


| | |
|---|--|
|  | <p>Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)</p> |
|---|--|

ФАКУЛЬТЕТ _____ ИУ _____

КАФЕДРА __ Программное обеспечение ЭВМ и информационные технологии _____

ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

Студент Гасанзаде Мухаммедали Алиназим оглы
фамилия, имя, отчество

Группа ИУ7-46

Тип практики Стационарная

| | | |
|---------|----------------------|-----------------------|
| Студент | _____ | <u>Гасанзаде М.А.</u> |
| | <i>подпись, дата</i> | <i>фамилия, и.о.</i> |

| | | |
|-----------------------|----------------------|----------------------|
| Руководитель практики | _____ | <u>Куров А.В.</u> |
| | <i>подпись, дата</i> | <i>фамилия, и.о.</i> |

Оценка _____

2019 г.

ОГЛАВЛЕНИЕ

| | |
|--|-----------|
| Введение | 3 |
| Задание на выполнение курсового проекта | 4 |
| 1. Описание предметной области | 4 |
| 2. Описание алгоритмов | 4 |
| 3. Анализ выбранного метода | 5 |
| 4. Критерии выбранного метода | 8 |
| 5. Модель освещения | 9 |
| Заключение | 13 |
| Список литературы | 14 |

ВВЕДЕНИЕ

Цель практики: разработать аналитическую и конструкторскую части курсового проекта по дисциплине компьютерная графика.

Задачи:

1. Определить и описать предметную область работы.
2. Рассмотреть существующие методы и алгоритмы, подходящие для решения поставленной задачи.
3. Выбрать реализуемые методы или алгоритмы, обосновать выбор.
4. Сформулировать постановку проблемы в общем виде.
5. Разработать алгоритм метода моделирования, используемые структуры данных.

В задачу современной компьютерной графики обязательно входит разработка алгоритмов представления трехмерных объектов и их оптимизация. Безусловно, рост производительности по закону Мура уменьшает необходимое сокращение вычислений, однако возможности графических карт не безграничны. Алгоритмы рендеринга созданы несколько десятков лет назад, имеют длинную историю совершенствования и применяются до сих пор.

Целью данной работы является анализ методов рендеринга. Необходимо подобрать методы, наиболее подходящие для реализации курсового проекта, проанализировать наиболее удобные из них, описать минусы и сделать выбор исходя из подобранных критериев.

Аналитическая часть

ЗАДАНИЕ НА ВЫПОЛНЕНИЕ КУРСОВОГО ПРОЕКТА

Разработать модель распространения света на границе трёх сред: воздушной, стеклянной и водной. Объект аквариум, стеклянная среда — его стенки, водная среда — заполняет аквариум, у аквариума каменистое дно (гравий). Вода в аквариуме стоячая и прозрачная. Цвет света выбирается пользователем, источник бесконечно удалённый — в каждой точке сцены поток света одинаковый (интенсивность), пользователь должен иметь возможность задавать любое произвольное положение и количество источников.

ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

Компьютерная графика - очень значимый раздел в IT сфере. Алгоритмы и методы, представляющие данную область, являются необходимыми для решения множества задач, актуальных в наше время. В частности, решение задачи моделирования распространения света на границах сред - водной, стеклянной и воздушной - является крайне важным для построения реалистичных изображений, так как распространение света на границе двух сред встречается очень часто, например, когда мы ставим стакан с водой под лампу.

ОПИСАНИЕ АЛГОРИТМОВ

Существует 4 группы методов рендеринга:

1. Растеризация
2. Рейкастинг
3. Трассировка лучей
4. Трассировка пути

Ниже рассмотрим каждый из этих алгоритмов, проанализируем их минусы и плюсы. При кратком анализе перечисленных методов рендеринга можно сразу отказаться от двух из них:

1. Рейкастинг: данный метод не подходит из-за того, что он не вычисляет новые тангенсы лучей света (*которые возникают после пересечения луча, идущего от глаза к источнику света с поверхностью, а у нас задача исследовать преломление света*), что приводит к

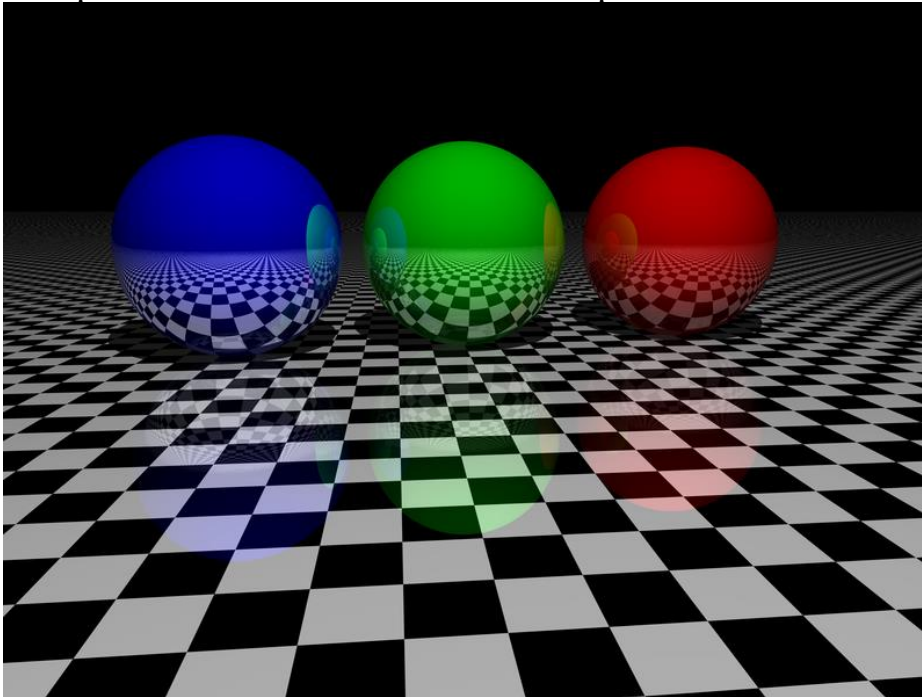
невозможности точно отрендерить отражения, преломления и естественные проекции теней. Этот метод обычно используется для визуализации поверхностей.

2. Трассировка пути: Трассировка пути является наиболее простым, наиболее точным с физической стороны и наиболее медленным по производительности методом рендеринга. Трассировка пути естественным способом воспроизводит множество оптических эффектов, которые тяжело достижимы или вообще недостижимы другими методиками рендеринга: построение теней, глубина резко изображаемого пространства и т.д. Осуществление этих оптических эффектов при помощи трассировки пути намного проще, чем при помощи других методик. Данный метод не подходит из-за требовательности к ресурсам.
- Растеризация имеет преимущество по производительности, но с учетом требований к качеству изображения и наличию множества оптических эффектов на сцене, был выбран алгоритм трассировки лучей.

Вследствии этого, можно сделать вывод, что для построения реалистичной модели распространения света на границе сред наиболее подходящими является алгоритм **Трассировки лучей**.

АНАЛИЗ ВЫБРАННОГО МЕТОДА

Трассировка лучей - технология, предназначенная для построения изображений 3d моделей в компьютерах.



Существует **II** метода трассировки лучей - **forward tracing** (прямая трассировка) и **backward tracing** (обратная трассировка). В основе обоих методов лежит **concept of shooting rays** (концепт стреляющих лучей).

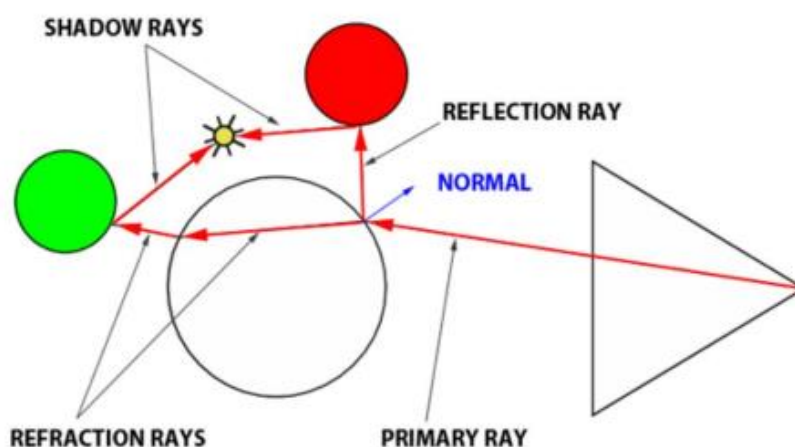
Но **forward tracing** является крайне неэффективным методом. Его суть состоит в том, что в источнике света генерируется пучок лучей, распространяющихся во всех возможных направлениях от источника. И лишь малая часть этих лучей в результате движения (преломления, отражения и тд.) попадает на камеру (приемник). Также, в результате работы отслеживается изменение энергии всех лучей (включая те, которые в конечном итоге так и не попадут на камеру). Вышеперечисленные причины делают метод прямой трассировки малоэффективным и слишком объемным в плане расходования компьютерных мощностей.

Поэтому целесообразнее подробно рассмотреть метод **обратной трассировки**. В сущности, данный метод работает только с теми лучами, которые в конечном итоге пришли в камеру. Т.е. мы генерируем лучи не от источника света, а от камеры. Количество лучей, подвергаемых трассировке, ограничено. Это очень сильно уменьшает объем ресурсов, необходимых для рассчитывания и моделирования сцены. В результате рейтрейсинга мы получаем изображение, собранное из пикселей.

Мы проводим линию от камеры до центра следующего пикселя (проходим по всему экрану). Так как мы знаем направление луча, мы можем найти все пересечения данного луча с объектами сцены. Если объектов, пересекающих луч, несколько, выбирается объект, находящийся ближе остальных к камере. Далее из точки пересечения проводится линия к источнику света. Если на пути есть объект, то точка затенена. В противном случае, освещена. Для вычисления освещенности необходимо сложить количество света от каждого источника, приходящее в рассматриваемую точку, а затем умножить результат на цвет поверхности в данной точке.

Но, так как нам важна фотореалистичность, для решения нашей задачи нам понадобится учесть характеристики материалов, то есть количество отражаемого и преломляемого ими света. Этот метод является рекурсивным.

В начале мы рассчитываем направление отражения. Для этого нам необходимо знать исходное направление луча (от камеры к середине пикселя) и нормаль к поверхности в точке пересечения. Узнав направление отражения, мы отправляем новый луч в этом направлении. Далее делаем тоже самое с преломлением, но для этого нам также понадобится индекс преломления материала.



Явным недостатком данного метода является то, что для остановки рекурсии приходится вводить лимит по интенсивности света. Каждый раз, отражаясь или преломляясь, свет теряет интенсивность, и именно это помогает избежать случая, когда луч будет бесконечно прыгать по сцене. Также к недостаткам нужно отнести тот факт, что наибольшая часть вычислений в данном алгоритме уходит на расчет пересечений лучей с поверхностями. Это делает выбранный алгоритм неэффективным в сложных сценах с большим количеством объектов.

Плюсами являются:

1. Высокое качество изображения
2. Простота в реализации

3. Возможность корректно создавать большинство оптических эффектов, таких как тени, преломления, диффузные отражения. Диффузное отражение необходимо, если объект имеет неровности на поверхности. В таком случае, луч отражается в сцену равномерно во всех направлениях. В качестве примера можно привести, например, грубую ткань. Ее цвет, то есть свет, отраженный от нее, одинаков в любом месте.

КРИТЕРИИ ВЫБОРА МЕТОДА

При выборе метода для реализации курсового проекта учитывались 3 основных критерия:

1. Реалистичность и качество итогового изображения
2. Производительность
3. Удобство реализации

Конечно, очень важна требовательность метода к производительным мощностям компьютера. Для решения поставленной задачи нет необходимости использовать запредельные мощности, необходимые, например, для современных компьютерных 3d игр. Наоборот, в нашем случае технология должна быть наименее требовательной к железу (насколько это возможно, поэтому отказались от метода «трассировка пути» т.к. в данном случае нужно было бы использовать мощное железо). Но главное, конечно - качество изображения. Нам необходимо получить наиболее фотореалистичное изображение. На нашей сцене будет необходимо изобразить множество визуальных эффектов, таких как тень, отражение, преломление. Все эти эффекты должны выглядеть максимально качественно и натурально. И, по приоритетам, это важнее, чем производительность, так как основная цель курсовой работы - получение реалистичного изображения, а не реализация наиболее производительного алгоритма. Нам не подходит метод, который, скажем, делает невозможным четко отобразить отражение, но зато быстро работает. Также, при выборе учитывается удобство реализации, ведь от этого во многом зависит результат. Если один алгоритм предусматривает какие-то специфические особенности, которые не предусматривает другой алгоритм - это явно является большим плюсом в сторону первого.

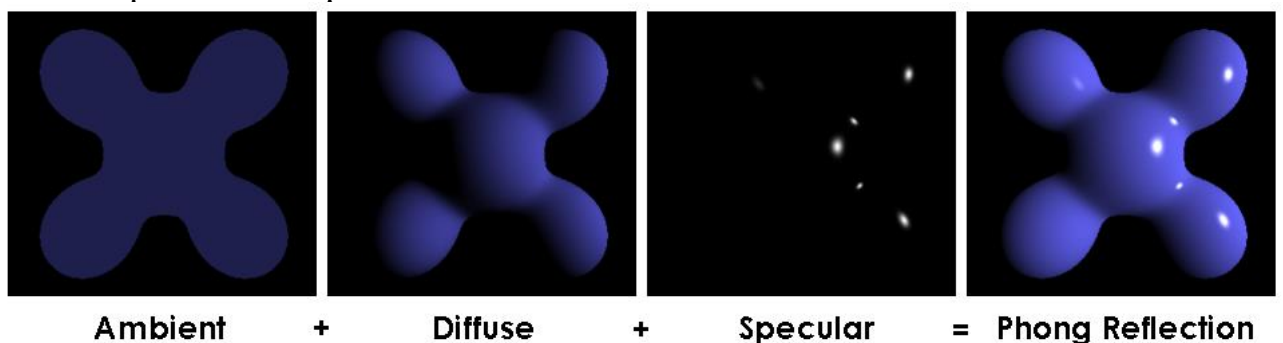
МОДЕЛИ ОСВЕЩЕНИЯ

В сцене может находиться несколько источников света. В основном используется три модели освещения:

- **Модель Ламберта**
Самая простая модель освещения, учитывает только диффузную составляющую. Делает поверхность матовой, без бликов. Так как такая модель является базовой, то она не дает правильного освещения, поэтому использовать ее мы не будем.
- **Модель Фонга**
Состоит из диффузного освещения по модели Ламберта и зеркального освещения. В этой модели на объекте появляются блики. Падающие и отраженные лучи лежат в одной плоскости.
- **Модель Блинна-Фонга**
Данная модель похожа на модель Фонга, но имеет отличие, которое расписано ниже.

В модели Фонга для света выделяют три составляющих:

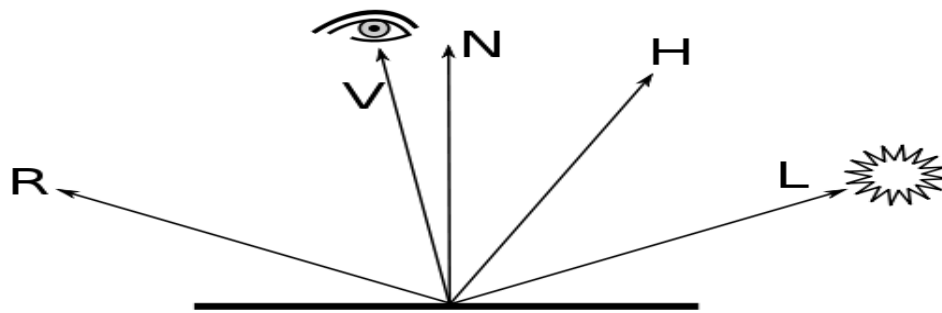
- Фоновое освещение - Ambient
- Диффузное - Diffuse
- Зеркальное - Specular



Модель освещения Фонга.

Перед описанием моделей следует рассмотреть векторы для модели Блинна-Фонга:

- **N** - нормаль к поверхности
- **L** - направление к источнику света
- **R** - направление отраженного луча
- **V** - направление к наблюдателю
- **H** - дополнительный вектор для упрощения подсчета



Векторы в модели Блинна-Фонга.

Фоновое освещение используется для общего освещения сцены, как дополнительный компонент. И вычисляется по формуле:

$$I_a = k_a \cdot i_a$$

k_a – коэффициент фонового освещения,
 i_a – интенсивность.

Зеркальное освещение показывает блики на объекте и хорошо передает расположение источника света. Вычисляется по формуле:

$$I_s = k_s (R \cdot V)^\alpha i_s$$

k_s – коэффициент зеркального отражения,
 i_s – интенсивность зеркального освещения,
 R – вектор отраженного луча,
 V – вектор на наблюдателя,
 α – коэффициент материала.

Улучшение модели Блинна-Фонга состоит в том, что не используется вектор отраженного луча, который сложен для вычислений. Вместо этого используют вектор H . Вычисляется по формуле:

$$H = \frac{L + V}{|L + V|}$$

Таким образом, формула интенсивности имеет вид:

$$I_s = k_s \cos^\alpha (\vec{N}, \vec{H}) i_s$$

Диффузное освещение - рассеянный свет, распространяется равномерно.

Интенсивность диффузного освещения вычисляется по формуле:

$$I_d = k_d \cos(\vec{L}, \vec{N}) i_d = k_d (\vec{L} \cdot \vec{N}) i_d$$

k_d — коэффициент диффузного освещения,

i_d — интенсивность диффузного освещения,

L – вектор на источник,

N - вектор нормали в точке.

На каждом шаге нужно проверять сумму полученных интенсивностей и сравнивать ее с интенсивностью входящего луча. Как я описал выше, выходящие лучи не могут превосходить входящие.

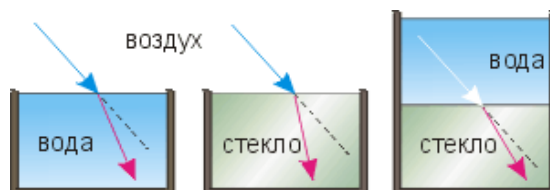
Сложение полученных интенсивностей дает хороший результат для освещения сцены. Освещение от данных моделей можно программно варьировать с помощью коэффициентов. При этом получается более точная модель освещения. В данной работе буду использовать освещение Блинна-Фонга из-за его упрощения по сравнению с моделью Фонга.

Для получения цвета используется модель RGB, где каждый цвет задан с помощью интенсивностей базовых цветов: синий цвет может быть определён как (0,0,255), красный как (255,0,0), фиолетовый (255,0,255), чёрный (0,0,0), а белый (255,255,255).

Поэтому для подсчета интенсивности следует учитывать каждый из трех базовых цветов.

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} A_r r_a \\ A_g g_a \\ A_b b_a \end{bmatrix} + \sum_{i=1}^N \begin{bmatrix} I_r \left(r_d(\hat{\mathbf{n}}_i \cdot \hat{\boldsymbol{\ell}}_i) + r_s(\hat{\mathbf{r}}_i \cdot \hat{\mathbf{v}})^n \right) \\ I_g \left(g_d(\hat{\mathbf{n}}_i \cdot \hat{\boldsymbol{\ell}}_i) + g_s(\hat{\mathbf{r}}_i \cdot \hat{\mathbf{v}})^n \right) \\ I_b \left(b_d(\hat{\mathbf{n}}_i \cdot \hat{\boldsymbol{\ell}}_i) + b_s(\hat{\mathbf{r}}_i \cdot \hat{\mathbf{v}})^n \right) \end{bmatrix}$$

Преломлением света называют изменение направления луча на границе раздела двух сред, при котором свет переходит во вторую среду. Например, на рисунке мы изображён пример преломления светового луча на границах воздуха и воды, воздуха и стекла, воды и стекла (те 3 границы которые мы и будем рассматривать).



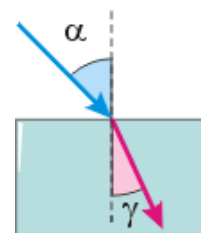
Из сравнения рисунков следует, что пара сред «воздух-стекло» преломляет свет сильнее, чем пара сред «воздух-вода». Из сравнения рисунков видно, что при переходе из воздуха в стекло свет преломляется сильнее, чем при переходе из воды в стекло. То есть, пары сред, прозрачные для оптических излучений, обладают различной преломляющей способностью, характеризующейся относительным показателем преломления. Он вычисляется по формуле, которая будет указана ниже, поэтому может быть измерен экспериментально. Если в качестве первой среды выбран вакуум, то получаются значения:

| | | | | |
|--------|--------|--|----------|-----------|
| Вакуум | 1 | | Вода | 1,33 |
| Воздух | 1,0003 | | Глицерин | 1,47 |
| Лёд | 1,31 | | Стекло | 1,5 – 2,0 |

Эти значения измерены при 20 °С для жёлтого света. При другой температуре или другом цвете света показатели будут иными. Поскольку показатель преломления воздуха почти не отличается от единицы, влияние воздуха на распространение света практически незаметно.

В входных данных будет таблица с показателями для разных цветов, но при постоянной температуре, так как кипение воды в данной задаче не рассматривается. Также будет учитываться то, что стекла аквариума с потёртостями.

Закон преломления света. Чтобы рассмотреть этот закон, введём определения. Угол между падающим лучом и перпендикуляром к границе раздела двух сред в точке излома луча назовём *углом падения* (α). Аналогично, угол между преломлённым лучом и перпендикуляром к границе раздела двух сред в точке излома луча назовём *углом преломления* (γ).



При преломлении света всегда выполняются закономерности, составляющие **закон преломления света**:

1. Луч падающий, луч преломлённый и перпендикуляр к границе раздела сред в точке излома луча лежат в одной плоскости.
2. Отношение синуса угла падения к синусу угла преломления – постоянная величина, не зависящая от углов(именно поэтому метод **рейкастинга** не был рассмотрен):

$$\frac{\sin \alpha}{\sin \gamma} = n$$

n – относительный показатель преломления

α – угол падения луча

γ – угол преломления луча

Применяют и качественную трактовку закона преломления света: при переходе света в оптически более плотную среду луч отклоняется к перпендикуляру к границе раздела сред. И наоборот.

ЗАКЛЮЧЕНИЕ

В ходе практики разработана аналитическая часть курсовой работы по теме «Моделирование распространения света на границе сред: воздушной, стеклянной и водной», сделан анализ методов, подходящих для решения поставленной задачи, а также был выбран метод, которым будет реализовываться курсовая работа.

СПИСОК ЛИТЕРАТУРЫ

1. What's the Difference Between Ray Tracing and Rasterization? by Brian Caulfield. [Электронный ресурс]. - URL: <https://blogs.nvidia.com/blog/2018/03/19/whats-difference-between-ray-tracing-rasterization/>
2. Computer Graphics from scratch by Gabriel Gambetta. [Электронный ресурс]. - URL: <http://www.gabrielgambetta.com/computer-graphics-from-scratch/introduction.html/>
3. DirectX Raytracing: трассировка лучей в реальном времени, Алексей Берилло. [Электронный ресурс]. - URL: <https://www.ixbt.com/3dv/directx-raytracing.html> 13
4. Rasterization: A Practical Implementation [Электронный ресурс]. - URL: <https://www.scratchapixel.com/lessons/3d-basic-rendering/rasterization-practical-implementation>
5. Introduction to Ray Tracing: A Simple Method for Creating 3D Images [Электронный ресурс]. - URL: <https://www.scratchapixel.com/lessons/3d-basic-rendering/introduction-to-ray-tracing/how-does-it-work>
6. Raytracing vs Rasterization by Philipp Slusallek and David Kirk [Электронный ресурс]. - URL: <http://www.scarydevil.com/~peter/io/raytracing-vs-rasterization.html>
7. Introduction to Realtime Ray Tracing by Philipp Slusallek, Peter Shirley, Bill Mark, Gordon Stoll, Ingo Wald [Электронный ресурс]. - URL: <https://pubweb.eng.utah.edu/~cs6958/papers/a1-slusallek.pdf>
8. Hugo Pacheco - Ray Tracing in Industry
An up-to-date review of industrial ray tracing applications and academic contributions. [Электронный ресурс]. - URL: <https://paginas.fe.up.pt/~aas/pub/Aulas/DiCG/HugoPacheco.pdf>
9. Ray Tracing Illumination [Электронный ресурс] - URL: <http://www.cs.otago.ac.nz/cosc342/2017-notes/342-2017lect16.pdf>