ на компютъра както за произвеждане на изображение така и за манипулация на изображенията от потребителя.

# 3.Технически средства на компютърната графика

Под технически средства на компютърна графика разбираме ПУ на компютъра,чрез които можем да постигнем заявената цел.От гледна точка на функциите,които изпълняват ПУ в една диалогова система различаваме три групи технически средства:

1. Графопостроители
2. Диалогови устройства
3. Дисплейни системи,чрез които се визуализира манипулацията на изображенията,съдържат както 1 ,така и 2

Графопостроител наричаме ПИУ,което посредством „пишещ” инструмент оставя следа в някаква среда за рисуване.Оставената следа наричаме изображение.От гледна точка на продължителността на живота на изображение графопостроителите се делят на:

1. Произвеждащи твърдо копие
2. Подходящи за диалогова работа

Типичен представител на графопостроителите,произвеждащи твърдо копие е плоският плотер.Средата за рисуване е лист хартия формат А4-обикновено тази среда се нарича поле за работа.Пишещият инструмент е молив или флумастер,който е закрепен върху греда.Може да бъде придвижван по оста х на работното поле,а гредата може да бъде придвижвана по оста у на работното поле.Пишещият инструмент може да бъде вдигнат или спуснат.Само в спуснато положение той оставя следа.Тогава една отсечка може да бъде изрисувана чрез постъпково придвижване по оста х и оста у.Големината на стъпките е такава,че да може да бъде използван недостатъка на човешкото зрение- стълбичка от отделни стъпала се вижда като права линия.От гледна точка на процеса на рисуване моментното местоположение на молива се нарича текуща точка.Оттук произтичат два начина за изчертаване на отсечки.При първия начин в командата от дисплейния файл се задават в явен вид координатите на първата и последната точка на отсечката.Това означава,че при изпълнението на тази команда плотерът:

1. Дига молива
2. Придвижва се от текущата точка до първата точка от отсечката
3. Спуска молива
4. Придвижва се до втората точка на отсечката

Другия тип команда за рисуване на отсечка е с относителна адресация т.е. в командата е описан само векторът,по който трябва да се придвижи молива от текущата точка.

По този начин една група команди от езика на дисплейния файл са команди за придвижване на молива включително командата за вдигане и слагане на молива.Друга група команди са свързани със смяната на молива.Това означава,че имаме възможност един и същи графичен примитив да бъде изрисуван с различни визуализационни характеристики(дебелина,цвят).

Върху тези две групи команди обикновено могат да се направят попдпрограми,записани в постоянната памет на плотера,чрез които се рисуват по-сложни графични примитиви-графеми,символи.

Рулонният плотер се различава от плоския само по това,че средата му е лист с „безкрайна” дължина.Гредата му е неподвижна ,а движението по оста у се реализира чрез придвижване на средата за рисуване посредством въртене на барабана.Принтерите също са графопостроители,произвеждащи твърдо копие.Терминът „твърдо копие” обикновено се свързва с невъзможността да бъде поправено част от изображението,т.е. при всяка промяна изображението трябва да бъде прерисувано изцяло.От тази гледна точка електронно лъчевите тръби без регенерация на изображението също са графопостроители,произвеждащи твърдо копие.Но възможностите на ЕЛТ ще обсъдим като цяло като обсъждаме графопостроителите,подходящи за диалогова работа!

Графопостроители,подходящи за диалогова работа-необходимо е произвеждането на изображение да не влияе върху работата на художника/ върху диалога между потребителя и системата.

Представител на този тип графопостроител са ЕЛТ.Една ЕЛТ представлява стъклена вакуумирана бутилка с форма на круша.Предната страна е фосфорициращо вещество,наречено екран:има 2 характеристики-време за послесветене и цвят. Теоретично това означава,че приемайки квант светлина или осветено за определено време ,то поглъща светлината,след което започва да я излъчва с определена интензивност,определяща времето за светене с определен цвят.

Електронна пушка-произвежда облак от електрони.Този облак бомбардира екрана,в резултат на което целият екран примигва хаотично.

Фокусиращи лещи-те концентрират облака в тънък лъч електрони.Този лъч е пишещ инструмент.В тази ситуация образът е точка в центъра на екрана.

Усилватели за хоризонтално и вертикално отклонение-чрез тези устройства се постига движение на пишещия инструмент,т.е. при тези ЕЛТ има ограничение върху сложността на рисуваното изображение(броя отсечки),тъй като ако броят им е твърде голям при постигане на рисуване на н-та отсечка първата ще е избледняла.

Поради това обикновено се взима твърде голямо време за послесветене от порядъка на 1-2 часа,но това пък поражда проблеми с малки промени в изображението,тъй като изтриването на изображението изисква време и е съпроводено с неприятни за зрението ефекти.Поради тази причина в практиката са се наложили ЕЛТ с регенерация на изображението.Това означава,че:

1. Времето за послесветене е от порядъка на 1/25 от секундата
2. Електронният лъч обхожда екрана по строго определен ред
3. Екранът е покрит или изграден от матрица от точки луминофор.Обикновено лъчът обхожда тази матрица ред по ред отгоре надолу.

Тъй като времето за обхождане е константно,то независимо колко сложно изображение ще бъде произведено,то ще се рисува за едно и също време.Обикновено размера на тази матрица се нарича разширителна способност на екрана.Могат да бъдат възпроизведени 2м\*н черно-бели изображения.Много се използва недостатъкът на човешкото зрение.Такова изображение обикновено се записва като низ от битове като изображението между тази матрица и този низ е биактивно.

Ако можем да променяме интензитета на лъча можем да постигнем монохромно изображение.

За да постигнем цветно изображение всяка точка луминофор е заменена от три точки с различен цвят.Електронният лъч е тройка успоредни лъчи.По този начин в битовата карта всеки бит се заменя с три бита.В най-простия си вариант цветните ЕЛТ могат да произвеждат 8 цвята.Обикновено за добра визия битовата карта се рисува поне 25 пъти в секунда.

За нашите нужди отсега нататък под графопостроител ще разбираме

1. Плотер
2. ЕЛТ с регенерация на изображението

Диалогови устройства-чрез които се въвеждат данни и се подпомага диалога

Различаваме различни диалогови устройства от гледна точка на типа данни,които се въвеждат чрез тях:

1. Превключватели(0,1)-чрез които се въвеждат цифрите 0/1.Типичен представител са клавишите на клавиатурата.Обикновено клавишите се делят на две групи:с моментно и продължително действие.Моментно действие е равнозначно на неравноправни състояния,а продължителното на равноправни
2. АЦП(0,1,2…)-въвежда се число от някакъв интервал
3. Сензитивна мрежа(0,1,2…)(0,1,2…)-въвеждат се наредени двойки числа

Обикновено ВУ на компютъра са комбинация от тези основни типове устройства,например мишката е 2\*1 +2\*2.

Дисплейни системи-една типична дисплейна система се състои от

1. Изходно устройство-графопостроител ЕЛТ с регенерация на изображението
2. Входно устройство(мишка)
3. Дисплейна памет(видео карта)
4. Дисплеен процесор

Функциите на дисплейния процесор са:

1. Да рисува съдържанието на дисплейната памет със скорост 1/25 от секундата
2. Да следи за промяна на състоянието на ДУ
3. Да рисува върху екрана специално изображение,наречено показалец(курсор).Местоположението му се управлява от мишката и се нарича текуща екранна точка.

Взаимовръзката на дисплейния процесор и централния процесор е от 2 типа:

1. Централният процесор може да промени незабелязано за дисплейния процесор съдържанието на дисплейната памет
2. Дисплейният процесор информира централния за настъпило събитие(промяна на състоянието на ДУ,реакцията на което не е в компетентността на дисплейния процесор.

Под дисплеен процесор ще разбираме част от диалогова система.

# 4.Методология за графично моделиране

Съвкупността от няколко виртуални изчислителни машини(Програма с явна функционалност), взаимовръзката между тях,както и начините за изграждане и поддържане на диалогова система за работа с графична информация наричаме методология за графично моделиране.

Изграждането и поддържането на диалогова система е предмет на ръководството за упражнения.

Състава и структурата на диалоговата система е предмет на лекциите.

Съставът на диалоговата система е следният:

1. Дисплеен процесор-непосредствена софтуерна обвивка над диалоговата система.Има две основни задачи:

А)да преобразува дисплейния файл в битова карта

Б)да разпознае указаната текуща точка от екрана

1. Геометричен процесор-функционалността му е да реализира геометрични преобразувания и операции върху геометрични обекти
2. Структурен процесор-неговата основна функция е да поддържа структурата на изображението,такава,каквато я желае потребителят(чрез командите групиране и разгрупиране)
3. Семантичен процесор-неговата функция е да свързва/групира неграфичната информация с графичните обекти
4. Диалогов процесор-функцията му е да разпознава заявките(командите на потребителя) и да активира дейност по тяхното изпълнение

Би трябвало да разглеждаме потребителят на една такава система и като ВИМ(процесор).Тези 6 процесора могат да бъдат свързани по различен начин,но за нашите нужди ние избираме циклична взаимовръзка.

Връзката от диалоговия процесор до потребителя наричаме права,а обратната-обратна.По обратната връзка потребителят задава команди или заявки,а по правата връзка получава резултатите.

# 5.Изображение.Дисплеен процесор.

От гледна точка на техническите средства на КГ можем да формулираме 2 дефиниции за понятието изображение.

Дефиниция 1 Изображение наричаме подмножество на средата за рисуване-векторна графика

Дефиниция 2 Изображение наричаме описанието на конкретно състояние на средата за рисуване-растерна графика

Дисплейният процесор е софтуерната обвивка на диалоговата система.Основни функции на дисплейния процесор са :

1. По правата връзка-да конвертира,преобразува,преведе векторното изображение в растерно
2. По обратната-да разпознае посочената точка от растерното изображение от кое(структурирано)подмножество на векторното изображение е

Растерното изображение се дефинира в първи квадрант на целочислената равнина в лявоориентирана координатна система.Наричаме област на изобразяване.Едно представяне на векторното изображение е дисплейният файл.Той е линейна последователност от команди за:

1. Команда за установяване на статус-всички команди,които описват геометрично подмножество след тази команда до достигане на друга подобна команда притежават указания статус.Пример за статус е забрана на за работа(посочване) с изрисуван върху екрана примитив.
2. Установяване на визуализационни характеристики-до срещане на следващата команда от този тип всички множества притежават тези характеристики.Пример-цвят,дебелина на линия.
3. Рисуване на графичен примитив-такова подмножество,което съответния графопостроител може да изрисува
4. Графичен обект-под графичен обект разбираме съвкупност от графични примитиви,която за потребителя е неделима или едно цяло
5. Символ-под символ разбираме подпрограма на дисплейния файл,която завършва очевидно с команда за възврат.Символът не трябва да съдържа команди за статус и визуализационни характеристики.Те се наследяват от графичния обект,активирал тази подпрограма.Изображението,получено от активирането на такава подпрограма се нарича екземпляр на символ.

Дисплейният файл се изпълнява команда по команда като командите за графичен примитив произвеждат множество от точки,които принадлежат на този примитив,но имат целочислени координати ,и ги записва в битовата карта.Този алгоритъм за апроксимация на графичен примитив не е лесна математическа задача.Обратната връзка,указана точка от растерно изображение се разпознава като точка от графичен примитив по следния алгоритъм:

Дисплейният файл отново се изпълнява команда по команда,но вместо произведеното множество от точки за примитив да се запише в битовата карта СЕ ПРОВЕРЯВА ДАЛИ НЕ СЪДЪРЖА УКАЗАНАТА ТОЧКА.За един начинаещ програмист е ясно как оттук може да се изведе точния адрес на указания графичен примитив в дисплейния файл.

# 6.Геометричен процесор

Не е задължително ГП да бъде обособен като ВИМ,но за нашите нужди го разглеждаме като такъв.По аналогия с ДП той има задача,извършвайки геом.преобразувание и операции да преобразува някакви структури от данни в дисплеен файл от каквато подструктура на данните е бил произведен.Концентрираме нашето внимание върху две задачи,които са части от алгоритъм.Въвеждаме понятието World Domain(WD),което е реалното двуизмерно множество с дясноориентирана декартова координатна система.Потребителят в своите мисли работи в WD.Векторното изображение е подмножество на WD.В учебника са описани представени в WD графични примитиви точка и вектор ,както и начинът за реализация на геометричното преобразуване (проекция) смяна на големината на единична отсечка.Графичните примитиви се представят чрез хомогенни(примитивни) координати,които са вектор ред като третата координата за точка е различна от 0,а за вектор е 0.Преходът към декартови координати се нарича нормализация на представяне.Това представяне,умножено отдясно със съответната трансформационна матрица реализира описаните геометрични преобразувания като съответните подматрици имат съответния смисъл:чрез матрицата АБЦД се описва ротацията,мащабирането и огледалното отражение.Чрез матрицата МН се описва транслацията.Чрез РО се описва преобразувания,като ако те са различни от 0 резултатът трябва да се нормализира и чрез R се реализира смяна на дължината на единичната отсечка,но в учебника е обобщено мащабиране.

Освен геометрични преобразувания ГП решава още две задачи,свързани с прехода на WD към SD.За този преход се въвеждат още два прозореца:

1. В екранната област на прозореца се нарича отверстие на гледане и от съображение за простота и скорост при изчисление този прозорец е прав със страни,успоредни на координатните оси
2. Прозорецът се нарича прозорец на гледане и пак от съображения за простота и скорост е центрирано в началото на координатната система –прозорецът е със страни, успоредни на координатните оси

Важно е двата прозореца да бъдат подобни фигури.Ако не са,ще се получи деформация на изображението.Двете задачи,които ГП трябва да реши са:

1. Отрязване по WWD
2. Налагане на WWD върху WSD

В нашите термини това означава да открием графични примитиви или части от тях,които са изцяло в прозореца на гледане(точката е граница на отсечката) трябва да бъде подложена на последователност от геометрични преобразувания:

1. Мащабиране с мащабен множител (М/А,Н/Б,1)
2. Отрязване на дробната част
3. Преминаване от дясноориентирана към лявоориентирана координатна система
4. Позициониране върху отверстието на гледане

# 7.Структурен процесор

Основна функция на структурния процесор е да поддържа структурата на изображението такава,каквато я желае потребителят.Има много съображения,най-вече като мислене на преобразуването,според което в структурата на изображението се налага изискването да бъде йерархия от множества,като под множество се разбира графичен обект,който е изграден от графични примитиви и/или други графични обекти.За да се поддържа тази йерархия от множества е достатъчно структурния процесор да поддържа дадената структура „ацикличен ориентиран граф”(„дърво”).Това означава,че в графа няма цикли и само един възел няма предходник,т.е той е корен на дървото.Листата на този граф са възли,които нямат наследници,описват графични примитиви и в такъв смисъл най-ниското ниво на графа можем да наречем геометрично.Ако тази даннова структура се поддържа и от семантичния процесор,то над геометричния слой могат да бъдат различени още два слоя-семантичен и структурен.За нашите нужди ние сливаме семантичния и структурния слой и казваме,че структурния слой съдържа и възли,описващи имена на графични обекти.По правата връзка този ацикличен ориентиран граф би трябвало да се преобразува в някаква структура от данни,с която работи геометричния процесор,а по обратната връзка да бъде разпозната посочената структура и да бъде преобразувана до граф.

Тези задачи се решават с обхождане на ацикличен ориентиран граф.Ако се вземат предвид и възможностите на дисплейния файл обикновено върху геометричния и структурния слоеве могат да се наложат съглашения с цел по-бързо получаване на дисплеен файл с добри характеристики като скорост на рисуване.Върху връзките и възлите на структурния слой може да се наложи някаква типизация.За нашите нужди е необходимо да наложим понятието рамка=(NWD,WSD).Това позволява едно и също изображение(различни части от него) да бъдат изобразявани едновременно в различни отверстия.Понятието рамка може да бъде включено в ацикличен ориентиран граф.Задачата за указване(разпознаване) на посочена рамка се свежда до разпознаване на указана точка на кое отверстие на гледане принадлежи .В системата Windows всеки прозорец е графичен обект,чието работно поле е отверстие на гледане в някаква рамка.

# 8.Смислов процесор/семантичен процесор

Основна функция на семантичния процесор е да придаде смисъл на изображението.Това означава да свързва неграфичната информация с части от изображението и/или да преобразува неграфичната информация в графично изображение.За нашите нужди считаме,че семантичния процесор е този,който поддържа имена на части от изображението.От по-обща гледна точка всеки полезен модел на реалността е математически.Всеки математически модел има геометрична интерпретация.Всеки геометричен модел може да бъде преобразуван в графично изображение.От тази обща гледна точка семантичния процесор като същност би трябвало да притежава следните три основни функции:

1. Да представя математически обекти
2. Решаване на математически задачи
3. Визуализация на математически обекти

Налага се да направим следното уточнение:от гледна точка на практиката под СИСТЕМА ЗА ГЕОМЕТРИЧНО МОДЕЛИРАНЕ се разбира ДП и ГП,а структурния,семантичния и диалоговия процесор обикновено се разглеждат като надстройки(обвивки),съобразени с конкретната приложна област в тази система.Например ОС Windows можем да дефинираме като диалогова система за работа с графична информация,чийто семантичен процесор изпълнява функциите на ОС.

# 9.Диалогов процесор

Диалогов процесор е виртуална машина,която се явява партньор на потребителя.Този процесор инициира правата връзка,след като е разпозната заявката на потребителя от обратната връзка.Диалог наричаме алтернативно сменяща се последователност от права и обратна връзки.

Това означава,че потребителят трябва да знае езика на системата ,а не обратното.Това разбиране за диалог обикновено се реализира чрез 4-стъпковата схема:

Избор

Подсказка

Изчисление

Ехо

Първоначално по правата връзка се извежда подсказка и потребителят прави своя избор.Изборът означава заявка за изпълнение.Тя се разпознава от диалоговия процесор,който активира съответното изчисление за реализацията и́.Резултатите от това изчисление се предоставят на потребителя по правата връзка във вид на ехо.Например ако вие кликнете върху икона,системата реагира на вашето кликане чрез ехото инвертиране на цвета на иконата.Ако езикът за диалог е автоматен,то обикновено изчислението привежда схемата в ново състояние,в което потребителят има някакви възможности за заявка,.Обикновено в този случай ехото и новите възможности се включват в подсказката ,т.е. започва нов цикъл.За да видим реализационно как става диалога трябва да обсъдим три понятия:

1. Устройство за диалог
2. Събитие
3. Рамка

Устройства за диалог- така наричаме диалогово устройство на компютър.Едно устройство е пасивно,ако може да бъде точно в едно от няколко равноправни състояния.Промяната на състоянието на такова устройство не носи информация на диалоговата система,но трябва да бъде прихванато и обработено от дисплейната система.Устройството за диалог е активно,ако може да бъде в точно едно от две неравноправни състояния(например бутона на мишката)Промяната на такова устройство носи информация на диалоговата система.Събитие за диалоговата система е промяна на състоянието на активно устройство.Това събитие се прихваща от дисплейната система хардуерно и по обратната връзка достига до диалоговия процесор във вид на указан обект или част от изображение.Съгласно езика на диалоговия процесор тази посочена част от екрана + състоянието,в което се намира системата са достатъчни за диалоговия процесор да реши какво изчисление да извърши и какво ехо да произведе по правата връзка.Работата с рамки може да бъде обособена като подфункция на диалоговия процесор,тоест част от диалоговия процесор да бъде рамков процесор.За нашите нужди е достатъчно да изясним как може да бъде открита посочена рамка.Това става чрез т.нар. алгоритъм на Худ,който е част от Windows.За простота предполагаме,че имаме списък от рамки,които се реализират,като се рисуват последователно на екрана.Това означава,че списъкът от рамки се обхожда в ред,обратен на рисуването и първата рамка,на която принадлежи указаната точка е рамката,посочена от потребителя.

В съвременните системи за графично моделиране диалоговия и семантичния процесор реализират т.нар. графичен интерфейс.Това понятие включва основни 3 характеристики:

1. Възможност за комуникация човек-компютър през прозорец(прозореца в Windows отверстие на гледане,което е работно поле на прозореца в системата)
2. Възможност за задаване на команда чрез посочване на пиктограма(картинка на графичен обект)
3. Възможност за манипулация и работа с графични обекти,наречени прозорци

# 10.Стандартизация на графичен вход/изход

С този въпрос обсъждаме преносимостта и независимостта на диалоговата система от гледна точка на свързаността на и́ с техническите устройства на КГ.За да се постигне тази независимост обикновено се реализират т.нар стандарти.Стандартът е абстрактен модел на някакво устройство,което реалните/физически устройства материализират.За нашите нужди различаваме три типа стандарти:

1. За графичен изход
2. За графичен вход
3. За графичен обмен

Под стандарт за графичен изход се разбира виртуален графопостроител,притежаващ възможностите на реалните графопостроители,които се намират в момента на пазара или ще се появят в обозримо бъдеще.Обикновено тези стандарти се реализират като библиотеки от подпрограми към език за програмиране,т.е. графичните примитиви са достъпни във вид на подпрограми,чието извикване изрисува съответния графичен примитив.

Стандарт за графичен вход-обикновено се разбира виртуално диалогово устройство,притежаващо възможностите на диалоговите устройства към настоящия момент или обозримото бъдеще.Обикновено тези възможности се класифицират от гледна точка на вътрешните системни функции.От тази гледна точка в GKS се въвежда понятието логическо устройство,което се реализира като подпрограма като се различават 3 основни и 2 спомагателни устройства.Основните са:

LOCATOR,който дава /връща координатите на посочената екранна точка(пиксел)Т.е.,това е т.нар позициониращо устройство

Следващото е PICK-то връща името на указан обект

CHOICE –връща цяло число от зададено предварително множество цели числа.Чрез това логическо устройство се реализира избор.

VALUATOR 5->’101’

String 5->’5

Обмен(файлов формат)-целта е да се обменя информация,данни между диалогови системи за графично моделиране с различна предметна насоченост.Всяка диалогова система да поддържа конвертиране от/към други диалогови системи.Обикновено собственото представяне на графична информация се обозначава като файлов формат,тъй като се разнасят чрез различни носители.

Геометрично моделиране

# 1.Подход за геометрично моделиране

Термина моделиране-ще обсъдим класическия за информатиката модел на дейността моделиране,който е атрибутивен.Разглеждаме конкретния човек като мисли и всичко останало извън тези мисли наричаме реалност.Части от тази реалност човекът възприема в мислите си като система от свойства.За нашите нужди казваме,че той ги възприема като съвкупност от свойства.Няма да дефинираме понятието свойство,тъй като тази дефиниция е извън областта на информатиката,т.е всеки има интуитивна представа за това що е свойство,но поради различни съображения,човекът знае,че обектът на неговите мисли притежава и други свойства,които не са достъпни непосредствено за него и/или техните стойности не могат да бъдат изчислени/изведени от познатите му свойства.Това означава,че обектът А е непредсказуем за човека от гледна точка на използването му за някаква цел.В тази ситуация човекът открива или създава друг обект от реалността(означаваме с Б),който притежава свойствата на обекта А и освен това недостъпното в обекта А свойство е достъпно в обекта Б.В тази ситуация изчисленото в обекта Б свойство се пренася(в мислите) върху обекта А,т.е. чрез тази дейност обектът А става използваем.При тази дейност обектът А се нарича обект-оригинал,а обектът Б - обект-модел(на А,по отношение на целта на моделиране).Дейността по създаването и използването на модели се нарича моделиране.Тъй като реалният обект А притежава много повече свойства,отколкото тези,които са необходими на човека-моделиер,то атрибут наричаме само онези свойства,които са необходими за конкретното моделиране.Различаваме атрибутна стойност и атрибутен клас.В термините на релационните бази данни(релационна таблица) атрибутна стойност е стойността на конкретно поле от тази таблица.Типът на колона от релационна таблица( множеството от стойности,които могат да приемат полетата от тази колона) наричаме атрибутивен клас.Синоним на термина атрибутен клас е математическия термин измерителна скала.В термините на програмирането-тип величина.Един ред от релационната таблица(запис ) всички полета на които са остойностени,се нарича представяне(модел) на единичен реален обект.Отговорът на въпроса защо смятаме,че изчисленото в модела свойство се приближава към оригнала,защото Б е модел на А,това е тълнодизъм -словоформа,която за говорещия може би има смисъл,а за слушащия,който притежава необходимите знания и владее езика на говорещия е неразбираемо,защото това е потвърждавано многократно от дейността на моделиера и теорията ,въз основа на която той извършва своята дейност.

Казаното дотук е извън областта на информатиката.Като модел на казаното би могъл да бъде възприет машинният език на един компютър.В рамките на това съглашение ще обсъдим един език за програмиране от високо ниво,който обозначаваме като подход за атрибутивно геометрично моделиране.Като първа стъпка се създава системен модел на изследваната реалност.Този системен модел като същност е съвкупност от интересуващите ни атрибутни класове и взаимовръзките между тях.2-ра стъпка-система от атрибутивни класове.Езикът,на който се материализира този модел е строг език,но не е необходимо да бъде формален език.Например подмножеството на естествения език ,в което липсват двусмислия.Като част от този подход на системния модел се съпоставя математически модел.Това е моделът,който гарантира изчислимост ,т.е постигане на целта на моделирането.Езика, в който се описва математическия модел е езикът на математиката.

Като 3-та стъпка на математическия модел (който е алгоритъм,чрез който се постига решение на някаква задача)се съпоставя информатически модел.Това е програма(алгоритъм +структура от данни).Данните са представени на атрибутни стойности,а алгоритъмът описва начина за изчисление на интересуващата ни атрибутивна стойност.Езикът,на който се описва информатическия модел е език за програмиране.От гледна точка на геометрично моделиране ни остава да уточним онези атрибутни класове,формиращи системен модел,чиято съвкупност обикновено означаваме като геометрия и реалните триизмерни твърди тела.Във въпрос №2 обсъждаме системния и математическия модел,необходими за геометрично моделиране.Във въпрос № 3 обсъждаме възможностите на математическия модел.Във въпрос №4 обсъждаме прехода от математически към информационен модел като се акцентира върху данните,т.е. като се нарича представяне на геометрична информация за реално триизмерно твърдо тяло,а всички други въпроси касаят конкретни схеми за реализация на този преход.

# 2.Геометрична информация

Обсъждаме проблема за изясняване на множество от атрибутни класове,които в нашите мисли наричаме геометрия на реални твърди тела.В исторически план първо е създаден и използван математическия модел на РТТ и чак след това в по-ново време е формиран в явна форма системния модел.Преди всичко,всеки човек свързва с реалността понятието пространство.От тази гледна точка математическия модел на реалното пространство е евклидово пространство с дясноориентирана декартова координатна система.От тази гледна точка модели на реалните твърди тела са множества от точки.Геометричното място на точки от евклидовото пространство,които се намират на разстояние по-малко от някакво число r от една конкретна точка(център).Тези точкови множества се характеризират от следните атрибутивни класове:

1. Местоположение-от гледна точка на математиката това е местоположение на конкретна точка(която е асоциирана с множеството и не е задължително да бъде от него) и се нарича полюс
2. Ориентация в пространството-в математическия модел обикновено това е локална координатна система,чийто център съвпада с полюса и относно която се извършва описанието на множеството
3. Метрика(размери)в нашата дефиниция размерът е радиуса.Най-общо казано под метрика се разбират разстоянията или ъглите между някакво избрано крайно подмножество от точки на описаното множество
4. Форма- дефиниция на диск наричаме множеството от точки/корените на неравенството (x-а)2+(y-b) 2-R2<=0 Тук (а,б) е полюсът R-размера.Формулата наричаме форма на множество от точки,които в разговорен език наричаме диск(х,у)-форма.

Съвкупността от 4-те атрибутивни класа<местоположение><ориентация><метрика><форма> наричаме геометрична информация за реалното твърдо тяло диск,което в реалността има някаква пренебрежимо малко дебелина,а в съответния математически модел дискът е двуизмерен.От втората дефиниция за диск следва ,че представянето на геометрична информация на диск удовлетворява целите на моделирането(например визуализацията на диска) само ако формата му разрешава чрез включване в неравенството да се определят корените на това неравенство.От гледна точка на информатическия модел реализацията на такова направление като проверка дали дадена точка принадлежи на съответното множество се нарича предикат за принадлежност на точка към множество.

Извод:За да бъде едно представяне на 4те атрибутивни класа геометрична информация е достатъчно за нашите нужди от това представяне да е възможно да се реализира предикат за принадлежност.В този случай представянето е модел на твърдо тяло .Възниква въпросът кои подмножества на Евклидовото пространство могат да бъдат използвани като модели на реалните тела.За да изясним това е необходимо да определим кои качества на части от реалността формулират в нашето мислене понятието РТТ.Такива качества са:

1. Отделимост
2. Ограничимост - в термина на математиката всяка околност на множество сфера с крайни граници,на чиято вътрешност тялото принадлежи
3. еднозначност на границите-нещо,което разделя пространството на 2 части-вътрешност и допълнение
4. хомогенна триизмерност –за всяка вътрешна точка съществува околност,която няма сечение с границите
5. свързаност-за всеки две вътрешни точки съществува път или крива между тях,която не пресича границите
6. твърдост-неизменност на формата и размерите по отношение на някои въздействия върху множества от точки например по отношение на ротацията и транслацията всяко множество е твърдо
7. крайност на описанието- в геометрията на всяко РТТ е възможно да бъде описано чрез краен брой знаци от някаква азбука по отношение на така формираните от качества математически модели на реалните тела

По дефиниция R множествата са ограничени,затворени (и от практически съображения полуалгебрични) подмножества на евклидовото пространство.Най-простите примери за такива множества са произволни многостени с равнинни лица.

# 3.Математическо пространство

Класическите задачи на аналитичната геометрия са да се изследват свойствата на R множествата с известна форма.Целта на геометричното моделиране е да се конструират/изграждат множества с нова форма.От тази гледна точка нас ни интересува какво предлага математиката като инструмент за постигане на тази цел.Преди всичко,в математиката представянето на точка е или в декартови или в проективни(хомогенни)координати.Това означава,че точка се представя с наредени числа,където 4-тата координата е различна от 0,а векторите чрез наредена 4-ка числа,където 4-тото е 0.Преобразуванията върху тези числа се описват чрез матрицата Т

A b c p

D e f q

G h I r

N u l s

В която чрез подматрицата

A b c

D e f

M n I

Се описват геометричните преобразувания ротация около произволна права, мащабиране, огледално отражение относно равнина.Чрез (2)

p

q

r

се описва транслацията(преместване в права линия).Чрез 3 (m n l се описват проективни преобразувания,след извършването на които може да се получи проекция на точка върху равнина.Чрез (4) s се описва операцията смяна на дължина на единична отсечка на координатната система,което в учебника се нарича обобщено мащабиране,защото декартовата координатна система има една и съща единична отсечка за всички оси и да намалим дължината на тази отсечка означава за КГ да мащабираме с реципрочен> от s коефициент .Изпълнението на компютъра на геометричното преобразувание се реализира като вектор-редът точка или вектор се умножи отдясно с матрицата Т.

Тези геометрични преобразувания могат да се реализират по аналогичен начин и върху някои прости точкови множества,например равнина.,чрез вектор стълб от коефициенти може да бъде подложено на същите преобразувания,ако представянето му се умножи отляво с обратната на Т.

По отношението на генерирането на нови форми в геометричното моделиране най-често се използват следните математически операции:

1. Теоретико-математически операции- обединение,сечение,разлика.Класическите ТМО не са затворени по отношение на 3-мерна хомогенност на резултата.Поради това в геометричното моделиране се използват регуляризирани ТМО,които гарантират 3-мерна хомогенност,но не и свързаност на резултата
2. Слепване-вариант на обединение,за което се изисква слепваните тела да нямат сечение на вътрешните си и сечението на границите им да бъде хомогенно двумерно
3. Запълваща позицията-обикновено на всяка точка от пространствата,които са принадлежали на дадено множество по време на неговото движение.Най-прост пример е векторното параметрично уравнение на отсечка,използвано за тази операция.

# 4.Схеми за представяне на геометрична информация

Представяне на геометрична информация наричаме всеки низ от знаци,породен от дадена формална граматика.Съвкупността от всички такива низове наричаме език,породен от дадената граматика и за нашите нужди този език наричаме пространствено представяне(геометричен обект в аспект от данни)Езикът е структура на данните,а всяко конкретно представяне конкретна структура от данни.От гледна точка на математиката и математическия модел е съвкупност на всички възможни R множества,които могат да бъдат конструирани чрез някакви правила за използване на математически операции.

Формална граматика в математиката-правилата,по които на R се съпоставя представяне (R множеството се изобразява като представяне)се нарича схема за представяне.

Свойства на схемите за представяне от гледна точка на геометричното моделиране:

1. Мощност на схемата-мощността на множеството от подмножества,които могат да бъдат представени и описани чрез тази схема
2. Валидност на схемата за представяне-ако всички нейни представяния в информатическия модел са действителни едно представяне е действително,ако описвано представяне R множество.Обикновено е много трудно да се създаде език(граматика),чийто представяния да са валидни.От тази гледна точка всички схеми като области от стойности имат някакво подмножество от области от езика на информатическия модел.

Правота за R множество на всички действителни представяния наричаме още семантика или смисъл на това представяне.Други свойства:

Еднозначност-схемата за представяне е еднозначна,само ако е от тип 1:1

Недвусмисленост 1:Н-ако е от този тип 1 множество може да се представи чрез няколко различни представяния като тези представяния се наричат синоними.

Схемата притежава свойството двусмисленост,ако е от типа М-1,М-Н-тогава за нуждите на ГМ двусмислените схеми не са полезни.Тези свойства се наричат още формални свойства. И подлежат на компютърни изчисления.Неформални свойства са сбитост(касае размера,дебелината, броя знаци) на действителните представяния на схемата.Обикновено много сложните схеми за представяне на състоянието на представяне е трудно,но изчисляването на свойствата на това представяне е бързо.Това свойство касае размера на данните,които използва една програма.

Ефективност в приложенията-касае проблема колко бързо се изчислява желаното свойство от съответното представяне.При неефективните схеми обикновено представянията са сбити,в термините на програмирането ефективността се свързва с броя на елементарните(процесорни) операции,които трябва да се извършат за всички изчисления на дадено свойство.

Лекота на създаване на представянето-най-общо това е доколко лесно от някакво представяне на R множество(дали в мислите на човека или в някакъв документ) може да бъде създадено валидно/действително представяне за нуждите на моделирането.

Съществен е проблема с конвертирането на представяния.Това е проблемът за точен превод от един език на друг.Ясно е ,че преводът е точен,ако прообразите на входното и изходното представяне съвпадат.В противен случай преводът е приближен.Някои приближени преводи наричаме неточни.

# 5.Схеми с екземпляри на чисти примитиви

Това е основополагаща схема за представяне не само в областта на ГМ.Чист примитив наричаме знанието за изчисляването на конкретно свойство от представяне на R множество.Обикновено тази схема е достъпна като библиотека от подпрограми,всяка от които:

1. Името и по някакъв начин отразява най-малко свойството,което изчислява(върху какво представяне) на какъв тип R множество
2. Като входни параметри фигурират представянията на атрибутните класове на геометричната информация за съответното R множество
3. Като изходен параметър обикновено фигурира само величина за изчисленото свойство,но е възможно списъкът от изходни параметри да бъде разширен в аспект точно описание на ситуацията,невъзможно за изчисляване

Всяко изпълнение на такава подпрограма,респективно резултатът,който се получава се нарича екземпляр на чист примитив .От гледна точка на ГМ под чист примитив ще разбираме представяне на геометрична информация + зададен в явна/неявна форма предикат за принадлежност.

# 6.Схеми”изброяване на заетото пространство”

В исторически план тази схема се използва за нуждите на материалознанието. В този си вид схемата гласи:под представяне на R множество се разбира крайно подмножество от точки на описаното R множество,което притежава желана(хомогенна) гъстота(близост на точките).Предикатът за принадлежност гласи:”Една точка принадлежи на описаното множество,ако съществува точка от представянето,чиято околност(сфера с предварително известен радиус) съдържа класифицираната точка.”Казаното означава,че се използва топологичен подход,който от гледна точка на геометричното моделиране не удовлетворява свойството хомогенна тримерност(гарантирано представяне).Това е причината в литературата по ГМ да се използва изчислително-геометричен подход.Това означава,че понятието околност се замества с понятието „клетка”(примитив),като полюсите на клетките са точките от описания топологичен подход.При тези съглашения схемата гласи:”Представянето е списък от еднородни клетки,чиито полюси принадлежат на описаното R множество,нямат сечение на вътрешностите си,а слепването им гарантира хомогенна тримерност.За конкретност ще обсъдим една такава схема за описание(представяне) на двумерни R множества.Построено е евклидово пространство с дясноориентирана декартова координатна система.Предполагаме,че трябва да опишем R множество,позиционирано в първи квадрант.Само от съображение за изчислимост в термините на естествените числа чрез въвеждане на равномерна мрежа със стъпка единица,първи квадрант е покрит с клетки:

1. Имащи формата на квадрат
2. Имащи размерите 1
3. Имащи местоположение(полюс),дефиниран като точка от клетката,най-близка до началото на координатната система(долния ляв ъгъл)
4. Имащи ориентация еднаква с тази на координатната система(т.е като следствие от б))локалната координатна система е еднакво ориентирана с глобалната.

При съглашението,че клетка принадлежи на описаното R множество само когато нейния полюс принадлежи на това множество,като представяне се получава таблица от две колони,като всеки ред на тази таблица представя полюсната клетка,принадлежаща на описаното множество.При въвеждане на съглашение на понятието съседни клетки информатическото представяне може да бъде тестване за хомогенна тримерност(или в този случай двумерност).Предикат за принадлежност:

-върху действителното представяне точка принадлежи на R множеството ,ако принадлежи на някоя от описаните клетки.Ясно е,че при тази схема представянето е приближено(неточно)Повишаване на точността на представянето се постига чрез сгъстяване на мрежата,например чрез деление на отсечка на 2.Това означава,че дължината на представянето се учетворява.Поради тази причина се казва,че тази схема е сложна,т.е.,че не е сбита.Задачи,които можем да решим при такова представяне са:

1. Лице на описаното подмножество
2. Дължина на периметър
3. Изчисляване на броя на дупките

# 7.Схеми”Разбиване на клетки”

Като идея чрез тази схема се цели преодоляването на многослойността на схемата (6) чрез използването на повече от една клетка.Това означава описаното множество да бъде (за 2Д покрито,за 3Д уплътнено) с клетки(примитиви) от някакво предварително известно множество от типове клетки,така,че тяхното слепване(те нямат сечение на вътрешностите си) да гарантира хомогенната двумерност(тримерност).Като пример за двумерния случай можем да изберем клетките на играта тетрис.Задачата за покриване не е от лесните математически задачи.Като пример от реалността това е задачата на крояча,който разполага в парче плат някакво множество от клетки,наречени кройки.Целта му е

1. Да разположи всички клетки върху това парче плат
2. Да ги разположи така,че отпадъкът да бъде минимален(което означава клиентът да купи по-малко плат)
3. Да ги разположи така,че отпадъкът да притежава някакви допълнителни свойства (например размер,форма)

Обикновено този тип задачи се наричат задачи за оптимален разкрой и са предмет на математическото оптимиране.

Ако игнорираме сложността на получаване на представянето,то представянето е списък от клетки,който в термините на програмирането е таблица с 4 колони от следния тип:

1. Форма,където се записва името на клетката
2. Полюс
3. Ориентация-обикновено за нашия случай това е завъртане на 90 градуса
4. Метрика(размери)-на практика много често тази колона не се използва,тъй като „добрите програмисти” неясно защо приписват метриката към формата

Ясно е,че конвертирането(преобразуването) от (7) към (6) е елементарна задача.

Обратната задача е класическата математическа задача за оптимален разкрой.Това означава,че всички задачи ,които могат да бъдат решени върху представяне (6),могат да бъдат решени и върху представяне(7).

Предимствата на (7) пред (6) са

1. Представянето е по-компактно
2. Позволява осмислянето и решаването в явен вид на много реални задачи

# 8.Схеми”Кодиране с осмично дърво”

Тази схема цели преодоляването на общия на (6) и (7) недостатък,а именно многослойността и невъзможността да се работи с клетки с еднаква форма,но с различни размери.При тази схема клетката е една-куб като форма,но този куб може да има различно местоположение и размери.Предполага се,че ориентацията му съвпада с ориентацията на главната координатна система.На идейно ниво представянето е представяне на множество от кубове,чието слепване гарантира хомогенна тримерност на описаното множество.От тази гледна точка представянето би могло да бъде и таблица на схема (7),но теоретично пр. тази в представянето се кодира начина на изчисляване на клетките,формиращи представянето.

Ще обсъдим тази схема в двумерния случай,където схемата се нарича четвъртично дърво.В 3Д имаме 8 октанта,а в 2Д имаме 4 квадранта.Отново предполагаме евклидово пространство и квадрат с дължина 2к.Пак порадисъщите съображения така дефинирания квадрат е обвивка на R множеството,което ще описваме.Правим следните съглашения:ако 1 квадрат няма сечение с R множеството,казваме,че е празен.Ако квадратът е подмножество на R множеството,казваме,че е пълен.Ако имат сим. разлика класифицираме като частично запълнен.Първоначално квадратът е запълнен частично и в това състояние Р се съпоставя възел от едно дърво,което има 4 полета,съответстващи на 4-те подквадрата,което се получава чрез делението наполовина на квадрат,класифициран като Р.Поредността на тези полета във възела съответства на избран предварително начин на обхождане на получените 4 квадранта.Ние започваме от долния ляв и се движим обратно на часовниковата стрелка.Класифицираме всички получени подквадранти и записваме резултата в съответното поле на съответния връх от дървото-ниво 0 на дървото.За всеки възел от ниво к и всяко негово поле,съдържащо Р се генерира възел наследник от следващото ниво на дървото и за всеки такъв възел се повтаря обсъдения вече процес.Е и F нямат наследници.Този процес на строене на дървото продължава докато

1. Никое от полетата на възлите от текущо ниво не съдържа Р
2. Когато се генерират и остойностят всички възли от ниво к(полетата НА ВЪЗЛИТЕ ОТ НИВО К ПРЕДСТАВЯТ ИЛИ ОПИСВАТ КВАДРАТИ СЪС СТОЙНОСТ 1)

В четвъртичното дърво не присъстват в явен вид размерът и местоположението на квадрат,описан чрез даден възел на дървото,при което чрез извършване на следните две дейности

1. Деление наполовина на глобална величина,имаща стойност 2к при преход от ниво с по-малък номер към ниво с по-голям номер.Респективно умножаване по 2 при възходящ(обратен) преход
2. Използване на свойствата по отношение на полюса на възприетото обхождане на квадрантите при прехода от възел към възел,които са наследници на 1 и същ възел от предходното ниво ,за да изчислим полюса

Казаното ни дава възможност да определим алгоритъма на предиката за принадлежност:

Четвъртичното дърво се обхожда като всеки посетен възел(представяш/описващ) еднозначно квадрат с определено местоположение и размери при посещението на всеки възел се реализират предикат за принадлежност към такава клетка.Ако такъв предикат върне стойност истина,то обхождането се прекратява с тази стойност.Ако резултатът е лъжа,обхождането продължава.Приключването на обхождането на дървото означава,че класифицираната точка не принадлежи на описаното чрез дървото R множество.Конвертирането на четвъртичното дърво към(6) както и обратното не са трудни задачи.Какво означава конвертиране от осмично дърво към (6)-за всеки описан в дървото квадрат(куб),който има размери 2к,да се изчислят всички единични квадрати.Ясно е,че ако на ниво к има ,които са класифицирани като Р,то това представяне е приближено>Все пак ние не уточнихме алгоритъма,по който един квадрат се класифицира като F,E,P.това вече е въпрос на конкретната схема.Процедура по делене на единичната отсечка на 2 (което влияе само върху увеличаването на дълбочината на дървото с още едно ниво).Или се приема съгласно някакъв критерий,че това приближено представяне е достатъчно за целта на моделирането и класификациите Р се заменят е с една от другите.

# 9.Схеми”Конструктивна геометрия с твърди тела”

Това е схемата,при която се използват (регуляризираните) теоретико-множествени операции.За нашите нужди ще говорим само за обикновените ТМО-сечение,обединение и разлика.Те не гарантират 3Д хомогенност на резултата,но целта ни е да постигнем идеята за представяне.Класически идеята за представяне в тази схема е геометричен израз.Това е израз,изграден от представяне на примитиви(клетки) и ТМО.Резултатът от изчисляването на такъв израз би трябвало да бъде R множество.Но всеки израз може да бъде преобразуван във вид на дърво.От тази гледна точка в теорията на ГМ се казва,че представянето на тази схема са двоични дървета,които се наричат конструктивни дървета.Има различни варианти на тази схема.Например ако клетките са затворени множества, не са ограничени(например полупространства/равнини),то тази схема се използва за дисциплините линейно оптимиране и линейна алгебра.

За нуждите на ГМ най-вече в областта на машиностроенето т.нар CAM/CAD примитивите на R множеството с някаква обикновена проста форма,най-често наложена от гледна точка на възможността за автоматизирано производство. Пример:сфера,цилиндър,паралелепипед,пирамида

Всеки такъв примитив е описан като форма относно своя полюс,но полюсът,както и местоположението и размера най-често не фигурират в представянето.Идеята е,че примитивът с единични размери е разположен по някакъв начин относно главната координатна система и върху този начален примитив се извършват геометрични преобразувания, ротация, транслация, мащабиране спрямо глобалната координатна система на местоположение,ориентация и метрика.По този начин в конструктивното дърво се оформят няколко типа възли

1. Възли,описващи името на примитив
2. Възли,описващи пар. На геометричния примитив
3. Възли,описващи вида на геометричното преобразувание
4. Възли,описващи вида на ТМО

Възлите от а и б да листа на дървото,а В) и г) са всички останали възли.Този начин на фиксиране на R множествата позволява в конструктивното дърво да бъде отразен и закодиран начинът(логиката),по който се изгражда желания модел.Това е причината тази схема да се нарича още интерфейсна и да бъде предпочитана в системите за геометрично моделиране.В конструктивното дърво всеки възел за геометрично преобразувание или ТМО е корен от конструктивното дърво,описващо някакъв геометричен обект..Алгоритъмът за принадлежност на точка към множество конструктивно дърво е

1. Обхождане на ляв наследник
2. Ако е необходимо обхождане на десен наследник
3. Комбиниране на получените клас. резултати в следствие от изчисленията на клас. резултати в зависимост от типа и вида на посочения възел
4. Представяне на получения резултат към предходника на посетения възел

Казаното означава при посещение на възел,описващ примитив се изпълнява предиката за принадлежност на този примитив по отношение на класифицирана точка и този резултат се предава към предходника.При посещение на възел,описващ геометрично преобразувание-ако посещението е низходящо класифицируемата точка се подлага на геометрично преобразувание,което е обратно на описаното от възела и тази нова точка се предава към новия наследник.При възходящо посещение класифицираната точка се подлага на преобразувание ,описано във възела и получената точка заедно с класифицирания резултат се предава към предходника.При посещение на възел за ТМО съгласно правилото за обхождане се обхожда първо левия наследник,ако е необходимо след това десния,изчислява се резултатът от двете посещения,които заедно с резултат с е предават към предходника

Върху това представяне от практическа гледна точка е необходимо да бъдат решени още 2 задачи:

Опит за минимализиране на дървото,което означава,че от гледна точка на интерфейса конструкторът не е действал по най-оптималния начин

Откриване и отстраняване на поддървета,описващи празното множество.

# 10.Схеми”Запълваща продукция”

При схемите,обсъдени досега(без чистите примитиви)резултатът се получава мигновено-в смисъл,че някакво множество от примитиви мигновено се включва в някаква система и в този момент се получава и резултатът .От тази гледна точка тази схеми са статични.Запълваща продукция е динамична схема,защото резултатът се получава в резултата на действие(операция),протичащо за някакъв определен интервал от време.Най-простата дефиниция за тази схема е следата,която оставя движещо се в пространството тяло.От математическа гледна точка под следа се разбира множество от точки,които в някой момент от времето,през което се извършва движението са принадлежали на движещото се тяло.Идеята на запълващата продукция най-явно се характеризира чрез т.нар векторно параметрично уравнение на отсечка.

Теоретично се казва,че в тази схема представянето е съвкупност от

1. Представяне на тялото,което се движи
2. Закона за движение на тялото

Проблем е предикатът за принадлежност,защото той е свързан със съществуването и изчислимостта на закон,обратен на закона за движение.Поради това за практически нужди най-често се използва равномерно движение.В геометрията и аналитичната механика равномерното движение се нарича механично движение.Когато по време на движение тялото не променя своята форма или размери схемата се обозначава като Simple Sweeping.Но ако е възможно размерите(метриката) и/или формата на движещото се тяло да бъдат функция на времето,то схемата се нарича General Sweeping/обобщена запълваща продукция

В този случай се казва,че движението най-общо на тялото е суперпозиция от две движения

Деформиращо движение,което променя формата и/или размерите(резултат от което се подлага на механично движение)

Тази схема е много удобна/полезна за добрите математици,но е много неефективна в приложенията.

# 11.Схеми”Описание на границите”

Обсъдените дотук схеми(без схемата”чисти примитиви”) се наричат още обемни.Причината е,че ако точка принадлежи на тяло,то са необходими допълнителни изчисления,за да се определи дали тя принадлежи на граница на тялото и ако да,как принадлежи.Например какъв е нормалния вектор към границата в тази точка.Тази схема се нарича повърхнина,защото целта е да се опише границата на тялото,която е повърхнина в Е3.Това означава,че ако точка не принадлежи на границата на тялото,са необходими допълнителни изчисления,базирани върху някакво съглашение,за да се определи дали точката е вътрешна или не.Оттук нататък в лекциите ще използваме свободно термина тяло като синоним на R3 множество и термина повърхност на тяло,респективно като повърхнина в Е3.Налага се да обясним математическата причинност на тази схема.Границата на едно тяло може да бъде от един от следните два типа повърхнини

1. Сферичен тип-например повърхността на сфера или паралелепипед,т.е. тялото няма дупки
2. Тороидален - геврек или безкрайно дълъг цилиндър.Тези тела с к-свързани,като к е броят на дупките

Третият единствено възможен тип повърхнина,която разделя Е3 на две части е планарния тип повърхнина,например равнина,но двете части са неограничени множества,следователно не могат да бъдат използвани като модели на тялото,за да бъде една повърхнина от един от тези 3 типа,означава,че тя чрез непрекъсната деформация(мачкане на глина) може да бъде съвместена с някой еталонен представител на тези три типа.Например ,вземаме глинен паралелепипед или куб и чрез мачкане го привеждаме до сфера.Това означава,че повърхността на паралелепипеда е в сферичен тип.В математиката непрекъснатата деформация е изображение с определени качества,дефинирано върху плътно множество.Съглашението за вътрешност на тяло,на което познаваме само границата е следното:за всяка точка и за всяко тяло съществува равнина,инцидентна с точката,която разделя тялото на две непразни множества.Тогава всеки лъч,който е компланарен пресича границата на обекта четен или нечетен брой пъти(0 е четно).Под сечение се разбира интервал или точка(или списък от такива).Ако броят на сеченията с границата на обекта е нечетен,точката принадлежи на вътрешността,респективно ако е нечетен,точката е външна за обекта.Изричното условие е точката да не принадлежи на границата.Ние няма да обсъждаме тела,които имат кухини поради два факта

1. Границата на такива тела разделя Е3 на повече от две части
2. Границата на такива тела е несв. Множество
3. За такива тела може да се използва конструктивна геометрия

Трябва да обсъдим начин за представяне на повърхнини от сферичен и тороидален тип.От гледна точка на сложността на формите се е избрал подход за описание,базиран върху понятието лице.

Границата на някакво тяло е множество от лица(в информатиката списък от лица).Всяко такова множество,както и неговите елементи лицата трябва да удовлетворяват следните условия

1. Всяко лице е подмножество на границата,респективно повърхнината на описания обект
2. Обединението на всички лица трябва да съвпада с границите на обекта
3. Всяко лице е подмножество на примитивна повърхнина.Една повърхнина е примитивна,ако познаваме нейната форма.Обикновено като примитиви се избират онези повърхнини,за които формата и размерите на лицето и не зависят от местоположението и ориентацията му в повърхнината.Например лице върху сфера от технически съображения като примитив се използват равнината,сферата и цилиндъра(линейчати повърхнини)
4. Всяко лице е топологично еквивалентно на диск.

При тези условия б) не ограничава възможността лицата да имат сечение на вътрешностите си.Това означава,че имаме два различни типа гранично представяне:

1. Когато лицата нямат сечение на вътрешностите си се използва математическата операция обединение и обикновено такива схеми(но не само по тази причина) се наричат скулпторни,а лицата кръпки.ние няма да обсъждаме тези схеми.
2. Когато никои две лица нямат сечение на вътрешностите си.Това означава,че се използва частен случай на обединение,което ние нарекохме слепване.Тези схеми са предмет на нашето обсъждане

Освен това от съображения за яснота и краткост на изображението избираме примитивните повърхнини да бъдат равнини,а лицата-многоъгълници.Най-простият многоъгълник е триъгълник.От този момент ще обсъждаме представянето на многостени с равнинни лица.Прилагаме същите разсъждения по отношение на представянето на R множество-лица на многоъгълници.това означава,че границата на всеки многоъгълник е списък от отсечки.В термина на тази схема отсечките наричаме ребра.Всяко лице се явява списък от ребра.Прилагаме същия подход към R множествата списъка с ребра,които са отсечки,че границите на отсечките са точки,които в тази схема наричаме върхове.списъкът от точки наричаме геом. на границата на описано тяло,а останалите списъци и взаимовръзки между тях наричаме структура(топология) на границите на описваното тяло.От съображения за скорост на предиката за принадлежност към вътрешността на тяло обикновено чрез списъка на ребрата на едно лице на многоъгълник се постига някаква еднородна наредба на върховете на всички многоъгълници,т.е в това представяне се”скрива” наредбата на върховете на всяко лице.Ако съществува наредба и за n-мноогоъгълник имаме такава наредба,то лесно чрез съглашението може да бъде изчислен нормален вектор към примитивната повърхнина на това лице по такъв начин,че тези нормални вектори за всички лица да сочат вътрешността(респективно външността) на тялото.Това представяне е „теоретично” и може да бъде реализирано по много различни начини.Едно такова BR представяне е валидно,ако удовлетворява формулите

1. За сферичен тип [E]+2=[F]+[V]
2. За тороиден тип [e]+2=[F]+[V]+2[H]

Възможно разширение на тази схема е свързано с използването на специфични лица,наричани ципи(хралупи,бърлоги и др.).Границата за всяка точка от границата на такова лице всяка нейна околност съдържа точки от вътрешността на тялото,а за всяка от вътрешността на ципата съществува околност,която не съдържа точки от вътрешността на тялото.Ципите са технически реализационен инструмент,чрез който е възможно да се опишат(представят) тела с кухини,а границите им да бъдат свързано множество.

Схемата позволява всеки многоъгълник да бъде покрит с триъгълни клетки,като допълнителни ребра се наричат още фалшиви,защото по отношение на визуализацията на такъв обект,те не би трябвало да се визуализират.От тази гледна точка в представянето не може да има 2 съседни ципи и една и съща носеща примитивна повърхнина.

# 12. Хибридни(смесени схеми)

Дефиниция:Геометрично моделиране наричаме съвкупност от теории,методи и системи,фокусирани върху създаването на „информационно пълнене” представяния(модели) на реалните твърди тела,като тези модели позволяват да бъде изчислено(по възможност автоматично) всяко добре дефинирано геометрично свойство на реалните им праобрази на обекта-оринигнал(изображението на един обект също е негово свойство),като свойствата обем и лице на границата.От обсъдените от нас теми само схемата „чисти примитиви” е „чиста” схема,защото всички останали я използват в явен или неявен вид.Въпреки това всички тези схеми в теорията на геометричното моделиране се наричат чисти схеми.Може би причината е,че при обособяването им като схеми са се имали предвид предимно общите понятия математическа операция и вътрешност и граница на R множество.Но в практиката тези схеми твърде рядко се използват сами например CSE+Sweeping+BR.Казаното означава,че в практиката много често се конструират/създават нови схеми за представяне,в които се използват и други математически операции от гледна точка на дефиницията за ГМ,чрез тези схеми се цели

1. По-добро дефиниране на геометрично свойство
2. По-лесно изчисляване на добре дефинирано геометрично свойство
3. Това е причината-създаването на нови схеми във въпрос 4 да обсъдим формалните и неформални свойства на схемите за представяне

# 13.Системи за геометрично моделиране

Система за геометрично моделиране наричаме програма,притежаваща следната функционалност

1. Наличие(поддържане) на достатъчно количество интерфейси(схеми на представяне)
2. Наличие(поддържане) на достатъчно количество изчислителни схеми
3. Наличие на конвертори,чрез които е възможно конвертиране между всеки две от поддържаните схеми
4. Отвореност на системата в аспект,че множеството от схеми и конвертори между тях могат лесно и бързо да се променят

Това ни позволява оттук нататък да предполагаме,че съществува схема за ГМ,от която потребителите(човек или програма) могат да получат достатъчно бърз и точен отговор на всеки свой въпрос относно свойство на обект,независимо какво е представянето на обекта в термините(мислите) на потребителя.Това съглашение ни освобождава в трета част на конспекта да си мислим за визуализация на множества от многостени да обсъждаме алгоритмите за визуализация в термините на познати за студентите от курса по АГ операции с подмножества на Е3.

Алгоритми за визуализация

# 1.Обща постановка на задачата за визуализация

Алгоритъм за визуализация наричаме действие от преобразуване (проектиране) на пространствена сцена(множество от многостени) в графично изображение(равнинна рисунка),което създава у наблюдаващия го илюзия за тримерност.Съществуват две основни техники за постигане на илюзия за тримерност:

1. „чертожна” техника,основаваща се върху централното им перспективно проектиране и отстраняване на невидимите за наблюдателя ребра.Т.е. рисуват се само видимите,проектирани перспективно върху средата за рисуване.Развитие на тези 2 техники за бинополярно проектиране,при което се произвеждат 2 изображения,много близки едно до друго,по едно за всяко око и мултипликацията,свързана с произвеждане на снимки(изображения) на движещ се обект(обикновено въртяща около ос със скорост между 25 и 30 изображения в секунда.
2. Произвеждане на фотореалистични изображения-изображение,в което се отчита и светлината,осветяваща сцената и резултатът от такъв алгоритъм е съпоставим с фотоснимка,направена на прообраз на визуализационния модел.За производството на реалистични изображения към геометричния модел трябва да бъдат добавени и физични характеристики,касаещи разпространението на светлината.Например вид на материала,от който е създадено тялото,качества на повърхността на тялото,характеристики на средата,през която се разпространява светлината,местоположение и качество на източника на светлина и други

Въпросите 2-5 обсъждат чертожната техника,а 6-8 касаят фотореалистичната техника.За да се получи резултат на един алгоритъм за визуализация са необходими следните входни параметри:

1. Пространство –Е3
2. ГКС-дясноориентирана декартова координатна система
3. Пространствена сцена-предполагаме,че тя се поддържа от GMSys ,а телата в нея са групирани около центъра на ГКС
4. Наблюдател на сцената,чийто модел за алгоритъма е точка на гледане,която за удобство приемаме,че е върху оста z и направление на гледане,което е по посока на z
5. Проекционна равнина α-върху която ще получим изображението.От съображения за разбираемост на изображението се приема,че α е между наблюдателя и ПС,т.е. няма сечение с телата от сцената и е перпендикулярна на оста z.Това ни позволява с α да свържем локална координатна система
6. Вид проектиране.Използват се два вида проектиране а) паралелно(в случая ортогонално),при което се създава права от точката,която проектираме и направлението на гледане прохода на тази права с α в Р се нарича проекция на Р при паралелно проектиране с вектор б)централно(перспективно) проектиране-при него се прокарва права м/у деф. От проектираната точка и точката на гледане.Прохода на тази права с α наричаме централна проекция на т. Р.При централно проектиране центъра на пр. е точката на гледане W.
7. Прозорец на гледане WWD=α.Пак от съображения за яснота на изображението се приема този прозорец да е правоъгълник,страните му да са паралелни на координатните оси и да е центрирано в пресечната точка на диагоналите да съвпада с центъра на локалната координатна система
8. Обем на видимост-вида на проектирането и прозореца на гледане формират обем на видимост

По отношение на обема за видимост алгоритмите се делят на 2 типа.При първия тип се визуализира само онази част от пространствената схема,която е с обема на видимост.При втория тип алгоритми първо се проектира пространствената сцена,след което се извършва изрязване на прозореца за видимост.Обсъждаме 2 вид алгоритми,тъй като отрязването по прозорец на видимост обсъдихме в 1.6.Възможно е освен въведената предна проекционна равнина да бъде проектирана и задна проекционна равнина.В този случай обемът на видимост става R множество,тоест права през пирамида.Това са параметрите,необходими за реализация на алгоритъм с чертожна илюзия за тримерност.Допълнителните параметри за постигане на фотореалистични изображения ще обсъдим в 3.6.

# 2.Алгоритъм на Робъртс

Този алгоритъм е изграден от 2 части-в първа част се отстраняват лицата(ребрата),които се закриват за наблюдателя от самото тяло.Във втората част пространствената сцена се разглежда като списък от потенциално видими лица,които могат да се закриват взаимно за наблюдателя.

Първа част-идеята на този алгоритъм се основава върху наблюдението,че ако изпъкнал многостен,който е сечение на равнинни полупространства,е представен така,че нормалния вектор към примитивните равнини на неговите лица са ориентирани еднопосочно,за конкретност приемаме,че всички са насочени навън с възможност да бъдат открити,закривани от многостени лица(невидими за наблюдателя и на тази основа да бъдат отстранени и невидимите за наблюдателя ребра.Идеята е проста,ако нормалния вектор към лицето и вектора на гледане сключват ъгъл >90 градуса,-видимо,=90 градуса-частично видимо,което по съображения причисляваме към невидими<90 градуса-невидими.В схемата BR всяко ребро е сечение на точно 2 лица.От представянето да отстраним невидимите лица и невидимите ребра.-как да открием и отстраним невидимите върхове-Тази първа част на алгоритъма важи и за неизпъкнали многостени и многостени с дупки.След извършване на този тест обаче останалите лица,респективно ребра,са потенциално видими,тоест,могат да се закриват от други лица на тялото,на което те са принадлежащи.

Класифицирането на едно лице като видимо или невидимо се извършва лесно чрез скаларното умножение на нормалния вектор и вектора на гледане.Със списъка от потенциално видими лица се пристъпва към втора част на алгоритъма.Това означава,че всеки две лица от този списък се подлагат на следната обработка:

1. Те се проектират в α
2. Търси се и ако е налично се открива евентуално сечение
3. Съгласно вида проектиране(за нас е централно) през вътрешна за сечението точка се пропуска сонда за дълбочина.За нас това е лъч с начало точката на гледане,който преминава през фиксирана точка на сечението.Този лъч можем да наречем антипроектиране.Сондата задължително пресича и двете лица.Лицето,чийто проход е по-близък до наблюдателя(разстоянието се мери по лъча) закрива другото лице.Първото наричаме видимо,второто частично видимо.Частично видимото се обработва като откритото в равнината перпендикулярно сечение се проектира върху него и се извършва операцията изваждане.Така обработеното лице се включва отново в списъка за изследване с други лица.

В резултат се получава множество от видими за наблюдателя лица,чиито проекции върху α създават у наблюдателя желаната илюзия за тримерност.

Алгоритъмът на Робъртс се определя като геометрично векторен и мястото му като метод за изчисляване на свойства е в геометричния процесор(Съгласно философското разбиране на структурата дърво всяка ДСРГИ е основа/корен на GMSys т.е предполагаме оттук нататък,че функциите на системата за геометрично моделиране са разпределени между процесорите от 1 раздел.

# 3.Алгоритъм на Варнок.

Алгоритъмът на Варнок е геометрично растерен алгоритъм,т.е. мястото му е в дисплейния процесор.Като идея той се основава върху наблюдението,че всеки човек задържа по-дълго време зрението си(зрението е психофизиологичен процес,управляван от централната нервна система въз основа на получаване чрез паралелен интерфейс усещания(които наричаме картинки),произвеждани от рецептор,наречен ретина) върху онези части на пространствената сцена,които за него са по-сложни.Тази психологично условена закономерност Варнок интерпретира по следния начин като алгоритъм.Входни параметри на алгоритъма са

1. Пространствената сцена като списък от лица(приемаме,че лицата са триъгълници,тъй като за Варнок границата и вътрешността на лицата имат едни и съши визуализационни характеристики и статус
2. Прозорец на гледане WWD .Предполага се,че WWD е пикселизиран неявно(пиксел наричаме прозорец за гледане,който се изобразява в една екранна точка)Тази неявна пикселизация се изчислява на растера(множество от екранни точки,принадлежащи на вектора на гледане и задачата за налагане на прозорци от 1.6)

При тези начални условия алгоритмът на Варнок гласи:

Разглежда се проекцията на пространствената сцена в прозореца на гледане.Тази проекция се класифицира по отношение на прозореца на гледане като проста или сложна.Според Варнок цялата е проста,когато съществува прост линеен алгоритъм за изрисуването и върху растерна ЕЛТ.Ако сцената е сложна Варнок предлага прозорецът на гледане да се раздели на 4 равни части и за всеки подпрозорец да се приложи същия алгоритъм.В учебника е описан алгоритъма в термините на деление на 4 еднакви подпрозореца,но в тази си част делението на прозореца трябва да бъде функция на пикселизацията,т.е прозорците могат да не бъдат еднакви,и могат да не бъдат 4,а 2.Възможно е също при делението да се отчете локализация на проекцията с цел по-бързо достигане до прозорците,съдържащи проста проекция.Варнок дефинира следните възможности:

1. Никой ред,лице от проекцията няма сечение с прозореца
2. Точно едно лице от сцената има сечение с прозореца
3. Точно едно лице от сцената принадлежи на вътрешността на прозореца
4. Прозорецът принадлежи на вътрешността на проекцията

В оригиналната публикация се обсъжда и 5-та възможност,която е 4) с допълнение,че всички останали лица от сцената са по-далече(зад това лице).Но тази възможност изисква допълнителни изчисления по обработка на цялата схема и затова я игнорираме.

Обработката при визуализация на проста схема е ясна.Необходимо е сечение на лицето с произволен цвят да бъде представено чрез схемата 2.6 като клетката в случая е пиксел.Предполага се,че центърът(на тежестта) на всеки пиксел съответства на екранната точка,в която той се е изобразявал.Делението на WWD на подпрозорци продължава,докато се стигне до прозорец пиксел,който съдържа сложна проекция.В тази ситуация могат да се приложат различни техники за асоцииране на визуални характеристики и статус за съответния пиксел,но обикновено се избира следното просто решение.От точката на гледане през центъра на пиксела(защото имаме централно проектиране) се пропуска сонда за дълбочина и резултатът на това сондиране се присвоява на съответния пиксел.При това сондиране има възможност сондата да няма сечение с проекцията.Тази последна особеност определя алгоритмът на Варнок като приближен. Алгоритъм за визуализация(алгоритмът на Робъртс е точен). Алгоритъм на Варнок има много варианти за реализация(предварително сортиране на лицата от схемата при прецизни алгоритми при обработката на пиксел) като целта им е единствено ускоряване на алгоритъма.

# 4.Алгоритъм,използващ Z буфер

Този алгоритъм е растерен,тъй като има хардуерна реализация,следователно мястото му е в хардуерната дисплейна система от 1 раздел.Идеята е следната:

Входни параметри на алгоритъма са:

1. Пикселизиран прозорец на гледане
2. 1 лице от пространствената схема-това означава,че алгоритъма не познава пространствената сцена,а реагира(обработва) всяко постъпило лице,като обработката означава промяна на текущото му вътрешно състояние.

Вътрешното му състояние като структура от данни се състои от две числови матрици с размерите на пикселизацията.Първата матрица се нарича цветови буфер и той е дисплейната памет.Втората матрица се нарича буфер за дълбочина или z буфер,тъй като поради съображения за скорост се предполага,че наблюдателят е по оста z.Първоначално двата буфера имат следните стойности:

1. Цветовия буфер е запълнен с числото,кодиращо цвета на фона
2. Z буферът е запълнен с максималното цяло число,което означава ,че св. Сонда през съответния пиксел няма сечение със сцената.

При постъпване на лице(триъгълник) алгоритъма извършва следните дейности:

1. Търси се сечение на всеки пиксел с проекцията на това лице.Ако не бъде открито сечение се преминава към обработката на следващ пиксел
2. Откритото сечение се нарича фрагмент,т.е. в тази част алгоритъма обработва фрагменти
3. За всеки фрагмент обикновено чрез сонда за дълбочина се изчислява неговата отдалеченост от наблюдателя(ясно е,че ако центъра на пиксел няма сечение с фрагмента няма нужда от сонда за дълбочина,т.е. пиксела и проекцията нямат сечение)
4. Изчислената дълбочина се сравнява с дълбочината,записана в съответната на този пиксел клетка на z буфера.Ако новоизчислената дълбочина е по-малка от старата,записана в z буфера,то клетката се актуализира с новата дълбочина,а в съответната клетка на цветовия буфер се записва цвета на проектираното лице.Този алгоритъм също е приближен алгоритъм.

# 5.Алгоритми за поредово сканиране

Тези алгоритми също са векторно растерни.,т.е. в някакъв смисъл могат да бъдат разглеждани като варианти на алгоритъма на Варнок.Входните им параметри за следните:

1. Пространствена схема като списък от лица
2. Пикселизиран прозорец на гледане

За разлика от алгоритъма,използващ z буфер,този алгоритъм цели намаляването на размера на буфера като обикновено за се използва буфер с големината на един ред пиксели.От гледна точка на пикселизацията,прозорецът на гледане е разделен на редове,като алгоритмът обработва сцената ред по ред.Обработката на пикселен ред има две части:геометрична и растерна.

Геометрична част:пикселния ред се разглежда като прозорец,(т.е. понятието пиксел изчезва),през средата на който е прекарана права,наречена сканираща права.Границите на прозорците на лицата разделят тази сканираща права на отсечки,като всяка една отсечка принадлежи или има сечение с вътрешността на 0,1,или повече от 1 лица.Чрез сонда за дълбочина,пусната през центъра на такава отсечка се установява най-близкото до наблюдателя лице и неговите визуализационни характеристики и статус се асоциират със съответната отсечка.Казваме,че я превръщат в цветови сегмент.Като заключителна дейност в тази част на алгоритъма е възможно да се извърши обработка по слепване на съседни цветови сегменти с еднакви визуализационни характеристики.При тази геометрична обработка е възможно да се използва т.нар. кохерентност на сцената-малки изменения в малки околности.Това означава,че е необходимо да се пазят резултатите от сканиращата права на предходния ред(както и цветовия буфер на предходния ред).

Растерна част-върху множество от цветови сегменти се наслагват пикселите от този ред.Това означава,че за всеки пиксел се изчислява сечението му с цветовия сегмент.Това сечение може да бъде изградено(съставено) от 1 или повече цветови сегменти.

Когато пикселът има сечение само с един цветови сегмент,то визуализационните характеристики и статуса на сегмента се асоциират с пиксела.В противен случай(сечение с 2 или повече сегмента) има две основни техники за оцветяване на пиксела:

1. Избират се характеристиките на сегмента,който съдържа центъра на пиксела
2. Чрез някаква формула,зависеща от дължините и визуализационните характеристики на сегмента,имащи сечение с пиксела се изчислява някакъв цвят,който се асоциира с пиксела.Обикновено формулата е линейна.

Въведената класификация на алгоритъма за визуализацията като геометричен и растерен цели единствено да се демонстрира чрез разбираем пример.Технологията,чрез която се осъществява нарастването на „интелигентността” на хардуерните реализирани спец. Процесори.В информатиката,в частност компютърни науки,ако функционалността на 1 ВИМ или на един алгоритъм е подмножество на функции на друга ВИМ се казва,че А е по-малко интелигентна от 4 АСИ.(…)В живота неинтелигентен е отношение/първоначално еднопосочно на един човек към друг човек.

# 6.Прост модел на осветяване

Ако една цветна дигитална фотоснимка е съизмерима с резултата от работата на алгоритъм за визуализация,обработващ модел на реалността на фотоснимката,се казва,че алгоритъмът произвежда фотореалистични изображения.Предполагаме,че на всеки е ясен донякъде механизмът за фотозаснемане и той е,че отразената от пространствената сцена светлина,която попада върху някаква специфична среда за рисуване(фотоплака) и оставя следа,която наричаме снимка.Това означава,че алгоритмите за фотореалистични изображения трябва да използват някакъв модел на светлината,за да могат в някаква степен да имитират процеса на фотозаснемане.Заимстваме един прост модел на светлината(в аспект свойства на разпространението и) от оптиката,която е част от общата физика.Целта ни е да трансформираме този модел в термините на геометрична задача.Ние(програмистите) не знаем какво е светлина.Физиците казват,че светлината е нещо(вълна,частици),което притежава

1. Две основни измерими характеристики,наречени цвят и интензивност.В математическия модел,който ще създаваме това са две величини-1 от тип цвят,една от целочислен тип
2. Светлината за нашите нужди на фотозаснемането се движи по абсолютно права линия с пределна(мигновена) скорост.Това ни дава право да моделираме разпространението на светлината чрез лъч,чието начало наричаме светлинен източник,а посоката му е посока на разпространение
3. По време на своето разпространение в някаква среда светлината променя своите характеристики,за нас основно интензивността,като формулата за пресмятане на интензивността се влияе от качествата на средата за разспространение(наречено обикновено коефициент на затихване) и дължината на изминатия път.Ясно е,че пътят не зависи от времето,тъй като скоростта е пределна,т.е. дължината на лъч светлина е функция на възможността за достигане на преграда,т.е.л среда,през която не може да се разпространява
4. При достигане на такава преграда(с определени характеристики)лъчът светлина може да бъде:

-погълнат-означава,че тази среда(тяло)поглъща интензитета и цвета,т.е. лъчът изчезва.Такива тела се наричат абсолютно черни тела(черни дупки) и тях човек не може да види

-отразена-означава,че светлината се поглъща от тялото,след което излъчва във всички посоки равномерно светлинни лъчи с интензитет и цвят,чиято сума е по малка или равна на погълнатата светлина.Това отражение се нарича дифузно като върху качествата на отразената светлина влияе основно материала,от който е изградено тялото(и донякъде качеството на неговата повърхност).Ако повърхността е идеално гладка е възможно да се получи лъч огледално отразена светлина.За неговите характеристики,както и за характеристиките на всеки лъч дифузно отразена светлина в оптиката има формули.Ако наблюдателя се достигне от огледално отразен лъч,в общия случай се получава така нареченото заслепяване(пълно/частично) .Ако направлението на гледане не съвпада с отразения лъч върху отразената повърхност се наблюдава ярко петно,независимо какъв е цветът на повърхността.

-пропусната от т.нар прозрачни тела(например стъкло).

В модела,който ще използваме,ще използваме формулите за изчисляване на цвят и интензивност само на пропусната(дифузна) светлина.При този модел като входни параметри на алгоритъма са необходими още:

1. Местоположението и качеството на светлинните източници,осветяващи сцената.За нашите нужди ги моделираме чрез точка и ги наричаме точкови светлинни източници
2. Модел на средата,в която се разпространява светлината(ЗА НАШИТЕ НУЖДИ ПРЕДПОЛАГАМЕ,ЧЕ ТОЗИ МОДЕЛ В ВКЛЮЧЕН ПО НЯКАКЪВ НАЧИН ВЪВ ФОРМУЛИТЕ ЗА КАЧЕСТВАТА НА ОТРАЗЕНАТА/ПРОПУСНАТАТА СВЕТЛИНА)
3. Допълнителна физическа информация относно материята и качествата на повърхността на телата в пространствената сцена.Предполагаме,че тази информация се поддържа от системата за геометрично моделиране и е достъпна като параметри на формулите за отразена и пропусната светлина.

При тези съглашения най-простият алгоритъм за фотореалистични изображения е следния:

През центъра на всеки пиксел(от пикселизирания прозорец на гледане) се пропуска сонда за дълбочина.От резултата в аспект най-близка до наблюдателя точка от пространствената сцена се пускат сонди за осветеност към всеки точков светлинен източник.Една сонда за осветеност се обработва за осветеност в последствие,ако между началото и светлинния източник няма други тела от сцената.

Обработката на успешна сонда за осветяване означава да се изчислят съгласно възприетия модел цветът и интензивността на светлината,достигащи от един светлинен източник началото.Това означава,че тези точки(резултати от сонди за дълбочина)могат да бъдат наречени виртуални светлинни източници(луни).Като втора стъпка се изчисляват характеристиките на светлината,достигаща до точката на гледане от такава луна.(Съществено е,че луните могат да не излъчват светлина във всички посоки).Този най-прост модел за фотореалистични изображения има редица недостатъци от гледна точка на нивото на „фотореализъм”.Например губи се илюзията за дълбочина на телата отс . светлината,излъчена от луните,която осветява пространствената сцена допълнително.

# 7.Сенки

Макар,че простият модел от 6 често се нарича и алгоритъм чрез трасиране на лъчи,е възможно чрез предварителна обработка на сцената да се постигне изчисление на луните и тези изчисления да се използват в последствие при по-добрия алгоритъм за получаване на фотореалистични изображения чрез трасиране на лъчи.Казаното означава,че е необходимо предварително да се изчислят онези части от повърхността на всяко тяло от пространствената сцена,които имат еднаква осветеност(като количество и качество на светлинните източници,които ги осветяват непосредствено).От геометрична гледна точка сянка е централна проекция на едно тяло с център светлинния източник на тялото върху повърхността на друго тяло.Ако имаме повече от един светлинен източник като сенки се формират повече области с различна осветеност.Областта,която не е осветена от нито един източник се нарича собствена(плътна) сянка.Областите,които са осветени от подмножеството светлини източници се наричат полусенки,а всички останали части,осветени от всички светлинни източници се наричат осветени.Казаното означава,че по отношение на сенките всяко лице може да бъде разделено на светлинни фрагменти.светлинен фрагмент наричаме част от лице,осветена от известни като качества светлинни източници.Това ни позволява светлинни фрагменти да бъдат трансформирани като виртуални светлинни източници(луни)(съгласно приетия модел на осветяване) преди върху така конструираната сцена да се изпълни алгоритъмът за получаване на фотореалистично изображение чрез трасиране на лъчи.

# 8.Глобален модел на осветяване чрез трасиране на лъчи

Този алгоритъм е продължение на простия алгоритъм от 6.Идеята е ,че при откриване чрез сонда за дълбочина за виртуален точков светлинен източник,то е ясно(тъй като е точка от светлинен фрагмент) какви качества биха попаднали в точката на гледане от него.,т.е. в този момент е необходимо за този точков виртуален светлинен източник да се изчисли само светлината,с която го осветяват евентуално други светлинни фрагменти от сцената.Това означава,че в тази точка се прилага моделът за осветяване в обратна посока,т.е. точката на гледане се разглежда като светлинен източник,облъчващ сцената с „неутрална” светлина,моделирана чрез сондата за дълбочина като тази неутрална светлина в точковия ВСИ би могла съгласно модела на осветяване да бъде отразена или пропусната.Това означава,че сондата за дълбочина формира 2 нови лъча светлина на пропусната и отразена светлина в тази точка.Тези лъчи отново се разглеждат като сонди за дълбочина и се проследяват от същия алгоритъм.Този процес на проследяване на лъчите,породени от сонда за дълбочина се фиксира в структура от данни двоично дърво,чийто корен е точката на гледане.

Възлите на това дърво са ВСИ,а ребрата му са от два типа-за пропусната и за виртуално отразена светлина.Проследяването на сонда за дълбочина(респективно изграждане на съответното дърво) приключва когато

1. По технически причини т.е когато времето,отделено за изграждането му се изчерпа
2. Когато всички листа на дървото обозначават или светлинен източник или пътят от листото до точката на гледане е достатъчно дълъг,така че излъчената от листото светлина не влияе върху осветеността на корена на дървото

Така изграденото листо се обхожда от листата към корена(всички листа са точкови ВСИ ,чиито светлинни характеристики или са собствени,или се назначават служебно в началото на алгоритъма за обхождане) като при това обхождане във всеки възел се прилага модела на осветяване и полученият резултат се предава към предходника.Полученият в корена на дървото резултат се асоциира с пиксела на просп. Сонда за дълбочина.